



Apron Management System

AI Turnaround Intelligence

Overview

All figures are planning estimates, not quotes. Actual costs depend on airport size, existing infrastructure, data policies, and vendor selection.

1. Executive Summary

The Apron Management System (AMS) – AI Turnaround Intelligence is an AI-powered video analytics platform that monitors aircraft turnarounds in real time and automatically generates detailed operational reports.

It uses a local detector and an LLM report engine to analyse apron videos, detect safety and compliance events, identify process bottlenecks, and produce a structured turnaround report — all without manual observation.

The platform is built on an **event-centric, tiered data logging foundation** designed for long-term AI investigation, trend analysis, and model retraining.

Proven pipeline example:

- Total analysis time: 644.94s
- Observation window: ~6 min 40s
- Frames scanned: 8,175
- Key frames sent to LLM: 6
- LLM call time: 245.16s (ChatGPT subscription ~2700s)



- Detector: AI, local mode (<http://localhost:9001/detect>)
- Outputs: turnaround report, structured event log (CSV/Parquet/SQL), run ID tracking

Deployment timeline: Pilot in 1–3 months. Full rollout in 3–12 months. Accuracy improves from 80–90% at day one to 95–99% within 8 weeks.

2. Opportunity

Segment	Opportunity
Real-time apron management market	\$1.2B (2024) → \$2.8B (2034), ~9% CAGR
Consulting demand	Growing need for feasibility, ConOps, vendor selection, implementation support
Key buyers	Airports, ground handlers, airlines, regulators, investors

Market drivers:

- Rising passenger volumes and stand congestion
 - Safety and regulatory compliance pressure
 - Labour shortages and need for operational efficiency
 - Digitalisation of airport ground operations
 - Demand for data-driven turnaround performance
-

3. The Problem

Manual apron monitoring is:

- **Subjective** — different observers, different standards
- **Reactive** — issues found after delay or incident
- **Incomplete** — safety and compliance gaps missed
- **Slow** — no real-time bottleneck alerts
- **Expensive** — manual audit and reporting costs

Example from a real analysis report:

- Fuel truck bottleneck: present ~2 minutes continuously, total fueling >3 minutes in a 6.6-minute window
 - Jet blast detected 3 times — critical safety event
 - Chocks never detected — compliance gap
 - FOD never detected — safety blind spot
 - Safety cones detected at start, then detection gap
-

4. The Solution

AI Turnaround Intelligence turns existing or new apron cameras into a real-time monitoring and reporting system.

What it detects

- Aircraft presence
- Ground crew
- Baggage carts
- Passenger boarding stairs



- Fuel truck
- Belt loaders
- Jet blast
- Safety cones
- Boarding bridge
- FOD (Foreign Object Debris)
- Safety vests
- Chocks
- Airport individual requirements

What it delivers

- Current turnaround activity — live stage and status
 - Aircraft Turnaround Monitor — e.g., Stage 4/4: Done in 644.94s
 - Timeline analysis — activity windows and durations
 - Safety analysis — jet blast, ground crew proximity, cones
 - Compliance analysis — stairs, bridge, vest, chocks, FOD
 - Process optimisation — bottlenecks, inefficiencies, delays
 - **Long-term AI investigation** — event-centric data logging, trend analysis, model re-training
-



5. Product Features

Feature	Description
Real-time activity monitor	Tracks turnaround stage and current activity
Automated timeline	Detects start/end of each activity window
Safety event detection	Jet blast, crew proximity, FOD, cone gaps
Compliance audit	Stairs, bridge, vest, chocks, FOD
Bottleneck detection	Fuel truck, baggage, crew, equipment delays
LLM narrative report	Human-readable turnaround analysis
Event-centric logging	Aggregates frame detections into tracks/events; CSV/Parquet/SQL
Tiered storage	Hot, warm, cold retention for cost-efficient long-term investigation
Model/config versioning	Logs model version, config hash, prompt hash, run ID for reproducibility
Embeddings & semantic search	Enables AI-searchable knowledge base over frames, events, and reports
Configurable pipeline	<code>config.json</code> , run IDs, detector URL
Local or cloud AI	Flexible deployment for data privacy



6. Technical Architecture

Text of process description:

Apron Cameras



Video Ingestion



Local AI Detector (<http://localhost:9001/detect>)



Frame Scanning (e.g., 8,175 frames total)



Key Frame Selection (e.g., 6 frames)



LLM Report Engine (e.g., 245.16s call time)



Turnaround Report



Event-Centric Data Logging Layer (Parquet + SQL + object storage)



Tiered Storage: Hot / Warm / Cold + Vector DB



AI Investigation, Dashboards, Retraining

Pipeline status example:

- Run ID: c850e99d-043f-4f9e-8a66-71699b441ee4
- Total time: 644.94s
- Source: video
- Scan mode: full
- Detector backend: MASI-AI
- Detector mode: local
- Frames scanned: 8,175
- Key frames sent to LLM: 6
- LLM call time: 245.16s



- Config: /home/ai-user/ApronManagement/config.json
- CSV log: /home/ai-user/ApronManagement/aircraftTurnaround/detections.csv

Aircraft Turnaround Monitor

Stage 4/4: Done in 644.94s

Based on the provided detection report and key frames, here is the analysis of the turnaround activity.

TIMELINE

- [t+0.37s → t+399.62s]: Total observation duration is approximately 6 minutes and 40 seconds.
- [t+0.37s → t+93.78s]: Initial aircraft presence detected. Ground crew and baggage carts are active immediately, indicating the aircraft is already on-stand and servicing has commenced.
- [t+10.43s → t+166.94s]: Passenger boarding stairs are deployed and visible for the majority of the first half of the clip, suggesting passenger flow is active during this window.
- [t+158.61s → t+270.41s]: A fuel truck is detected in a continuous, long-duration presence (approx. 112 seconds), indicating the fuelling operation is the central activity of the mid-turnaround phase.
- [t+286.97s → t+304.60s]: A second distinct window of activity involving the fuel truck and belt loaders, suggesting a final top-up or equipment repositioning.
- [t+329.11s → t+383.83s]: Final fuel truck presence detected before departure preparations.
- [t+399.62s]: The video feed cuts to an interior terminal view, ending the external ramp observation.



SAFETY

- [t+163.66s → t+177.56s]: Ground crew detected in high-visibility vests (conf 0.93) operating in the immediate vicinity of the aircraft fuselage and engines.
- [t+284.15s → t+285.94s]: **Jet blast** detected. This is a critical safety event indicating engine spool-up or testing while ground personnel may still be in the vicinity.
- [t+361.79s → t+364.33s]: **Jet blast** detected again.
- [t+366.22s → t+367.88s]: **Jet blast** detected again.
- [t+0.37s → t+8.95s]: Safety cones detected at the start of the clip.
- [t+26.23s]: A single scan detects a cone, but for the majority of the clip, the system fails to detect cones, despite them being visible in the key frames (e.g., Image #2, #3, #4). This suggests a detection gap rather than a physical absence.
- [window t+0.37s .. t+399.62s]: **FOD (Foreign Object Debris)** was never detected by the system.

COMPLIANCE

- [t+89.79s → t+92.80s]: **Boarding bridge** detected. The system identifies the bridge, but it is not consistently tracked, likely due to the camera angle or occlusion.
- [t+13.93s → t+45.57s]: **Stairs** are deployed. In a compliant turnaround, stairs and bridges are rarely used simultaneously unless for specific crew access or emergency egress.
- [window t+0.37s .. t+399.62s]: **Safety vest** class was never detected as a standalone object, though "ground crew" detections (e.g., at [t+ 163.66s]) show high confidence scores (0.93) implying high-vis clothing is present.
- [window t+0.37s .. t+399.62s]: **Chocks** were never detected. This is a compliance gap; the system failed to identify the wheel chocks clearly visible in Image #1 and #2.

PROCESS OPTIMIZATION

- **[t+158.61s → t+270.41s]: Bottleneck Detected.** The fuel truck is present for nearly 2 minutes. This is a significant portion of the total 6.6-minute observation window.
- **[t+14.28s → t+81.75s]:** Baggage cart presence is continuous for 67 seconds.
- **[t+182.28s → t+230.34s]:** Ground crew presence is continuous for 48 seconds, coinciding with the fueling operation.
- **[t+286.97s → t+304.60s]: Inefficiency Detected.** A gap in fuel truck activity occurs between [t+270.41s] and [t+286.97s] (16.5 seconds), followed by a return of the truck. This suggests a pause in the process or a repositioning delay.
- **[t+329.11s → t+383.83s]:** The fuel truck is present for another 54 seconds. The total fueling time (sum of major intervals) exceeds 3 minutes, which is high for a 6-minute observation window, suggesting the aircraft is either large (wide-body, as seen in images) or the operation is slow.

Pipeline Status

Run ID: c850e99d-043f-4f9e-8a66-71699b441ee4

Total time: 644.94s

Source: video

Scan mode: full

Detector backend: AI

Detector mode: local

Detector URL: http://localhost:9001/detect

Frames scanned: 8175

Key frames sent to LLM: 6

LLM call time: 245.16s

Video source: <https://www.youtube.com/watch?v=PcjDL2d-xLo&t=2157>

Config: /home/ai-user/ApronManagement/config.json

CSV log: /home/ai-user/ApronManagement/aircraftTurnaround/detections.csv

6.1 Data Logging & AI Investigation Optimization

Optimizing long-term data logging for AI investigation means moving from a simple CSV append model to a **tiered, event-centric, versioned data platform**. The goal is to keep the data complete enough for root-cause analysis, trend detection, and model retraining — while controlling storage cost and query speed.

Your current `detections.csv` is a good start, but it is not enough for multi-year AI investigation. You need structured events, metadata, model versions, and tiered storage.

6.1.1 Start with the investigation use cases

Log only what you will actually investigate. Typical apron AI investigations:

Investigation	Data needed
Why did turnaround take longer?	Event timeline, fuel truck, baggage, crew, stairs
Was there a safety incident?	Jet blast, FOD, crew proximity, cones
Are we compliant?	Chocks, vests, bridge, stairs, FOD
Predict delays	Historical KPIs, weather, aircraft type, stand
Retrain models	Key frames, labels, model version, confidence

If you log everything at full fidelity forever, cost explodes. If you log too little, AI investigation becomes guesswork.



6.1.2 Use a layered data model

Separate raw, derived, and semantic data.

Layer	Examples	Retention	Storage
Raw video	Full camera feed	7–30 days	Object storage
Frame detections	Bounding boxes, classes, confidence	3–12 months	Parquet
Tracks/events	Fuel truck present, jet blast	3–7 years	DB / lakehouse
Key frames	6 frames sent to LLM	1–3 years	Object storage
LLM reports	Turnaround analysis	7+ years	DB / lakehouse
KPIs/metrics	Turnaround time, delays	Permanent	Time-series DB
Model/config versions	LLM, config hash	Permanent	Metadata DB

This gives you fast queries on events, cheap storage for raw frames, and reproducibility for AI.

6.1.8 Add embeddings and semantic search

For long-term investigation, store:

- **Frame embeddings** — search visually similar events.
- **Report embeddings** — search “fuel truck delay” across years.
- **Event embeddings** — cluster recurring bottlenecks.

This turns your log into an AI-searchable knowledge base.

6.1.9 Data quality and governance

- **Schema registry** — enforce data contracts.
- **Validation** — reject missing `turnaround_id`, bad timestamps.
- **Deduplication** — avoid double-counting events.
- **Time sync** — use NTP/PTP, UTC, monotonic timestamps.
- **Privacy** — anonymize faces/plates, GDPR retention policies.
- **Access control** — role-based access to raw vs. aggregated data.
- **Monitoring** — alert on missing logs, high latency, low confidence.

6.1.10 Cost optimization techniques

Technique	Saving
Parquet + Zstd	5–10x storage reduction
Event aggregation	10–100x less data
Tiered retention	50–80% cost reduction
Key frames only	90%+ video cost reduction
Partition pruning	10–100x faster queries
Pre-aggregated KPIs	Avoid full scans
Shared GPU for inference	90% AI cost reduction



6.1.11 Implementation roadmap

Phase	Time	Action
1	0–1 month	Define schema, IDs, retention policy
2	1–3 months	Move CSV → Parquet + DB, add event aggregation
3	3–6 months	Add lakehouse (Iceberg/Delta), tiered storage
4	6–9 months	Add embeddings, vector DB, feature store
5	9–12 months	Add dashboards, labeling, retraining loop

This gives you a data foundation that is cheap to store, fast to query, and rich enough for AI investigation over years — not just a folder of CSV files.

7. Value Proposition

Benefit	Impact
Safety	Real-time jet blast, FOD, crew proximity alerts
Compliance	Automated checks for chocks, cones, vests, stairs, bridge
Efficiency	Identifies bottlenecks like fuel truck delays
Cost reduction	Less manual monitoring, fewer delays
Scalability	One system can monitor multiple stands



Benefit	Impact
Auditability	Every report and detection is logged
Data-driven decisions	Turnaround trends, stand performance, KPIs
Long-term AI investigation	Event-centric, tiered, versioned data platform

8. Target Customers

- **Airports** — single terminal to multi-terminal hubs
 - **Ground handlers** — turnaround service providers
 - **Airlines** — station performance and safety
 - **Regulators** — compliance and safety oversight
 - **Consulting firms** — AMS feasibility and implementation
 - **Investors** — airport technology and infrastructure funds
-

9. Deployment & Timeline

Phase	Duration	Notes
Proof of Concept / Pilot	1–3 months	1–2 stands, validate accuracy
Core software implementation	~6 weeks	For some off-the-shelf solutions
First release rollout	12–18 weeks	Milestone capture + reporting



Phase	Duration	Notes
Fast airport deployment	~139 days	Small Airport
Full platform rollout	6–14 months	Advanced integrations, billing, telematics
Large-scale hub	12–18+ months	Complex Airport
Accuracy optimisation	8 weeks	80–90% → 95–99%

Key timeline factors: number of gates, existing cameras, AODB integration, cybersecurity, data residency, vendor model.