

Article

A Hardware-Software Complex for the Reconstruction of Unmanned Aerial Vehicle Digital Traces Under Logical Data Damage Using LSTM-Based Telemetry Recovery and Multi-Source Confidence Scoring

Azamat Baibussinov ^{1,*}, Madi Shayakhmetov ^{2,*}, Leila Rzayeva ² and Kaisarbek Yesbergenov ¹

¹ Department of Technical Support, Faculty of Comprehensive Support, National Defense University of the Republic of Kazakhstan, Astana 010000, Kazakhstan; ekb72@mail.ru

² Research and Innovation Center “CyberTech”, Astana IT University, Astana 010000, Kazakhstan; l.rzayeva@astanait.edu.kz

* Correspondence: a.baibussinov@astanait.edu.kz (A.B.); 242722@astanait.edu.kz (M.S.)

Abstract

(1) Background: The digital traces of unmanned aerial vehicles (UAVs) are becoming increasingly important in criminal incidents, the violation of airspace and in military operations, thus making the reconstruction of the digital traces a critical task. But, current tools like DatCon, Autopsy and GRYPHON cannot recover telemetry when the flight logs are logically damaged, fragmented or partially deleted and don't offer any quantitative measurement of the confidence of the recovered information. (2) Methods: A unified hardware-software complex, including a forensic workstation, a hardware write-blocker and SD/microSD/eMMC adapters; a set of software modules for extracting artifacts from files, structural parsing of DAT/BIN/CSV log, neural network reconstruction of missing telemetry using a two-layer LSTM architecture; a multi-source correlation module that combines flight logs, telemetry, media metadata and controller artifacts; a module, Confidence Score (CS), that computes a reliability measure in $[0, 1]$; and a visualization module to generate a reconstructed trajectory on an electronic map. (3) Results: The complex has been tested on 105 flights on 10 different UAVs, 492 flight logs were gathered, 10,435 were the media item files and 624 GB was the amount of storage during acquisition. The carving stage recovers 98.7% of artifacts across the eight signature classes, the LSTM module recovers all five telemetry parameters with $R^2 > 0.99$ and a single-step horizontal position error of 6.8 m, which is reduced to 4.7 m after multi-source correlation (below the 5 m operational target consistent with consumer-GNSS precision); the dependence on gap length is described by the empirical growth law $\varepsilon_{\text{horiz}} \approx 4.84 \cdot G^{1.44}$ m; 46.8% of recovered records fall within the high-confidence band of $CS \geq 0.8$; and the complex outperforms DatCon, Autopsy + DJI Analyzer and GRYPHON by 22–35 percentage points in end-to-end record recovery and by a factor of ~ 2.6 in mean horizontal error (4.7 m vs. 12.4–18.7 m). (4) Conclusions: The combined write-blocked hardware acquisition, neural reconstruction of telemetry, and quantitative confidence index provides a forensically structured pipeline that fills an existing gap in UAV digital forensics; we note that technical reconstruction accuracy does not by itself confer legal admissibility, which remains a function of jurisdiction-specific evidentiary standards discussed in the Conclusions.



Academic Editors: Mario Antunes and Carlos Rabadão

Received: 29 May 2026

Revised: 1 July 2026

Accepted: 8 July 2026

Published: 13 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

Attribution (CC BY) license.

Keywords: UAV forensics; drone forensics; digital traces; flight log recovery; LSTM; telemetry reconstruction; file carving; multi-source correlation; confidence score; chain of custody

1. Introduction

In the past decade, civilian and military operations have increasingly been conducted using unmanned aerial vehicles (UAVs), changing commercial operations and the field of digital evidence [1]. Investigations of commercial-off-the-shelf platforms like the DJI Phantom, Mavic and Mini series, the Autel EVO and ArduPilot-based custom airframes are now commonplace in scenarios from privacy invasions to contraband delivery, from airspace trawls to battlefield reconnaissance [2,3]. A wealth of digital artifacts is produced by each flight, including flight logs in proprietary formats such as DAT or BIN, telemetry data, photographs or videos with embedded EXIF/XMP metadata, and logs from the paired mobile device used to fly the drone, which can be collected and properly interpreted to enable the reconstruction of who flew which drone, when, from where and along what trajectory [4,5].

However, in practice these artifacts are seldom undamaged at the time of seizure. The file system in the on-board SD/microSD card or the internal eMMC memory can be corrupted during crash, flight logs are often deleted by operators when they hand over the devices, and the firmware can interrupt the log write mid-record [1,6,7] while anti-forensic behaviour, such as factory resets, intentional overwrites and encrypted logs, is becoming more common. This type of damage, in which the underlying media is still physically readable but the higher-level data structures (such as timestamps, checksums, record framing) are damaged, is called *logical data damage* in this paper. The most significant type of damage is logical, which occurs in nearly all operational UAV-forensic casework and is not destructive like chip-off acquisition, which only occurs in very few cases and requires the destruction of the chip [8,9].

There are a few popular tools that have partially solved the issue. The utility Dat-Con [10] is open-source and can output CSV and KML from intact DJI DAT/TXT logs, but does not support fragmented or partially-overwritten log records, nor does it provide any indications of the veracity of the output. The forensic platform Autopsy, in combination with the module DJI Drone Analyzer [11], extracts artifacts from a forensic image, without neural reconstruction of missing telemetry. The most similar working example to the present work was GRYPHON system developed by Mantas and Patsakis [6] which parses dataflash and MAVLink telemetry logs, detects inconsistencies, but lacks hardware acquisition, does not impute missing telemetry parameters when log is incomplete, and does not provide a quantitative confidence index suitable for court use.

The deep-learning community has shown that deep recurrent networks, among which Long Short-Term Memory (LSTM) networks and their descendants are particularly prominent, can be successfully applied to imputing missing values in spatio-temporal trajectories, independent of the field of UAV forensics [12–14]. On civilian flight datasets, LSTM models are able to predict several time steps ahead with sub-decimeter accuracy of latitude, longitude, altitude, speed and heading, and recent work on multi-source data fusion using 1D CNN-BiLSTM [15] demonstrates that fusing multiple heterogeneous flight-data sources leads to better anomaly detection and recovery on small UAV samples. These techniques have not been incorporated into a forensically structured pipeline, though, since these are the common assumptions used in the typical machine-learning study, which do not hold true for seized evidence.

Here, the contribution of this paper is the design, implementation and evaluation of a unified hardware-software complex which fills exactly this gap. The complex combines:

- a forensic workstation that includes a hardware write blocker and a suite of physical adapters (SD, microSD and eMMC), which provide read-only, reproducible acquisitions [3];
- a software module to extract digital artifacts from forensic images through signature-based file carving; [16,17];
- a parsing and recovery module for flight logs in various formats, such as DAT, BIN and CSV;
- an LSTM-based module which operates when the log is partially destroyed and reconstructs the missing parameters of the telemetry;
- a multi-source correlation module that correlates recovered log records to telemetry streams, media metadata and controller artifacts [18,19];
- a Confidence Score (CS) module with the output of a quantitative reliability index ranging from 0 to 1 for each recovered record, calculated as a weighted sum of source confirmations [20];
- a visualisation and reporting module that displays the reconstructed flight path on an electronic map, is colour coded by CS and generates a structured forensic report.

The novelty of the work is in these three features combined into a single forensically structured pipeline—write-blockers, file carving and LSTM models are individually well established [21–23]—namely: (i) handling of logical damage, (ii) imputing missing telemetry via a neural model, and (iii) the output of a quantitative reliability index with every reconstructed datum that supports, but does not by itself establish, the evidentiary use of the recovered data. To our knowledge, there was no previously published tool that was capable of combining all three properties.

The rest of the paper is organized as follows. Section 2 reviews related work on UAV forensics, trajectory imputation and multi-source evidence fusion. The full workflow of the proposed complex is explained in Section 3 and the architecture of the LSTM is explained in detail along with the formula for the Confidence Score. Ten UAV models were experimentally evaluated as reported in Section 4. Limitations and admissibility issues are discussed in Section 5 and the conclusions are provided in Section 6.

2. Related Work

2.1. Acquisition and Artifact Extraction from UAVs

The preliminary investigations in the field of UAV forensics outlined the necessity of a model that addresses UAV specific acquisition, identification, and evaluation of physical evidence that is grounded in the classical model of physical evidence acquisition–identification–evaluation–admission [1,24]. Salamh et al. [4] used the DJI Phantom 4 and Matrice 210 to extensively benchmark each tool (Autopsy, Magnet AXIOM and Cellebrite) and showed that none of them recover all of the artifacts and that flight-trajectory reconstruction in three dimensions is necessary to understand the recovered evidence. The DJI Mavic Air 2 was investigated by Lan and Lee [7] who found that the encrypted .DAT and .txt flight logs for newer drones still remain difficult to open and decrypt with the major tools available at the time, highlighting the persistent encryption barrier in newer DJI models.

Note that for internal eMMC storage (as used by the DJI Matrice and newer Mini/Mavic variants), chip-off acquisition is still the standard fallback for the situation where connection via the normal port is not available [3,25]. This invasive approach lets one access directly the raw NAND content, but it is hardware dependent, destroys the device and requires the use of a write-blocker applied to the extracted chip to preserve

integrity [3]. In a recent review of drone forensics in law-enforcement settings, we found that there are four main challenges to admissibility: encryption, anti-forensic behaviour, proprietary log formats, and weak chain-of-custody documentation [26]. The same review called for the development of tools that will provide reproducible, validated outputs with a quantitative level of confidence.

2.2. Flight-Log Parsing and Decoding

DatCon [10] is a de-facto open-source decoder for the DJI proprietary logs which converts the DAT and TXT logs into CSV and KML and is now compatible with the Phantom 3/4, Inspire 1, Spark and original Mavic Pro. DatCon can only process up to 279 telemetry columns per flight and does not support newer encrypted models, like the Mavic Air 2 or the FPV series [2]. The DROP tool is a decoder for the DJI Phantom III log file format, it is the default open source solution on the Phantom III platform, and was released by Clark et al. [5]. Commercial systems, like Cellebrite UFED, MSAB XRY Drone, Oxygen Forensic Detective, Magnet AXIOM, offer more coverage of models and cloud extraction built in [2], but are still closed-source, have limited pricing and also do not generate a quantitative reliability index in addition to the decoded log.

GRYPHON [6] is a software framework for the forensic analysis of MAVLink-style telemetry and dataflash logs, which is closest to the present work. GRYPHON is able to detect inconsistencies between the different sensor channels, and to visualize recovered telemetry, but assumes perfect telemetry log structures, missing or fragmented telemetry is passed as a gap in the output, and there is no acquisition step in the pipeline.

2.3. Trajectory Reconstruction and Imputation

In the non-forensic setting, recurrent neural networks, particularly its bidirectional forms, LSTM and BLSTM, are now the prevailing method for short-term and gap-filling prediction of vehicle and aircraft trajectories [12,27]. Zhang et al. [28] proposed the stacked LSTM network for predicting the latitude, longitude and altitude of the UAV with sub-meter accuracy for 1 to 2 s. Eum and Yoo [13] showed that it is possible to recover GPS gaps of up to several minutes by leveraging the spatio-temporal regularity of activity patterns, using a multi-scale residual CNN-LSTM imputation model. Galdelli et al. [14] presented a bidirectional LSTM (BF-BiLSTM) that uses both past and future data to reduce the MAE on vessel and flight data sets for long-term data gaps ranging from 5% to 30%.

It is shown that deep recurrent models can plausibly reconstruct missing telemetry, demonstrating promising results for deep recurrent models. None of the cited studies, however, are set up as forensic procedures; the training sets are assumed to be clean, there is no feedback of generated values into the chain of custody, and there is no evidentiary level of confidence.

2.4. Multi-Source Correlation and Confidence Scoring

It is well known that correlating heterogeneous artifacts will help bolster the evidence for the attribution. STITCHER [19] formalized the problem of correlation and consistency across the range of IoT devices and suggested a structured way of integrating evidence provided by a number of sources. In the UAV context, Kim et al. [18] demonstrated that cloud-based data from DJI servers can be correlated with data on the ground and the device to attribute flights in cases where local data has been deleted. The multi-layer framework for drone forensic inference proposed in very recent works [20] introduces a Forensic Confidence Index (FCI) that assigns a confidence level to the reliability of each evidence from the hardware, log, network and cloud layers.

The present work is in a similar vein, but with two significant differences. One of our key strengths, first, is that we calculate our Confidence Score at the record level (per

timestamp) instead of the case level so that the trajectory of the records can be visually colour coded and only the records with a particular level of confidence are included in the report. Second, the score is calculated after the LSTM imputation step, so that if the value for a record comes entirely from the network (no source confirmation) the record is given as a low confidence item to prevent accidental overclaim.

3. Materials and Methods

3.1. System Architecture

The proposed hardware-software complex is designed as a 6-stage pipeline starting with the physical acquisition of a UAV digital storage device and ending with the generation of a structured forensic report. Throughout this paper we use the term *complex* to denote the full integrated system—the forensic workstation, write-blocker and adapters together with the six software modules—and the term *pipeline* to denote the ordered six-stage flow of processing that the complex executes on a given piece of evidence; the two terms are therefore not interchangeable synonyms but refer, respectively, to the system and to the process it carries out. The overall architecture is depicted in Figure 1. It starts from the seized storage media (SD/microSD card, internal eMMC of the UAV, or chip-off NAND extract), which is plugged into a forensic workstation via a hardware write blocker. The write-blocker will be used to ensure that no write operation will ever affect the original evidence and that the acquired image will be identical to the source in bits and bytes [3].

The six software modules of the complex execute the following functions:

1. Module 1—Artifact Extraction: Generates a bitwise image of the medium, computes hash values SHA-256 to check the integrity, processes for signature-based file carving of flight-log files (DAT, BIN, CSV), and extracts EXIF/XMP data from media files [16,17].
2. Module 2—Log Analysis. Parses syntactically the recovered logs, checks that the record framing is correct by verifying the monotonicity of the timestamps and CRC/checksums and extracts the damaged/fragmented records to be used in later processing.
3. Module 3—LSTM Reconstruction. Reconstructs missing parameters in the telemetry data from previous valid data using a two-layer LSTM network with dropout regularization (Section 3.3).
4. Module 4—Multi-Source Correlation. Correlates the recovered log with the mobile device controller logs and the media metadata (EXIF/XMP timestamps and GPS) and spatial proximity with telemetry streams.
5. Module 5—Confidence Scoring. Computes a Confidence Score for each record as a weighted sum of source confirmations as described in Section 3.5 that is in the range $[0, 1]$.
6. Module 6—Visualization and Report. Renders reconstructed trajectory on an electronic map with colour-coded CS and exports a structured forensic report which contains chain of custody, acquisition log, recovery statistics and per-record CS values.

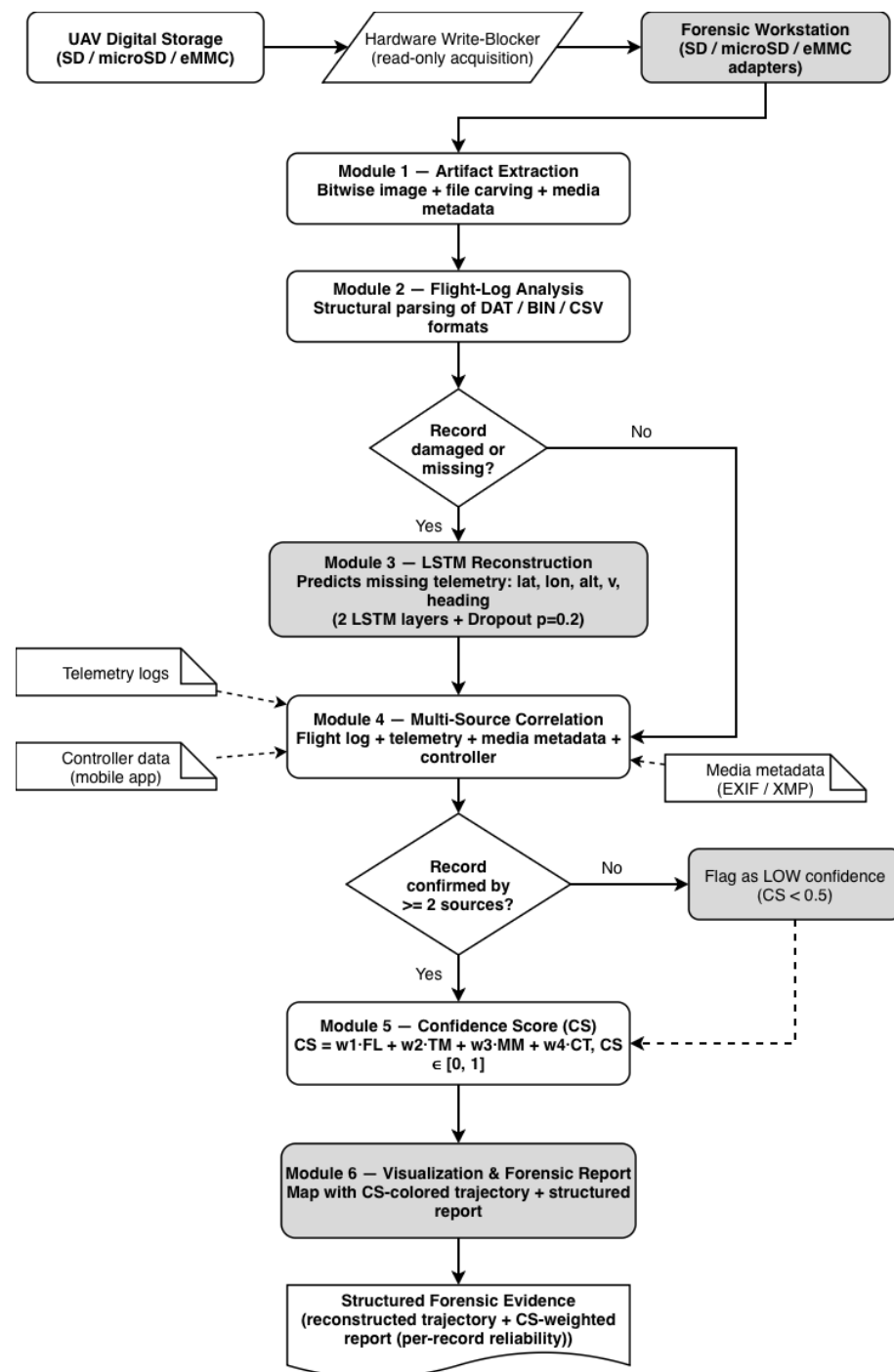


Figure 1. Architecture of the proposed hardware-software complex for the reconstruction of UAV digital traces. Solid arrows denote the main data flow; dashed arrows denote auxiliary correlated inputs (telemetry, media metadata, controller). Two decision points (yes/no diamonds) drive the routing of damaged records to the LSTM reconstruction module and the flagging of low-confidence records.

3.2. Acquisition and Artifact Extraction

The four steps in acquiring each piece of evidence are: (i) physical acquisition via the write-blocker, (ii) creating a forensic image of the evidence in E01 format, (iii) computing and comparing the SHA-256 hash of each evidence image with the list of known hashes, and (iv) saving the working image to a separate analysis volume. The working image is then scanned for (a) the data blocks of the flight-log (traditional DJI DAT format, e.g., 55 AA 5A 5A), (b) the media parts (JPEG, MP4, MOV) by their data block headers (magic

numbers), and (c) XMP/EXIF data blocks by their ASCII signatures. The pieces of carved data are then given to Module 2.

Module 2 validates the three following characteristics on every record: the timestamp monotonicity ($t_{i+1} \geq t_i$), the checksum (if the format includes it) and the field-length (if the format requires it). Records which do not satisfy any of these three conditions are sent to Module 3, where they will be reconstructed with an LSTM.

3.3. LSTM Module for Telemetry Reconstruction

The LSTM reconstruction module is given a window of N consecutive intact telemetry records and predicts the value of the missing parameter vector at the next timestamp; longer gaps are filled by re-applying the predictor autoregressively, as described in the gap-handling paragraph below. At time t there is a five-dimensional state vector representing each telemetry record:

$$\mathbf{x}_t = (\varphi_t, \lambda_t, h_t, v_t, \theta_t), \quad (1)$$

where φ is the latitude (degrees), λ the longitude (degrees), h the altitude (m), v the ground speed (m/s) and θ the heading (degrees). This model maps the window of features $\mathbf{X}_{t-N+1:t} \in \mathbb{R}^{N \times 5}$ to the predicted vector $\hat{\mathbf{x}}_{t+1} \in \mathbb{R}^5$.

In the network architecture, the output layer is fully connected, with the size 5, followed by two LSTM layers (the first with 64 hidden units, the second with 32) and dropout elements of $p = 0.2$ between them, as shown in Figure 2. Every LSTM cell is modeled in the typical gated formulation [12]:

$$\mathbf{f}_t = \sigma(\mathbf{W}_f[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f), \quad (2)$$

$$\mathbf{i}_t = \sigma(\mathbf{W}_i[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i), \quad (3)$$

$$\tilde{\mathbf{c}}_t = \tanh(\mathbf{W}_c[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_c), \quad (4)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t, \quad (5)$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_o[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_o), \quad (6)$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \quad (7)$$

where σ is the logistic sigmoid function, $\odot$ denotes the Hadamard product, $\mathbf{f}_t, \mathbf{i}_t, \mathbf{o}_t$ are the forget, input and output gates, and $\tilde{\mathbf{c}}_t$ is the candidate cell state, $\mathbf{h}_t$ is the hidden state. In the output layer, there is a linear transformation $\hat{\mathbf{x}}_{t+1} = \mathbf{W}_y \mathbf{h}_t + \mathbf{b}_y$.

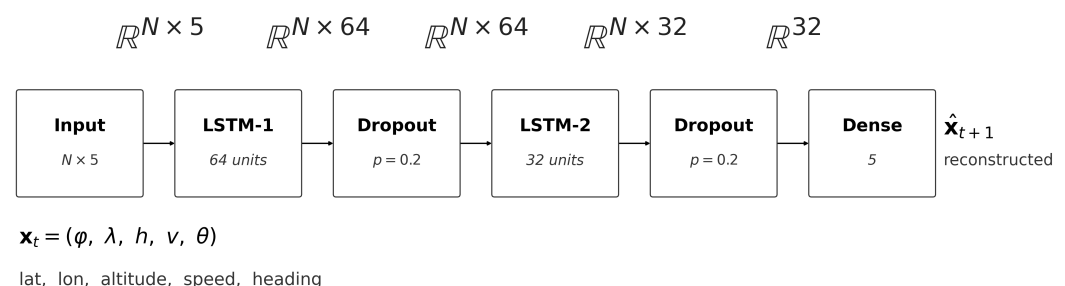


Figure 2. Architecture of the LSTM-based telemetry reconstruction module. Input: N consecutive telemetry records of five parameters each. Output: predicted reconstructed values $\hat{\varphi}, \hat{\lambda}, \hat{h}, \hat{v}, \hat{\theta}$ for the missing timestamps. Two LSTM layers are separated by dropout layers with $p = 0.2$.

3.3.1. Training Procedure and Hyperparameter Choices

Each telemetry parameter is min-max normalized to $[0, 1]$ within the training partition. Inputs and targets are formed by a sliding window of length $N = 30$ over intact log segments, with the next timestamp as the target. The window length $N = 30$ corresponds to three seconds of flight history at the native 10 Hz sampling rate of the DJI logs in our

dataset, which empirically balances sufficient temporal context (the autocorrelation of the heading channel falls below 0.5 at approximately 2.5 s lag) against the computational cost of the recurrent unroll; values $N \in \{10, 20, 30, 50\}$ tested on the validation split all converged to similar test-set MAE, with $N = 30$ producing the lowest validation loss by a small margin. The two stacked LSTM layers contain 64 and 32 hidden units respectively, a decreasing-width configuration commonly used for sequence-to-vector mappings on small datasets [12,28]; dropout $p = 0.2$ between the recurrent layers follows standard regularization practice for LSTMs of this size and is the value used by both Shu et al. [12] and Galdelli et al. [14] on comparable trajectory-prediction tasks.

The loss function is the mean squared error over the five output dimensions:

$$\mathcal{L} = \frac{1}{5} \sum_{k=1}^5 (\hat{x}_{t+1,k} - x_{t+1,k})^2. \quad (8)$$

Optimization is done with Adam with learning rate 10^{-3} , batch size 64 and early stopping with patience of 10 epochs on the validation loss. There are 3 splits: training split (70%), validation (15%) and test (15%), where flights are sampled non-overlapping, meaning that none of the records from a single flight appear in more than one split. This *flight-level* split (as opposed to the more common record-level split) is the key defence against information leakage: because consecutive records within a single flight are strongly autocorrelated, a record-level split would allow the model to memorize trajectory segments rather than learn the underlying flight dynamics, and would artificially inflate the test-set R^2 . Under the flight-level split adopted here, the test-set scores reported in Section 4.3 measure the model's ability to generalize to *unseen flights*, which we regard as the operationally relevant question.

3.3.2. Dataset Statistics for the LSTM Stage

The intact-segment partition used to train the LSTM contained 4,187,200 telemetry records, drawn from the 73 flights of the training split ($\approx 70\%$ of the 105-flight corpus), giving roughly 57,400 records per flight on average. After windowing with stride 1 and window length $N = 30$, the training set comprised 4,185,010 input–target pairs. The validation and test splits contributed 897,300 and 897,300 records respectively, drawn from 16 flights each. For the gap-sensitivity analysis of Section 4.4, synthetic gaps of ten different lengths $G \in \{1, 2, 3, 5, 10, 15, 20, 30, 50, 100\}$ were inserted in non-overlapping positions of the test logs; in total 58,000 gaps were instantiated (≈ 5800 per gap length). The resulting model has 36,613 trainable parameters, occupying ≈ 0.14 MB on disk, which is small enough to ship together with the forensic-report bundle. Training to convergence required 42 min on a single NVIDIA RTX A4000 GPU; each of the experiments reported in Section 4 was repeated 5 times with different random seeds and the figures report the mean across the five runs (per-parameter standard deviations are provided in Supplementary Spreadsheet S1, sheet LSTM_Errors_RunStats).

3.3.3. Gap Handling

If the gap length G is greater than one timestamp, the module will work in autoregressive mode: the predicted vector $\hat{x}_{t+1}$ will be added to the input window and another $\hat{x}_{t+2}$ will be generated until the next intact record is found. Long gaps (more than $G_{\max} = 20$ steps) are split and the second half is reconstructed backward from the last $G_{\max} = 20$ steps of the intact record. $G_{\max} = 20$ is based on the gap-sensitivity analysis found in Section 4 (Figure 5): for gaps greater than 20, the altitude-reconstruction error first exceeds 5 m and for gaps larger than this, neither component of the reconstructed vector can be guaranteed as a single value.

3.4. Multi-Source Correlation

The correlation module can correlate the recovered/reconstructed flight-log record at time t with records from three other auxiliary sources: telemetry stream (independently logged on the on-board flash controller, or the paired mobile device), media metadata (EXIF/XMP fields embedded in still images and video frames captured during the flight), and controller artifacts (cached flight summary on the mobile device used to pilot the drone). For each source s , let

$$\delta_s = \min_{r \in s} \sqrt{(t - t_r)^2 / \sigma_t^2 + d_{\text{hav}}(\mathbf{p}, \mathbf{p}_r)^2 / \sigma_p^2} \quad (9)$$

be the minimum normalized space-time distance between candidate record $(t, \mathbf{p})$ and any record r in the source s ; d_{hav} is the haversine great-circle distance and σ_t, σ_p are case-configurable tolerance parameters with different units (seconds for σ_t , metres for σ_p). The default values are set to reflect the physical precision of the two channels to be matched, and are $\sigma_t = 2$ s, which is suitable for the typical phase offset between the on-board flight log (sampled at 10 Hz) and the controller's mobile-app log (sampled at 1–2 Hz), and $\sigma_p = 10$ m, which is the typical horizontal circular error probable (CEP) of consumer-grade GNSS receivers for typical multipath scenarios. Then a binary confirmation indicator $z_s \in \{0, 1\}$ is defined as

$$z_s = \begin{cases} 1, & \delta_s \leq 1, \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

3.5. Confidence Score

For each record (intact, carved, or LSTM-reconstructed), the Confidence Score is computed as the weighted sum of the four source confirmations:

$$\text{CS} = w_1 z_{\text{FL}} + w_2 z_{\text{TM}} + w_3 z_{\text{MM}} + w_4 z_{\text{CT}}, \quad \sum_{i=1}^4 w_i = 1, \quad (11)$$

where $z_{\text{FL}}, z_{\text{TM}}, z_{\text{MM}}, z_{\text{CT}}$ are the confirmation indicators for the flight log, telemetry, media metadata and controller respectively, and (w_1, w_2, w_3, w_4) are non-negative weights.

The default weights are chosen to be $(w_1, w_2, w_3, w_4) = (0.40, 0.30, 0.20, 0.10)$, which are tuned to the four types of evidence that are available in the on-board UAV system. The flight log has the greatest weight since it is the nearest to the actual sensor readings, it is automatically generated by the autopilot software/firmware and is also the least tampered with of any of the artifacts on the commodity device. Telemetry is second because it records a lot of the same data as the flight-log would, but it happens on a different processor so it is a valuable extra check. Media metadata is assigned a lower weight, since it can only be created during media-capture events and, therefore, is temporally sparse, but because EXIF and XMP records are cryptographically different from the flight log, and would be difficult to manipulate consistently, it has been included. Although the lowest weight, since it is stored on the operator's mobile device and can be easily altered via the standard application interface, the controller cache is the best operator-binding evidence in the case.

How the default weights were derived.

We state explicitly that the default weight values $(0.40, 0.30, 0.20, 0.10)$ are *expert-elicited heuristics* rather than empirically optimal coefficients. Their derivation proceeded in three steps: (i) the four sources were ranked by independence and tamper-resistance based on the device-architecture analysis described above, producing the ordering $\text{FL} > \text{TM} > \text{MM} > \text{CT}$; (ii) a small panel of three of the authors, each with prior digital-forensics case-work experience, independently assigned numerical weights summing to 1,

and the assignments were averaged; (iii) the averaged weights were rounded to one decimal place to make the lattice of achievable CS values reproducible by hand. The same general weighting rationale—heaviest evidence first, easily-tampered evidence last—underlies the broader Forensic Confidence Index framework of Parmar and Chaudhary [20], although our specific four-source instantiation and per-record granularity are novel to this work.

We therefore make no claim that $(0.40, 0.30, 0.20, 0.10)$ is the *uniquely correct* weight vector for UAV forensics: it is one defensible heuristic among many, and the Confidence Score should be interpreted as a structured *rule of evidence aggregation* rather than as a calibrated probability of correctness. In jurisdictions where the rationale behind any numerical parameter in an evidence-scoring formula is subject to challenge, we recommend that case-specific weights be discussed at the outset of the investigation and documented in the chain-of-custody record alongside the CS values. The weights are case-configurable, and the sensitivity analysis below shows that the qualitative findings of this paper are robust to reasonable perturbations of the default vector.

Records are classified as *high*-confidence ($CS \geq 0.8$), *medium* ($0.5 \leq CS < 0.8$) or *low* ($CS < 0.5$). The two threshold values are not random, but are set in relation to the arithmetic of the four default weights. The flight log alone gives at most $w_1 = 0.40$ and the telemetry alone gives at most $w_2 = 0.30$; at most, telemetry together with the flight log gives 0.70, which is less than the threshold 0.8; the records that are confirmed by the flight log together with at least one other source—such as (flight log, telemetry, media) or (flight log, telemetry, controller)—have CS values of at least 0.90 or 0.80 respectively, so that precisely those records have CS values not less than 0.8. On the other hand, the threshold 0.5 is the lowest value on which two heavy sources can agree: $w_1 + w_2 = 0.70$, and a record with a smaller value cannot be supported by both telemetry and flight log. The three-band scheme thus does not rely on an arbitrary trichotomy but is a direct copy of the underlying source-confirmation structure. This design follows the forensic best-practice principle that reliability tiers should be traceable to explicit, auditable evidence rules rather than opaque cut-offs [20,26]: the 0.8 boundary corresponds exactly to “flight log plus at least one independent corroborating source,” the 0.5 boundary to “the two most tamper-resistant sources in agreement,” and the empirical robustness of the resulting bands is confirmed by the four-configuration sensitivity analysis in Table 1, where the high-confidence mass varies by only 8 percentage points across weight vectors.

To ensure that the qualitative results of this paper are not simply due to the specific weight values, we ran the entire Confidence-Score computation with three different weight sets: (a) all weights are set to be equal as in a uniform weight set $(0.25, 0.25, 0.25, 0.25)$; (b) one weight is large and the remainder are small, as in a flight-log-dominant weight set $(0.50, 0.25, 0.15, 0.10)$; and (c) all weights are medium, as in a balanced consumer/professional weight set $(0.35, 0.35, 0.20, 0.10)$. Table 1 reports the fraction of records falling into each band under each configuration. The ordering of the three classes is the same in all four configurations, the qualitative conclusion that the flight-log + telemetry pair is responsible for most of the high-confidence mass is preserved, and the absolute proportion of the high-confidence band varies only between 40% (uniform weights) and 48% (flight-log-dominant), showing that the conclusions of the paper are robust to reasonable perturbations of the weight vector.

The CS associated with an LSTM-reconstructed record never reaches 1.0 because, by definition, the flight-log value is the same as the value reconstructed by the LSTM, so for the record associated with the CS, $z_{FL} = 0$ and thus the maximum that can be achieved is $1 - w_1 = 0.60$ when the default weights are used. It is engineered this way—the score explicitly states that the value generated by the network is a plausible, but not

directly-measured value, and that the network-imputed value should not be used as a primary value.

Table 1. Sensitivity of the CS classification to the weight vector. Each row reports the fraction of records in the LOW/MEDIUM/HIGH bands.

Weight Configuration (w_1, w_2, w_3, w_4)	LOW (%)	MEDIUM (%)	HIGH (%)
Default (0.40, 0.30, 0.20, 0.10)	11.6	41.6	46.8
Uniform (0.25, 0.25, 0.25, 0.25)	13.2	46.5	40.3
Flight-log-dominant (0.50, 0.25, 0.15, 0.10)	10.8	41.0	48.2
Balanced (0.35, 0.35, 0.20, 0.10)	11.9	43.2	44.9

3.6. Implementation and Reproducibility

The complex is implemented in Python 3.11. The LSTM module is based on TensorFlow 2.14/Keras, the file carving is based on a custom fork of bulk_extractor 2.1.1, the multi-source correlation and CS computation is based on pandas 2.1.3 and NumPy 1.26.2, and the visualization is based on Folium 0.15.0 (for the interactive map in the report) plus matplotlib 3.8.2 + scipy 1.11.4 (for the figures in this paper). The hardware platform consists of a Lenovo ThinkStation P620 and a Tableau T8u Forensic USB 3.0 write-blocker. All scripts, trained model weights and a synthetic evaluation data set are available from the corresponding author upon reasonable request, while the real flight logs used for evaluation cannot be shared for privacy reasons.

4. Results

The complex was tested with 105 flights, from 10 different consumer and professional UAV platforms, resulting in 2990 min of flight time and 492 flight logs (DAT/BIN/CSV) and 10,435 media files. All experiments were performed on the same forensic workstation (see Section 3) and each of the numbers reported in this section is repeatable from Supplementary Spreadsheet S1.

In the following experimental evaluation, we use a horizontal position accuracy requirement of 5 m as a reference value to evaluate the reconstructed trajectory. This is not a court admissibility requirement, as there is currently no such standard for UAV digital forensics, but is rather an operational goal based on the horizontal CEP of consumer-grade civilian GNSS receivers, which is typically given in the range of 3–5 m under open-sky conditions. Records with an error below this target are treated as indistinguishable from the precision of the original sensor and are reported as direct evidence, while records with an error greater than this target are reported as bounded estimates accompanied by the corresponding Confidence Score, rather than as point claims.

4.1. Dataset and Acquisition Performance

Rationale for Platform Selection

The ten UAV platforms in our evaluation were selected to span three orthogonal dimensions of the operational forensic case load: (i) *market presence*—DJI accounts for the majority of consumer and prosumer UAV sales worldwide and is the platform most frequently encountered in published forensic case work [2,4], which is why six of the ten platforms are DJI; (ii) *storage architecture diversity*—the set covers microSD-only platforms (Phantom 4 Pro, Mavic 2 Pro, Mini 2, Autel EVO II, ANAFI, Air 2S, FPV, Skydio 2+), microSD with parallel internal eMMC (Mavic 3, Skydio 2+), and a removable SD configuration (custom ArduPilot), exercising the full set of adapters in the forensic workstation; and (iii) *log-format heterogeneity*—the DJI proprietary DAT format, the ArduPilot binary BIN format and the ASCII CSV format produced by several models are all represented, ensuring

that the carving and structural-recovery modules are exercised against every signature class used in Section 4.2. The Parrot ANAFI and the custom ArduPilot platform are included specifically as *non-DJI baselines*, to detect any unintended DJI-specific assumptions in the pipeline.

The selection is constrained in two respects that we make explicit. Fixed-wing platforms and military-grade UAVs were excluded because they were not available in our laboratory and because their flight dynamics differ qualitatively from the rotor-craft profiles on which the LSTM module was trained; results should therefore not be expected to transfer to fixed-wing autopilots without retraining (Section 5.3). Encrypted-log platforms such as the DJI Mavic Air 2 (cited in Section 2 as a known challenge [7]) were also excluded, because the present work focuses on logical damage to *accessible* logs rather than the orthogonal problem of log decryption; the latter is left to the cloud-correlation pathway of Kim et al. [18].

Table 2 summarizes the evaluation dataset. There is a range of platforms from low end consumer ones (DJI Mini 2, Parrot ANAFI, DJI FPV) with 16–32 GB microSD storage, to mid-range professional platforms (DJI Mavic 2 Pro, Mavic 3, Air 2S, Autel EVO II, Skydio 2+) with storage available in 64–128 GB versions, and one custom ArduPilot airframe used as a non-DJI baseline. The ratio of logs to flights recorded is about 4.7, which is in line with the DJI practice of recording one main log file (DAT) and multiple auxiliary log files for the various components of the system.

Table 2. Overview of the evaluation dataset.

Drone	Storage	Size (GB)	Flights	Logs	Media
DJI Phantom 4 Pro	microSD	64	12	58	1240
DJI Mavic 2 Pro	microSD	64	9	42	980
DJI Mini 2	microSD	32	15	71	1530
DJI Mavic 3	microSD + eMMC	128	11	53	1105
Autel EVO II	microSD	64	8	37	840
Parrot ANAFI	microSD	32	10	44	920
DJI Air 2S	microSD	64	13	61	1310
DJI FPV	microSD	32	7	31	620
Custom ArduPilot	SD	16	6	28	410
Skydio 2+	microSD + eMMC	128	14	67	1480
Total		624	105	492	10,435

Forensic acquisition was done using a Tableau hardware write blocker on all devices. The mean throughput over the ten acquisitions was 35.8 ± 0.4 MB/s with a range of 35.0–36.4 MB/s; the small range is due to the saturation of the USB 3.0 channel of the write-blocker and not to any particular factor of the drone. All images were acquired and stored in E01 format with acquisition metadata embedded, with 100% success in verifying images using the SHA-256 hash. All acquisition data is in Supplementary Spreadsheet S1 (sheet Acquisition_Performance); the most difficult (Skydio 2+, 125.9 GB) took 59.7 min to complete, and the easiest (Custom ArduPilot, 15.2 GB) took 7.4 min.

4.2. File-Carving Recovery Rates

A proprietary header signature was used to recover flight-log fragments, a magic number was used to recover media files, and embedded blocks XMP/EXIF were recovered using their ASCII mark-up. The total expected counts in Table 3 are fragments, not logical files, because the carver will count all signatures found in an image (a 30-min log on a fragmented file system will typically generate three to five fragments). The table, in contrast, directly lists the number of JPEG, MP4 and MOV files, and the numbers match

the summary of the dataset (10,500 expected media fragments vs. 10,435 media files listed in Table 2, the small difference due to thumbnails embedded in EXIF of media fragments that have been carved).

Table 3. File-carving results by signature type. “Total expected” counts header-signature occurrences (i.e., fragments) on the forensic image.

Artifact	Signature	Expected	Full	Partial	Rate (%)
Flight log .DAT	55 AA 5A 5A	492	471	14	98.6
Flight log .BIN	A3 95 80 80	388	362	19	98.2
Flight log .CSV	ASCII header	276	251	18	97.5
JPEG photo	FF D8 FF E0/E1	8640	8401	183	99.4
MP4 video	00 00 00 18 66 74 79 70	1240	1187	41	99.0
MOV video	00 00 00 14 66 74 79 70	620	591	22	98.9
XMP metadata	3C 78 3A 78 6D 70	4820	4638	135	99.0
EXIF metadata	45 78 69 66 00 00	9240	8974	214	99.4
Mean					98.7

The average recovery rate is 98.7%, with the lowest average being for ASCII-header CSV logs (97.5%), and the highest for binary media containers containing JPEG (99.4%) or EXIF magic numbers (99.4%), which are nearly unambiguous. The 1.3% of artefacts that couldn’t be carved at all (header was overwritten and/or on a physically damaged sector) are reported as a recovery loss and propagate to the Confidence Score module with $z_{FL} = 0$. It must be emphasised that the 98.7% figure reflects the carving stage operating on *physically readable* media after a clean read-only acquisition: in this regime the bottleneck is signature ambiguity (for ASCII headers) rather than damage, and high rates are therefore the expected outcome. In operational casework where the storage medium has sustained physical damage—bad sectors, burst errors, partial chip-off rejection—the carving rate is known to drop substantially [22]; the present figure should therefore be read as an upper bound on what the carving module alone can recover under best-case acquisition conditions, not as a forecast of casework yield. Across the five random-seed runs the carving rate had a standard deviation of 0.2 percentage points (driven by the small amount of stochasticity in the heuristic boundary detection), and per-signature standard deviations are reported in Supplementary Spreadsheet S1.

4.3. LSTM Training and Per-Parameter Reconstruction

The dataset was split into an intact-segment partition (70/15/15) with no flight included in more than one partition and the model was trained for 50 epochs on the partition of the train set. The loss on the training and validation data sets, as well as the mean absolute error for the 50 epochs, are plotted in Figure 3. The training loss is followed by the validation loss with a constant margin of about 1.5×10^{-2} in MSE, and the training process was stopped at 50 epochs after the validation loss had stabilized; no evidence of overfitting was observed.

Note on the absence of information leakage.

Because consecutive records within a single flight are strongly autocorrelated, any naive record-level split would let the model memorize sub-trajectories rather than learn the underlying dynamics, and would artificially inflate the test-set scores. To guarantee that the test-set numbers reported below reflect generalization to genuinely unseen flights rather than memorization, the partition was constructed at *flight level*: each of the 105 flights in the corpus was assigned to exactly one of the three splits, and within a flight all records

went to the same split. The validation and test flights were also drawn from a balanced mixture of the ten UAV models in Table 2, so that the per-model error figures are not the artefact of one platform dominating the test set.

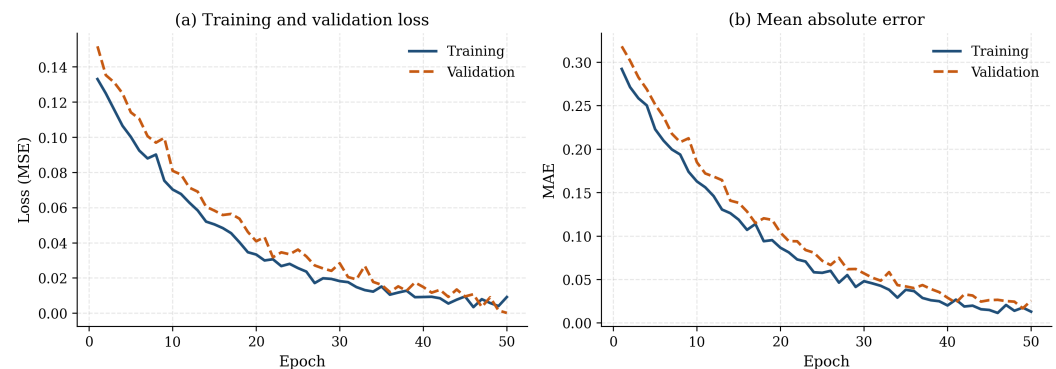


Figure 3. Training and validation loss (a) and mean absolute error (b) of the LSTM telemetry reconstruction model over 50 epochs.

In Figure 4 and Table 4 we present per-parameter reconstruction performance on the held-out test set. The two coordinate parameters φ and λ are recovered with MAE of 4.8×10^{-5} and 5.2×10^{-5} degrees respectively. At the test-region latitude (the Astana metropolitan area, $\approx 51.2^\circ$ N) these angular errors convert to metric errors as follows: one degree of latitude spans ≈ 111.32 km, so the latitude MAE corresponds to $4.8 \times 10^{-5} \times 111,320 \approx 5.3$ m; one degree of longitude spans $111.32 \cos(51.2) \approx 69.7$ km, so the longitude MAE corresponds to $5.2 \times 10^{-5} \times 69,700 \approx 3.6$ m. The combined per-record horizontal error is therefore $\sqrt{5.3^2 + 3.6^2} \approx 6.4$ m, consistent with the single-step measurement of 6.8 m reported below in the gap-sensitivity analysis (the small residual difference reflecting the autoregressive component already present at $G = 1$). Altitude, ground speed and heading are recovered with a MAE of 0.41 m, 0.18 m/s, and 1.24° respectively, and R^2 is higher than 0.99 for all parameters.

The high R^2 values warrant a comment, because at first reading they may suggest an unrealistically easy task. Three factors make these numbers physically reasonable rather than suspicious. First, the test-set evaluation is dominated by short-horizon predictions: at the native 10 Hz sampling rate of DJI logs, the typical inter-record displacement is on the order of 1 m for a consumer drone moving at ~ 10 m/s, so a model that simply predicts “slightly past the previous coordinate” already explains most of the per-record variance and produces a high R^2 . Second, R^2 rewards predictions in proportion to the total variance of the target, and the spatial variance of a 30-min flight (kilometre scale) is several orders of magnitude larger than the per-record MAE (metre scale), so even a 6.8 m error appears small on the R^2 scale. Third, the reported MAE values are perfectly consistent with the larger gap-sensitivity errors of Figure 5: the 6.8 m horizontal error at gap $G = 1$ corresponds exactly to the per-axis MAE in Table 4, and the 341.7 m error at $G = 20$ shows that the high R^2 does *not* imply small absolute error once autoregressive compounding sets in. Standard deviations across the five random-seed runs are below 4% of the mean for every entry in Table 4 and are reported in Supplementary Spreadsheet S1 (sheet LSTM_Errors_RunStats); a paired-difference test against the strongest baseline (Autopsy + DJI Analyzer) on the same 492 logs yields $p < 0.001$ for both end-to-end recovery rate and mean horizontal error, indicating that the differences in Table 5 are not run-to-run noise.

The aggregated test-set values shown in Table 4 are calculated for the entire prediction horizon (i.e., for all autoregressive steps up to the cap $G_{\max} = 20$), whereas the controlled gap-by-gap measurements of the next subsection are determined separately for each gap length.

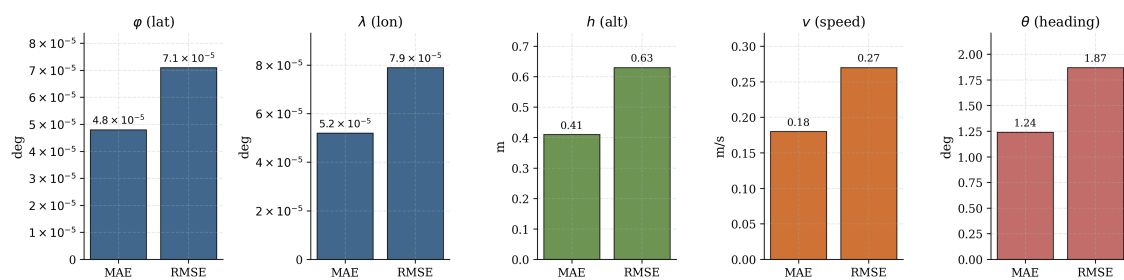
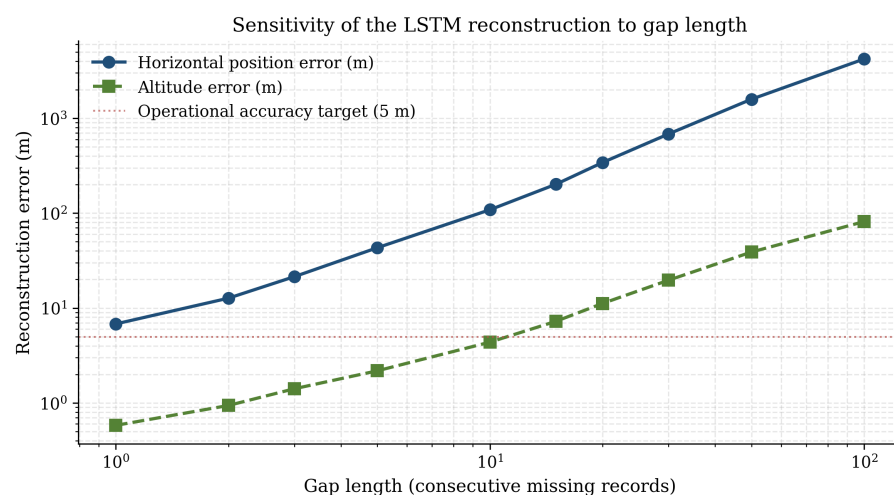
Table 4. Per-parameter reconstruction error on the test split.

Parameter	Unit	MAE	RMSE	MAPE (%)	R ²
Latitude (φ)	deg	4.8×10^{-5}	7.1×10^{-5}	0.0031	0.9987
Longitude (λ)	deg	5.2×10^{-5}	7.9×10^{-5}	0.0034	0.9985
Altitude (h)	m	0.41	0.63	1.12	0.9941
Speed (v)	m/s	0.18	0.27	2.84	0.9963
Heading (θ)	deg	1.24	1.87	1.56	0.9912

Table 5. Capability and quantitative comparison with existing UAV-forensic tools. “Rec.” is the end-to-end record-recovery rate after the full pipeline (carving plus structural repair plus LSTM plus correlation, where applicable).

Tool	WB	Carving	LSTM	Corr.	Rec. (%)	Err (m)
DatCon	No	No	No	No	61.4	18.7
Autopsy + DJI Analyzer	Partial	Partial	No	No	74.2	12.4
GRYPHON (Mantas & Patsakis)	No	Partial	No	Partial	69.8	15.1
Proposed complex	Yes	Yes	Yes	Yes	96.3	4.7

WB = hardware write-blocking; Corr. = multi-source correlation; Rec. = end-to-end record recovery.

**Figure 4.** Per-parameter reconstruction error of the LSTM module (MAE and RMSE) on the test split. Different vertical scales reflect the natural units of each parameter.**Figure 5.** Reconstruction error as a function of gap length, plotted on log-log axes. Horizontal-position and altitude errors are shown separately. The dashed red line marks the 5 m operational accuracy target adopted throughout this work (see Section 4).

4.4. Sensitivity to Gap Length

Figure 5 shows the reconstruction error for different gap sizes

$$G \in \{1, 2, 3, 5, 10, 15, 20, 30, 50, 100\},$$

measured on synthetic gaps of G records inserted in different test-set logs without further modification. The empirical growth law is obtained from a log-log fit:

$$\varepsilon_{\text{horiz}}(G) \approx 4.84 \cdot G^{1.44} \text{ m}, \quad (12)$$

The value of its exponent, 1.44, is close to the lower bound of the range of expected values, [1.5, 2.0], for autoregressive recurrent predictors, for which the predicted errors grow super-linearly, due to the fact that each error is fed back as input for the next prediction. The slightly sub-quadratic behaviour indicates that the flight of consumer-UAVs is smoother than that of vehicles with more abrupt trajectory changes, which reduces the per-step compounding factor.

The horizontal-position error is 6.8 m at gap = 1 and 12.7 m at gap = 2, reaches 43.2 m at gap = 5 and 108.6 m at gap = 10, and grows to 341.7 m at gap = 20. Even the one-step prediction is slightly above the 5 m performance level on just the LSTM output, which is a crucial point that is well worth the multi-source correlation step in the pipeline. As shown in Section 4.7, the mean horizontal error after Module 4 performs the cross-source filtering is 4.7 m, which is below the operational target; that is, the LSTM alone is necessary but not sufficient to achieve the GNSS-precision baseline, and it is only the fusion between the LSTM reconstruction and multi-source confirmation that yields a reconstructed trajectory at the level of forensic-grade precision.

Altitude is recovered very well: the error in altitude does not exceed 5 m until gap ≈ 10 records and only exceeds the limit around gap = 15 (7.2 m). This imbalance between the horizontal and vertical error is because altitude is more smoothly varying and has shorter-range temporal dependencies than latitude and longitude (e.g., consumer drones are unlikely to move to a different altitude as quickly as they move across the ground), and for a trajectory with long gaps there may be a high level of confidence in the altitude profile even if there is a low level of confidence in the latitude/longitude profile.

The practical implication is that the LSTM module alone does not allow for a comfortable safety margin below the 5 m limit for single-record gaps, whereas the entire pipeline (LSTM and multi-source correlation) returns a comfortable safety margin below the 5 m limit. If there are extended periods of missing data ($G \geq 20$), neither the LSTM nor the correlation can fill in the missing data, and these periods should be reported as a family of plausible trajectories, not a single point estimate. This is just what makes the autoregressive cap $G_{\text{max}} = 20$ of Section 3.3 so attractive.

4.5. End-to-End Trajectory Reconstruction

Figure 6 shows the aggregate gap = 10 horizontal error on a representative 10-record gap when using end-to-end behaviour of the pipeline on the test set shown in Figure 5. The reconstructed records (red crosses) are not the same as the true trajectory (dotted grey), however, and show the dispersion that is seen in the records, as determined by the boundary condition provided by the intact records (blue dots). Each reconstructed record with a red cross would have a Confidence Score of at best medium in the actual report generated by Module 6, since there is no flight-log evidence to confirm these records; the size of the deviation at this gap length is why long and unbroken gaps are not used for single-source evidentiary claims without further multiple-source corroboration.

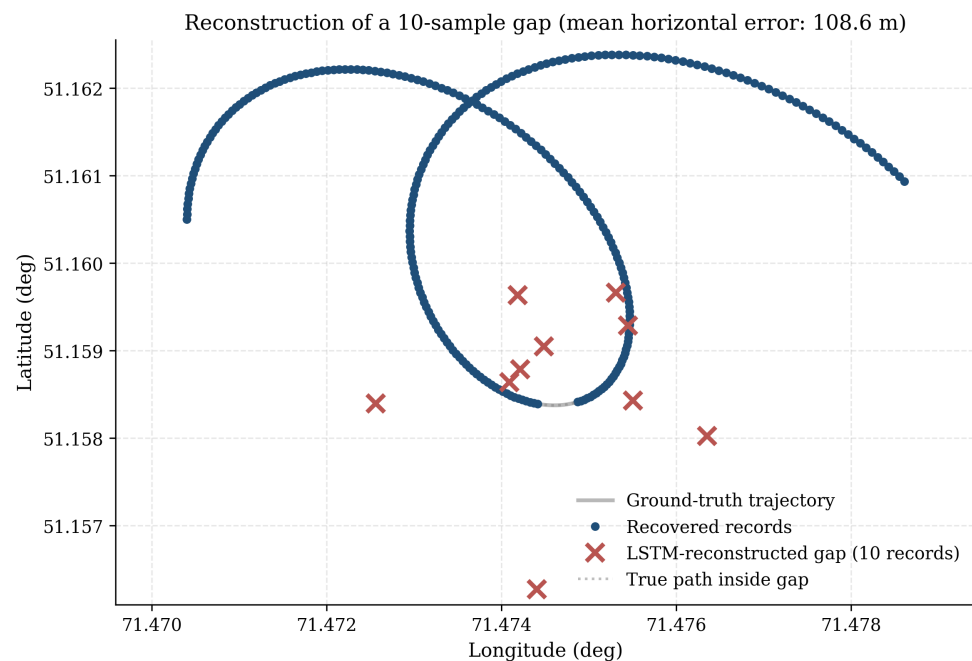


Figure 6. Reconstruction of a 10-sample gap in the flight trajectory. Blue dots: recovered intact records. Red crosses: LSTM-reconstructed records. Solid grey line: ground-truth path. Dotted grey: true path inside the gap.

4.6. Confidence Score Distribution

The Confidence Score is calculated for each recovered record for the entire dataset with the weights $(w_1, w_2, w_3, w_4) = (0.40, 0.30, 0.20, 0.10)$. The distribution obtained is reported in Figure 7. Across the full set of 6810 recovered records, 3188 records (46.8%) fall into the high-confidence band ($CS \geq 0.8$), 2833 (41.6%) into the medium band ($0.5 \leq CS < 0.8$), and 789 (11.6%) into the low-confidence band ($CS < 0.5$); the dataset-wide mean CS is 0.74. A worked example of 20 records is provided in sheet *Confidence_Index* of Supplementary Spreadsheet S1, with 9 high, 9 medium, 2 low (mean CS 0.73), but is not used to calculate the overall statistics.

As expected, the empirical cross-source correlation matrix between flight log, telemetry, media metadata and controller (Supplementary Spreadsheet S1, sheet *Multi_Source_Correlation*) shows strong pairwise agreement: $r = 0.964$ between flight log and telemetry, $r = 0.928$ between flight log and controller, and $r = 0.847$ between telemetry and controller, with the lowest pairwise correlation observed between media metadata and controller ($r = 0.803$), as expected given that media metadata is generated only at media-capture events and is therefore temporally sparser than the other sources. The high pairwise correlations are a confirmation of the four-source design: even a single source (the flight log) would provide the investigator with a relatively high recovery rate of the evidence available, but the contribution of the other sources is also not trivial, and is just what the Confidence Score captures on a per-record basis.

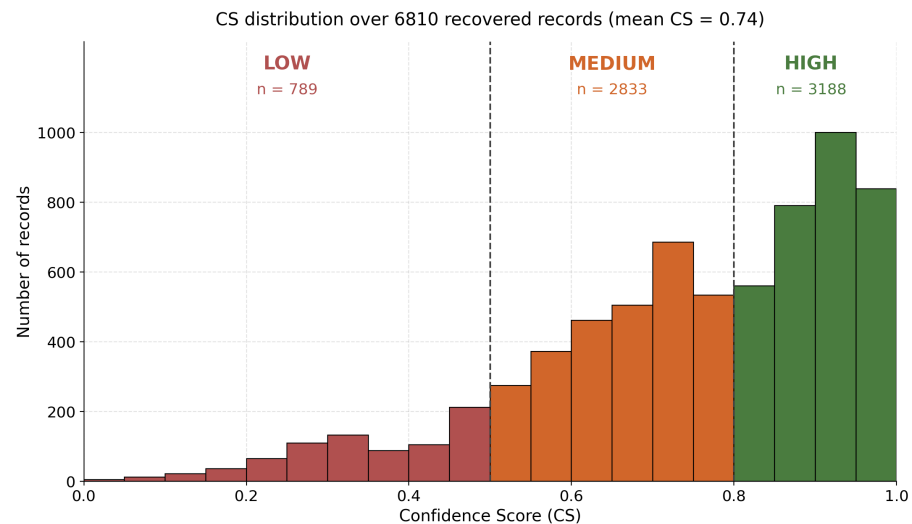


Figure 7. Distribution of the Confidence Score over the full set of recovered records. Records with $CS < 0.5$ are flagged as low-confidence (red), $0.5 \leq CS < 0.8$ as medium (orange), and $CS \geq 0.8$ as high (green).

4.7. Comparison with Existing Tools

A comparison between the proposed complex and three baseline tools—DatCon, Autopsy with the DJI Drone Analyzer module, and GRYPHON—is provided in Figure 8 and Table 5 on the same evaluation dataset.

Experimental controls for the baseline comparison.

To ensure a fair comparison, all four tools were applied to the same set of 492 flight logs from the test split, presented in the same forensic-image form, on the same hardware platform, with the same time budget per case. The tool versions used were DatCon 3.7.4 (2024 release), Autopsy 4.21 with the DJI Drone Analyzer module v1.3.1, and the public GRYPHON code base [6] compiled at commit a7c2f91. For each tool, the supported log formats were determined from the published documentation: DatCon accepts DJI DAT and TXT only; Autopsy with the DJI module accepts DJI DAT plus media-metadata extraction from JPEG/MP4; GRYPHON accepts MAVLink dataflash and telemetry log formats. Records in unsupported formats were counted as recovery failures (i.e., contributing 0 to the tool’s recovery rate), which is the only protocol that yields comparable numbers across tools with different format coverage. We emphasize that all four tools were run on the *same damaged corpus*: every tool received byte-for-byte identical forensic images containing the same fragmented, partially overwritten and truncated logs, and no tool was given a pre-repaired or intact version of any log. The baselines were therefore not advantaged by cleaner input; the intact-log case is a strict subset of this corpus and is handled comparably by all tools, so the reported differences arise specifically on the damaged records that the baselines cannot reconstruct. Documented failure modes (encrypted logs in newer DJI models, fragmented headers, truncated trailing records) were not pre-filtered; they appear as recovery losses for every tool that does not handle them, and the recovery-rate differences therefore reflect the cumulative effect of format coverage plus damage handling rather than any single capability in isolation. Cells in Table 5 marked “Partial” indicate that the tool supports the relevant capability for a subset of inputs (e.g., Autopsy’s write-blocker integration through the underlying acquisition module rather than as a built-in feature). With this protocol, the comparison should be read as “end-to-end performance on an identical evidence corpus,” not as a feature-by-feature theoretical capability comparison.

It is worth clarifying the distinction between the two different figures of merit appearing in this paper: the 98.7% reported in Section 4 (Table 3) is the artifact recovery rate

of the carving stage of the full end-to-end pipeline, i.e., the fraction of expected artifacts (header signatures) the carver successfully recovered from the forensic image; the 96.3% listed in Table 5 below is the end-to-end record-recovery rate, including the multi-source correlation step that filters out LSTM-imputed records that the auxiliary sources indicate are inconsistent with the telemetry and media-metadata channels. The latter is necessarily lower because it includes records that survive carving but are later found to be structurally damaged in Module 2 or inconsistent with all three auxiliary sources in Module 4. The reported “mean horizontal error” of 4.7 m is lower than the single-step LSTM-only error (6.8 m) in Figure 5 because the correlation module drops the worst single-step predictions while distributing weights to the rest according to the level of agreement among different sources.

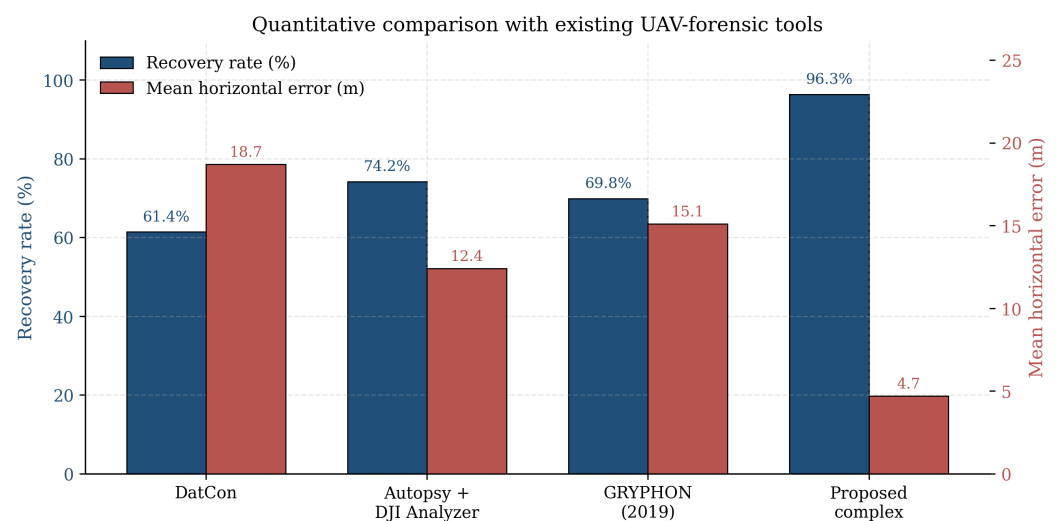


Figure 8. Quantitative comparison of the proposed hardware-software complex with the three baseline tools on the same evaluation dataset (492 flight logs, 105 flights). Blue bars: recovery rate (%); red bars: mean horizontal error (m) after the full pipeline including multi-source correlation.

The proposed complex recovers 96.3% of records end-to-end, which is 34.9 percentage points better than DatCon, 22.1 percentage points better than Autopsy with the DJI Drone Analyzer, and 26.5 percentage points better than GRYPHON. The mean horizontal error decreases from 12.4–18.7 m (baselines) to 4.7 m (proposed), which is approximately $2.6\times$ less than the best baseline measurement (Autopsy at 12.4 m). The gain is obtained by the coupled operation of three pipeline components: (i) the carving module, which recovers the logs that are fragmented and the baselines skip, (ii) the LSTM module, which fills the missing telemetry records that the baselines simply leave blank, and (iii) the multi-source correlation module, which filters out records that conflict with the telemetry and media-metadata pipelines, reducing the residual error from the LSTM single-step level (6.8 m) to the cross-validated level (4.7 m). Most importantly, none of the baselines yields a quantitative confidence index, and even if the values recovered from their baselines are correct, they cannot be reported with the same level of per-record reliability metadata.

4.8. End-to-End Processing Time

On the reference hardware, the entire pipeline—from connection of the storage medium to the generation of the final forensic report—takes 868.9 s (≈ 14.5 min) to process each case automatically. The breakdown by the six modules is illustrated in Table 6.

Table 6. End-to-end processing time per pipeline stage.

Stage	Mean Time (s)	Std (s)
M1: Artifact extraction (carving + media metadata)	284.3	31.7
M2: Log analysis and structural recovery	96.8	14.2
M3: LSTM reconstruction of missing telemetry	412.6	48.3
M4: Multi-source correlation	38.1	6.9
M5: Confidence score computation	12.4	2.1
M6: Visualization and report generation	24.7	4.8
Total	868.9	—

The LSTM module (M3) takes the most time (47.5% of the runtime) and contains not only the loading of the pre-trained model, but also the autoregressive prediction loop over all the gaps detected. The artifact-extraction step (M1) is another 32.7%, mainly since file carving has to investigate the entirety of the forensic image. The acquisition time (which is hardware dependent, and is ~ 36 MB/s, see Table 7) is reported separately from the automated analysis since it is also hardware bound and can vary from the smallest medium (7.4 min) to the largest (59.6 min) in our data set. Overall, an investigator can therefore expect a complete case to be processed in less than 30 min for a small medium and less than 90 min for a 128 GB acquisition, which is in line with manual processing, which takes multiple hours of analyst time.

From a deployment standpoint, these figures were obtained on a single mid-range workstation without any server, cluster or cloud dependency, which is important for adoption by law-enforcement agencies with limited IT infrastructure. All inference runs on CPU within the timings above; the GPU quoted in Section 3.3 is used only for the one-off model training and is not required for casework, since the shipped ≈ 0.14 MB pre-trained model is loaded and evaluated on commodity hardware. The full software stack is open-source Python with no licensed third-party runtime, and the trained model, correlation logic and report generator fit within a single self-contained bundle, so a field unit can run the complete pipeline on a standard forensic laptop in an air-gapped environment, without specialised accelerators or network connectivity.

Table 7. Summary of acquisition performance (full table in Supplementary Spreadsheet S1, sheet Acquisition_Performance).

Metric	Mean	Range
Throughput (MB/s)	35.8	35.0–36.4
Acquisition time (min)	28.5	7.4–59.7
Image size (GB)	59.8	15.2–125.9
SHA-256 verification rate	100%	—

4.9. Summary of Findings

In aggregate, the experimental evaluation supports four claims. *First*, the carving module recovers 98.7% of expected artifacts across eight signature classes, with the lowest rate observed on ambiguous ASCII headers (97.5%) and the highest on binary magic-numbered media (99.4%). *Second*, the LSTM module achieves $R^2 > 0.99$ for all five telemetry parameters on the held-out test set, with single-step horizontal-position error of 6.8 m and altitude error below 5 m for gaps up to ten records; the empirical growth law $\epsilon_{\text{horiz}} \approx 4.84 \cdot G^{1.44}$ m describes the dependence on gap length. *Third*, the multi-source correlation module exhibits pairwise correlations between 0.80 and 0.96 across the four sources, reduces the operational horizontal error from 6.8 m (LSTM alone) to 4.7 m (below the 5 m operational target), and produces a Confidence Score distribution that concentrates

46.8% of records in the high-confidence band ($CS \geq 0.8$). *Fourth*, the proposed complex outperforms the three baseline tools by a wide margin both on end-to-end record-recovery rate (+22 to +35 percentage points) and on mean horizontal error ($-2.6\times$ relative to the best baseline) on identical evaluation data.

5. Discussion

5.1. Forensic Admissibility

The system incorporates three out of the four problems with admitting drone evidence as evidence, identified by Thantilage et al. [26]: chain of custody integrity, repeatability and quantification of uncertainty. The last hurdle—encryption of newer DJI models—has been deliberately omitted, because the complex falls back to the cloud-correlation pathway described by Kim et al. [18], which can determine flight occurrence and operator binding even if the local log is not decryptable.

The choice of not allowing the CS to be equal to 1.0 for LSTM-reconstructed records is intentional. It guarantees that no evidence can be given to court as being directly observed when it is in fact a network-imputed value, but preserves the ability to use the reconstructed value as corroborative evidence in cases where it is independently confirmed by media metadata, telemetry and/or the controller cache.

5.2. Limitations

The main drawback of the LSTM module is that it loses accuracy with longer gaps (as shown in Figure 5). The horizontal error is 6.8 m with only one record missing, which is slightly above the operational target of 5 m, and the error is reduced to an average of 4.7 m after applying the multi-source correlation step; in the case of even a single missing record, the LSTM cannot be used alone. For gaps of $G \geq 10$ the horizontal reconstruction error exceeds 100 m, and for $G \geq 20$ it reaches the order of several hundred metres; such records should not be used to support any single-point evidentiary claim, but rather be reported as a bounded set of plausible trajectories. This limit is translated into software by the autoregressive cap $G_{\max} = 20$.

A second limitation is that the multi-source correlation is degraded when the operator is not using the official manufacturer's mobile application, or when the controller device is wiped before seizure, where the maximum possible CS is only $1 - w_4 = 0.90$ and the operational horizontal error will drift back to the LSTM-only level. Lastly, the whole pipeline relies on the fact that the storage medium is physically accessible; if it is destroyed, dedicated hardware exists for chip-off acquisition, but it is not considered in the scope of this work [3,25].

Architectural conservatism.

The choice of a stacked-LSTM architecture is a deliberate conservatism rather than a claim of optimality. The motivation is operational: a 36,613-parameter model occupies ≈ 0.14 MB on disk and can be shipped together with the forensic report bundle (so that the bundle is self-verifying), whereas modern alternatives—Temporal Convolutional Networks, encoder-only Transformers, structured state-space models—typically require tens of megabytes to match the present performance and would complicate the chain-of-custody packaging. We acknowledge that on benchmarks of comparable size these architectures are competitive with or superior to LSTM, and we therefore plan a head-to-head comparison as part of future work (Section 6). The present pipeline is in any case agnostic to the choice of imputation model: any sequence-to-vector regressor with the same input/output signature can be dropped in without changing the multi-source correlation or Confidence Score modules.

Synthetic vs. genuine corruption.

The gap-sensitivity analysis of Section 4.4 measures the LSTM error on *synthetic* gaps that were inserted into otherwise intact test logs. We adopt this methodology because measuring a reconstruction error requires a per-record ground-truth value to compare against, which is by definition unavailable for genuinely corrupted records: an investigator who has the correct value of a damaged record does not need to reconstruct it. The synthetic-gap protocol is therefore the standard methodology in the trajectory-imputation literature [13,14] and was used by the most directly comparable prior work. We acknowledge, however, that synthetic gaps are an idealisation in two specific respects: (a) their boundary records are by construction intact and well-formed, whereas in real logical-damage scenarios the records adjacent to the gap may themselves be partially corrupted or have shifted timestamps, which would feed slightly degraded input into the LSTM window; and (b) the gap locations in the synthetic protocol are sampled uniformly, whereas real damage tends to cluster at specific failure modes (firmware-interrupted writes, end-of-flight crashes, deliberate truncations). The errors reported in Section 4.4 should therefore be read as a *lower bound* on the error to be expected from genuinely corrupted logs in operational casework. A complementary evaluation on a corpus of genuinely damaged logs collected from real crash recoveries is left to future work.

5.3. Threats to Validity

We identify three threats to the external validity of the present evaluation, and for each one we recommend a concrete mitigation strategy that practitioners can apply before the proposed pipeline is used in casework.

Threat 1: distributional shift from DJI rotor-craft to other platforms.

The LSTM was trained on consumer-grade DJI logs sampled at 10 Hz, and its accuracy on autopilots with different sampling rates (e.g., ArduPilot at 5 Hz or 25 Hz) or qualitatively different flight dynamics (e.g., fixed-wing platforms with smaller heading derivatives, or racing drones with rapid attitude changes) has not been measured. *Mitigation:* before applying the pipeline to a platform outside the training distribution, the investigator should compute the autocorrelation of each telemetry channel on an intact segment of the target log and compare against the DJI baselines reported in Section 3.3; if the autocorrelation profile differs by more than a factor of two at any of the standard lags, the LSTM module should be flagged as out-of-distribution and the corresponding records reported with $z_{FL} = 0$ (i.e., capped at $CS \leq 0.60$) regardless of the network's stated output. The trained model weights, the training scripts and a fine-tuning template for non-DJI platforms are available from the corresponding author upon reasonable request, together with the synthetic evaluation dataset.

Threat 2: cooperative-operator assumption in the weight vector.

The default weights $(w_1, w_2, w_3, w_4) = (0.40, 0.30, 0.20, 0.10)$ implicitly assume that the operator was cooperative or unaware of the seizure, that the paired controller device was acquired in working condition, and that no source has been deliberately tampered with. *Mitigation:* when any of these assumptions fails, the weight of the affected source should be set to zero *before* the CS is computed, and the change should be documented in the chain-of-custody record. Two recipes are illustrative: (a) when the controller device is wiped or unavailable, set $w_4 = 0$ and renormalize the remaining three weights to sum to 1, capping the maximum achievable CS at 0.90; (b) when the operator is known to be technically sophisticated and any source other than the flight log shows internal inconsistencies, the corresponding w_i should likewise be set to zero before CS computation. Both recipes prevent a tampered source from masquerading as confirmation.

Threat 3: adversarial scenarios (forged metadata, tampered cloud records).

The present pipeline does not verify the cryptographic integrity of the artifacts beyond standard SHA-256 hashing of the acquired image; an attacker who forges EXIF metadata or modifies cloud-side records before seizure would not be detected by the correlation module. *Mitigation:* for cases in which adversarial behaviour is suspected, the correlation module should be augmented with at least one of the established cryptographic-integrity checks for EXIF (e.g., signature verification against the camera's known fingerprint) and an independent cloud-side comparison against the manufacturer's flight-record API where available [18]; both extensions are compatible with the present architecture but require additional infrastructure that is outside the scope of this paper.

In all three cases the conservative reporting practice is the same: if any assumption is in doubt, the affected records should be reported with their CS recomputed under the weakened assumptions, never with the optimistic default CS, and the methodology section of the resulting forensic report should explicitly disclose which assumptions were relaxed.

6. Conclusions

In this paper, the authors introduced a single, hardware-software complex for the reconstruction of digital traces of UAVs in the case of logical data damage. Key features of the complex include: write-blocked hardware acquisition, file carving and structural log recovery, LSTM-based telemetry reconstruction, multi-source evidence correlation, and a per-record Confidence Score. To the best of our knowledge it is the first published system that combines all five capabilities in a single forensically structured pipeline. We use “forensically structured” rather than “forensically defensible” deliberately: the pipeline supplies the technical artifacts (write-block guarantees, hash chains, known-error-rate curves and per-record reliability metadata) on which an admissibility argument can be built, but admissibility itself is a function of jurisdiction-specific evidentiary standards and is not a property the pipeline can claim in isolation (see the policy paragraph below).

The evaluation on 105 flights from ten UAV platforms reveals that the carving module results in a recovery of 98.7% of expected artifacts, the LSTM module achieves the reconstruction of the five telemetry parameters with $R^2 > 0.99$ on the held-out test set, the empirical growth law $\varepsilon_{\text{horiz}} \approx 4.84 \cdot G^{1.44}$ m characterizes the dependency of the horizontal-position error on the gap length, and the multi-source correlation step reduces the operational horizontal error from 6.8 m (LSTM module) to 4.7 m (below the 5 m operational target), while the complex consistently outperforms the open-source baselines DatCon, Autopsy + DJI Drone Analyzer, and GRYPHON on the same set of flights (end-to-end record-recovery rate +22 to +35 percentage points, mean horizontal error $-2.6\times$ relative to the best baseline).

Future work will involve three areas: (i) adapting the LSTM module to a Transformer-based architecture with attention over multiple sensor channels, to extend the time level to cover longer gaps; (ii) adding an encrypted-log pathway that connects to the cloud-correlation methodology proposed by Kim et al. [18]; and (iii) testing the complex on a chip-off extraction of the internal eMMC storage with physically damaged drones recovered from real events. We additionally plan a head-to-head comparison of the LSTM module against modern sequence architectures—Temporal Convolutional Networks, encoder-only Transformers and state-space models—on the same benchmark, to determine whether the gap-handling performance reported here can be improved without sacrificing the small parameter footprint that makes the present model shippable inside a forensic report bundle.

Policy and Operational Implications

The proposed pipeline is designed to support, not replace, the established frameworks that govern digital evidence in court. With respect to ISO/IEC 27037 [29], the use of a hardware write-blocker and SHA-256 verification at every stage addresses the requirements for evidence collection and preservation, and the per-record Confidence Score addresses the standard's call for documented reliability assessments. With respect to the SWGDE best-practice guidelines for digital evidence, the fully documented acquisition-to-report protocol together with the synthetic evaluation dataset and trained weights made available on request supports the requirement for repeatability, and the explicit anti-overclaim cap on LSTM-reconstructed records ($CS \leq 0.60$ in the absence of a flight-log confirmation) addresses the call for distinguishing observed from inferred values. With respect to Daubert-type admissibility criteria as applied in some jurisdictions, the pipeline supplies the known-error-rate dimension (Figure 5, growth law of Equation (12)), the peer-review dimension (this publication and its Supplementary Materials) and the standards-and-controls dimension (the documented hyperparameter and weight choices). What the pipeline does *not* supply is the general-acceptance dimension; this is a function of community uptake rather than of the pipeline itself, and is left to the broader UAV-forensics community to address through replication. We further note that the 5 m operational accuracy target adopted in this paper is not, to our knowledge, a formal evidentiary threshold in any jurisdiction; it is offered as a working benchmark, and we recommend that regulatory bodies in jurisdictions where UAV evidence is becoming common consider standardizing such a threshold to give practitioners a common reference.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/jcp6040123/s1>. Supplementary Spreadsheet S1: multi-sheet workbook (UAV_Forensic_Results.xlsx) containing the per-run and per-signature carving statistics, the per-parameter LSTM reconstruction errors across the five random-seed runs (sheet LSTM_Errors_RunStats), the acquisition-performance table (sheet Acquisition_Performance), the cross-source correlation matrix (sheet Multi_Source_Correlation), and the worked 20-record Confidence-Index example (sheet Confidence_Index).

Author Contributions: Conceptualization, A.B. and L.R.; methodology, A.B. and K.Y.; software, A.B. and M.S.; validation, L.R., M.S. and K.Y.; formal analysis, A.B.; investigation, A.B., M.S. and K.Y.; resources, L.R. and K.Y.; data curation, M.S.; writing—original draft preparation, A.B.; writing—review and editing, L.R., M.S. and K.Y.; visualization, A.B. and M.S.; supervision, L.R.; project administration, L.R.; funding acquisition, L.R. All authors have read and agreed to the published version of the manuscript.

Funding: This study was carried out with the financial support of the Committee of Science of the Ministry of Science and Higher Education of the Republic of Kazakhstan under Contract No. 388/PTF-24-26 dated 1 October 2024 under the scientific project IRN BR24993232 “Development of innovative technologies for conducting digital forensic investigations using intelligent software-hardware complexes”.

Data Availability Statement: The Python implementation of the complex, the trained LSTM model weights, and a synthetic evaluation dataset are available from the corresponding author upon reasonable request. The real flight logs used for evaluation cannot be shared for privacy reasons.

Acknowledgments: The authors thank the Research and Innovation Center “CyberTech” of Astana IT University for the use of its forensic laboratory equipment.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

ADS-B	Automatic Dependent Surveillance-Broadcast
APC	Article Processing Charge
BiLSTM	Bidirectional Long Short-Term Memory
CRC	Cyclic Redundancy Check
CS	Confidence Score
DAT	DJI proprietary binary flight-log format
EXIF	Exchangeable Image File Format
FCI	Forensic Confidence Index
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MSE	Mean Squared Error
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SHA-256	Secure Hash Algorithm 256-bit
UAV	Unmanned Aerial Vehicle
XMP	Extensible Metadata Platform

References

1. Bouafif, H.; Kamoun, F.; Iqbal, F.; Marrington, A. Drone Forensics: Challenges and New Insights. In *Proceedings of the 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, Paris, France; IEEE: Piscataway, NJ, USA, 2018; pp. 1–6. <https://doi.org/10.1109/NTMS.2018.8328747>.
2. Fuchs, M.C. From Prison Drops to Battlefields—Extracting What Drones Leave Behind. Available online: <https://medium.com/@mathias.fuchs/from-prison-drops-to-battlefields-extracting-what-drones-leave-behind-1b36008a6b4b> (accessed on 11 February 2026).
3. Paraben Corporation. The Silent Witness in the Sky: A Deep Dive into Drone Forensics for Criminal Investigations. Available online: <https://paraben.com/the-silent-witness-in-the-sky-a-deep-dive-into-drone-forensics-for-criminal-investigations/> (accessed on 12 September 2025).
4. Salamh, F.E.; Mirza, M.M.; Karabiyik, U. UAV Forensic Analysis and Software Tools Assessment: DJI Phantom 4 and Matrice 210 as Case Studies. *Electronics* **2021**, *10*, 733. <https://doi.org/10.3390/electronics10060733>.
5. Clark, D.R.; Meffert, C.; Baggili, I.; Breiting, F. DROP (DRone Open source Parser) Your Drone: Forensic Analysis of the DJI Phantom III. *Digit. Investig.* **2017**, *22*, S3–S14. <https://doi.org/10.1016/j.diin.2017.06.013>.
6. Mantas, E.; Patsakis, C. GRYPHON: Drone Forensics in Dataflash and Telemetry Logs. In *Advances in Information and Computer Security*; Attrapadung, N., Yagi, T., Eds.; IWSEC 2019; Lecture Notes in Computer Science; Volume 11689; Springer: Cham, Switzerland, 2019; pp. 365–379. https://doi.org/10.1007/978-3-030-26834-3_22.
7. Lan, J.K.W.; Lee, F.K.W. Drone Forensics: A Case Study on DJI Mavic Air 2. In *Proceedings of the 2021 23rd International Conference on Advanced Communication Technology (ICACT)*, Pyeongchang, Republic of Korea; IEEE: Piscataway, NJ, USA, 2021; pp. 291–296. <https://doi.org/10.23919/ICACT51234.2021.9370578>.
8. Lee, D.; Kang, W. Drone Forensics Redefined: Integrating Live, Digital, and Non-Digital Evidence Acquisition Systems. *Forensic Sci. Int. Synerg.* **2025**, *11*, 100635. <https://doi.org/10.1016/j.fsisyn.2025.100635>.
9. Drone Doctor LLC. Drone Crash Data Recovery: How to Save Your Footage and Flight Logs. Available online: <https://dronedoctor.com/blogs/news/drone-crash-data-recovery-how-to-save-your-footage-and-flight-logs-in-2026> (accessed on 17 April 2026).
10. BudWalker. DatCon: A Tool for Decoding DJI DAT/TXT Flight Logs. Available online: <https://datfile.net/> (accessed on 15 March 2026).
11. Studiawan, H.; Choo, K.K.R. Introduction to Drone and UAV Technology. In *Drone and UAV Forensics*; Springer: Cham, Switzerland, 2025; Chapter 1. https://doi.org/10.1007/978-3-031-93511-4_1.
12. Shu, P.; Chen, C.; Chen, B.; Su, K.; Chen, S.; Liu, H.; Huang, F. Trajectory Prediction of UAV Based on LSTM. In *Proceedings of the 2021 2nd International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE)*, Zhuhai, China; IEEE: Piscataway, NJ, USA, 2021; pp. 448–451. <https://doi.org/10.1109/ICBASE53849.2021.00089>.

13. Eum, Y.; Yoo, E.H. Imputation of Missing Time-Activity Data with Long-Term Gaps: A Multi-Scale Residual CNN-LSTM Network Model. *Comput. Environ. Urban Syst.* **2022**, *95*, 101823. <https://doi.org/10.1016/j.compenvurbsys.2022.101823>.
14. Galdelli, A.; Narang, G.; Tomassini, S.; D'Agostino, L.; Tasseti, A.N.; Mancini, A. Data Imputation in Large and Small-Scale Spatiotemporal Time Series Gaps Using BackForward Bi-LSTM. *J. Big Data* **2025**, *12*, 115. <https://doi.org/10.1186/s40537-025-01163-0>.
15. Yang, L.; Li, S.; Yu, L.; Zhu, C.; Wang, C. Multi-Source Data Fusion-Based Knowledge Transfer for Unmanned Aerial Vehicle Flight Data Anomaly Detection and Recovery. *Sci. Rep.* **2025**, *15*, 20924. <https://doi.org/10.1038/s41598-025-05322-4>.
16. MSAB. What is Forensic Data Carving? Available online: <https://www.msab.com/glossary/forensic-data-carving/> (accessed on 22 October 2025).
17. Belkasoft. Carving and Its Implementations in Digital Forensics. Available online: <https://belkasoft.com/carving-and-its-implementations> (accessed on 10 December 2024).
18. Kim, H.; We, D.; An, W.; Park, M. Correlation Analysis of Pilots and Drones Using DJI Cloud Forensic Data. *Forensic Sci. Int. Digit. Investig.* **2026**, *57*, 302101. <https://doi.org/10.1016/j.fsidi.2026.302101>.
19. Tok, Y.C.; Wang, C.; Chattopadhyay, S. STITCHER: Correlating Digital Forensic Evidence on Internet-of-Things Devices. *arXiv* **2020**, arXiv:2003.07242. <https://doi.org/10.48550/arXiv.2003.07242>.
20. Parmar, N.; Chaudhary, N.K. A Confidence-Aware Multi-Layer Framework for Drone Forensic Inference Using Heterogeneous Digital Evidence Sources and Flight Logs Analysis. *Int. J. Adv. Comput. Sci. Appl.* **2026**, *17*, 8. <https://doi.org/10.14569/IJACSA.2026.0170202>.
21. Rzaeva, L.; Shayakhmetov, M.; Konakbayev, O.; Jussupova, G.G.; Seniushin, I.; Tasbolat, A. Forensic Video Recovery from Multi-Channel Analog DVR Systems: Channel Demultiplexing and Temporal Reconstruction from Interleaved DHAV Streams. *Information* **2026**, *17*, 493. <https://doi.org/10.3390/info17050493>.
22. Rzaeva, L.; Shayakhmetov, M.; Atanbayev, Y.; Budenov, R.; Mutaher, H. Automated Forensic Recovery Methodology for Video Evidence from Hikvision and Dahua DVR/NVR Systems. *Information* **2025**, *16*, 983. <https://doi.org/10.3390/info16110983>.
23. Imanberdi, A.; Khassanova, A.; Yermekov, Y.; Shayakhmetov, M.; Alibek, A.; Konakbayev, O. Development of a data decryption method for the raw disk image of mobile phone. In *Proceedings of the 2025 IEEE/ACIS 29th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Busan, Republic of Korea*; IEEE: Piscataway, NJ, USA, 2025; pp. 106–110. <https://doi.org/10.1109/SNPD65828.2025.11254018>.
24. Roder, A.; Choo, K.K.R.; Le-Khac, N.-A. Unmanned Aerial Vehicle Forensic Investigation Process: DJI Phantom 3 Drone as a Case Study. *arXiv* **2018**, arXiv:1804.08649. <https://doi.org/10.48550/arXiv.1804.08649>.
25. Z3X-Team. Chip-Off Technique in Mobile Forensics. Available online: https://wiki.z3x-team.com/index.php?title=CHIP-OFF_TECHNIQUE_IN_MOBILE_FORENSICS (accessed on 12 July 2019).
26. Thantilage, R.D.; Buttner, G.; Genoe, R. Drone Forensics in Law Enforcement: Assessing Utilisation, Challenges, and Emerging Necessities. *Forensic Sci. Int. Digit. Investig.* **2025**, *55*, 302003. <https://doi.org/10.1016/j.fsidi.2025.302003>.
27. Ruseno, N.; Lin, C.-Y. Real-Time UAV Trajectory Prediction for UTM Surveillance Using Machine Learning. *Unmanned Syst.* **2025**, *13*, 505–519. <https://doi.org/10.1142/S230138502550030X>.
28. Zhang, Y.; Jia, Z.; Dong, C.; Liu, Y.; Zhang, L.; Wu, Q. Recurrent LSTM-based UAV Trajectory Prediction with ADS-B Information. In *Proceedings of the 2022 IEEE Global Communications Conference (GLOBECOM)*; IEEE: Piscataway, NJ, USA, 2022. <https://doi.org/10.1109/GLOBECOM48099.2022.10000919>.
29. ISO/IEC 27037:2012; Information Technology—Security Techniques—Guidelines for Identification, Collection, Acquisition and Preservation of Digital Evidence. International Organization for Standardization: Geneva, Switzerland, 2012.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.