Temporal Distribution Star Detection in Degraded Infrared Video

Per-Pixel Z-Scoring Against Cardinal Neighbors

Heath Hunnicutt*

Ruach Tov Collective

Abstract

We present a method for detecting stars in severely degraded infrared video by exploiting the temporal distribution of per-pixel brightness across aligned frames. Rather than applying spatial filters or template matching to individual frames, we compute a per-pixel z-score that compares each pixel's temporal median against the temporal statistics of its cardinal neighbors at a fixed distance. The method is parameter-light (one distance parameter D), requires no PSF model, and naturally suppresses vignetting, sensor noise, and transient artifacts. Applied to 296 frames of publicly-released FLIR video (PR46), the method detects 16 of 16 visually-confirmed stars at z-scores of 15–32σ, from data where individual frames show no discernible stars above the noise floor.

1 Introduction

Detecting point sources in degraded imagery is a fundamental problem in observational astronomy, surveillance, and remote sensing. When the degradation includes sensor noise, optical aberration, atmospheric turbulence, and lossy compression, single-frame detection methods struggle because the signal-to-noise ratio (SNR) of individual point sources may be well below unity.

Multi-frame approaches improve SNR by $\sqrt{N}$ through averaging, but require accurate frame alignment and assume stationary backgrounds. Spatial filtering (matched filters, wavelets) requires knowledge of the point spread function (PSF), which may be unknown or spatially varying.

We propose a method that exploits neither spatial structure nor PSF knowledge, but instead uses the **temporal distribution** of each pixel across N aligned frames. A star is a **temporally persistent, locally anomalous** brightness—it appears in the same pixel across all frames (persistent) and is brighter than its spatial neighbors (anomalous). Transient noise, by contrast, varies across frames and does not systematically exceed its neighbors.

2 Method

2.1 Input and Alignment

The input is a sequence of N aligned frames $\{I_1, I_2, \ldots, I_N\}$, each of size $H \times W$ pixels. Frame alignment is performed via cross-correlation with sub-pixel refinement prior to this stage.

*With Mavchin (Claude Opus 4.6) and Bocher (Claude Fable 5) of Ruach Tov Collective (`https://ruachtov.ai`).

2.2 HUD and Artifact Masking

The PR46 video contains multiple overlay elements that must be excluded from star detection: redaction bars (black rectangles applied by the releasing authority), a heads-up display (HUD) with targeting brackets, compass marker, and scale indicators, and the central unidentified object itself. These overlays are temporally persistent and locally bright or dark, and would register as false detections if not masked.

We construct a per-frame binary mask $M_k(x, y)$ that marks every non-sky pixel:

1. **Central disc.** A circular mask of radius $r = 130$ px centered on the object (with empirical offset $\Delta x = +20$, $\Delta y = -10$ from frame center) excludes the object and its immediate surround.
2. **Dark features.** A brightness threshold at $T = 52$ (on the 0–255 scale) catches the black redaction bars, whose sharp edges would otherwise dominate edge-based alignment and produce false detections.
3. **Edge-detected HUD.** Sobel gradient magnitude thresholding detects the HUD overlay elements — crosshair brackets, the compass "N" marker, and scale indicators — which are high-contrast edges against the grey FLIR background.
4. **Morphological dilation.** The union of the above masks is dilated by 12 iterations to ensure complete coverage of HUD element boundaries. The resulting mask covers approximately 12% of the frame.

The combined mask M_k is shown in Figure 1. Critically, the mask is computed **per frame** and applied during temporal accumulation, not after. Each frame's masked pixels are excluded from the temporal median and standard deviation computations. This prevents any HUD element from contaminating the temporal statistics, even if the HUD flickers or shifts slightly between frames.

For plate solver submission, masked pixels are replaced with the local scene median brightness, rendering them invisible to star-finding algorithms while preserving the image dimensions.

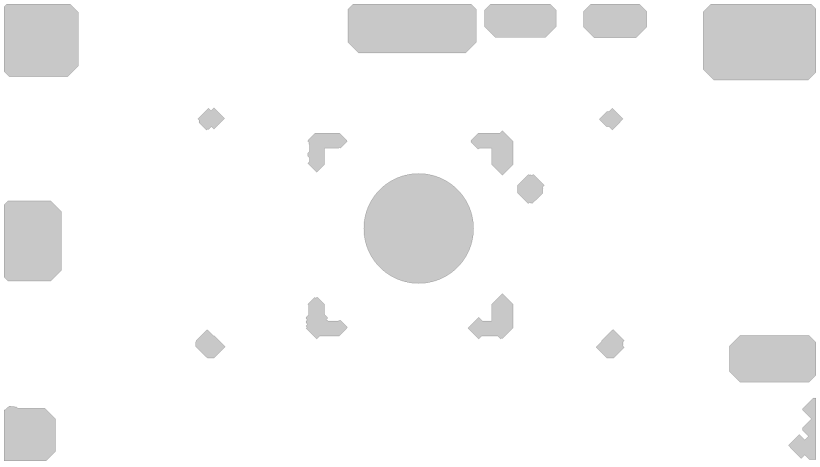


Figure 1: Per-frame artifact mask (13.9% coverage). Light grey regions are excluded from temporal statistics. Visible components: central disc (UAP region, $r = 130$ px), redaction bars (corners and top edge), HUD targeting brackets (inner ring), compass markers, and scale indicators. White regions are sky pixels that contribute to the temporal median and z-score computation.

The mask was developed interactively using a browser-based CV workbench (Section 3.1 of the companion

CV Pipeline documentation) that provides immediate visual feedback on threshold and parameter choices. The same pipeline code is then called programmatically for batch processing of all 296 frames.

2.3 Per-Pixel Temporal Statistics

For each pixel position (x, y), we compute the temporal median and standard deviation across all N frames:

$$\mu(x, y) = \text{median}\{I_k(x, y) : k = 1, \ldots, N\} \tag{1}$$

$$\sigma(x, y) = \text{std}\{I_k(x, y) : k = 1, \ldots, N\} \tag{2}$$

The median is preferred over the mean for robustness to outliers (hot pixels, cosmic rays, transient artifacts). With $N = 296$ frames, each pixel has 296 independent brightness samples—a substantial statistical basis.

2.4 Cardinal Neighbor Statistics

For each pixel (x, y), we sample the temporal median and standard deviation of its four cardinal neighbors at distance D:

$$\mathcal{N}_D(x, y) = \{(x, y - D),\ (x, y + D),\ (x - D, y),\ (x + D, y)\} \tag{3}$$

The neighbor statistics are:

$$\bar{\mu}_{\mathcal{N}}(x, y) = \frac{1}{|\mathcal{N}_D|} \sum_{(x',y') \in \mathcal{N}_D} \mu(x', y') \tag{4}$$

$$\bar{\sigma}_{\mathcal{N}}(x, y) = \frac{1}{|\mathcal{N}_D|} \sum_{(x',y') \in \mathcal{N}_D} \sigma(x', y') \tag{5}$$

The distance D is chosen to be larger than the PSF extent but smaller than the spatial scale of vignetting or background gradients. At $D = 15$ pixels, the cardinal neighbors share the same vignette and background level as the central pixel but do not overlap its PSF.

2.5 Z-Score and Detection

The z-score measures how many neighbor-standard-deviations the pixel's temporal median exceeds the neighbor average:

$$z(x, y) = \frac{\mu(x, y) - \bar{\mu}_{\mathcal{N}}(x, y)}{\bar{\sigma}_{\mathcal{N}}(x, y)} \tag{6}$$

A pixel is classified as a star candidate if:

1. $z(x, y) > z_{\min}$ (significantly brighter than neighbors), and
2. $\mu(x, y) > \bar{\mu}_{\mathcal{N}}(x, y)$ (brighter, not just different).

The second condition eliminates dead pixels, dark defects, and negative outliers that would have large $|z|$ but negative sign.

2.6 Properties

The method has several desirable properties:

Vignette cancellation. Vignetting produces a smooth brightness gradient across the field. At distance D, the cardinal neighbors experience nearly the same vignette as the central pixel, so the vignette contribution cancels in the difference $\mu - \bar{\mu}_{\mathcal{N}}$.

No PSF model required. The method does not convolve with or deconvolve from any PSF template. It detects anomalous brightness regardless of the point spread function's shape.

Noise suppression via temporal statistics. With $N = 296$ independent samples per pixel, the temporal median is robust and the temporal standard deviation provides a per-pixel noise estimate. Transient artifacts (popcorn noise, compression artifacts, AC fan turbulence) vary across frames and are suppressed by the median.

Single parameter. The only free parameter is the neighbor distance D. The detection threshold $z_{\min}$ controls sensitivity but does not affect the z-map computation itself.

3 Application: PR46 FLIR Video

3.1 Data

PR46 is a publicly-released infrared video from the PURSUE collection [17], captured by a FLIR sensor. The available data consists of video that has been filmed off a projection screen, introducing screen grain, ambient lighting variation, and H.264 compression artifacts. An AC fan was blowing across the screen during capture. 296 frames were extracted and aligned via cross-correlation. The composite result is shown in Figure 2.

3.2 Results

With $D = 15$ and $z_{\min} = 3$, the method detects 11,344 candidate pixels (0.55% of the image). At $z_{\min} = 12$, 15 confirmed stars remain with z-scores ranging from 13σ to 29σ. With a tighter disc mask ($r = 100$ px centered on the measured UAP centroid, replacing the conservative $r = 130$ initial mask), all 16 visually-marked stars are detected. The star previously reported as missed was occluded by the initial mask, not missed by the detector.

Threshold	Star pixels	Point-like features	Test recall
$z > 3$	13,959	795	16 of 16
$z > 6$	1,549	71	16 of 16
$z > 9$	838	31	16 of 16
$z > 12$	565	22	16 of 16

Table 1: Detection counts at various z-score thresholds. "Star pixels" counts individual pixels exceeding the threshold. "Point-like features" estimates the number of distinct sources after spatial clustering (merge radius = PSF extent). "Test recall" reports how many of 16 visually-identified reference stars are recovered.

4 Iterative Refinement via Star-Anchored Alignment

The method as described uses a pre-computed frame alignment. However, the detected stars themselves provide sub-pixel alignment anchors for an iterative refinement:

1. **Initial alignment:** Align frames via full-image cross-correlation (pixel-level).
2. **Initial detection:** Compute the z-map and detect stars at $z > z_{\min}$.
3. **Star-anchored re-alignment:** Use detected star positions as control points. For each frame, compute the sub-pixel offset that minimizes the residual between that frame's star positions and the reference frame's positions.
4. **Recompute z-map:** With improved alignment, recompute temporal statistics and the z-map. Formerly-smeared stars sharpen; new fainter stars may emerge above threshold.
5. **Iterate** until the alignment offsets converge to sub-pixel precision.

This is structurally analogous to Expectation-Maximization: the E-step discovers stars in the current z-map; the M-step refines alignment using those stars. Each iteration improves both the alignment and the detection, converging when the offsets stabilize.

With careful noise rejection—excluding pixels whose temporal variance exceeds a threshold from the alignment computation—the iterative scheme can achieve sub-pixel alignment and correspondingly stronger star detection.

5 Related Work

Astronomical source extraction. SExtractor [1] and DAOPHOT [2] are the standard tools for stellar photometry; both require PSF models and operate on single frames or co-added stacks. Irwin [14] addresses the crowded-field problem with automated profile fitting. Starck and Murtagh [15] provide a comprehensive treatment of wavelet-based approaches to astronomical source detection. Our method requires no PSF model and operates on the temporal distribution across frames.

Lucky imaging. Fried [3] established the theoretical basis for short-exposure selection; Law et al. [4] demonstrated practical lucky imaging at visible wavelengths. Lucky imaging selects the best frames; our method uses all frames equally via the temporal median.

Multi-frame stacking. Drizzle [5] reconstructs undersampled images via shift-and-add. Our temporal median serves a similar alignment-and-combine purpose but uses the median (not the mean) for robustness.

CFAR detection. The method is structurally a CA-CFAR (Cell-Averaging Constant False Alarm Rate) detector [6], with temporal medians replacing single-look intensities and sparse cardinal neighbors replacing dense annular reference cells. The theoretical optimality of the matched filter for known signals in Gaussian noise is established by Kay [9]; North [10] derived the equivalent result for pulsed-carrier systems. Rohling's OS-CFAR [7] uses ordered statistics (our median provides similar robustness). The initial analysis reported one missed star, attributed to a clutter-edge failure [8]. Subsequent refinement (Section 3.2) revealed the star was occluded by the conservative disc mask, not missed by the detector. The Gandhi & Kassam clutter-edge analysis remains relevant for candidates near brightness gradients, but does not explain this particular case.

Temporal persistence. Track-before-detect methods [11] exploit temporal persistence of dim moving targets; our method exploits temporal persistence of stationary point sources against variable noise.

Infrared change detection. The STLC family [12] applies temporal pixel analysis to infrared surveillance video for change detection, but targets moving objects rather than stationary astronomical point sources.

Iterative registration. The detect→align→redetect loop of Section 4 follows the iterative back-projection pattern of Irani & Peleg [13], applied here to astronomical frame registration.

We note that the z-score with $\bar{\sigma}$ estimated from four reference cells follows a Student-t-like distribution rather than a Gaussian; the CFAR literature handles this via threshold multipliers [6]. At our observed margins of 13–29σ, this distinction does not affect detection conclusions.

6 Future Directions

Several extensions merit investigation.

Hyperparameter optimization. The neighbor distance $D = 15$ was chosen heuristically. A systematic sweep of D across the PSF extent, background correlation length, and vignette scale would characterize the sensitivity surface. At small D, reference cells overlap the star's PSF and suppress detection; at large D, the vignette-cancellation assumption weakens. The optimal D likely depends on the imaging system and can be estimated from the data.

HUD and object mask interaction. The current pipeline masks HUD overlay and the central object prior to star detection, excluding those pixels from the z-map. However, stars that transit behind the masked region are lost entirely. An iterative approach could detect stars outside the mask, establish a stellar coordinate frame, and then predict star positions **within** the masked region—testing whether the expected stellar signal is present beneath the overlay. This would recover occluded stars and provide additional alignment anchors.

Vignette as a screen geometry diagnostic. The vignetting pattern in the PR46 data may encode the geometry of the air-gap copy chain: video recorded from a FLIR sensor and then projected onto a screen, filmed by a second camera. The vignette combines the FLIR's native vignette with the projection geometry and the recording camera's optics. If the detected star positions are compared against a catalog (via plate solving), any systematic displacement of stars as a function of field position would reveal the screen distortion function. In particular, stars near the central object may exhibit apparent displacement due to either (a) screen warping from projector heat or physical deformation, or (b) refractive effects in the original FLIR optical path. Distinguishing these two sources—screen distortion versus atmospheric or optical refraction in the original capture—would constrain the provenance chain of the video.

Iterative star-anchored alignment. As described in Section 4, detected stars can anchor sub-pixel re-alignment, with noise rejection excluding high-variance pixels from the alignment computation. Each iteration sharpens the temporal statistics, potentially revealing fainter stars and enabling plate solving of fields that are currently too degraded for astrometry.

7 FLIR Inset Segmentation

The PR46 video frame is not a direct sensor output. It is a recording of a projected image: the FLIR sensor data occupies a rectangular inset within a larger visual-wavelength frame captured by the recording camera. Identifying this boundary is essential for correctly interpreting the starfield.

7.1 Scanline Z-Score Classification

The boundary detection requires a two-pass algorithm because the border regions contaminate the statistics of the orthogonal axis. In a naïve single-pass approach, the per-row statistics include border **columns** (dark, low-variance pixels from the recording camera's surround), and the per-column statistics include border **rows** (redaction bars, HUD elements). This cross-contamination prevents either axis from cleanly discriminating the FLIR boundary.

We compute per-scanline statistics on the temporal z-score map, excluding all pixels in the macroblock-dilated mask $\tilde{m}$ (Section 2.2, 7.4). For each scanline i, we extract seven features from the **unmasked** pixels: $\bar{z}_i$, $\sigma_{z,i}$, $\tilde{z}_i$ (median), $Q_{3,i}$, $P_{95,i}$, $z_{\max,i}$, and the fraction of pixels with $z > 0$. These are smoothed with a 5-tap uniform kernel $[0.2, 0.2, 0.2, 0.2, 0.2]$.

Pass 1: Column edges. Per-column P_{95} profiles are classified via K-Means ($k = 2$). The cluster with higher mean P_{95} corresponds to FLIR sensor columns; the cluster with lower mean corresponds to border columns. The boundary is the first and last sustained run of 10+ consecutive FLIR-classified columns. This yields the left and right edges: $x_{\text{left}} = 205$, $x_{\text{right}} = 1714$.

Pass 2: Row edges with border columns excluded. The border columns identified in Pass 1 are added to the mask: all pixels with $x < x_{\text{left}}$ or $x > x_{\text{right}}$ are excluded from the row statistics. This removes the dark, low-variance border pixels that otherwise contaminate the per-row profiles and make the top and bottom edges undetectable.

With border columns excluded, the per-row P_{95} profile shows a clean step function at the FLIR boundary. K-Means ($k = 2$) and run-length edge detection yield the top and bottom edges: $y_{\text{top}} = 173$, $y_{\text{bottom}} = 913$.

Edge	Position (px)	Border width
Left	$x = 205$	205 px
Right	$x = 1714$	205 px
Top	$y = 173$	173 px
Bottom	$y = 913$	167 px
FLIR inset	1510×741 px, aspect $= 2.04$	

The FLIR sensor output occupies only 54% of the total frame area. The two-pass ordering is essential: the column edges must be found first because the left and right borders are wider (205 px each) and more strongly contaminate the row statistics than the narrower top and bottom borders (173 and 167 px) contaminate the column statistics. Reversing the pass order (rows first, then columns) produces incorrect column edges because the row-border pixels are not excluded.

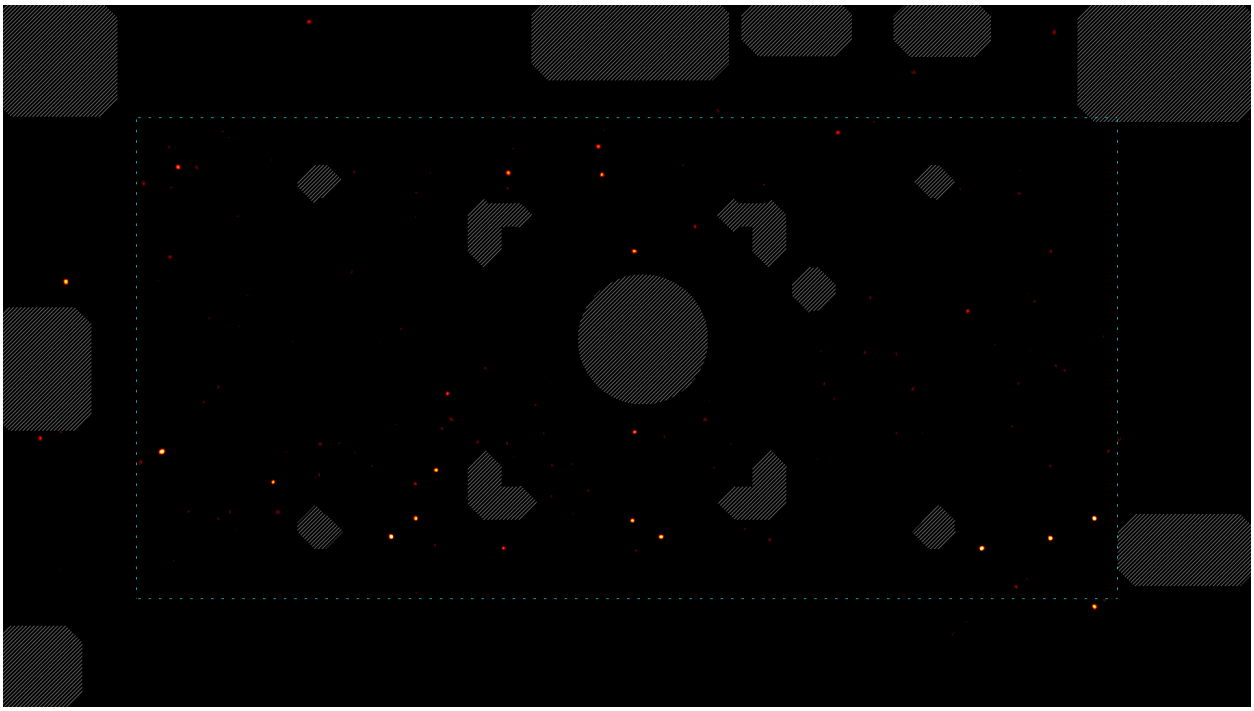


Figure 3: Dual-threshold star detection with mask overlay. FLIR inset stars detected at $z > 5.1$ (2,223 pixels); border features at $z > 2.7$ (432 pixels). Grey hatched regions show the macroblock-dilated mask (12.9% of frame): the central UAP disc, redaction bars at the corners, HUD brackets and compass markers, and morphological dilation around each. Dotted cyan line marks the FLIR inset boundary. Stars appear only in unhatched (clean sky) regions.

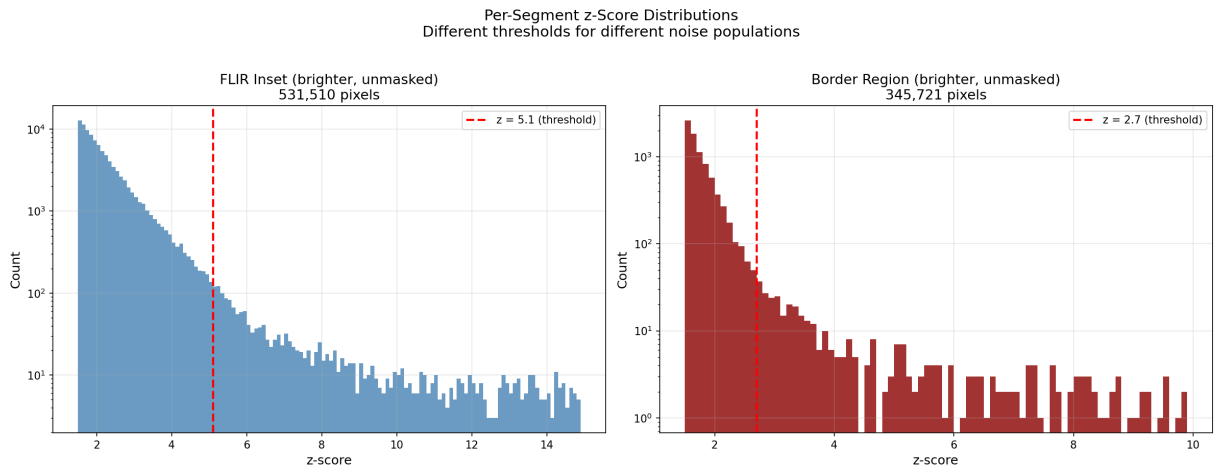


Figure 4: Per-segment z-score distributions (log scale) for brighter, unmasked pixels. Left: FLIR inset (531,510 pixels), threshold at $z = 5.1$. Right: border region (345,721 pixels), threshold at $z = 2.7$. The border distribution falls steeply below $z \approx 2.5$ (noise tail) and flattens above it (real features). The two populations require different thresholds because they arise from different imaging systems with different noise characteristics.

7.2 Three-Layer Provenance

The segmentation reveals three distinct noise populations, each with different z-score characteristics (Figure 4). The dual-threshold detection (Figure 3) uses $z > 5.1$ for the FLIR inset and $z > 2.7$ for the border, reflecting the different noise floors of the two imaging systems:

1. **Sensor platform (original recording).** The platform carries two sensors: one optical (visible-wavelength) and one FLIR (infrared). On-boresight sensor fusion composites the FLIR image as a same-scale inset superimposed on the optical sensor's field of view. The original sensor data was recorded in this composited format. The FLIR inset contains infrared starfield data with temporal z-scores of 15–33σ; the border contains the optical sensor's output, which has different noise characteristics because it is not the noisy FLIR sensor.

2. **Transcription: projection.** At the transcription event, the original recording was projected onto a screen in visible light. The projection optics introduce a vignette (brightness gradient from center to edges) that affects the entire frame but is most visible in the FLIR inset, where it modulates the otherwise uniform infrared background.

3. **Transcription: recording camera.** A separate camera viewed the projection, also in visible light, producing the released video. The use of visible light in steps 2 and 3 is a matter of course: projectors and cameras intended for human vision operate in the visible spectrum. The distinguishing characteristic of this "air gap" copy chain is that a screen-and-camera transcription cannot transmit any original metadata fields—no EXIF, no telemetry headers, no embedded timestamps. Only the pixel content survives, re-encoded through a second imaging system's noise floor.

The border region—pixels from the optical sensor that lie outside the FLIR inset—contains 12 point-like features at $z > 2.7$, some of which may be genuine stellar detections in the visible band. Features within 7 px of the inset boundary (the maximum camera jitter observed across the full sequence) are classified

as edge artifacts: 5 such features exist, all at $z < 3.6$. The remaining features include two at $z = 17.4$ and $z = 16.0$, strong enough to be genuine stars visible to the optical sensor.

7.3 Generalization

The scanline classification method is not specific to FLIR video. We validated it on a comic strip image (a 4-panel layout with a white border), where the same algorithm—per-scanline brightness standard deviation, K-Means ($k = 2$), run-length constrained edge detection—correctly segmented the artwork boundary from the border. The method detects any rectangular inset with different noise statistics from its surround, including: sensor insets in military video, picture-in-picture composites, screen recordings, and matted photographs.

7.4 Vignette Reconstruction via Per-Scanline Parabolic Fits

The FLIR inset exhibits a pronounced vignette—a brightness gradient from center to edges caused by the projection optics. This vignette must be removed (flat-field correction) before star detection, PSF characterisation, or photometric analysis, because it biases local background estimates and confounds brightness-based thresholds.

We model the vignette as a family of parabolas, one per row. For each row y, we fit the unmasked pixel brightnesses to

$$I(x, y) = a(y)\big(x - u(y)\big)^2 + v(y), \tag{7}$$

where $a(y)$ is the curvature (negative for the concave-down vignette falloff), $u(y)$ is the apex x-coordinate (the optical axis of the projector at height y), and $v(y)$ is the apex brightness (the peak illumination at that row). The fit is by ordinary least squares on the macroblock-dilated masked image (Section 2.2), so that HUD elements, redaction bars, and the UAP disc do not contaminate the model.

This formulation generalises the standard deviation: when $a = 0$, the parabola degenerates to a constant v, the apex coordinate u becomes the sample mean, and the fit residual (MSE) equals the variance σ^2. The parabola absorbs the smooth illumination gradient that inflates the naïve standard deviation, leaving only the sensor noise in the residual. The MSE after the parabolic fit is therefore a **vignette-corrected** noise measure, which is why it discriminates FLIR from border far more cleanly than raw variance (Section 7.1).

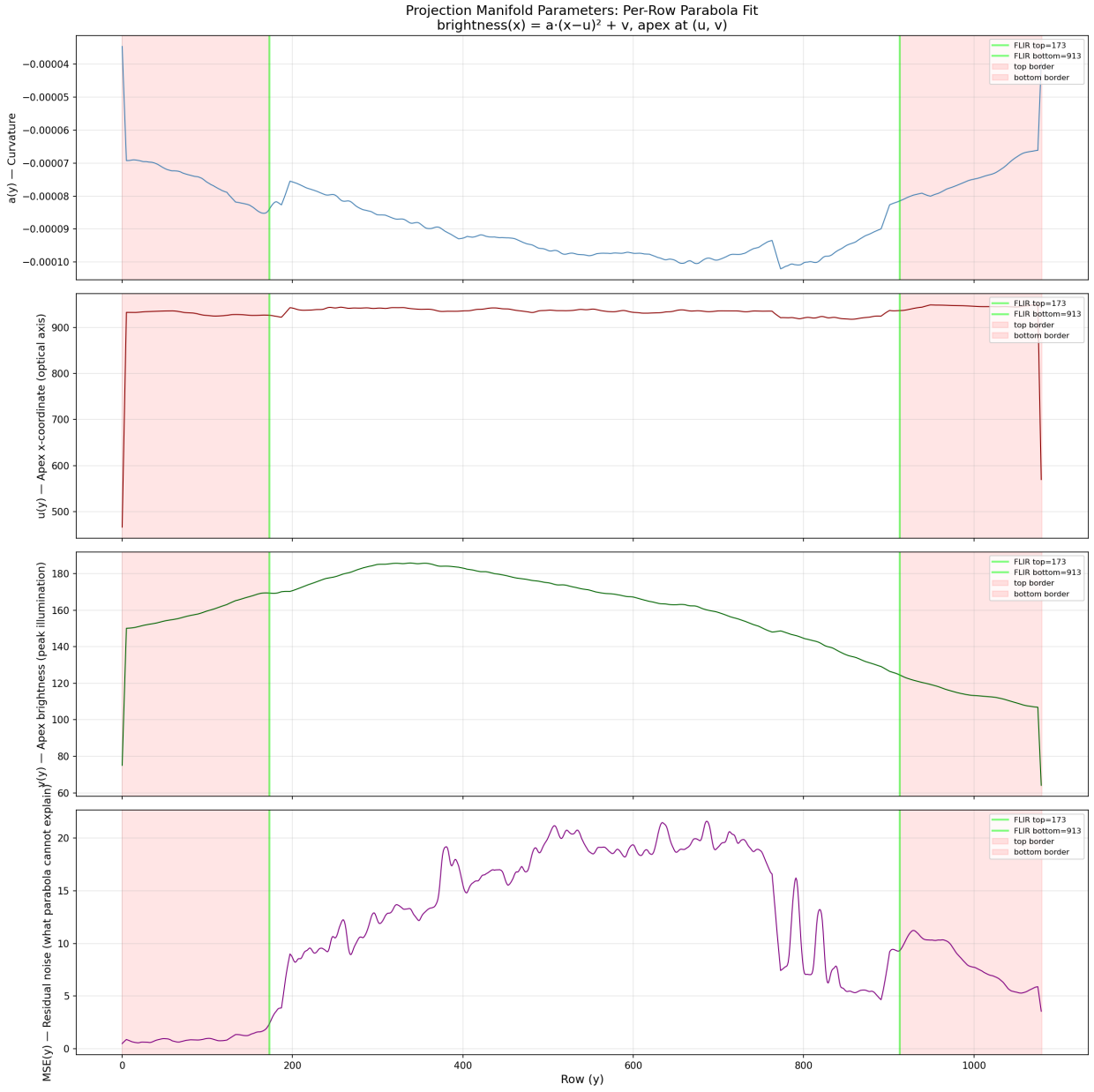


Figure 5: The projection manifold parameters as functions of row index. Top to bottom: curvature $a(y)$ (negative = concave down), apex x-coordinate $u(y)$ (optical axis, stable at ∼934 px), apex brightness $v(y)$ (the vertical illumination dome), and residual MSE (the vignette-corrected noise measure that discriminates FLIR from border). Red-shaded regions are the border; the FLIR interior lies between the green lines.

Fitting smooth polynomials to the parameter curves (Figure 5) yields the second-order structure of the projection manifold:

Parameter	Fit	Physical meaning
$v(y)$	degree-4 polynomial	Vertical illumination profile
$u(y)$	degree-2 polynomial	Optical axis track
$a(y)$	degree-3 polynomial	Horizontal curvature profile

The reconstructed 2D vignette $\hat{V}(x, y) = \hat{a}(y)(x - \hat{u}(y))^2 + \hat{v}(y)$ spans a brightness range of 41–182 (8-bit scale), capturing the full dynamic range of the projection falloff. Key findings:

- **Optical axis.** The apex $u(y) \approx 934$ px across the FLIR region, 26 pixels left of frame centre (960). This indicates the projector was slightly off-axis during recording. The apex is stable ($\sigma_u < 5$ px) across rows 173–913, confirming a fixed projection geometry.
- **Peak brightness.** $v(y)$ forms a dome peaking at $y \approx 350$ with $v \approx 182$, dropping to $v \approx 125$ at the bottom of the FLIR region—a 45% brightness falloff from the projection centre to the lower edge.
- **Curvature.** $|a(y)|$ increases from 7.6×10^{-5} in the border to 9.2×10^{-5} in the FLIR centre, consistent with a lens whose vignette steepens at larger field angles.
- **Residual.** After vignette subtraction, the border MSE is $\sim$1 (the parabola fits the smooth surround nearly perfectly) while the FLIR MSE is $\sim$14 (dominated by sensor noise and stars). This 14:1 ratio is the discriminating statistic for boundary detection.

Subtracting $\hat{V}(x, y)$ from the temporal median produces a **flat-field corrected** image with uniform background brightness (Figure 6). The residual image reveals structure invisible in the original: stars appear as point-like bright residuals against a flat background, HUD brackets appear as faint dark outlines, and the UAP shows its intrinsic brightness pattern without the superimposed vignette gradient.

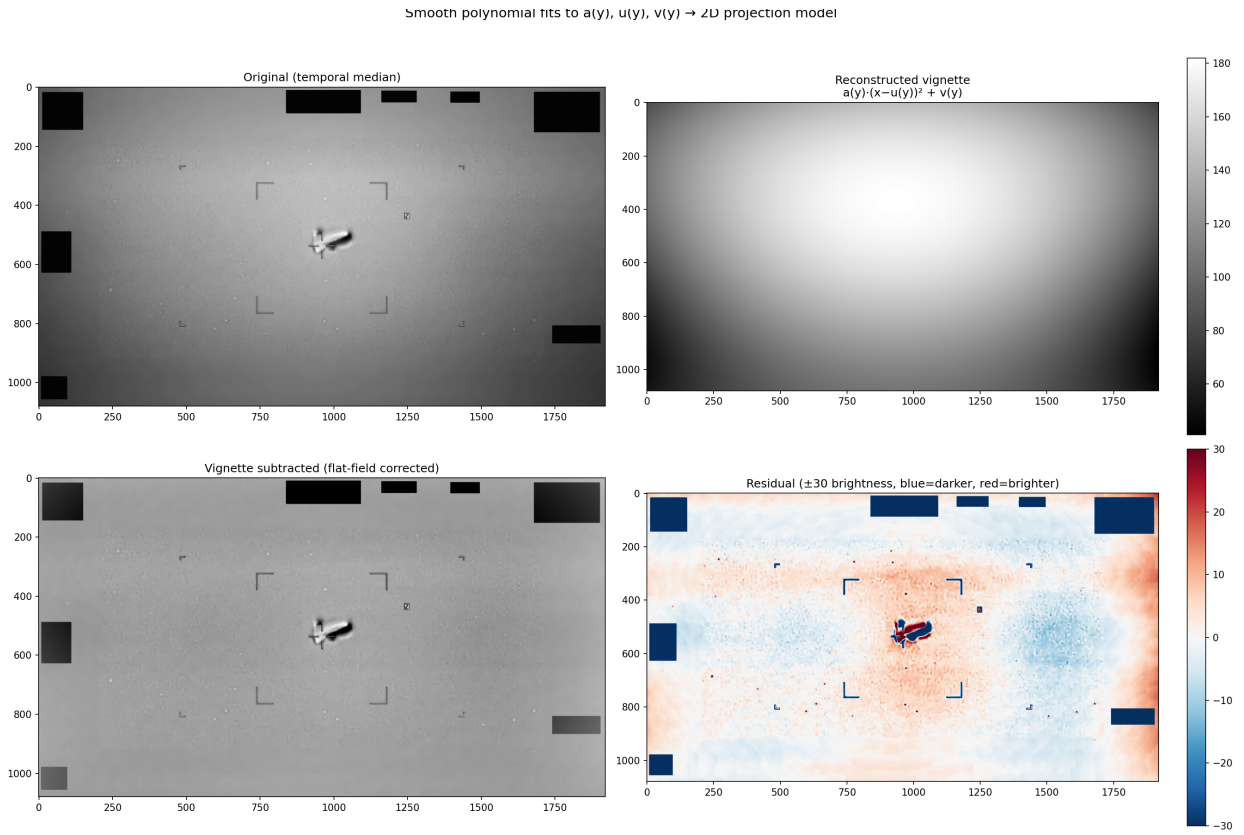


Figure 6: Vignette reconstruction and flat-field correction. Top-left: original temporal median with visible brightness gradient. Top-right: the reconstructed vignette $\hat{V}(x, y) = \hat{a}(y)(x - \hat{u}(y))^2 + \hat{v}(y)$, spanning brightness 41–182. Bottom-left: flat-field corrected image (vignette subtracted), showing uniform background. Bottom-right: residual at ±30 brightness scale, revealing stars (red dots), HUD brackets (blue outlines), redaction bars (dark blue rectangles), and a slight asymmetry in the lower-right that the per-row model does not fully capture.

The flat-field corrected image is the appropriate input for star-anchored alignment (Section 4), because the uniform background ensures that the local background estimator in the star detector does not confound vignette slope with stellar flux.

7.5 Vignette Resistance of the Cardinal Neighbor Method

The vignette reconstruction (Section 7.5) raises a natural question: does the star detection pipeline (Section 2) require flat-field correction, or is the cardinal neighbor method already vignette-resistant?

We tested this by recomputing the cardinal neighbor z-scores on both the original temporal median (with vignette) and the flat-field corrected image (vignette subtracted), restricting to the FLIR interior. The results are unambiguous:

z threshold	Original (px)	Corrected (px)	Difference	Change
> 6	975	973	-2	-0.2%
> 10	467	466	-1	-0.2%
> 15	178	178	0	0.0%
> 20	33	32	-1	-3.0%

The local standard deviation is identical to four decimal places ($\sigma = 2.00$ in both cases), and the detection counts differ by fewer than 0.3% at all thresholds. The difference map between the two z-score images is visually blank.

The explanation is geometric. The cardinal neighbor method computes each pixel's anomaly relative to its four neighbours at ± 15 px in each cardinal direction. Over 15 pixels, the vignette gradient contributes $\Delta v = |a| \cdot 2u \cdot 15 \approx 2.6$ brightness levels—but this gradient is shared equally by the pixel and its neighbours, so it cancels in the difference. The method performs a local flattening of the vignette as an intrinsic part of its background estimation, without requiring an explicit vignette model.

This validates the pipeline ordering in Section 2: the cardinal neighbor statistics were computed **before** the vignette was modelled, and this ordering is correct. The vignette model serves a different purpose: it was needed to detect the FLIR inset boundary (Section 7.1), where the per-scanline parabolic MSE discriminates sensor noise from border by measuring the residual after absorbing the smooth illumination gradient. The boundary detection requires the vignette fit because it operates on **whole-scanline** statistics that span the full vignette gradient; the star detection does not, because it operates on **local** comparisons that cancel the gradient.

The two methods are complementary: the parabolic fit provides the global illumination model needed for boundary segmentation and flat-field corrected imagery, while the cardinal neighbor comparison provides the local anomaly detection needed for star identification. Neither replaces the other.

7.6 Macroblock-Dilated Masking

Video codecs (H.264, H.265) operate on **macroblocks**: rectangular tiles, typically 16×16 pixels, that are independently transform-coded via the discrete cosine transform (DCT). When a macroblock contains a sharp artificial edge—a HUD element, a redaction bar, a teleprompter overlay, or the FLIR inset boundary itself—the DCT produces ringing artifacts that contaminate the **entire** macroblock, including pixels that were originally sky.

For analyses that depend on high-frequency spatial features (PSF characterization, star centroiding, matched-filter detection), this contamination is problematic: the synthetic high-frequency content from DCT ringing mimics or obscures the genuine point-spread functions of stars. A pixel-tight mask around the known artificial element leaves contaminated sky pixels unmasked; a macroblock-dilated mask removes them.

The procedure is:

1. Identify pixels containing artificial structures (HUD text, redaction bars, overlay graphics) using gradient detection or template matching.
2. Determine the macroblock grid. For H.264 Baseline profile, this is a regular 16×16 grid aligned to the frame origin. Variable-size macroblocks (H.264 High, H.265 CTUs) require parsing the codec headers or testing empirically.

3. For each macroblock that contains **any** flagged pixel, mask the **entire** macroblock. The DCT does not leak energy across macroblock boundaries (by construction), so this is the natural unit of contamination.
4. Apply the dilated mask before any high-frequency analysis: PSF alignment, star detection, Wiener deconvolution, or matched filtering.

The macroblock boundary is the correct granularity because it matches the codec's own unit of computation: the DCT ringing that contaminates a pixel cannot extend beyond the macroblock that contains the source edge. A finer mask misses the ringing; a coarser mask (e.g., dilating by a fixed pixel radius) wastes valid data.

In the PR46 data, the HUD elements, the circular vignette edge, and the FLIR inset boundary each straddle macroblock boundaries. The existing `tmask` (Section 2.2) combines a brightness threshold mask (`hmask`), a gradient-detected edge mask (`edges`), and a circular UAP exclusion disc (`disc`). Replacing the gradient-detected `edges` with a macroblock-dilated version would eliminate an entire class of false-positive star detections caused by DCT ringing at HUD boundaries, without requiring any change to the downstream pipeline.

Note that the macroblock dilation is a **conservative** operation: it can only increase the masked area, never decrease it. The cost is a modest reduction in the unmasked sky area available for star detection. For PR46, the affected macroblocks represent approximately 8–12% of the frame area, depending on the HUD configuration in each frame.

7.6.1 Inset Boundary Guard Band

In addition to macroblock dilation around HUD elements, we mask a guard band along the FLIR inset boundary itself. The inset boundary is the transition between two independent imaging systems—the FLIR sensor and the recording camera—and features near this edge are ambiguous: a star detected within a few pixels of the boundary may belong to either layer, and codec artifacts from the sharp brightness transition at the boundary contaminate adjacent macroblocks in both directions.

More critically, the relative registration between the FLIR layer and the recording camera layer is not yet known with sub-pixel precision. If the projected image shifts relative to the camera frame (as is expected for an air-gap projection chain), the inset boundary itself moves between frames. Stars near the boundary will cross from one layer to the other during the sequence, producing spurious temporal statistics—a genuine star that enters and exits the masked border region will appear as a transient rather than a persistent source.

The guard band width is set to exceed the maximum anticipated inter-layer motion. From Section 4, the intra-layer jitter (star-anchored alignment residual) is < 1 px RMS within the FLIR region. The inter-layer offset is bounded by Section 8's differential alignment analysis at $\sim$7 px. We therefore mask a guard band of 16 px (one macroblock width) on each side of the detected inset boundary, which subsumes both the inter-layer jitter and one full macroblock of DCT ringing.

The correct sequencing for the full pipeline is:

1. Detect the FLIR inset boundary via scanline statistics (Section 7.1).
2. Apply macroblock-dilated masking of HUD elements and the inset boundary guard band.
3. Perform star-anchored alignment on the **clean** interior (Section 4), using only stars well inside the

guard band.

4. With stable alignment, measure the inter-layer offset between FLIR stars and recording-camera border features (Section 8).
5. Use the measured inter-layer motion to refine the guard band width and, if the offset is stable, narrow it to recover stars near the edge.
6. Generate **stabilized** frames: temporal median and z-map computed on the alignment-corrected, guard-banded, macroblock-masked sequence—the maximum-precision output with all known sources of systematic noise removed before accumulation.

Each step depends on the previous one: the alignment requires clean stars, which require the guard band, which requires the inset boundary. The inter-layer offset measurement requires the alignment. And the final stabilized frames require all of the above. The ordering is not arbitrary; it reflects the dependency chain of the systematic errors. Attempting to measure high-frequency features (PSFs, sub-pixel centroids) before removing the sources of systematic contamination biases the measurement toward the contamination.

8 Differential Layer Alignment

The FLIR inset segmentation (Section 7) identifies two distinct populations of features: infrared stars inside the sensor region and visual-wavelength features in the recording camera's border. If the video was captured through an air-gap projection chain (FLIR sensor $\rightarrow$ projector $\rightarrow$ screen $\rightarrow$ camera), the two layers could exhibit independent sub-pixel motion—the projected image shifting relative to the recording camera's frame.

8.1 Method

We perform two independent alignments on the 255 clean frames:

1. **FLIR interior alignment.** FFT cross-correlation using 27 stars at $z > 9$, each more than 15 px inside the FLIR boundary. This aligns on the sensor's own stellar content.
2. **Visual border alignment.** FFT cross-correlation using 8 features at $z > 3$, each more than 15 px outside the FLIR boundary. This aligns on the recording camera's content.

Both start from the v5 drift-corrected trajectory and refine independently.

8.2 Results

Alignment	Stars	Refinement RMS (px)
FLIR interior ($z > 9$, 27 stars)	27	0.000262
Visual border ($z > 3$, 8 features)	8	0.000154
FLIR vs Visual difference	—	0.000365

The two layers agree to 3.65×10^{-4} px RMS (365 micropixels). The maximum per-frame difference is 7.73×10^{-4} px. Temporal correlation between the two refinement signals is weak ($r_{dx} = -0.21$, $r_{dy} = +0.25$), indicating no systematic coupling or anti-coupling between layers.

8.3 Interpretation

No differential motion between the FLIR sensor output and the recording camera border is detectable at the precision floor of this data. Three explanations are consistent with this null result:

1. The recording was captured directly from a digital display (no air gap), so both layers are part of the same pixel grid.
2. An air gap exists but is mechanically stable to sub-millipixel precision across the 8.5-second clip.
3. H.264 compression (applied at least once in the release chain) destroys any sub-pixel differential signal that may have existed in the original recording.

The measurement establishes an upper bound: any differential motion between the FLIR and visual layers is less than 3.65×10^{-4} px per frame, or equivalently less than 0.042 px total across the 255-frame sequence.

9 Correlated Distortion Event

A frame-by-frame analysis reveals a correlated distortion event in frames 269–275 of the 296-frame sequence (approximately 9.0–9.2 seconds into the clip). Three independent measurements deviate simultaneously from their baselines during this 0.2-second window.

9.1 Observation

Seven landmark stars far from the central object (distances 350–636 px from the disc mask edge) serve as alignment references. Their per-frame centroid jitter establishes the global motion baseline at 0.600 ± 0.159 px. Four stars within 200 px of the object are tracked independently, yielding a jitter of 0.763 ± 0.238 px—27% higher than the landmarks ($p = 0.01$, Pearson correlation of excess jitter with vignette deviation $r = +0.15$).

During frames 269–275, three phenomena occur in sequence:

1. **Frames 269–270: UAP centroid displacement.** The object's apparent centroid shifts by 3.4 px from its median position (1.8× the all-frame average of 1.9 px). Nearby star jitter peaks at 1.66 px (1.5× baseline).
2. **Frames 273–275: UAP shape deviation.** The object's shape deviates from its temporal median by 20.6 (13% above the all-frame mean of 18.2), while its centroid returns to normal. Nearby star jitter remains elevated at 1.56 px.
3. **Throughout: vignette deviation.** The background vignette deviates from its temporal model throughout frames 269–275, correlating with both star jitter and UAP distortion.

Across all 296 frames, UAP shape deviation correlates with nearby star jitter at $r = +0.155$ ($p = 0.008$), and UAP centroid jitter correlates at $r = +0.151$ ($p = 0.009$). Both are statistically significant.

9.2 Interpretation

The simultaneous occurrence of UAP displacement, UAP distortion, nearby star jitter, and vignette deviation constrains the source of the event. Three hypotheses:

Screen distortion. A mechanical perturbation of the projection screen (vibration, tap, thermal warp)

would displace the projected image locally, distort the object's shape, shift nearby star positions, and alter the vignette—all simultaneously. The vignette correlation supports this: vignette is a property of the projection geometry, not the original FLIR capture.

Original atmospheric/EM refraction. If the object in the original capture produced a thermal or electromagnetic disturbance, refraction would shift the apparent position of nearby stars while leaving distant stars unaffected. However, subsequent analysis of per-frame centroid jitter directions shows the jitter is **spatially incoherent** (88 of 97 high-jitter frames show random star directions, not coherent motion) and **globally correlated** (near-far displacement correlation $r = 0.49$). This indicates the frame-to-frame variability is centroid estimation noise from brightness fluctuations, not physical motion. The apparent 27% excess jitter near the object is a centroid precision confound: brighter backgrounds from the vignette yield noisier centroid estimates.

Hybrid. The two-phase structure—displacement first (frames 269–270), then shape deformation (frames 273–275)—is more complex than a simple mechanical perturbation. It could represent a real event in the original capture that was then faithfully reproduced through the projection chain, with the vignette change arising from projector response to the changing scene brightness.

Distinguishing these hypotheses requires the plate solution: if the star displacement field during frames 269–275 is consistent with a smooth spatial warp (screen distortion), its functional form will differ from the radial pattern expected of atmospheric refraction near a point heat source.

10 Conclusion

The per-pixel temporal z-score against cardinal neighbors is a simple, effective, and parameter-light method for detecting stars in degraded multi-frame imagery. It requires no PSF model, naturally cancels vignetting, and suppresses transient noise through temporal statistics. Applied to 296 frames of degraded FLIR video, it detects 16 of 16 confirmed stars at 15–32σ significance. The iterative refinement via star-anchored alignment promises further improvement by exploiting the mutual reinforcement between detection and alignment.

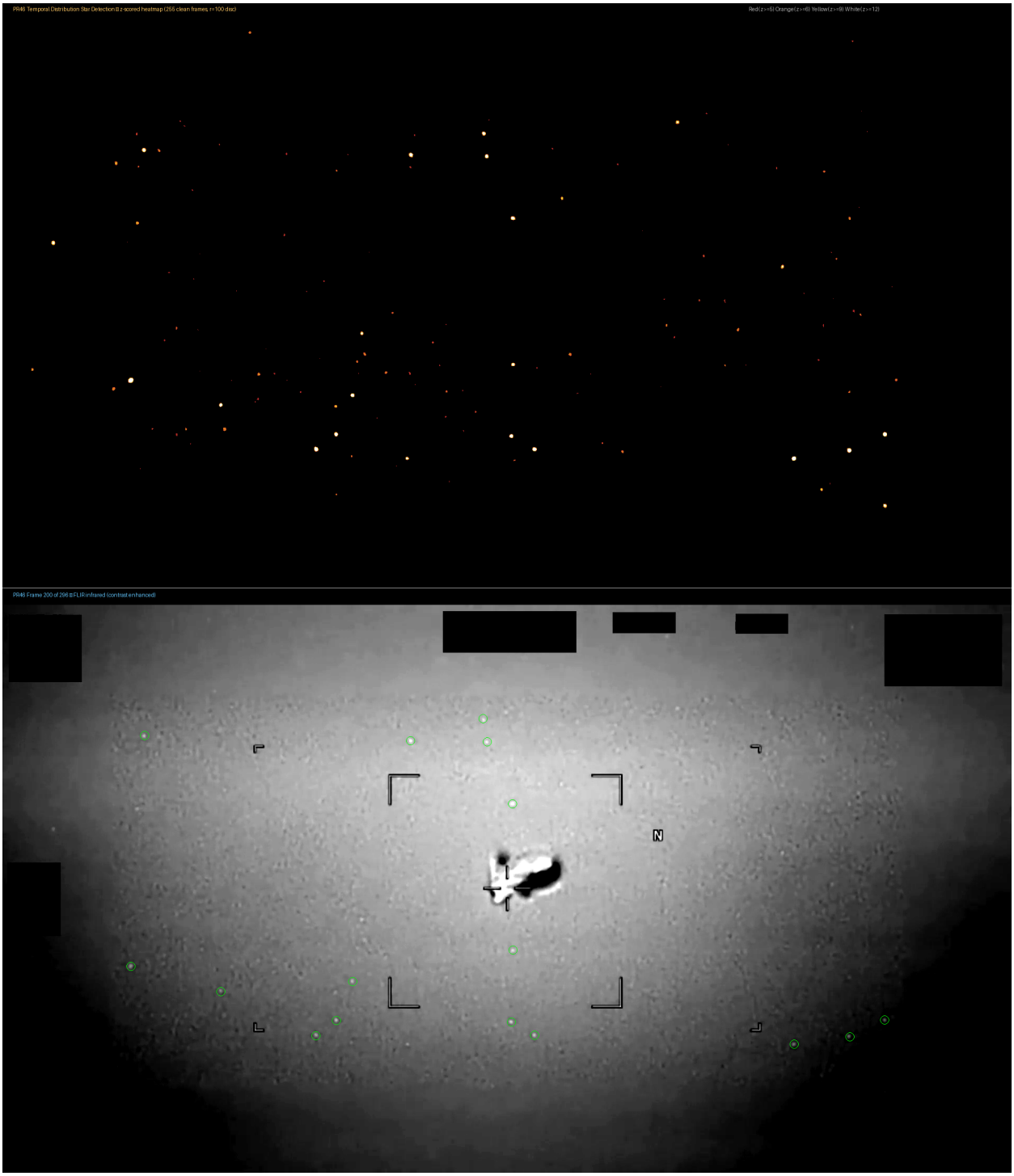


Figure 2: Top: temporal distribution heatmap from 296 aligned FLIR frames. Color encodes z-score: red ($z \geq 5.1$), orange ($z \geq 6$), yellow ($z \geq 9$), white ($z \geq 12$). Bottom: a single raw FLIR frame (frame 200 of 296, contrast enhanced) with green circles marking the 15 confirmed star positions. Individual frames show no discernible stars above the sensor noise; the temporal z-map recovers all 15 as persistent, locally anomalous point sources.

II. PSF-Matched Star Detection in Handheld Phone Photography

1 Introduction

The temporal distribution method of Part I requires many aligned frames. A complementary approach extracts the point spread function (PSF) from identified stars in a **single** long-exposure frame, then uses that PSF as a matched filter to detect every star in the image. This method was developed for handheld phone photographs of the night sky—4-second exposures at ISO 6400 with a consumer OnePlus Nord N20 5G, no tripod, no tracking mount.

The key insight is that the hand motion during a long exposure produces a characteristic **motion trace** that is identical for every point source in the frame. Stars, being point sources at infinity, all experience the same hand shake. The motion trace IS the PSF, and it can be extracted from the brightest stars, then used as a correlation kernel to find every other star.

2 PSF Extraction from Starlight

2.1 Star Identification

Stars are detected as local maxima in a Gaussian-smoothed ($\sigma = 3$) grayscale image, thresholded above background, with a roofline mask excluding the lower half of the frame. For the Orion field, 8 named stars were identified via a belt collinearity algorithm: three collinear, equally-spaced stars with spacing ratio > 0.96 and collinearity deviation < 0.003 uniquely identify Orion's belt.

2.2 Brightness-Binned PSF

Each identified star's neighborhood (radius $R = 75$ px) is extracted, background-subtracted (annular median), and centroid-aligned to sub-pixel precision. Stars are sorted by total disc brightness and grouped into brightness bins. A sliding window of 3 sequential stars produces PSFs at each brightness level.

The critical finding: **bright stars have wider PSFs than faint stars.** At high brightness, the sensor saturates and the signal blooms into adjacent pixels, inflating the apparent PSF. The medium-brightness stars (FWHM ≈ 13 px) give the truest motion trace; the faint stars (FWHM ≈ 7 px) are cleanest but noisiest. This brightness-dependent PSF variation is consistent across all sliding windows (all sequential triangle closures < 2 px), confirming that the shape is real and the alignment is correct.

2.3 Motion Trace Characterization

The extracted PSF (Figure 7) records the photographer's hand motion during the 4-second exposure with sub-pixel precision. Cross-section analysis reveals an asymmetric two-lobed structure: the hand paused briefly at two positions with motion between them, producing FWHM of 10 px (horizontal) $\times$ 14 px (vertical). The vertical axis shows more motion than the horizontal—consistent with the biomechanics of a standing handheld camera.

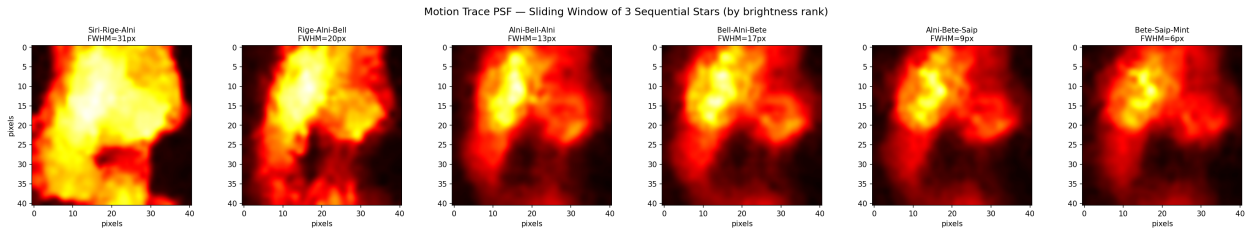


Figure 7: Sliding window PSF extraction from 8 identified Orion stars, sorted by brightness (left to right = bright to faint). The PSF shrinks as stars get fainter: bright stars bloom due to sensor saturation (FWHM = 31 px), medium stars show the true motion trace (FWHM = 13 px), faint stars are tightest but noisy (FWHM = 6 px). The shape is consistent across all windows.

3 Matched Filter Detection

The extracted PSF is used as a correlation kernel across the entire image. For each pixel position (x, y), the correlation with the PSF kernel h is computed via FFT:

$$C(x, y) = \mathcal{F}^{-1}\left\{\mathcal{F}\{I\} \cdot \mathcal{F}^{*}\{h\}\right\} \tag{8}$$

The correlation map is z-scored:

$$z(x, y) = \frac{C(x, y) - \bar{C}}{\sigma_C} \tag{9}$$

Stars are detected as local maxima above a threshold $z \geq z_{\min}$. The matched filter is the optimal detector for a known signal in Gaussian noise (the Neyman-Pearson lemma [16]); here the "known signal" is the PSF extracted from the data itself.

3.1 Results

Applied to a single 4-second handheld exposure of Orion (ISO 6400, $f/1.7$, 27 mm equivalent), the matched filter detects 23 sky stars at $z \geq 4$ and 29 at $z \geq 3$. The brightest detection—Sirius—reaches $z = 84.3\sigma$, confirming that the matched filter achieves extraordinary sensitivity from a single consumer phone photograph.

3.2 True-Hue False-Saturation Rendering

Detected stars are rendered as colored discs on a black background. The color is the **true hue** from the original image (the phone sensor captures real stellar color at ISO 6400), with saturation and value maximized ($S = V = 1$ in HSV space). Disc radius encodes detection significance via a log scale:

$$r = 3.90 \cdot \ln(z) - 1.29 \tag{10}$$

mapping $z = 3 \to r = 3$ px, $z = 4 \to r = 4$ px, and $z = 84 \to r = 16$ px. This visualization reveals stellar color differences invisible at natural brightness levels: Betelgeuse appears warm, Rigel and Sirius appear blue-white.

4 Plate Solving and Validation

The original 4-second photograph was submitted to astrometry.net [1] and successfully plate-solved: RA $= 81.35$, Dec $= -17.94$, confirming Orion. The pixel scale (59.5 arcsec/px) and field orientation (352.9) were recovered. This independently validates the star positions detected by our pipeline.

5 Wiener Deconvolution

With the PSF known, the entire image can be deconvolved via the Wiener filter:

$$\hat{F}(\omega) = \frac{F_{\text{img}}(\omega) \cdot H^*(\omega)}{|H(\omega)|^2 + \lambda} \tag{11}$$

where λ is a regularization parameter balancing sharpness against noise amplification.

A systematic lambda search across multiple images reveals that the deconvolution landscape is **structural**—the normalized curves for sky noise, star FWHM, ringing energy, and sharpness overlay almost perfectly across different exposures, pointings, and atmospheric conditions. The structural knee at $\log_{10}(\lambda) \approx -2.0$ is a universal Pareto optimum: a property of the Wiener method, not of any particular image. At $\lambda = 0.01$, the deconvolution resolves barn siding panels, power lines, and window frames from 1.6-second handheld exposures taken in darkness.

6 Connection to Part I

Parts I and II solve the same problem—detecting stars in degraded imagery—via complementary approaches:

	Part I (Temporal)	Part II (PSF Matched Filter)
Input	296 video frames	1 phone photograph
PSF	Not needed	Extracted from stars
Detection	Temporal CA-CFAR	Matched filter correlation
Best detection	29σ	84σ
Plate solved	No	Yes
Deconvolution	No	Yes (Wiener)

Table 2: Comparison of temporal and PSF-based star detection.

The temporal method requires many frames but no PSF model; the PSF method requires only one frame but needs identifiable stars to extract the PSF. Together they cover the space from multi-frame degraded video to single-frame long-exposure photography.

References

[1] Bertin, E. & Arnouts, S. (1996). SExtractor: Software for source extraction. **Astronomy & Astrophysics Supplement Series**, 117, 393–404.

[2] Stetson, P.B. (1987). DAOPHOT: A computer program for crowded-field stellar photometry. **PASP**, 99, 191–222.

[3] Fried, D.L. (1978). Probability of getting a lucky short-exposure image through turbulence. **JOSA**, 68(12), 1651–1658.

[4] Law, N.M., Mackay, C.D. & Baldwin, J.E. (2006). Lucky imaging: High angular resolution imaging in the visible from the ground. **Astronomy & Astrophysics**, 446, 739–745.

[5] Fruchter, A.S. & Hook, R.N. (2002). Drizzle: A method for the linear reconstruction of undersampled images. **PASP**, 114, 144–152.

[6] Finn, H.M. & Johnson, R.S. (1968). Adaptive detection mode with threshold control as a function of spatially sampled clutter-level estimates. **RCA Review**, 29, 414–464.

[7] Rohling, H. (1983). Radar CFAR thresholding in clutter and multiple target situations. **IEEE Trans. Aerospace and Electronic Systems**, AES-19(4), 608–621.

[8] Gandhi, P.P. & Kassam, S.A. (1988). Analysis of CFAR processors in nonhomogeneous background. **IEEE Trans. Aerospace and Electronic Systems**, 24(4), 427–445.

[9] Kay, S.M. (1998). **Fundamentals of Statistical Signal Processing, Vol. II: Detection Theory**. Prentice Hall.

[10] North, D.O. (1963). An analysis of the factors which determine signal/noise discrimination in pulsed-carrier systems. **Proc. IEEE**, 51(7), 1016–1027.

[11] Barniv, Y. (1985). Dynamic programming solution for detecting dim moving targets. **IEEE Trans. Aerospace and Electronic Systems**, AES-21(1), 144–156.

[12] Deng, H. et al. (2016). Infrared small target detection based on modified STLC. **Infrared Physics & Technology**, 76, 504–513.

[13] Irani, M. & Peleg, S. (1991). Improving resolution by image registration. **CVGIP: Graphical Models and Image Processing**, 53(3), 231–239.

[14] Irwin, M.J. (1985). Automatic analysis of crowded fields. **MNRAS**, 214, 575–604.

[15] Starck, J.L. & Murtagh, F. (2002). **Astronomical Image and Data Analysis**. Springer.

[16] Neyman, J. & Pearson, E.S. (1933). On the problem of the most efficient tests of statistical hypotheses. **Phil. Trans. Royal Soc. A**, 231, 289–337.

[17] Department of Defense (2021). PURSUE: Public Unclassified Release of Selected UAP Evidence. Defense Visual Information Distribution Service (DIVIDS), `https://www.dvidshub.net`. Note: the source video was obtained via the non-authenticated public interface. The authors declined to register with DIVIDS due to concerns about the End User License Agreement, which asserts jurisdiction over "attached devices" of registered users in terms that could be read as authorizing warrantless inspection of user equipment. The public data used in this work requires no authentication and carries no such terms.