# SYSTEMS STUDIO

# Jetson AI Telescope Node

Professional edge-AI architecture for dual-NIR observation, OpenCV object detection, image enhancement, telescope tracking, and Signals Platform integration

| | |
|---|---|
| **Recommended edge computer** | NVIDIA Jetson Orin NX 16 GB |
| **Primary real-time input** | Wide-field monochrome NIR camera with 200-300 mm lens |
| **Secondary imaging input** | Cooled monochrome NIR camera at telescope focus |
| **Core edge functions** | Detection, tracking, enhancement, buffering, mount correction |
| **Central platform role** | Sessions, evidence integrity, correlation, reporting, archive |

## 1. System Purpose

The Jetson node is the low-latency visual processor mounted on the telescope assembly. It should not be treated as a simple mini-computer attached to the optical tube. Its purpose is to close the real-time loop between the NIR cameras, the target tracker, the telescope mount, and the Signals Platform evidence system.

The edge node performs immediate computer-vision work locally, where internet latency cannot interfere with target acquisition or telescope motion. The VPS and Dell operator workstation remain responsible for session authority, long-term storage, evidence review, correlation, reporting, and remote supervision.

### Recommended dual-NIR architecture

```
Wide-field NIR camera + 200-300 mm lens
    |
    v
Jetson AI Node
    |- OpenCV / TensorRT object detection
    |- Target tracking and motion prediction
    |- Telescope mount corrections
    |- Enhanced operator preview
    |- Evidence metadata and event buffering
    v
Signals Platform on VPS / Dell operator station

Main telescope cooled NIR camera -> detailed imaging and confirmation
```

## 2. Recommended Jetson Platform

For the professional telescope node, the preferred platform is the NVIDIA Jetson Orin NX 16 GB. It offers a better balance of performance, memory, power consumption, physical size, and future expansion than the entry-level Orin Nano, while avoiding the weight and cooling burden of an AGX Orin.

| Platform | Recommended use | Assessment |
|---|---|---|
| **Jetson Orin NX 16 GB** | Dual-camera edge node, TensorRT inference, enhancement, buffering | Recommended final telescope node |
| **Jetson Orin Nano 8 GB** | Early development, one primary stream, lighter models | Good prototype or secondary node |
| **Jetson AGX Orin** | Several cameras, heavy segmentation, local large-model analysis | Powerful but heavy and thermally demanding |

# 3. Real-Time Object Detection and Tracking

- The wide-field global-shutter NIR camera should normally be the primary tracking input because it offers a larger field than the main telescope.
- The detector generates a bounding box, confidence score, class estimate, centroid position, apparent size, timestamp, and motion vector.
- A separate tracker should continue following the object between full neural-network detections. This reduces GPU load and prevents unstable tracking when a detection briefly disappears.

## Recommended processing chain

```
Detector -> Persistent tracker -> Motion estimator -> Mount controller
```

A practical operating target is a 60 fps camera stream, a tracker running at full frame rate, an AI detector at approximately 15-30 fps, and filtered mount-command output at approximately 10-20 Hz.

## Suitable tracking components

- Kalman filtering for motion prediction and noise reduction.
- Optical flow for short-term frame-to-frame movement.
- ByteTrack-style data association for maintaining object identity.
- Centroid and feature-point tracking for lightweight recovery.
- Confidence decay and lost-target recovery logic.

# 4. Pixel-to-Angle Conversion and Mount Control

The Jetson must convert the target's pixel displacement into an angular correction using the calibrated camera model. The required calibration includes sensor dimensions, active resolution, focal length, field of view, image rotation, distortion coefficients, and the measured offset between the wide-field optical axis and the main telescope axis.

```
Approximate horizontal angular error = pixel offset x horizontal field of view / image width
```

**Mount corrections should be velocity-based and filtered rather than repeated full GoTo commands. The control loop must include a deadband, maximum velocity, acceleration limit, confidence threshold, lost-target timeout, no-command state, and emergency stop.**

# 5. Image Enhancement Architecture

Image enhancement should be divided into operator-preview enhancement and evidence-preserving enhancement.

| Category | Purpose | Examples | Evidence rule |
|---|---|---|---|
| Operator preview | Make faint NIR targets easier to see in real time | Gamma, CLAHE, denoising, sharpening | May be visually aggressive, but must be labe |
| Evidence-preserving processing | Create traceable derived media for analysis | Dark subtraction, flat-field correction, temporal denoising, calibrated contrast | Original media is retained; every derivative |

## Recommended deterministic NIR pipeline

```
Raw NIR frame -> bad-pixel correction -> dark-frame subtraction -> flat-field correction ->
temporal denoising -> local contrast enhancement -> optional sharpening -> object detector
```

**Classical and calibrated image processing should be implemented before generative enhancement. Neural denoising, deblurring, super-resolution, and segmentation may be added later, but any model capable of inventing visual detail must be clearly marked as an interpretive processed variant rather than original evidence.**

# 6. Provenance and Evidence Integration

Every processed result created by the Jetson should be stored as a child of the original media item. The processing record should identify the source media, processor node, operation, parameters, model version, timestamps, output hash, and manifest hash.

```
{ "operation": "nir_contrast_enhancement", "processor": "JETSON-OPTICAL-01", "model":
"clahe-temporal-v1", "source_media_id": "IMG-...", "parameters": { "clip_limit": 2.0, "tile_grid":
[8, 8], "temporal_frames": 5 } }
```

# 7. Software Stack

| Layer | Recommended component | Function |
|---|---|---|
| **Operating system** | Jetson Linux / Ubuntu | Base operating environment |
| **Video acquisition** | GStreamer | Low-latency camera pipelines and transport |
| **Computer vision** | OpenCV | Preprocessing, tracking, geometry, overlays |
| **GPU inference** | CUDA + TensorRT | Accelerated object-detection and enhancement models |
| **Vision acceleration** | NVIDIA VPI | Optimized image-processing functions |
| **Control API** | FastAPI + WebSocket telemetry | Node registration, commands, status, live metadata |
| **Service management** | systemd | Automatic start, watchdog, restart and logging |
| **Multi-camera production** | DeepStream later | Scalable metadata and inference pipelines |

# 8. Physical Telescope-Mounted Node

The Jetson enclosure should be mounted on an upper Vixen or Losmandy rail, not attached directly to the optical tube using adhesive or a single camera screw. The rail should also provide strain-relief points for short USB or GigE camera cables and local power distribution.

- Aluminium enclosure with rigid mounting plate.
- Active heat sink and fan with thermal monitoring.
- Dust-resistant filtered airflow rather than a completely airtight housing.
- Vibration-resistant connectors and cable glands.
- Accessible NVMe storage and service Ethernet or USB.

- Status LEDs, physical power control, and safe shutdown capability.

# 9. Power and Storage

The node should be powered from a fused 12 V or 24 V distribution system through a regulated DC-DC converter. The Jetson, cooled camera, tracking camera, dew heaters, mount, and network equipment should have separate fused branches.

- Jetson Orin NX design allowance: up to approximately 40 W.
- 1 TB NVMe minimum; 2 TB preferred for field operation.
- Do not rely on microSD for continuous evidence recording.
- Include voltage, current, and temperature monitoring.
- Use short, strain-relieved cables to reduce mount drag and flexure.

## Recommended storage workflow

```
Continuous circular buffer -> detection event -> save pre-event + event + post-event footage -> hash original media -> generate processed variants -> synchronize with Signals Platform
```

A pre-event buffer of approximately 10-30 seconds is valuable because the detector may identify an event only after the event has already begun.

## 10. Signals Platform Node Registration

```
{ "node_id": "JETSON-OPTICAL-01", "node_type": "edge_vision", "hardware": "Jetson Orin NX 16GB",
"capabilities": [ "nir_object_detection", "multi_object_tracking", "contrast_enhancement",
"temporal_denoising", "frame_extraction", "annotated_video", "mount_tracking_output" ] }
```

The Jetson handles camera acquisition, inference, tracking, enhancement, buffering, and mount commands. The Signals Platform VPS remains responsible for the authoritative observation session, evidence index, correlation adapters, unified reports, operator interfaces, and long-term archive.

## 11. Operating Modes

| Mode | Function |
|---|---|
| **Search** | Wide field, elevated gain, moderate frame rate, scanning or stationary watch |
| **Acquire** | Detection confirmation, tracker initialization, confidence validation, mount correction begins |
| **Track** | High-rate centroid updates, predictive filtering, telescope follows target, main camera records |
| **Evidence** | Original streams preserved, processed previews generated, metadata and hashes written |
| **Lost target** | Short predictive continuation, expanded search region, reduced speed, return to search if not recovered |

## 12. Recommended Hardware Specification

- NVIDIA Jetson Orin NX 16 GB.
- Industrial carrier board.
- 1-2 TB NVMe SSD.
- Active heat sink and fan.
- Regulated 12/24 V input.
- At least two USB 3 or GigE camera interfaces.
- Gigabit or 2.5 Gb Ethernet.
- Wi-Fi only as a service-access option.
- Optional IMU.
- GPS/PPS time input strongly desirable.
- Aluminium telescope-mounted enclosure.

## 13. Development Sequence

| Phase | Development target |
|---|---|
| **J1** | Jetson node registration and heartbeat |
| **J2** | Wide-field NIR camera ingestion |
| **J3** | OpenCV motion detector |
| **J4** | Persistent target tracker |
| **J5** | Calibrated pixel-to-angle conversion |
| **J6** | Simulated AZ-EQ6 command output |
| **J7** | Real mount tracking with safety limits |

| | |
|---|---|
| **J8** | NIR enhancement pipeline |
| **J9** | Processed-media provenance integration |
| **J10** | Dual-camera handoff and field validation |

**Core design principle: The Jetson controls the real-time visual loop, while the Signals Platform controls the evidence, operator workflow, correlation, and investigation record.**