A Reproducible Pipeline for JunoCam Limb Analysis and the Detection of Detached Jovian Hazes as a Function of Planetocentric Latitude

Author: *Kevin J. Kelly, MFA*
Affiliations: *PCC, ASU, NASA Jet Propulsion Laboratory, Planetary Science Group #3222*
Team Lead: *Dr. Glenn Orton*
Date: January 21, 2026

Abstract

We present a reproducible, directory-agnostic, stage-aligned data-processing pipeline for the detection and classification of detached haze layers in Jupiter's atmosphere using images from the Juno spacecraft's visible-light camera, JunoCam. The pipeline executes all processing stages—from raw IMG ingestion and SPICE initialization through limb tracing, geometric rectification, brightness-gradient analysis, and fragment-level classification—using modular, version-pinned software designed for independent replication and extension.

Stages 1–7 define a conservative, fragment-based analysis pipeline operating on a single JunoCam image, emphasizing geometric integrity and statistical robustness over aggressive feature localization. Detached haze occurrence is assessed using brightness-gradient structure along the planetary limb and aggregated as a function of first-available SPICE planetocentric latitude, reporting detectability rather than atmospheric non-existence where geometric constraints preclude precise localization.

In Stage 8, the complete pipeline is applied without modification across all JunoCam images acquired during a single perijove (Perijove 14), producing the first fragment-weighted, Perijove-scale measurements of detached haze occurrence as a function of latitude and longitude. The Perijove-scale results reveal a pronounced enhancement in detached haze detectability at mid-latitudes, peaking near 30–35° planetocentric, consistently observed across multiple images. Longitude–latitude mapping further indicates spatial clustering under the perijove's sampling geometry, while explicitly preserving coverage limitations.

Together, these results demonstrate that detached haze detectability during Perijove 14 is not uniformly distributed along the limb and exhibits a robust latitude dependence that emerges only through statistically conservative aggregation across many independent limb fragments.

Introduction

Jupiter's upper-atmospheric haze plays an important role in the planet's radiative balance, photochemistry, and large-scale circulation. Detached haze layers—optically thin scattering features residing above the main cloud deck—have been observed intermittently in limb-viewing geometry and are thought to reflect latitude-dependent

atmospheric processes. However, their occurrence and spatial distribution remain difficult to quantify due to the geometric constraints of limb observations and the complexity of spacecraft imaging data.

NASA's Juno spacecraft provides repeated high-resolution limb views of Jupiter through its visible-light camera, JunoCam, across many perijove passes. These observations contain information about vertical atmospheric structure, but extracting that information reliably requires accurate geometric context, consistent sampling geometry, and careful handling of cases where limb crossings are ambiguous or poorly constrained.

Previous work (Kelly 2022) demonstrated that detached haze layers could be identified in JunoCam data by combining limb tracing with SPICE-derived geometry and brightness-profile inspection. While effective for exploratory analysis, such approaches are difficult to reproduce or scale and rely heavily on manual intervention. The present work replaces that workflow with a fully programmatic, stage-aligned pipeline that makes each analytical decision explicit, reproducible, and independently auditable.

This study introduces a modular, directory-agnostic JunoCam limb-analysis pipeline designed to produce conservative measurements of detached haze detectability as a function of planetocentric latitude. The pipeline prioritizes geometric integrity and statistical robustness over aggressive feature localization, preserving atmospheric structure while avoiding over-interpretation in regions where SPICE geometry is undefined. By aggregating fragment-level detections statistically across images rather than over-localizing individual features, the method enables robust comparison across framelets and imaging geometries while remaining explicit about the limitations imposed by limb viewing.

Methods: Pipeline Overview

All processing and analysis stages are executed on a per-IMG basis. For a given JunoCam image, all framelets, limb fragments, diagnostic plots, and statistical summaries are generated within IMG-specific directories. This design ensures full reproducibility, prevents output collisions when processing multiple images, and allows cross-image aggregation to be performed as a distinct analytical step outside the scope of this study.

Stages 1–7 operate on a single JunoCam image and define a complete, reproducible analysis pipeline; in Stage 8, this identical pipeline is executed independently on all images from a single perijove, and the resulting fragment-level outputs are aggregated in a distinct Perijove-scale analysis.

The haze classification pipeline consists of a sequence of modular Python stages, each of which consumes the outputs of the preceding stage while remaining independently executable for diagnostic or exploratory analysis.

Pipeline Structure and Stage Alignment

The pipeline is explicitly organized into numbered stages (Stage 01–Stage 07), each corresponding to exactly one Python module. No stage relies on logic described in later stages, and no section contains forward references. This structure is intentional and enforces reproducibility, traceability, and modular reuse.

Stage	Module	Description	Primary Output
01	framelets_and_spice.py	Ingests raw JunoCam IMG and LBL products, converts them to ISIS-compatible .CUB files, and attaches full SPICE geometry for subsequent analysis.	SPICE-initialized .CUB files
02	trace_limb_polyline.py	Detects and traces the planetary limb using gradient-based edge detection, producing continuous polyline representation of Jupiter's limb curvature.	Limb coordinate sets (.csv), visual verification (.png)
03	rectify_limb.py	Rectifies limb segments into a linearized reference frame and extracts perpendicular brightness cross-sections spanning the atmospheric limb.	Rectified limb images (.tiff) and cross-section data (.csv)
04	plot_perpendiculars.py	Computes brightness gradients along perpendicular limb fragment cross-sections and prepares intermediate diagnostic outputs.	Intermediate diagnostic plots and exploratory analysis outputs (.csv)
05	graph_profiles.py	Generates diagnostic plots of rectified brightness profiles.	Diagnostic profile graphs (.png) for visual verification
06	peakfinder.py	Identifies primary limb peaks and secondary detached-haze brightness gradients along rectified limb profiles.	Annotated limb-fragment classification tables (.csv) with peak locations, amplitudes, and haze flags
07	haze_analysis.py	Aggregates fragment-level classification across all framelets and bins detections by first-available SPICE latitude to quantify detached haze as a function of latitude	Latitude-binned occurrence tables (.csv) and summary analysis figures (.png)

Note well: All terminal commands are executed from the project's root directory.

Stage 0 — Computational Environment (Assumed)

This pipeline assumes a working installation of ISIS3 and locally available SPICE kernels. Installation, operating-system configuration, and kernel acquisition are treated as prerequisites and are not addressed here.

To ensure numerical and geometric reproducibility, we provide a fully pinned conda environment named **JHC (Juno Haze Classifier)** that reproduces the exact computational context used in this study, including Python version, third-party libraries, and the ISIS runtime bindings.

All SPICE geometry is resolved using locally synchronized kernels rather than remote services, ensuring deterministic limb geometry and reproducibility independent of external servers. The specific kernels attached to each image cube are recorded verbatim in per-cube manifest files, providing a complete provenance record for every geometric solution used in downstream analysis. From this point forward, all pipeline commands are executed within the activated JHC environment in the root directory. No further environment configuration occurs within the pipeline itself. All scientific decisions occur downstream of this assumption.

An additional Jupiter body-code kernel was supplied via the SPICE `extra` option to ensure consistent body-ID resolution during offline processing.

The directory tree we use for this pipeline is as follows:

Stage 0.1 Directory Contract

```
JHC (project_root)
└── src/
    └── python modules
└── data/
    └── IMG_NAME/
        ├── raw/
        │   ├── IMG_NAME.IMG
        │   ├── IMG_NAME.LBL
        ├── cub/
        │   ├── stage_01_framelets/
        │   ├── stage_02_traces/
        │   ├── stage_03_rectified/
        │   ├── stage_04_perp_samples/
        │   ├── stage_05_graphs/
        │   ├── stage_06_peaks/
        └── analysis/
            └── stage_07/
```

This pipeline is directory-agnostic; however, once an IMG root directory is defined, this internal structure is assumed. All stages read only from their preceding stage and write exclusively to IMG-scoped output directories.

Stage 1 — Data Ingestion and SPICE Initialization

1.1 Data Acquisition from the Planetary Data System

Raw JunoCam images are publicly available through NASA's Planetary Data System (PDS). Each image is distributed as an .IMG file accompanied by a corresponding .LBL metadata label, which contains spacecraft position, timing, and geometric information required for ISIS ingestion.

In this study, the pipeline follows Brueshaber (2021) and is demonstrated using JunoCam image `JNCR_2018197_14C00024_v02.IMG`, acquired during Perijove 14 on 16 July 2018. This image captures Jupiter's northern limb and serves as a concrete, reproducible

example throughout all subsequent stages. Users may substitute other JunoCam images following the same directory and labeling conventions.

1.2 Ingestion into ISIS

Once the raw data are available locally, each .IMG file is ingested into ISIS as a .CUB, and full SPICE geometry is attached to the resulting image products. In this pipeline, ingestion is performed by executing `01.framelets_and_spice.py`, which processes each JunoCam .IMG/.LBL pair, breaks the raw image into its constituent framelets, and attaches locally cached SPICE kernels to each framelet cube.

For reproducibility, the ingestion step records a per-framelet manifest that enumerates every SPICE kernel attached to that cube, ensuring that the complete geometric context of each product is preserved and auditable.

1.3 Execution

Executing Stage 1 ingests the raw JunoCam IMG/LBL pair, converts it to ISIS-compatible framelets, attaches full SPICE geometry, and records a manifest of all kernels used for each cube. To execute, run this terminal command from the environment's root directory:

```
python src/01.framelets_and_spice.py \
  --img data/JNCR_2018197_14C00024_v02/raw/JNCR_2018197_14C00024_v02.IMG \
  --outdir data/JNCR_2018197_14C00024_v02/cub/stage_01_framelets
```

Successful execution produces a set of SPICE-initialized .cub framelets and accompanying manifest files under `stage_01_framelets/`. Once these outputs are present, the pipeline proceeds to Stage 2: Limb Detection and Tracing.

Stage 2 — Limb Detection and Tracing

With SPICE geometry successfully attached, the next stage identifies Jupiter's planetary limb in each JunoCam framelet. The limb is defined as the boundary separating Jupiter's illuminated atmosphere from surrounding space and represents the geometric foundation for all subsequent limb-normal sampling and brightness-gradient analysis.

Limb detection is performed by the module `02.trace_limb_polyline.py`, which combines SPICE-derived camera geometry with image-based edge detection to produce a continuous, subpixel-resolution polyline describing the curvature of the visible atmospheric boundary in each framelet.

Each SPICE-attached framelet (.cub) is exported as a 16-bit TIFF (science fidelity) and an 8-bit, linearly stretched PNG (for quick-look reference). The gradient-based search seeds a starting limb-point and iteratively proposes subsequent points along the edge by maximizing brightness differential across a small angular neighborhood. The resulting (x,y) pixel coordinates of the polyline are saved as `<cub>_LIMBENDPOINTS.csv`, while an overlay with cyan limb points is saved as `<cub>_OVERLAY.png` in the same directory.

The seeding step no longer assumes a fixed limb location. Instead, the algorithm identifies the global maximum of the Sobel gradient magnitude within each framelet, providing orbit-agnostic initialization and ensuring compatibility with any JunoCam observation geometry. This update replaces the earlier orientation-specific heuristic (cf. Wen, 2021) and improves cross-orbit generalization, detecting both upper and lower limb fragments even in oblique lighting geometries.

2.1 Processing Scope and Assumptions

All limb tracing and rectification steps are performed on a per-framelet basis under fixed geometric assumptions. Sampling density, smoothing parameters, and perpendicular construction are held constant across all framelets within a given IMG to ensure comparability. No framelet-level tuning is performed, and all parameters are documented explicitly to support reproducibility.

2.2 Execution

```
python src/02.trace_limb_polyline.py \
  --cubdir data/JNCR_2018197_14C00024_v02/cub/stage_01_framelets \
  --outdir  data/JNCR_2018197_14C00024_v02/cub/stage_02_trace_polyline \
  --noshow
```

Executing Stage 2 automates limb detection across SPICE-initialized framelets and writes the resulting limb polylines and verification overlays to IMG-scoped output directories. These artifacts constitute the sole geometric inputs to the limb-rectification process performed in Stage 3 and define the valid sampling boundary for all subsequent analysis.

Figure 2.2: Limb Detection Across Successive Framelets

2.3 Mandatory Manual Cleaning (Non-Optional)

Certain JunoCam framelets do not contain a usable limb–space transition, either because Jupiter fully occupies the frame or because the image boundaries are dominated by noise and blur. These framelets must be removed manually prior to downstream processing.

These framelets contain no usable limb–space transition. Retaining them artificially inflates null detections and suppresses true atmospheric structure.

Users must therefore (1) delete spurious limb traces at noisy framelet edges, and (2) remove entire framelets in cases where Jupiter's disc fills the entire frame and therefore contains no visible limb-space to rectify and sample. Figure 2.3 represents one such framelet to illustrate the point. This intervention is deliberate and scientifically necessary, not a pipeline failure, and must be completed before proceeding to Stage 3.

Figure 2.3 — False Positive: JNCR_2018197_14C00024_V02_BLUE_0028

Stage 3 — Limb Rectification and Perpendicular Sampling Geometry

The limb traces produced in Stage 2 define the precise two-dimensional boundary between Jupiter's illuminated atmosphere and surrounding space. However, the curvature of this boundary varies across each framelet, making direct comparison of brightness profiles difficult in raw image coordinates.

Stage 3 transforms each detected limb segment into a limb-aligned coordinate system in which the curved atmospheric boundary is straightened and placed into a common reference frame. This rectification enables consistent, repeatable sampling of brightness variations perpendicular to the limb, corresponding to changes in scattering with altitude.

3.1 Limb-Aligned Rectification

Each framelet's limb polyline is resampled at regular intervals and locally approximated as a smooth curve. Using this curve, the original image is re-projected into a rectified coordinate system in which the limb appears as a straight horizontal feature (Figure 3.1). In this rectified image, horizontal position corresponds to distance along the limb, while vertical position corresponds to distance normal to the atmospheric boundary.

Rectification is performed independently for each framelet and restricted to the stable limb region, excluding end segments where the tangent orientation becomes ill-defined near image boundaries. This resulting rectified product is a narrow strip centered on the limb that preserves the local photometric structure across the atmosphere–space interface.

The limb-aligned rectification strategy employed here follows the general methodology described by Eichstädt (2020), who introduced image-space rectification techniques for analyzing Jupiter’s limb structure in JunoCam observations. We have adapted this process for fully automated batch processing and reversible coordinate mapping.

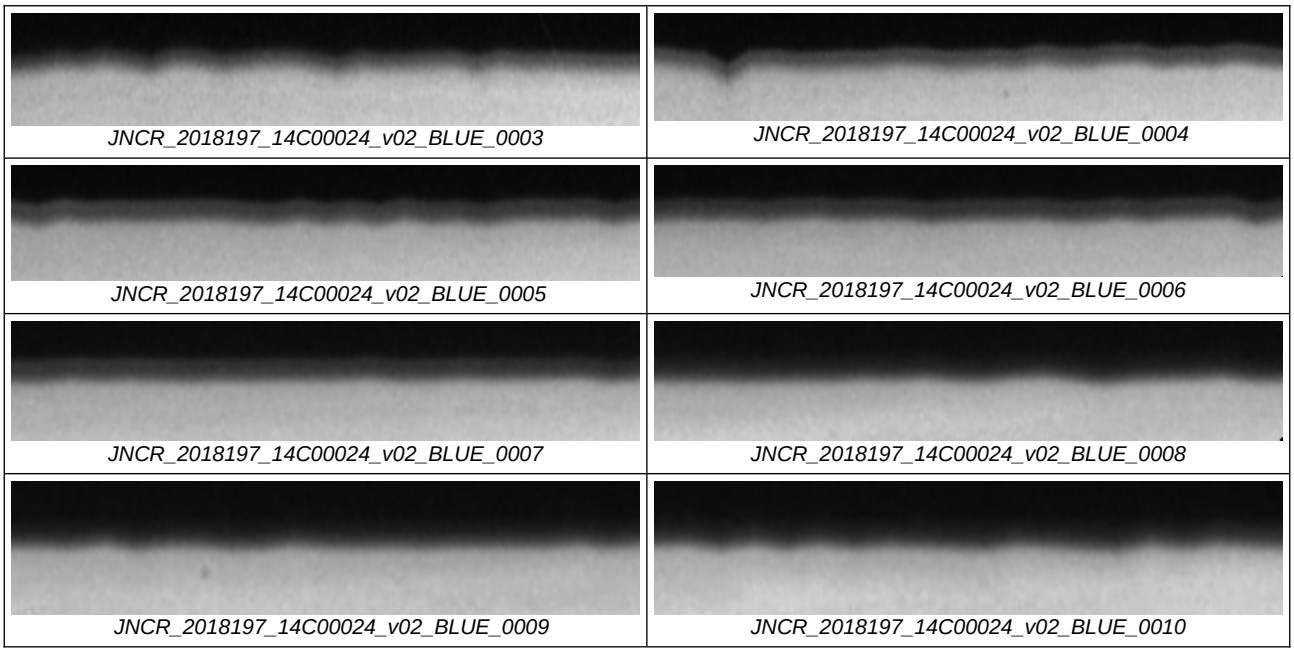


Figure 3.1 illustrates the geometric rectification process that transforms the curved limb into a flat, horizontally aligned reference frame, isolating the usable limb segment for brightness-altitude analysis.

3.2 Reversible Mapping Between Rectified & Original Coordinates

The rectification process is implemented as a fully reversible geometric transformation. For every pixel in the rectified image, the corresponding pixel coordinates in the original ISIS-calibrated framelet are recorded and stored in an explicit mapping table.

This rectified-to-original coordinate map ensures that any measurement performed in rectified space can be traced directly back to the original image pixels and associated SPICE geometry. As a result, no scientific interpretation relies solely on transformed imagery; all derived quantities remain physically anchored to the source observation.

For each framelet, Stage 3 produces:

- a 16-bit rectified limb image (*_RECTIFIED.tif) used for all quantitative sampling,
- an 8-bit visualization image (*_RECTIFIED.png) for diagnostic and publication figures, and
- a dense rectified-to-original coordinate mapping table (*_RECTIFIED_MAPPING.csv).

These products form the terminal outputs of Stage 3 and serve as the sole inputs to subsequent brightness-profile analysis.

3.3 Execution

Executing Stage 3 performs limb-aligned rectification for all validated limb traces produced in Stage 2. Each curved limb segment is transformed into a straightened reference frame centered on the atmospheric boundary (Figure 3.1), enabling consistent limb-normal sampling in downstream stages. This rectification step is required before any brightness-profile construction or scientific interpretation can occur. To achieve these results, execute the following terminal command from the project's root directory:

```
python src/03.rectify_limb.py \
  --indir data/JNCR_2018197_14C00024_v02/cub/stage_02_trace_polyline \
  --imagedir data/JNCR_2018197_14C00024_v02/cub/stage_01_framelets \
  --outdir data/JNCR_2018197_14C00024_v02/cub/stage_03_rectified
```

Successful execution produces limb-aligned rectified image strips, a visualization PNG, and a reversible mapping table linking rectified coordinates to original image pixels and SPICE geometry. These products define a stable geometric reference for all subsequent brightness sampling and analysis and constitute the sole valid inputs to Stage 4.

Users should verify that the rectified limb appears horizontally aligned and centered within each output image before proceeding. If rectified outputs are absent or malformed, downstream analysis must not be attempted.

Stage 4 — Perpendicular Sampling and Brightness Profile Construction

4.1 Perpendicular Brightness Sampling Geometry

With the limb geometry straightened, sampling paths perpendicular to the atmospheric boundary are defined directly in rectified space. Sampling lines are placed at uniform horizontal intervals along the rectified limb and extend vertically across the limb transition, probing from the dark background of space into Jupiter's upper atmosphere.

For each sampling position, two limb edges are identified in rectified coordinates, and their midpoint defines the local limb's centerline. From this midpoint, a vertical sampling path is constructed that spans a fixed number of pixels both outward into space and inward toward the planet. These rectified limb fragments are parallel, evenly spaced, and orthogonal to the limb by construction, independent of the original viewing geometry.

Each sampled rectified pixel is mapped back to its original framelet coordinates using the rectified-to-original mapping table, and its corresponding 16-bit pixel value is retrieved from the ISIS-calibrated TIFF image. For each framelet, a single consolidated table is produced containing rectified coordinates, original image coordinates, and pixel intensity values for every sampled location.

In addition to these numerical outputs, a diagnostic overlay image is generated for each rectified framelet (Figure 4.1), displaying the rectified limb strip with detected limb edges

and perpendicular sampling paths superimposed. These overlays serve both as validation artifacts and as publication figures illustrating the sampling geometry.

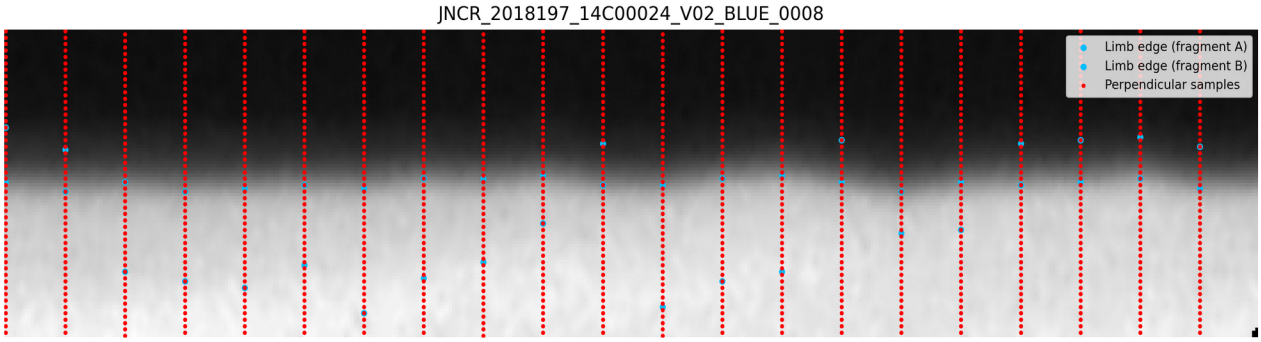


Figure 4.1. Rectified limb image with detected limb edges (blue) and perpendicular brightness sampling paths (red).

4.2 Association of Perpendicular Samples with SPICE Geometry

To associate each sampled pixel with physical planetary coordinates, spacecraft geometry is retrieved using the SPICE kernel system through the ISIS `campt` tool. For each sampled pixel location, SPICE is queried using the original unrectified image geometry, ensuring that physical coordinates are derived directly from calibrated spacecraft pointing and timing information.

For every sampling point along each perpendicular brightness profile, ancillary metadata—including planetocentric latitude, planetocentric longitude, spacecraft altitude, emission angle, incidence angle, phase angle, and slant distance—are retrieved and appended to the sampling records. These parameters are stored alongside the pixel brightness values without altering the sampling order established in rectified space.

This approach decouples geometric sampling from physical interpretation: brightness profiles are constructed and ordered according to their limb-normal distance in rectified image coordinates, while each sample is independently associated with its original corresponding physical location on Jupiter. As a result, brightness-gradient analysis and peak detection are performed in a stable geometric coordinate system, while all reported results can be expressed in physically meaningful planetary coordinates.

4.4 Execution

Executing Stage 4 constructs perpendicular brightness sampling paths in rectified space, maps each sampled pixel back to original image coordinates, and associates each sample with SPICE-derived physical geometry.

```
python src/04.plot_rectified_perpendiculars.py \
  --indir data/JNCR_2018197_14C00024_v02/cub/stage_03_rectified \
  --framelet-dir data/JNCR_2018197_14C00024_v02/cub/stage_01_framelets \
  --outdir data/JNCR_2018197_14C00024_v02/cub/stage_04_perp_samples \
  --batch
```

At the conclusion of Stage 4, each valid perpendicular brightness profile consists of a sequence of pixel samples ordered by limb-normal distance and annotated with full

SPICE-derived physical geometry. These profiles provide the foundation for brightness-gradient analysis, detached haze detection, and subsequent statistical analysis as a function of planetocentric latitude. They constitute the sole inputs to downstream diagnostic and classification stages and are written to IMG-scoped output directories for reproducibility

Stage 5 — Brightness-Gradient Visualization

Stage 5 constitutes the core analytical step of the Juno Haze Classification System, transforming geometrically rectified, SPICE-annotated limb samples into physically interpretable diagnostics for the detection and classification of detached haze layers. At this stage, the pipeline transitions from purely geometric processing to scientific interpretation, while remaining fully algorithmic, reproducible, and conservative with respect to geometric limitations near the limb. At this stage, profiles are visualized and prepared for analysis, but no classification decisions are made; all detection and labeling logic is deferred explicitly to Stage 6.

5.1 Construction of Limb-Normal Brightness-Gradient Profiles

For each rectified limb fragment produced in Stage 4, brightness is sampled along paths oriented perpendicular to the limb, yielding one-dimensional brightness profiles ordered by limb-normal distance (dy). These profiles probe the vertical structure of Jupiter’s upper atmosphere, extending from optically thin space, across the tangent limb, and into the upper cloud deck.

Raw pixel intensities are extracted from calibrated ISIS products and assembled into ordered brightness sequences. To suppress slowly varying background illumination and emphasize vertical structure, first-order brightness gradients (ΔBrightness) are computed along each profile as a diagnostic representation of vertical scattering structure. This transformation sharpens transitions associated with atmospheric boundaries and highlights secondary scattering layers above the cloud deck.

Internally, all profiles are constructed and analyzed strictly in dy-ordered space. This preserves uniform sampling and avoids ambiguities introduced by nonuniform physical spacing in latitude or slant distance. Where available, SPICE-derived physical coordinates (planetocentric latitude, longitude, slant distance, emission angle) are attached to each sampling point without altering the dy-based ordering.

For visualization purposes only, profiles may be plotted with the horizontal axis labeled in planetocentric latitude where SPICE geometry is available. Importantly, this relabeling does not alter the underlying index-space ordering used for peak detection or classification. The plotted latitude axis therefore serves as a physical annotation rather than a computational coordinate.

5.2 Smoothing and Noise Suppression

Brightness-gradient profiles often exhibit pixel-scale noise arising from photon statistics, local albedo variations, and residual detector artifacts. To suppress this noise while preserving physically meaningful extrema, all ΔBrightness profiles are smoothed using a Savitzky–Golay filter (quadratic polynomial, window size of 5 samples).

This window size represents the minimum smoothing scale capable of removing high-frequency noise without suppressing secondary maxima or shoulder-like features associated with detached haze layers. The smoothing parameters are fixed and applied uniformly across all profiles. No adaptive or profile-specific tuning is performed.

Both raw and smoothed profiles are retained. Raw profiles are preserved for archival integrity, sanity checks, and archival transparency, while smoothed profiles are used exclusively for peak detection and classification. Because detached haze signatures manifest as coherent extrema across multiple samples, the applied smoothing suppresses pixel-scale noise without erasing physically meaningful secondary structure.

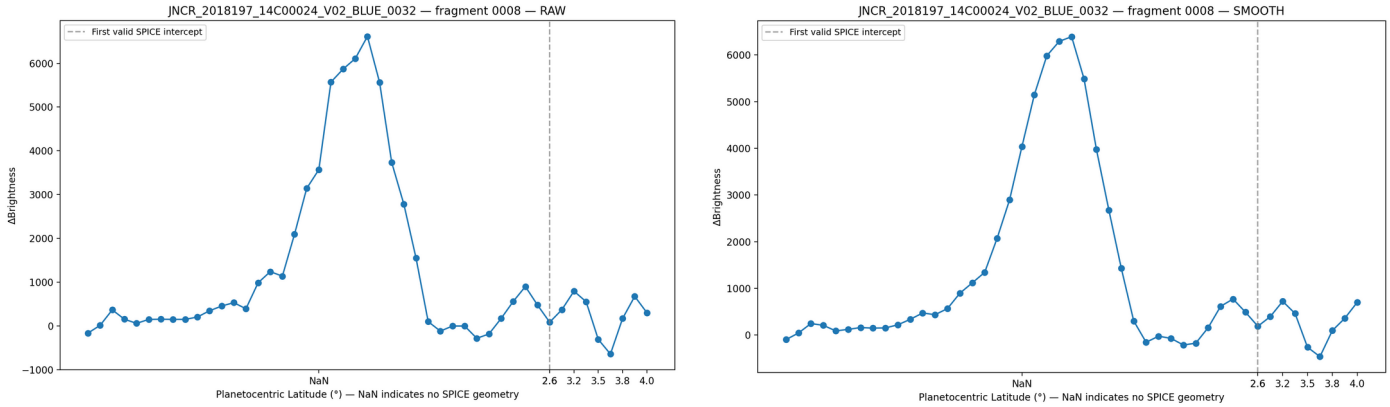


Figure 5.2 — Raw & Smoothed ΔBrightness Profiles illustrating noise suppression along the Jovian limb

5.3 Execution

Executing Stage 5 computes limb-normal brightness-gradient profiles for all SPICE-annotated perpendicular samples and generates diagnostic visualizations used to validate profile structure, smoothing behavior, and feature preservation prior to automated classification.

```
python src/05.graph_profiles.py \
  --indir data/JNCR_2018197_14C00024_v02/cub/stage_04_perp_samples \
  --outdir data/JNCR_2018197_14C00024_v02/cub/stage_05_graphs \
  --batch \
  --smooth 5
```

Stage 5 writes per-fragment ΔBrightness profile plots and associated diagnostic data to IMG-scoped output directories. These products provide visual confirmation of gradient structure and smoothing behavior but do not encode any detection or classification decisions. They constitute the sole inputs to Stage 6, where automated peak detection and detached haze classification are performed.

Stage 6 — Gradient Analysis and Peak-Finding

Stage 6 replaces earlier visualization-first workflows with a unified, physics-aware peak-detection system implemented in a single module (`06.peakfinder.py`) applied uniformly across all limb fragments. Diagnostic plots are generated directly from the same logic that performs classification, ensuring that all visualizations reflect the true decision path of the algorithm.

6.1 Peak Detection Logic and Physical Constraints

Stage 6 operates exclusively on the smoothed ΔBrightness(dy) profiles produced in Stage 5 and does not revisit geometric sampling or SPICE association logic established in earlier stages. Automated peak detection is performed on the smoothed ΔBrightness(dy) profiles using a constrained local-extrema search. The algorithm identifies candidate local maxima and classifies them according to physically motivated rules rather than purely mathematical prominence.

The dominant limb peak—corresponding to the sharp brightness transition at the atmospheric boundary—is identified as the **primary** peak. Candidate secondary peaks are only evaluated as potential detached haze signatures if they satisfy the following constraints:

1. **Outboard Requirement**: Detached haze features must occur on the spaceward (outboard) side of the primary limb peak in limb-normal (dy) sampling. Any extrema occurring at or interior to the primary limb peak are excluded. This is a known and defensible geometric limitation inherent to limb-normal sampling and is consistent with the physical expectation that detached hazes reside above the cloud deck, between the cloud tops and the tangent limb.
2. **Separation Constraints**: Secondary peaks must be separated from the primary peak by a minimum and maximum distance in dy index space, eliminating spurious noise near the limb transition and irrelevant interior structure.
3. **Near-Limb Atmospheric Constraint**: Secondary candidates must lie within the outer portion of the atmospheric column sampled. Features occurring too deep in the interior are rejected as unlikely to represent detached haze.
4. **Absolute and Relative Brightness Thresholds**: Candidate secondary peaks must exceed both an absolute ΔBrightness floor and a fractional threshold relative to the primary peak. This combination suppresses low-amplitude noise while retaining physically plausible weak haze layers.
5. **Valley (Shoulder) Test**: To prevent misclassification of post-limb decay shoulders or monotonic slopes, each secondary candidate must exhibit a sufficient local valley between itself and the primary peak. This allows the algorithm to accept both resolved secondary maxima and physically meaningful shoulder-like features while rejecting simple brightness decay.

Among candidates that pass all constraints, the strongest secondary feature is selected as the detached haze indicator for that limb fragment.

6.2 Geometric Limitations and SPICE Association

A critical geometric limitation arises near the tangent limb: not all detected atmospheric structure corresponds to a unique or stable intercept with Jupiter's reference shape model. In these regions, SPICE geometry cannot assign a reliable planetocentric latitude, even though optically detectable haze structure is present.

As a result, none of the detected peaks—primary or secondary—necessarily coincide with sampling points that have valid SPICE latitude solutions. This is not a failure of the pipeline and not a coding error. It is an inherent consequence of limb-viewing geometry, in which the line of sight intersects extended atmospheric structure without intersecting the modeled planetary surface.

To preserve scientific integrity, the pipeline does not interpolate, extrapolate, or infer latitude values for such detections. Instead, haze detections are explicitly labeled as either:

- **SPICE-resolved:** Secondary features whose sampling points possess valid planetocentric latitude values.
- **Optical-only:** Secondary features detected in ΔBrightness profiles that occur prior to the first valid SPICE surface intercept.

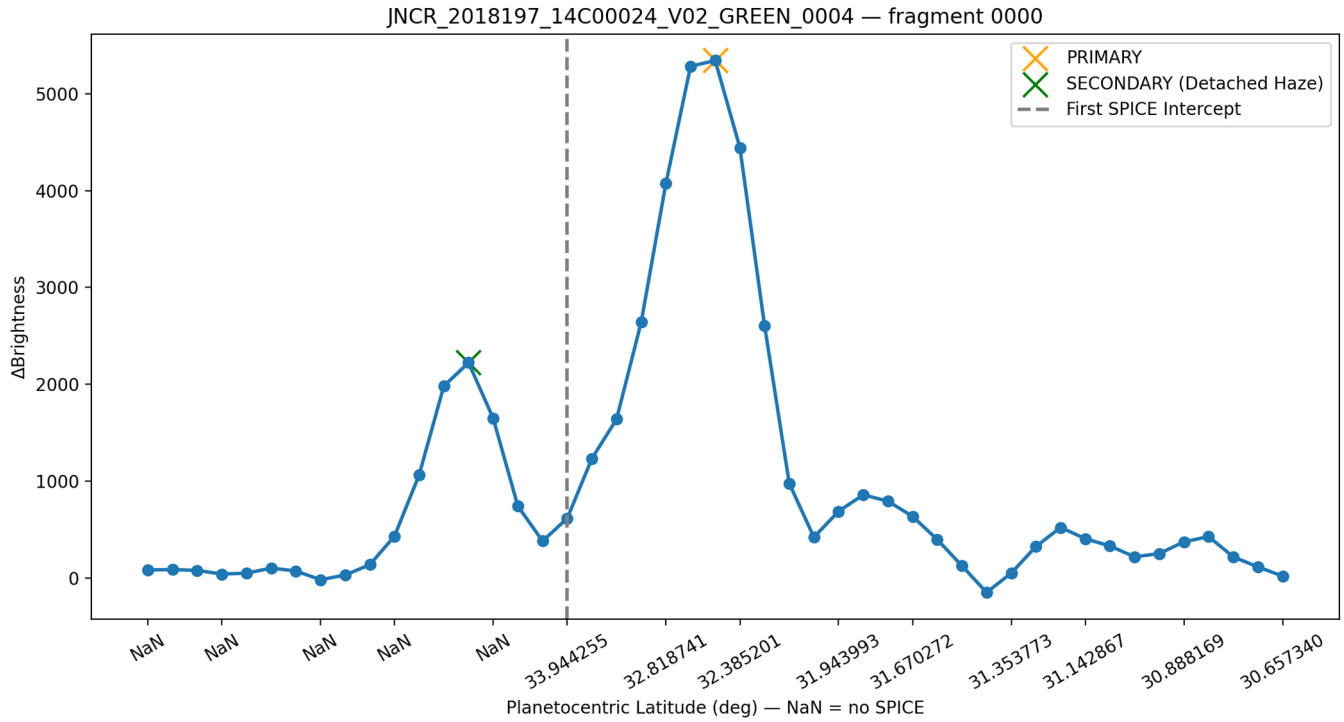
A vertical marker indicating the first valid SPICE intercept is included in all diagnostic plots. ΔBrightness structure occurring prior to this intercept represents optically detected haze for which no planetocentric latitude can be assigned. These detections are preserved rather than interpolated.

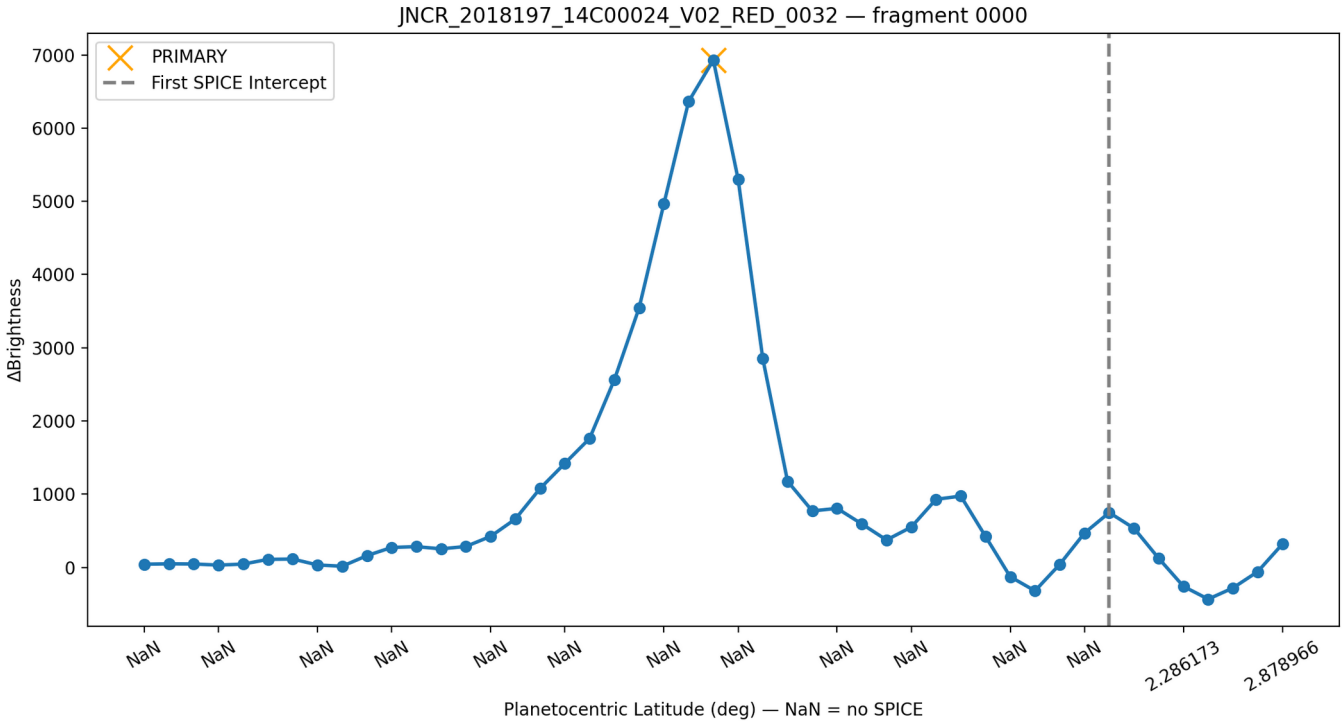
6.3 Visualization and Diagnostic Outputs

For each limb fragment, a diagnostic plot is generated showing the smoothed ΔBrightness profile, the identified primary limb peak, any detected secondary haze peak, and the location of the first valid SPICE intercept. Figure 6.3 presents representative examples illustrating both profiles: one with a single dominant limb peak and no detected secondary brightness gradient, and another in which secondary peaks are detected.

Profiles exhibiting a single dominant brightness-gradient peak are interpreted as sampling a continuous cloud deck with no resolved detached haze layer. Profiles exhibiting a secondary peak prior to the first SPICE surface intercept indicate optically detected atmospheric structure occurring above the visible limb, consistent with a detached haze morphology. These diagnostic distinctions motivate the statistical aggregation performed in Stage 7.

In all panels, orange markers denote the PRIMARY limb peak, while green markers denote SECONDARY peaks interpreted as candidate detached haze features. The vertical dashed line indicates the first valid SPICE surface intercept. Brightness-gradient structures occurring prior to this intercept correspond to optically detected atmospheric features for which no unique SPICE geometry can be assigned at the detected peak location.





Figures 6.3: Representative limb-normal brightness-gradient profiles illustrating single-peak and double-structure classifications.

6.4 Known Failure Modes and Algorithmic Scope

The peak-finding algorithm operates exclusively on brightness-gradient structure and geometric constraints. It has no intrinsic semantic understanding of cloud decks versus detached haze layers. In rare cases, a detached haze peak may be incorrectly selected as the primary peak when it rivals the cloud deck in ΔBrightness amplitude and occurs earlier along the sampling direction.

Such cases arise when:

- Two peaks have comparable prominence.

- The valley between them is shallow.
- Noise and smoothing subtly favor one feature.

Humans readily recognize these cases as a cloud deck plus detached haze system; the algorithm requires explicit geometric and SPICE-based rules to do the same. The constraints described above eliminate the majority of such failures, and residual ambiguities are flagged for manual review rather than silently excluded.

6.5 Execution

Executing Stage 6 applies automated peak detection and physical filtering to the brightness-gradient profiles produced in Stage 5, identifying candidate atmospheric features and classifying them as primary or secondary detections.

```
python src/06.peakfinder.py \
  --indir data/JNCR_2018197_14C00024_v02/cub/stage_04_perp_samples \
  --outdir data/JNCR_2018197_14C00024_v02/cub/stage_06_peaks \
  --smooth 5
```

Stage 6 writes per-fragment peak detection tables and diagnostic plots, including SPICE-resolved detections where geometry is available and optically detected features where it is not. These products constitute the sole inputs to Stage 7, where latitude-binned detached haze occurrence is computed.

Stage 7 — Analysis: Latitude-Dependent Detached Haze Occurrence

Stage 6 evaluates detached haze occurrence statistically across limb fragments rather than attempting precise spatial localization at the fragment level. Because haze signatures consistently occur prior to the first reliable SPICE surface intercept, individual detections lack uniquely assignable coordinates. Aggregation across fragments using first-available SPICE latitude therefore provides a conservative but physically meaningful measure of latitude-dependent haze occurrence. While individual limb fragments do not permit precise altitude localization due to limited SPICE coverage prior to the limb intercept, the presence or absence of detached haze can be assessed statistically across many fragments. This stage evaluates whether detached haze occurrence varies systematically with planetocentric latitude.

7.1 Proxy Latitude Assignment

For each limb fragment, the first available SPICE-derived planetocentric latitude encountered along the sampling direction is recorded as a conservative geographic proxy; preliminary tests using alternative latitude definitions yielded consistent trends, and limb-crossing latitude—being physically aligned with the limb-intersection geometry—was therefore adopted for all final analyses. Fragments lacking any valid SPICE latitude are excluded from statistical aggregation.

7.2 Binning and Occurrence Metric

Fragments are grouped into latitude bins of fixed width. Within each bin, the fraction of fragments exhibiting a secondary brightness-gradient peak is computed. This fraction represents the occurrence rate of detached haze candidates within that latitude band. Minimum fragment counts per bin are enforced to suppress small-number artifacts.

Detections are binned by planetocentric latitude where SPICE geometry is available, or by a proxy latitude defined at the first valid SPICE intercept along each limb-normal profile. Because earlier stages explicitly remove framelets lacking a resolvable limb–space transition, the occurrence metric reported here reflects detectability within the retained framelet set, not global atmospheric coverage. In latitude ranges where usable limb geometry is sparse or absent, low or zero occurrence values may therefore result from data exclusion rather than physical non-existence of detached haze. This effect is most relevant at low to mid latitudes for the IMG analyzed here and motivates the conservative interpretation adopted throughout this work.

7.3 Results

Detached haze is not uniformly present along the limb. Instead, its occurrence increases sharply at higher planetocentric latitudes, reaching near-continuous presence above approximately 30–35° in this observation.

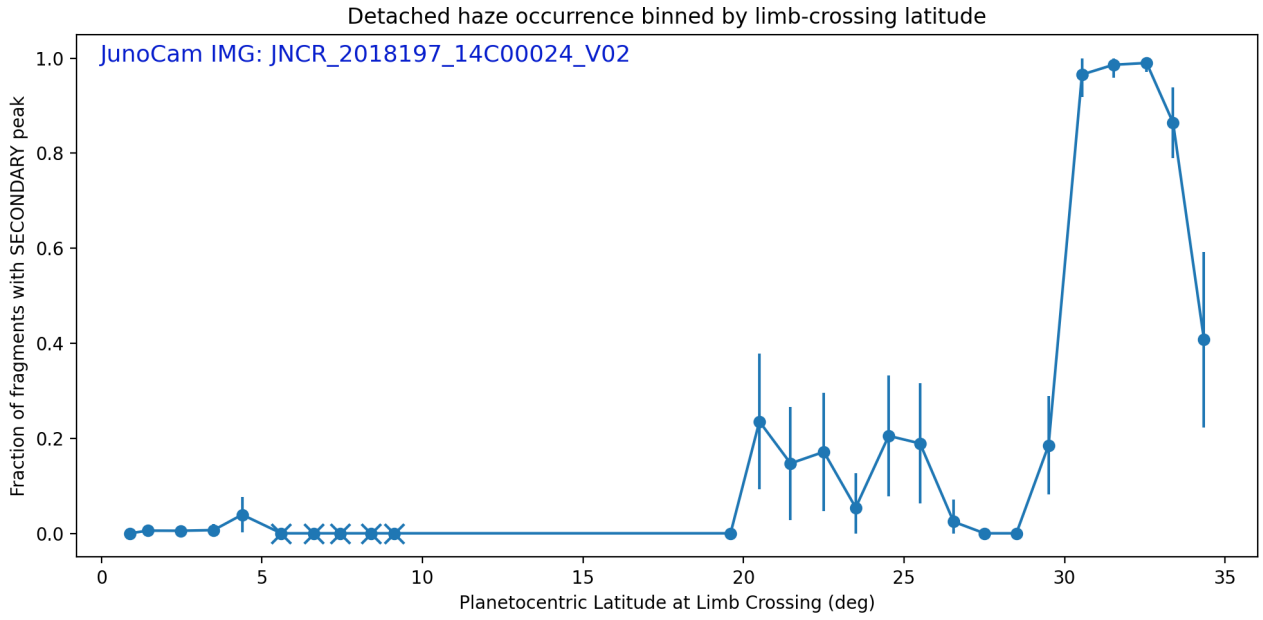


Figure 7.3 - Fraction of limb fragments exhibiting secondary brightness-gradient peaks (detached haze candidates), binned by first-available SPICE planetocentric latitude for JunoCam IMG JNCR_2018197_14C00024_V01.

7.4 Limitations and Interpretation

Because SPICE coverage does not extend to the altitude of detached haze layers, precise spatial localization is not possible at the fragment level. The observed trend therefore represents a statistical, not geometric, association. Nevertheless, the sharp transition in occurrence rate strongly supports a latitude-dependent haze structure.

7.5 Execution

Executing Stage 7 aggregates peak detections produced in Stage 6 into latitude bins and computes an occurrence metric describing the fraction of retained framelets exhibiting detached haze signatures as a function of latitude (Figure 7.3). This stage performs no additional feature detection and relies exclusively on classification outputs and geometric associations established in prior stages.

```
python src/07.haze_analysis.py \
  --imgdir       data/JNCR_2018197_14C00024_v02 \
  --stage6-root  data/JNCR_2018197_14C00024_v02/cub/stage_06_peaks \
  --outdir       data/JNCR_2018197_14C00024_v02/analysis/stage_07 \
  --bin-width-deg 1.0 \
  --min-fragments-per-bin 15
```

Stage 7 produces a latitude-binned occurrence table and a corresponding summary visualization depicting detached haze detectability across planetocentric latitude for the analyzed IMG. These aggregated products represent the terminal scientific outputs of the pipeline and form the basis for interpretation and discussion in the following section.

Stage 8 — Perijove-Scale Detached Haze Analysis

Because Stage 8 performs no new feature detection or geometric processing, it is presented here as a results-only aggregation of the validated Stages 1–7 pipeline.

Purpose

Stages 1–7 define a reproducible analysis pipeline operating on a *single* JunoCam image, producing fragment-level detections of detached haze features at limb-crossing locations. In Stage 8, this validated pipeline is applied without modification across *all* JunoCam images acquired during a single Perijove, enabling Perijove-scale aggregation of detached haze occurrence as a function of planetary latitude and longitude.

For this study, Stage 8 is applied to all available JunoCam images from Perijove 14 (PJ14).

Inputs and Aggregation Strategy

All Stage-7 fragment outputs from each JunoCam image in PJ14 are ingested. Each fragment contributes independently to the aggregate statistics, resulting in fragment-weighted occurrence estimates rather than image-level averages.

Fragment aggregation is subject to the following safeguards:

- Only fragments with finite SPICE-derived limb-crossing geometry (latitude, longitude, and slant distance) are retained.
- Latitude bins are masked unless a minimum number of contributing fragments is met, preventing interpretation in sparsely sampled regions, and latitude bins with insufficient fragment counts are excluded.

- Longitude–latitude cells are masked unless a minimum fragment threshold is met, preventing interpolation into unobserved regions.

Fragments lacking finite SPICE geometry at or near the detected limb-crossing location are excluded from spatial aggregation but retained in raw outputs for auditability.

Execution

Stage 8 is executed once per Perijove using the following command:

```
python src/08.perijove_analysis.py \
  --pjdir data/PJ14 \
  --pj-label "Perijove 14" \
  --lat-bin-width-deg 1.0 \
  --lon-bin-width-deg 5.0 \
  --min-fragments-per-lat-bin 30 \
  --min-fragments-per-img-lat-bin 8 \
  --min-fragments-per-lonlat-cell 10 \
  --map-lat-min -30 --map-lat-max 30
```

Results

8.1 Perijove-Scale Latitude Dependence

Figure 8.1 addresses the central question of this study: whether detached haze detectability exhibits a systematic dependence on planetocentric latitude when aggregated across an entire Perijove. The fragment-weighted occurrence curve reveals a pronounced mid-latitude enhancement, with detached haze detectability increasing sharply above ~30–35° planetocentric latitude. Below this transition, occurrence rates remain low or near zero, reflecting either reduced detectability or limited geometric coverage.

Confidence intervals widen toward the poles as fragment counts decrease, consistent with reduced sampling density and more restrictive limb geometry at extreme latitudes. Importantly, the observed mid-latitude enhancement persists well above the statistical uncertainty, indicating that the signal is not driven by small-number effects.This result demonstrates that latitude-dependent detached haze detectability emerges only when aggregating many independent limb fragments across images, and is not an artifact of any single framelet or localized observation.

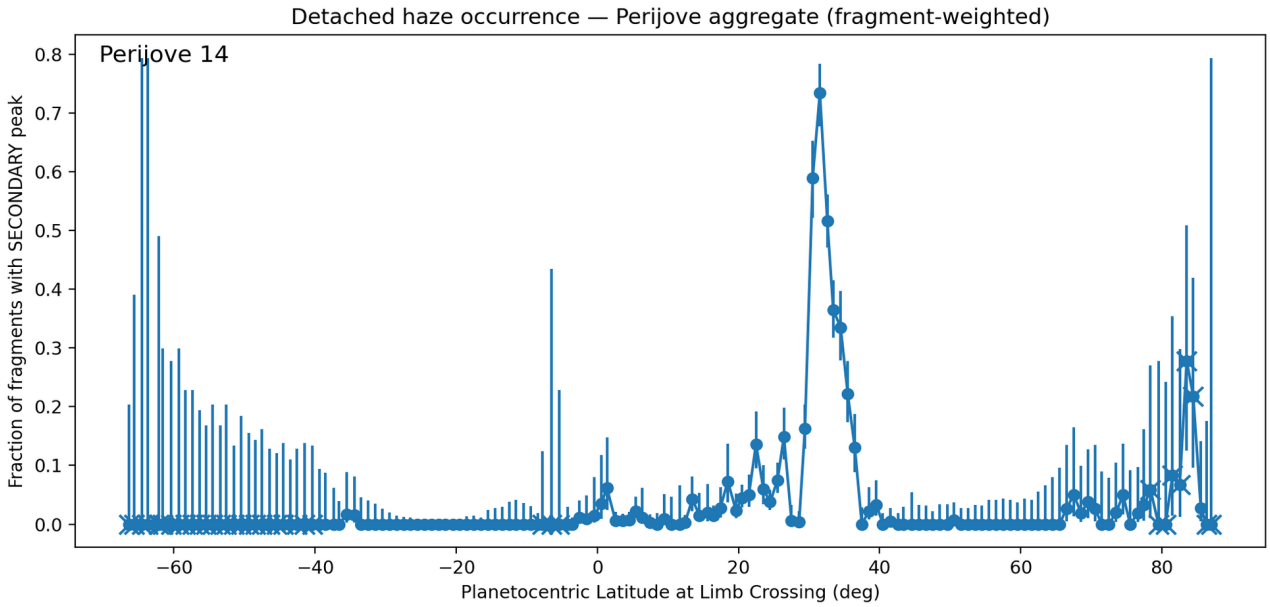


Figure 8.1 - Fragment-weighted detached haze occurrence versus planetocentric latitude for Perijove 14.

8.2 Robustness Across Images

Figure 8.2 evaluates whether the mid-latitude enhancement observed in Figure 8.1 is driven disproportionately by a single image or viewing geometry, or whether it represents a reproducible pattern across independent observations.

Detached haze occurrence is shown as a function of latitude for each individual JunoCam image acquired during Perijove 14. Although individual curves vary due to image-specific coverage and sampling density, the mid-latitude enhancement near ~30–35° is consistently present across multiple images.

No single image dominates the aggregate result, and the latitude of peak detectability remains stable across disparate limb geometries. This confirms that the Perijove-scale signal arises from repeated, independent detections rather than localized anomalies or statistical outliers.

The coherence of this feature across images supports the interpretation that detached haze detectability reflects a persistent atmospheric structure rather than transient observational conditions.

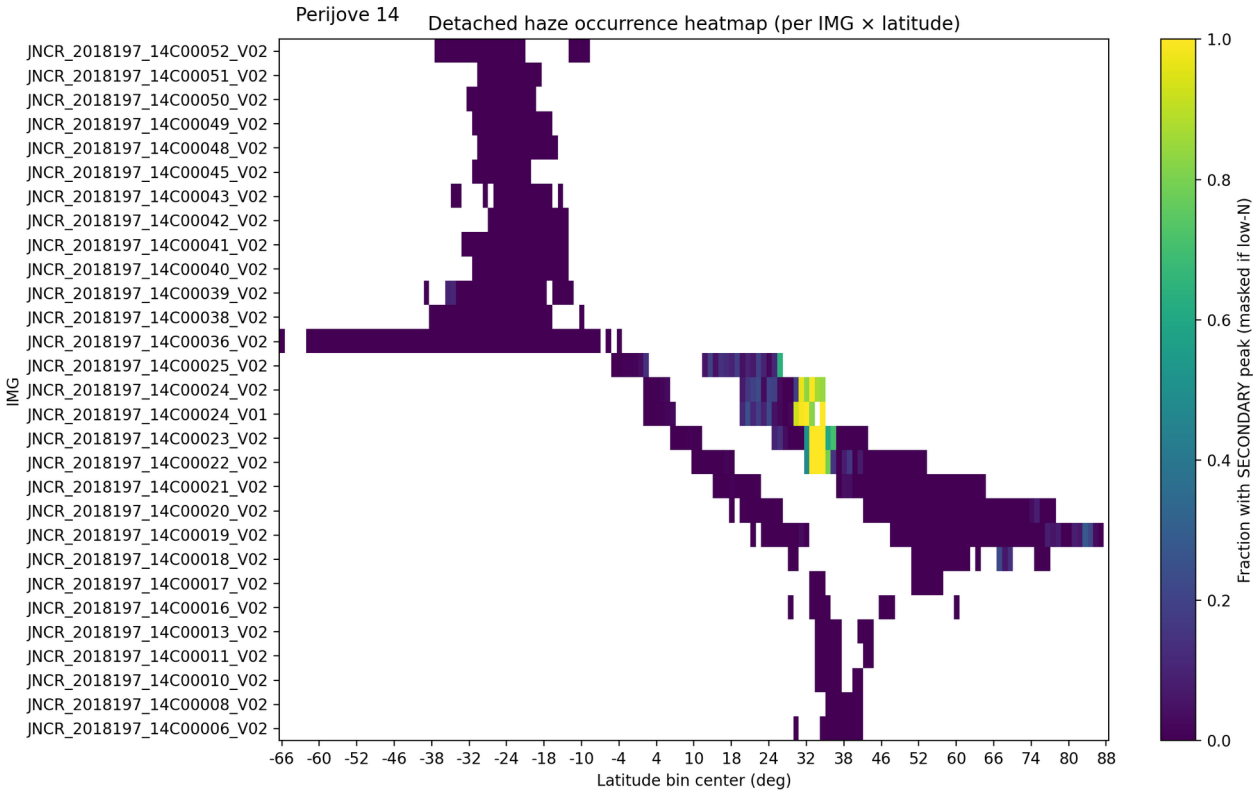


Figure 8.2 Figure Y shows detached haze occurrence binned by latitude for each individual JunoCam image in PJ14.

8.3 Longitude–Latitude Distribution

Figure 8.3 places the latitude-dependent results in a longitudinal context by mapping detached haze occurrence onto a longitude–latitude grid for Perijove 14. This figure addresses whether the observed enhancement is globally distributed or confined to specific longitudinal sectors.

Detached haze detections cluster within limited longitude ranges at mid-latitudes, while large portions of the map remain unpopulated. These gaps reflect the inherently sparse and non-uniform longitudinal coverage imposed by the spacecraft's Perijove trajectory rather than an absence of haze.

No interpolation is performed in unsampled regions, preserving a strict distinction between observed structure and coverage limitations. The resulting map therefore represents where detached haze was *measurably detectable* under PJ14's viewing geometry, not where haze is physically absent.

Together with Figures 8.1 and 8.2, the longitude–latitude map demonstrates that the latitude-dependent enhancement is not an artifact of longitudinal averaging, while simultaneously highlighting the need for multi-Perijove aggregation to assess global longitudinal structure.

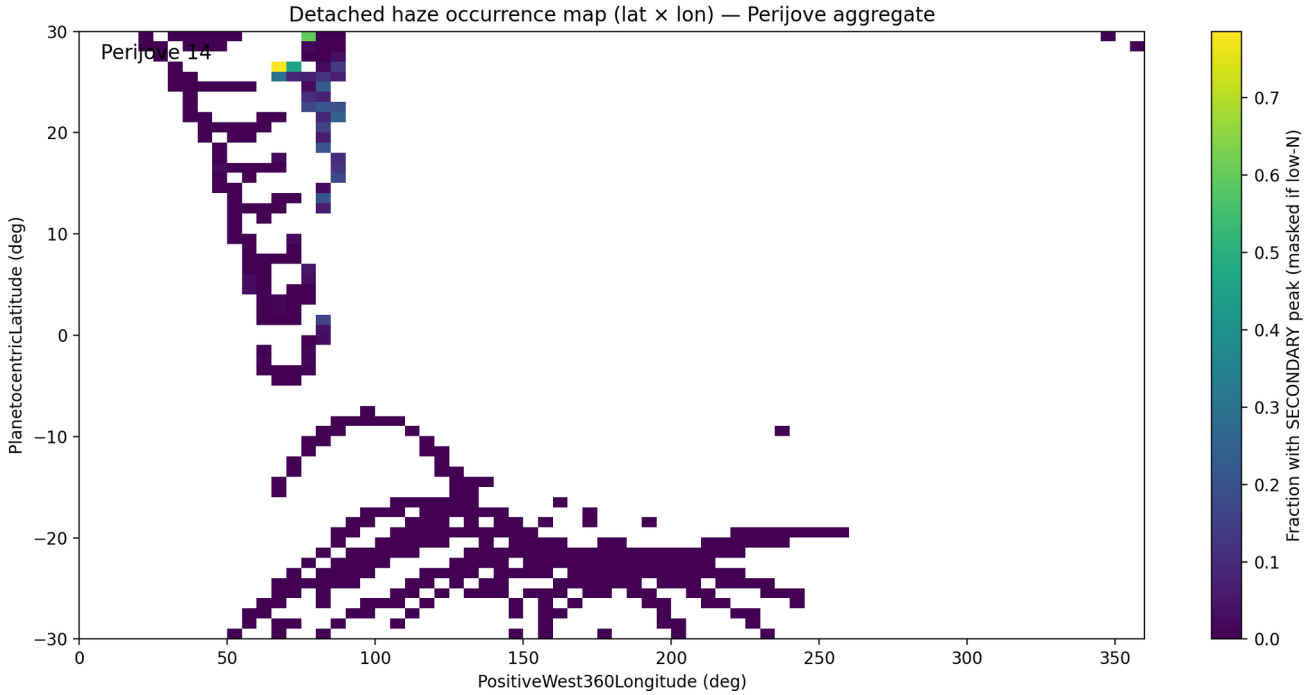


Figure 8.3 - Longitude–latitude occurrence map of detached haze for Perijove 14.
Colors indicate the fraction of fragments exhibiting detached haze within each longitude–latitude cell. Cells with insufficient sampling are masked.

Interpretation

Taken together, the Stage 8 results demonstrate that detached haze occurrence during Perijove 14 is not uniformly distributed with latitude, exhibiting a statistically significant mid-latitude enhancement that is consistently observed across multiple images. Longitude–latitude mapping further reveals structured—but coverage-limited—spatial clustering under the Perijove's viewing geometry.

Stage 9 — Discussion

Stage 9 interprets the Perijove-scale results of Stage 8 and does not introduce additional analysis or detection steps. The results presented here demonstrate a clear detectability-

based latitude-dependent transition in detached haze occurrence along the observed limb. Although some individual limb fragments lack precise SPICE geometry at the haze altitude, the aggregate behavior across fragments and framelets reveals a statistically robust pattern: detached haze features are rare or absent at low planetocentric latitudes and become increasingly prevalent toward higher latitudes, approaching near-continuous detectability above approximately ~30–35° in this observation.

The near-zero occurrence observed between approximately 10–20° planetocentric latitude does not imply the physical absence of detached haze. This latitude range corresponds to framelets in which usable limb geometry is sparse or entirely absent due to spacecraft motion, viewing geometry, and the necessary removal of images in which Jupiter fully occupies the frame.

The analysis therefore reports detectability, not atmospheric non-existence. Importantly, the sharp transition in detached haze occurrence above ~30–35° persists despite these conservative exclusions, indicating that the high-latitude signal is robust to data curation and geometric limitations.

This conclusion is deliberately conservative. Detached haze identification relies solely on brightness-gradient structure and geometric consistency, without assuming a priori vertical placement or atmospheric composition. The absence of SPICE surface intercepts for many haze-adjacent samples precludes precise altitude assignment at the fragment level. However, because first-available SPICE intercept latitude varies smoothly across fragments and framelets, it provides a meaningful proxy for large-scale latitudinal trends when analyzed statistically.

Importantly, this approach avoids over-interpretation of individual profiles while preserving sensitivity to genuine atmospheric structure. The observed transition therefore supports the hypothesis that detached haze occurrence is not uniform along the limb, but instead exhibits a strong dependence on planetocentric latitude. Future work aggregating multiple Perijoves or incorporating improved limb-geometry solutions may further constrain altitude and longitudinal persistence, but such refinements are not required to establish the primary latitude-dependent detectability trend reported here.

Stage 10 — Reproducibility and Data Availability

All analysis code used in this study—including limb tracing, rectification, peak finding, and statistical aggregation—are publicly available in a GitHub repository at https://github.com/kevinkell-y/juno-haze-classification. The repository contains versioned source code, execution instructions, and directory conventions required to reproduce all figures and tables in this paper. Intermediate products generated during processing are preserved in IMG-specific directories to ensure reproducibility and facilitate independent inspection. Processed data products (rectified limb profiles, classification tables, and diagnostic figures) are derived from publicly available JunoCam images distributed by NASA’s Planetary Data System. Instructions to reproduce each pipeline stage are provided inline in the Methods section. A citable

archival release of this pipeline and Perijove-scale outputs is provided via Zenodo to ensure long-term reproducibility independent of repository state.

Additionally, all Stage-8 Perijove-scale aggregation products, including fragment tables and spatial occurrence maps, are distributed alongside the single-image pipeline outputs.

Each JunoCam cube’s label and accompanying manifest file record the complete set of SPICE kernels used during initialization (LSK, PCK, SPK, CK, FK, IK, SCLK). These kernel sets were drawn from locally synchronized ISIS and Juno kernel repositories and are publicly available online. The manifest files stored alongside each .cub constitute the definitive provenance record for all geometry used in this study.

References

Brueshaber, S. (2021). *JunoCam Image Processing Document, Revision 2*. NASA Jet Propulsion Laboratory, Planetary Science Group Technical Report #3222.

Eichstädt, G., et al. (2021). *JunoCam Limb Geometry and Imaging Characteristics*.

Navigation and Ancillary Information Facility (NAIF). (2021). *SPICE Toolkit Documentation*. https://naif.jpl.nasa.gov/naif/

Kelly, K. (2022). *Haze Detection in the Atmosphere of Jupiter Using Planetary Limb Images from JunoCam*. NASA Jet Propulsion Laboratory.

Wen, D. California Institute of Technology. Personal communication, 2022.

Appendix A — Limb-Tracing Algorithm (Implementation Overview)

This appendix provides an implementation-level description of the limb-tracing procedure used in Stage 2. It is intended to document algorithmic behavior and reproducibility rather than derive formal edge-detection mathematics.

The limb-tracing module (02.trace_limb_polyline.py) combines SPICE-registered camera geometry with grayscale gradient analysis to identify the boundary between Jupiter's illuminated atmosphere and surrounding space. The algorithm operates entirely in image-coordinate space and makes no assumption about planetary orientation, illumination direction, or limb position within the frame.

A.1 Pre-processing

Input: SPICE-attached framelet F.cub
Output: Grayscale array I(x, y)

1. Convert .cub → 16-bit TIFF and 8-bit PNG using isis2std.
2. Read PNG and normalize pixel intensities to [0, 1].
3. Apply a 3×3 Gaussian blur to suppress single-pixel noise.

A.2 Gradient-field computation

The algorithm computes horizontal and vertical image gradients using a Sobel operator and combines them to form a gradient-magnitude field. The strongest global gradient in the framelet is used as an initial limb seed. This replaces earlier orientation-specific heuristics and ensures orbit- and geometry-agnostic limb initialization.

4. Compute Sobel derivatives:
   $G_x = dI/dx$, $G_y = dI/dy$
5. Compute gradient magnitude and orientation:
   $M = \sqrt{(G_x^2 + G_y^2)}$
   $\Theta = \arctan2(G_y, G_x)$
6. Identify global maximum of M as initial limb seed P_0.

This replaces the earlier fixed-orientation seeding heuristic (Wen 2021) with a geometry-agnostic initialization that adapts to any illumination direction.

A.3 Bidirectional edge propagation

7. Initialize limb polyline $L = \{P_0\}$.
8. For each propagation direction $D \in$ {forward, reverse}:
   repeat
   - Sample candidate points Q within a narrow angular window ($\pm\theta$) around $\Theta(P_0)$.
   - Select Q maximizing local gradient magnitude M(Q).
   - Append Q to L if distance(P_0,Q) < δ_{max}.
   - Update orientation $\Theta(P_0) \leftarrow \Theta(Q)$; set $P_0 \leftarrow Q$.

   until gradient < τ or boundary reached

Typical parameters:
$\theta \approx 10–15°$, $\delta_{max} \approx 3$ pixels, $\tau \approx 0.1 \times M_{max}$.
Both directions are traced independently and then merged, yielding a continuous limb polyline.

A.4 Polyline smoothing and export

9. Apply cubic spline or low-pass polynomial fit to L.
10. Save limb coordinates → <CUB>_LIMBENDPOINTS.csv
11. Overlay cyan markers on source image → <CUB>_OVERLAY.png

The resulting CSV provides ordered (x, y) positions along the detected planetary edge at sub-pixel precision. Each overlay PNG offers a rapid-visual confirmation of limb detection quality.

A.5 Performance and false-positive control

- **Runtime:** ≈ 0.5–1 s per 1600×128 framelet on a modern CPU.
- **Memory footprint:** < 200 MB per image.
- **False positives:** Occur when the scene lacks dark-to-bright transitions (planet fills frame). These cases are flagged during later tangent-normal sampling when no null-space data are detected.

A.6 Scientific use

The extracted limb polylines define the instantaneous geometric curvature of Jupiter’s visible atmosphere. They serve as the spatial reference for subsequent rectification and altitude mapping, enabling consistent geometric sampling for subsequent brightness-gradient analysis.

No assumption is made that the rectified limb represents a true planetary vertical; rather, rectification is performed in image-coordinate space to stabilize sampling geometry for comparative brightness-gradient analysis.

Appendix B – Environment Setup and Reproducibility Notes

Detailed operating-system requirements, ISIS3 installation procedures, SPICE kernel acquisition, and environment locking instructions are maintained in the project repository to ensure long-term accuracy and version control. These materials are intentionally excluded from the main manuscript, which assumes a working ISIS3 and SPICE installation, and focuses instead on the scientific pipeline and analytical methodology.