GRIME³

Complete User Guide

Version 0.1.0 RC8

User Guide Revision 1.5

Author

Dr. Kenneth W. Chapman

Adjunct Researcher, GRIME Lab

University of Nebraska-Lincoln

Software provided by GaugeCam

September 17, 2026

Document information

Program	GRIME 3 (software name: GRIME3)
Software version	0.1.0 RC8
User guide revision	1.5
Author	Dr. Kenneth W. Chapman
Author title	Adjunct Researcher, GRIME Lab, University of Nebraska-Lincoln
Software organization	GaugeCam
Guide date	September 17, 2026
Release source tag	`grime3-0.1.0-rc8`
Supported distributions	Windows x64 standalone; Linux x86_64 standalone; Conda; Linux/amd64 Docker/OCI
Authoritative build identity	See the packaged `BUILD_INFO.json` and distributed checksum for the actual release artifact.
License	Apache License 2.0

Key idea
This guide is written so that a careful new user can follow it without already knowing computer vision. It covers all supported RC8 distribution paths while preserving the scientific calibration and measurement workflow.

Revision history

Revision	Date	Description
1.5	2026-09-17	RC8 complete cross-distribution guide. Adds the supported CLI, curated CSV behavior, Site Configuration identity and portability, Conda installation, Linux standalone operation, Docker/OCI operation and persistence, and cross-distribution release provenance.
1.4	2026-09-15	RC6 update for grayscale inner-boundary calibration, projective facet selection, corrected grid and stage-zero geometry, and dynamic calibration-resolution guidance.
1.2	2026-09-11	RC5 rewrite around the canonical Site Configuration workflow.
1.1	2026-09-08	RC3 guide with large-image octagon safety and calibration diagnostics.

What this guide covers

This guide explains how to:

- install and verify the Windows release;
- start and stop GRIME3;
- understand the browser interface;
- choose image folders;

- load, create, save, and verify one Site Configuration JSON;
- understand the octagon ROI and waterline ROI;
- create and verify calibration;
- use the grid, staff gauge, ruler, octagon, waterline, and waterline-point overlays;
- measure one image;
- process a folder or folder tree;
- choose image timestamps from EXIF data or filenames;
- interpret Accepted, Not Accepted, and Error results;
- preserve CSV, JSON, decision-overlay, and run-manifest outputs;
- process collaborator data reproducibly;
- inspect exact build identity and licenses;
- report a problem so another person can reproduce it.

Contents

IMPORTANT - Calibration image resolution requirement. The detected calibration octagon must be at least **100 pixels wide AND 100 pixels high in the original source image**. GRIME3 rejects calibration when either dimension is smaller. Do not use resized or downsampled calibration images if resizing reduces the octagon below 100 by 100 pixels. **Force small-octagon detection mode does not override this minimum-resolution requirement.**

List of Figures

1 Start here: what GRIME3 is

1.1 The simple explanation

GRIME3 is a GaugeCam program for measuring water level from photographs. A camera records a scene containing a known calibration target and a visible water surface. GRIME3 uses the known target geometry to connect image pixels to real-world coordinates, then estimates the waterline.

The result is not merely a line drawn on a picture. A useful result connects several things:

1. the source image;
2. the site calibration;
3. the detected calibration target;
4. the waterline-search region;
5. the selected waterline;
6. the physical water-level value;
7. the software build that produced the result.

RC8 stores the persistent site-specific pieces of this chain in one **Site Configuration JSON** file.

1.2 Why the octagon matters

A camera image is a two-dimensional view of a three-dimensional scene. Perspective makes distances in different parts of the image look different. A known physical octagon gives GRIME3 geometry whose true shape and dimensions are known.

After the octagon has been detected, GRIME3 can construct the image-to-world mapping used for calibrated overlays and stage measurement.

1.3 The full workflow in one picture

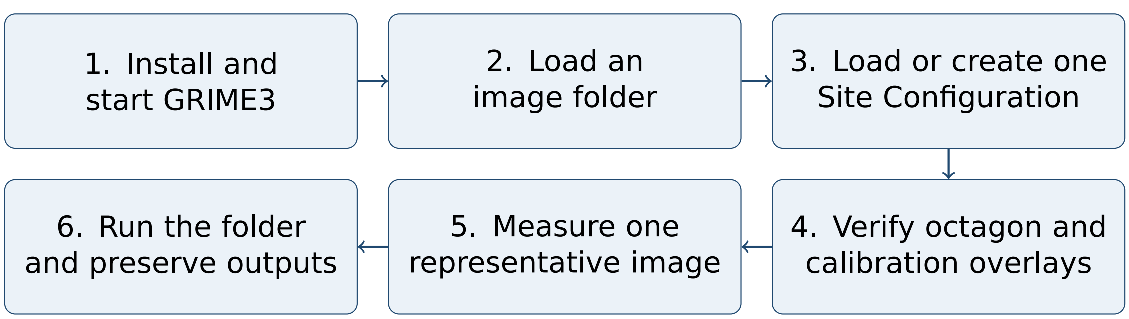

Figure 1: The recommended GRIME3 RC8 workflow.

1.4 Four words to understand early

1.4.1 ROI: Region of Interest

An ROI tells GRIME3 where to look.

RC8 uses two important regions:

- **Octagon ROI**: the region where GRIME3 searches for the physical calibration target.
- **Waterline ROI**: a quadrilateral region that tells GRIME3 where to look for the water surface.

1.4.2 Calibration

Calibration is the relationship between image coordinates and physical world coordinates. It uses the known octagon geometry, physical facet length, physical units, and known water-level datum.

1.4.3 Overlay

An overlay is a diagnostic drawing placed on the photograph. Examples include the found octagon, grid, ruler, staff gauge, found waterline, and candidate waterline points.

1.4.4 Site Configuration

The Site Configuration is the canonical persistent description of one configured site and its calibration. In RC8 it replaces the older practice of managing separate ROI and calibration JSON inputs.

> **Important**
> Do not trust a numerical stage value simply because the software produced one. First verify that the active geometry and selected waterline make physical sense on the image.

2 Windows installation from the ZIP file

2.1 What kind of Windows package this is

The RC8 Windows package is a **PyInstaller one-folder** distribution. It is not a traditional Windows installer.

The extracted release also contains these supporting documents:

- `RELEASE_NOTES.md`
- `WINDOWS_QUICK_START.md`
- `FIELD_EVALUATION_WORKFLOW.md`

They are generic release documentation intended to travel with the executable package.

The package contains:

- `GRIME3.exe`;

- `START_GRIME3.bat`;
- an `_internal` directory containing the packaged Python runtime and supporting libraries;
- release documentation;
- GRIME3's Apache 2.0 license;
- third-party package and native-runtime license information;
- build provenance.

Python, Conda, and a source checkout are not required to run the packaged application.

2.2 What you need before installing

For normal evaluation you need:

- a 64-bit Windows computer;
- permission to extract files;
- a web browser;
- access to the image data;
- permission to write to the chosen output directory.

For initial testing, normal local folders are easier to troubleshoot than network shares.

2.3 Step 1: put the release files together

Place these two files in the same easy-to-find folder:

- `GRIME3-0.1.0-rc8-windows-x64.zip`
- `GRIME3-0.1.0-rc8-windows-x64.zip.sha256`

2.4 Step 2: verify the SHA-256 checksum

This step verifies that the ZIP you received is exactly the ZIP that was validated for RC8.

Open PowerShell in the folder containing the ZIP and run:

```
Get-FileHash .\GRIME3-0.1.0-rc8-windows-x64.zip -Algorithm SHA256
```

The expected hash is:

```
0cc402ca481104890d7b6f5c8032b52eea84151dd194032e4a1cf95a583f5d9d
```

If even one character differs, do not use that ZIP as the validated RC8 release.

2.5 Step 3: extract the ZIP

Right-click the ZIP and extract it to a normal local folder. Keep the entire extracted directory together.

Do not copy only `GRIME3.exe` somewhere else. The executable depends on files stored beside it in the one-folder distribution.

A reasonable location is:

`C:\GRIME3\GRIME3-0.1.0-rc8-windows-x64\`

2.6 Step 4: start GRIME3

The preferred method is to double-click:

`START_GRIME3.bat`

Alternatively, double-click:

`GRIME3.exe`

Keep the console window open while using GRIME3.

The browser should open automatically at:

`http://127.0.0.1:8000/`

If the browser does not open automatically, type that address into a browser manually.

2.7 Windows SmartScreen or security warnings

RC8 is research/evaluation software and is not digitally code-signed. Windows may therefore display a SmartScreen or security warning.

The checksum is the important identity check for this release. Verify the ZIP hash before extracting and running the application.

2.8 How to stop GRIME3

Close the GRIME3 console window or press Ctrl+C in the console.

Do not close the process while a folder measurement is actively processing an image.

2.9 Updating or removing RC8

Because RC8 is a one-folder application:

- removing GRIME3 means removing the extracted program folder;
- upgrading normally means extracting a newer release into a separate folder;
- do not mix files from different release candidates.

3 Linux standalone installation

The RC8 Linux standalone package is a PyInstaller one-folder `.tar.gz` distribution for modern x86_64 Ubuntu/Xubuntu-class systems. Python and Conda are not required on the target computer.

3.1 Install and start

Verify and extract the complete archive:

```
sha256sum -c GRIME3-0.1.0-rc8-linux-x86_64.tar.gz.sha256
tar -xzf GRIME3-0.1.0-rc8-linux-x86_64.tar.gz
cd GRIME3-0.1.0-rc8-linux-x86_64
```

Start GRIME3:

```
./start-grime3.sh
```

or:

```
./grime3
```

The same executable provides the command line:

```
./grime3 measure --help
```

4 Conda installation and environment management

The RC8 Conda package provides the same GRIME3 browser and CLI within a managed Conda environment.

4.1 Create an environment

```
conda create -n grime3-rc8 -c conda-forge python=3.11
conda activate grime3-rc8
```

GaugeCam/GRIME software already uses the public `GRIMELab` organization on Anaconda.org. The RC8 package is intended to be published there under an explicit `rc8` label so release candidate software is not confused with a future production package.

After publication, create a dedicated RC8 environment with:

```
conda create -n grime3-rc8 \
  --override-channels \
  -c conda-forge \
  -c grimelab/label/rc8 \
  python=3.11 \
  grime3=0.1.0
```

The conda-forge channel is listed first so ordinary runtime dependencies are resolved from conda-forge while the GRIME3 package itself is obtained from the GRIMELab RC8 label.

The exact downloaded RC8 package artifact can also be installed directly:

```
conda create -n grime3-rc8 -c conda-forge python=3.11
conda install -n grime3-rc8 -c conda-forge \
  /path/to/grime3-0.1.0-py_0.conda
```

A later stable package can be promoted to the GRIMELab main label. Submission to conda-forge remains a later distribution-hardening option.

4.2 RC8 Conda runtime

The RC8 recipe uses conda-forge OpenCV 4.13.0 with the `headless_*` build and an LGPL FFmpeg build. Qt, PyTorch, and SAM2 are not required.

4.3 Verify and run

```
grime3 --version
grime3 measure --help
grime3 web --open-browser
```

4.4 Remove

Remove only GRIME3:

```
conda remove grime3
```

Remove the complete dedicated environment:

```
conda deactivate
conda env remove -n grime3-rc8
```

5 Docker and OCI container operation

The RC8 Docker image contains the same deterministic GRIME3 browser and CLI in a minimal Linux container.

The native RC8 validation target is Linux/amd64. Windows and macOS users can run the Linux image through Docker Desktop. Native ARM64 and multi-architecture images are later roadmap work.

5.1 Persistent paths

The container uses:

```
/data/config
/data/images
/data/results
/data/database
```

The container itself is disposable. Configuration, source images, scientific results, and persistent database material belong on mounted storage.

5.2 Browser

```
docker run --rm \
  -p 8000:8000 \
  grime3:0.1.0-rc8
```

Open:

```
http://127.0.0.1:8000/
```

5.3 CLI

```
docker run --rm \
  --user "$(id -u):$(id -g)" \
  --mount type=bind,src="/absolute/site",dst=/data/config,readonly \
  --mount type=bind,src="/absolute/images",dst=/data/images,readonly \
  --mount type=bind,src="/absolute/results",dst=/data/results \
```

```
  grime3:0.1.0-rc8 \
  measure folder \
    --images /data/images \
    --site-config /data/config/site_config.json \
    --output-dir /data/results
```

5.4 Docker Compose

The supplied Compose configuration maps the four persistent data locations and host port 8000.

On Linux:

```
mkdir -p data/config data/images data/results data/database
export GRIME3_UID="$(id -u)"
export GRIME3_GID="$(id -g)"
docker compose up --build
```

Docker Desktop users should use host paths supported by their platform.

6 A tour of the GRIME3 browser screen

6.1 The image is the center of the workflow

Most important operations eventually need visual confirmation on the image. The sidebar controls choose data and processing options. The image area shows the selected photograph and diagnostic overlays.

6.2 The four main tabs

The main browser interface is organized around:

Files Load an image folder and select images.

Calibration Load or create Site Configuration and inspect calibration.

Measurement Measure one image or process a folder.

About View release notes, Help, and licensing information.

6.3 The image toolbar

The image toolbar includes diagnostic displays such as:

- Ruler;
- Found Octagon;
- Grid;
- Staff Gauge;
- Found Waterline;
- Found Waterline Points.

These displays do not change the underlying photograph.

6.4 Zoom is not calibration

Browser zoom and image-view zoom change how large the picture looks on screen. They do not change the physical calibration.

A grid interval in physical coordinates must remain tied to calibration, not to how large the image happens to be displayed.

7 Files and the Site Configuration

7.1 Load an image folder

On the Files tab, type or edit the image-folder path and press **Enter**. GRIME3 loads the folder.

You can also select the ellipsis button to browse.

In the folder browser:

- **Load** accepts the selected folder and immediately loads it;
- **Cancel** closes the browser without changing the currently loaded folder.

7.2 Recursive folder processing

Enable recursive processing when images stored in subfolders should be included in a folder run.

Before a large recursive run, test a small representative subset first.

7.3 What changed in RC8

RC8 retains the canonical Site Configuration workflow and deterministic water-level measurement pipeline while substantially improving large-image calibration and calibration visualization.

The principal RC8 changes are:

- large-image octagon calibration is now grayscale-based and no longer depends on target color;
- the intended calibration perimeter is the inner octagon boundary. It lies between the calibration-target face and its black border;
- the outer edge of the black border, black-border width, and scene content beyond the border are not calibration geometry;
- full-resolution facet acquisition can retain multiple polarity-correct line hypotheses and select the globally consistent combination under one projective octagon model;
- measured adjacent-facet intersections remain the final precision corners;
- a completely closed detected contour is not required when sufficient distributed facet evidence remains;
- the yellow horizontal calibration grid is anchored to the physical bottom octagon facet;

- the blue zero-level line now represents actual calibrated water level 0.0 rather than raw calibration-world $Y = 0$;
- the zero-level line is labeled `0.0 water level`;
- calibration-resolution guidance is dynamic: red below the hard 100-pixel minimum, yellow near the minimum, and hidden when comfortably above it;
- Create-button eligibility is reevaluated immediately after ROI authoring or editing.

The Site Configuration remains the single canonical persistent description of site identity, ROI geometry, calibration, units, datum information, and related provenance.

7.4 Load an existing Site Configuration

There are three convenient ways to initiate a load:

1. type or edit the Site Configuration JSON path and press **Enter**;
2. enter the path and select **Load**;
3. select the ellipsis browser, choose an existing JSON file, and choose **Load**.

Choosing Load inside the browser immediately loads the file. A second external Load click is not required.

Cancel leaves the current Site Configuration unchanged.

7.5 Save

Save writes the currently loaded canonical Site Configuration back to its current JSON path.

Save does not silently rerun octagon detection or recalculate calibration.

7.6 Save As

Save As writes the currently loaded canonical Site Configuration to a different path.

The target filename may be new. If it already exists, GRIME3 requires explicit overwrite confirmation.

Save As does not silently recalculate calibration.

7.7 Create

Use **Create** when the current authoring state should generate a new/recalculated Site Configuration.

Create uses the current:

- selected calibration image;
- octagon ROI;
- waterline ROI;
- site name;

- operator;
- facet length;
- units;
- bottom-left octagon-corner waterlevel;
- grid divisions;
- small-octagon mode.

Create opens the Site Configuration save browser. The target filename may be new. Existing-file replacement requires confirmation.

Create availability. The browser reevaluates Create eligibility immediately when required calibration values or ROI geometry change. Defining, editing, clearing, or recreating an ROI therefore updates the Create-button state without requiring an unrelated field to be edited first.

8 Working with Site Configurations in RC8

8.1 What a Site Configuration represents

A Site Configuration is the persistent scientific description of one calibrated GRIME3 measurement configuration, not merely a group of GUI settings.

It contains or connects:

- persistent Site identity;
- persistent Configuration identity;
- representative source-image identity and provenance;
- octagon search ROI;
- quadrilateral waterline-search ROI;
- image-to-world calibration;
- physical octagon facet length;
- world units;
- water-level datum;
- grid-density and small-octagon settings;
- file, ROI, and calibration audit provenance.

RC8 Site Configuration schema version 2 uses persistent Site and Configuration UUIDs. A pathname is a storage location, not the logical identity.

8.2 Create new versus reuse existing

Create a new Site Configuration when the scientific camera/site geometry changes or a genuinely different calibrated configuration is required.

Reuse an existing configuration when the images belong to the same calibrated setup. Reuse preserves calibration, ROIs, units, datum, identity, and provenance.

Do not reuse a configuration merely because two scenes look similar.

8.3 Load, Save, Save As, and Create

Load Read an existing canonical Site Configuration and project it onto the selected image when possible.

Save Persist the loaded logical configuration to its current file without silently recalculating calibration.

Save As Write the same logical configuration to another pathname. A different pathname does not itself create a new scientific identity.

Create Calculate and persist a new or recalculated configuration from the current authoring state.

Cancel leaves the current configuration unchanged.

8.4 Browser and CLI use the same configuration

The browser and command line use the same Site Configuration model and the same scientific measurement implementation.

The browser maintains the loaded configuration interactively. The CLI requires the configuration explicitly:

```
grime3 measure folder \
  --images /data/site/images \
  --site-config /data/site/site_config.json \
  --output-dir /data/site/results
```

Measurement does not consume the configuration, so the same configuration can be reused across multiple runs.

8.5 Moving configurations between computers

The JSON configuration is intended to be portable. Copy the complete file instead of manually recreating calibration values.

Persistent UUIDs remain the logical Site and Configuration identities when the file moves to another pathname or computer.

After moving data:

1. preserve the Site Configuration;
2. preserve the source images;
3. preserve prior outputs and provenance;

4. load a representative image;
5. verify the octagon, waterline ROI, grid, datum, and units before processing a large dataset.

A recorded original source-image pathname is historical provenance and does not require the copied dataset to occupy that same pathname.

8.6 Recommended organization

For a simple site:

```
site_name/
    images/
    site_config.json
    results/
```

Users are not required to adopt that hierarchy. The important requirement is to preserve the relationship between source images, the configuration, and results.

8.7 Data Roots and portable site packages

RC8 contains the persistence foundation for the future Site Catalog.

A portable Data Root can carry:

```
.grime3-root.json
```

with a persistent Data Root UUID and provenance.

A portable site package can use:

```
.grime3/
```

for site metadata, configuration files, and explicit assignment rules.

The assignment model supports path-scope and time-range rules, providing a foundation for later hydrograph and unattended-station workflows.

> **Important**
> The full Site Catalog browser interface and automatic configuration assignment are later roadmap work. RC8 measurement still loads or names the intended Site Configuration explicitly.

8.8 Configuration/data mismatch

Stop and investigate if the projected octagon or waterline ROI is wrong, the datum or units are wrong, the configuration belongs to another camera geometry, calibration is rejected, or the resulting stage is physically implausible.

Correct the configuration/data relationship instead of accepting a measurement produced from mismatched geometry.

9 ROI geometry: teaching GRIME3 where to look

9.1 Why ROIs are useful

A field photograph can contain thousands or millions of pixels that are irrelevant to the calibration target or water surface.

ROIs reduce ambiguity by limiting the search.

9.2 The octagon ROI

A good octagon ROI:

- contains the complete physical target;
- includes enough surrounding context for detection;
- does not unnecessarily include a huge unrelated part of the photograph.

For high-resolution field images, including the complete target is especially important because RC8 contains dedicated large-ROI detection logic.

9.3 The waterline ROI

The waterline ROI is a quadrilateral. It should contain the section of water surface where a plausible waterline can be found.

Its geometry should follow the part of the scene that is scientifically meaningful for the measurement, rather than simply being a large box around everything.

9.4 Editing

Use the Calibration tab controls to edit or clear the octagon ROI and waterline ROI.

When editing, verify vertices on the image itself. Display zoom can help placement accuracy, but zoom does not alter calibration.

9.5 Persistence in RC8

The ROI geometry is persisted as part of the Site Configuration. It is not a separate normal browser configuration file in RC8.

10 Calibration: turning pixels into real measurements

> **IMPORTANT - Calibration image resolution requirement.** The detected calibration octagon must be at least **100 pixels wide AND 100 pixels high in the original source image**. GRIME3 rejects calibration when either dimension is smaller. Do not use resized or downsampled calibration images if resizing reduces the octagon below 100 by 100 pixels. **Force small-octagon detection mode does not override this minimum-resolution requirement.**

10.1 The big idea

Calibration establishes a geometric relationship between image coordinates and physical world coordinates.

GRIME3 combines:

- detected octagon geometry;
- known physical octagon size;
- physical units;
- the known water-level datum;
- perspective transformations.

The result supports physical measurements and calibrated diagnostic overlays.

10.2 The three important physical inputs

10.2.1 Octagon facet length

A facet is one side of the physical octagon. Enter its real physical length.

The value and units must agree. For example, do not enter a value in feet while labeling the units as centimeters.

10.2.2 Units

Examples include:

- ft
- m
- cm

The units become part of the Site Configuration and the resulting stage output.

10.2.3 Bottom left octagon corner waterlevel

This is the known physical water-level value assigned to the bottom-left octagon corner.

Conceptually:

```
stage = bottom-left-corner waterlevel + local calibrated height
```

This is the physical datum for stage calculation.

10.3 Loaded versus Active projection

The Site Configuration information panel can show:

Loaded Information persisted in the loaded Site Configuration.

Active projection The calibration as projected onto the currently selected image after the octagon has been located for that image.

These are related but not identical concepts.

The loaded configuration is persistent reference information. The active projection shows how that reference currently maps onto the selected image.

10.4 Creating a Site Configuration

A careful creation sequence is:

1. select a representative calibration image;
2. verify or edit the octagon ROI;
3. verify or edit the waterline ROI;
4. enter the site name;
5. enter the operator;
6. enter octagon facet length;
7. enter units;
8. enter Bottom left octagon corner waterlevel;
9. set Grid divisions per octagon;
10. select small-octagon mode only when appropriate;
11. select **Create**;
12. choose the destination JSON filename.

10.5 Grid divisions per octagon

The grid-density control ranges from approximately 0.5 to 5.0 divisions per octagon dimension, in 0.25 increments. The normal default is 1.5.

Grid density is a physical display parameter. It is not driven by browser zoom.

The horizontal reference remains anchored to the bottom octagon geometry.

10.6 Large-image calibration diagnostics

RC8 uses a protected grayscale large-image calibration path when the octagon ROI is large enough to qualify for adaptive processing.

The desired calibration perimeter is the **inner octagon boundary**. This is the boundary between the calibration-target face and its black border. The outer edge of that border, its width, and the scene outside the border are not used as calibration geometry.

A successful large-image calibration normally reports:

`adaptive_large_roi_grayscale_inner_facets`

The processing sequence is:

1. perform reduced-resolution grayscale coarse localization;
2. use the coarse result only to estimate approximate center, scale, and orientation;
3. search the full-resolution grayscale image broadly along the eight expected inner facets;

4. retain multiple polarity-correct line hypotheses when appropriate;
5. robustly fit distributed edge support for each candidate facet;
6. evaluate facet combinations under one globally projective octagon model;
7. select the globally consistent set of measured facet lines;
8. calculate the final precision corners from adjacent measured-line intersections.

The projective model is an acceptance and global-consistency mechanism. It does not replace the measured corners with idealized projected corners.

A fully closed contour is not required. Boundary sections may be interrupted or partly obscured when sufficient distributed support remains to determine the physical facet.

Edge polarity is an eligibility condition, not a complete decision rule. Unrelated scene edges can have the same polarity, so RC8 resolves competing facet hypotheses using the complete projective geometry of the octagon.

If an eligible large-image calibration cannot establish a credible inner octagon, GRIME3 fails calibration safely rather than falling back to an unrelated smaller octagon-like feature.

Calibration information includes:

- Detection path;
- Candidate count;
- Selected score;
- Octagon ROI size;
- Adaptive large-ROI eligible.

10.6.1 Native source-image resolution

The final detected calibration octagon must be at least **100 pixels wide and 100 pixels high in the original source image**.

The hard scientific acceptance rule remains 100 by 100 source-image pixels. Force small-octagon mode does not override it.

The browser presents the rule as:

- red when either detected dimension is below 100 pixels;
- yellow when both dimensions pass the hard minimum but at least one is below 125 pixels;
- no routine resolution message when both dimensions are at least 125 pixels.

The 125-pixel level is advisory only. Upsampling an image does not create missing calibration geometry.

11 Understanding the diagnostic overlays

11.1 Why overlays are important

An accepted result should still be visually auditable.

Overlays make important intermediate geometry visible so that a person can compare the algorithm's result with the physical scene.

11.2 Found Octagon

The Found Octagon overlay shows the detected/refined calibration target.

The current display uses green corner markers and red perimeter lines.

The overlay should follow the physical target rather than a smaller internal feature.

11.3 Grid

The Grid overlay shows calibrated physical spacing in yellow.

The horizontal grid family is anchored to the physical bottom facet of the octagon. One yellow horizontal line passes through the bottom-left and bottom-right octagon corners. All other yellow horizontal lines are spaced relative to that bottom-facet datum.

This is intentionally different from arbitrarily anchoring the grid to calibration-world $Y = 0$.

Grid divisions per octagon controls physical grid density. The setting is deterministic and is not driven by browser zoom. Changing Fit, 1:1, or zoom percentage changes only display scale; it does not alter calibration or physical grid spacing.

11.4 Staff Gauge

The Staff Gauge overlay is a synthetic calibrated scale drawn from the current Site Configuration. Its tick spacing adapts to the displayed calibration geometry while preserving the physical stage relationship.

The stage datum is defined by the entered **Bottom left octagon corner waterlevel**. Calibration-world Y and hydrologic water level therefore need not share the same zero.

If `bottomLeftWorldY` is the calibrated world-Y coordinate of the bottom-left octagon corner, then:

```
stage = bottomLeftOctagonCornerWaterlevel
        + (worldY - bottomLeftWorldY)
```

Actual stage 0.0 therefore occurs at:

```
worldY = bottomLeftWorldY
         - bottomLeftOctagonCornerWaterlevel
```

GRIME3 projects this true stage-zero location as a blue line and labels it:

```
0.0 water level
```

The blue zero-level line is distinct from the yellow calibration grid. Its label is positioned away from the octagon and, when practical, away from the synthetic staff gauge so that it does not obscure the calibration target.

11.5 Ruler

The ruler uses two adjustable endpoints.

It reports both image-space and calibrated physical information when a valid calibration is available.

11.6 Found Waterline

The selected waterline is drawn prominently on the image.

Rejected or alternate line diagnostics can also be displayed when available.

11.7 Found Waterline Points

Found Waterline Points shows evidence used in fitting/choosing the line.

The overlay may include:

- coherent candidate points associated with the selected result;
- rejected or outlier points marked separately.

Use this when a fitted waterline looks surprising.

12 Measuring one selected image

12.1 Why one image comes before a folder

A large batch should not be the first test of a new site configuration.

A single representative image lets the operator verify:

- the correct site is loaded;
- the octagon is found correctly;
- the active calibration is plausible;
- the waterline ROI is correctly projected;
- the selected waterline is physically credible;
- units and datum are correct.

12.2 Recommended single-image sequence

1. Load the image folder.
2. Select a representative image.
3. Load or create the correct Site Configuration.
4. Enable Found Octagon.
5. Inspect the Active projection.

6. Enable Grid or Staff Gauge if useful.
7. Verify the waterline ROI.
8. Select **Measure selected image**.
9. Inspect the result status.
10. Enable Found Waterline.
11. Enable Found Waterline Points when diagnostic detail is useful.
12. Confirm the reported stage and units are plausible.

12.3 Accepted, Not Accepted, and Error

Accepted The selected waterline passed current acceptance criteria.

Not Accepted Processing completed, but no candidate waterline passed those criteria.

Error Processing could not complete successfully.

Not Accepted and Error are not equivalent.

12.4 What to inspect when the waterline looks wrong

Check:

1. Found Octagon;
2. Active projection;
3. waterline ROI placement;
4. Found Waterline Points;
5. alternate/rejected line diagnostics;
6. image quality and reflections;
7. whether the Site Configuration belongs to the site/image geometry.

13 Measuring a folder or folder tree

13.1 What folder measurement does

Folder measurement repeats the selected measurement workflow across multiple images while writing run-level outputs and provenance.

13.2 Timestamp source

RC8 supports these timestamp choices:

Auto Try EXIF first, then filename when a filename format is supplied.

EXIF Use EXIF metadata.

Filename Parse the timestamp from the filename using the supplied format string.

None Do not derive an image timestamp.

13.3 Filename timestamp format

When Filename mode is used, enter a Python-style date/time format matching the filenames.

A common example is:

`%Y%m%d_%H%M%S`

The format must actually match the naming convention of the dataset.

13.4 Show EXIF for selected image

Show EXIF for selected image displays available EXIF metadata for the active image in Messages / Results.

This is diagnostic only. It does not change the selected image or measurement settings.

13.5 Configure outputs

Folder runs can write:

- summary JSON;
- summary CSV;
- decision-overlay images.

Decision-overlay scale can be adjusted from 10% through 100%.

Every folder run also writes:

`run_manifest.json`

13.6 Recommended small-folder test

Before processing a large dataset:

1. create a small representative subset;
2. run it with the intended Site Configuration;
3. inspect accepted and rejected cases;
4. verify output paths;
5. verify timestamps;
6. inspect decision overlays;
7. inspect the run manifest.

13.7 Starting and stopping

Select **Measure folder** to begin.

Results appear progressively in Folder / Batch Results.

Stop folder run lets the current image finish, then stops before starting the next image.

13.8 Reviewing completed rows

Select a completed result row to return to its source image and inspect the overlays associated with that result.

14 Understanding and preserving the outputs

14.1 The outputs are part of the measurement

A stage value without information about how it was produced is much less useful scientifically.

Preserve the machine-readable outputs and provenance with the data.

14.2 CSV

The CSV summary is convenient for spreadsheets, plotting, and later statistical analysis.

Preserve enough identifying information to connect each row back to its source image and run.

14.3 JSON

The JSON summary preserves structured output in a machine-readable form.

14.4 Decision overlays

Decision-overlay images provide a visual audit trail.

They are especially useful when reviewing:

- borderline accepted measurements;
- rejected measurements;
- images with reflections, foam, debris, or unusual lighting;
- suspected calibration problems.

14.5 run_manifest.json

The run manifest records run-level provenance such as:

- GRIME3 build identity;
- Site Configuration identity;
- run settings;

- timestamp settings;
- output settings.

Keep the manifest together with the result files.

14.6 Recommended output organization

A simple organization is:

```
site_name/
    images/
    site_config.json
    run_2026_09_11/
        run_manifest.json
        summary.csv
        summary.json
        decision_overlays/
```

The exact directory names can differ. The important point is to preserve the relationship between source data, configuration, and output.

15 A careful field-evaluation workflow

15.1 Goal

The goal is not merely to obtain numbers. The goal is to obtain results whose calibration and waterline decisions can be inspected and reproduced.

15.2 Step 1: organize by site

Keep images from different physical sites logically separated.

Do not reuse a Site Configuration merely because two sites look similar.

15.3 Step 2: choose a representative image

Choose an image where:

- the complete octagon is visible;
- the waterline region is visible;
- image quality is representative;
- the scene geometry is typical for that site.

15.4 Step 3: load or create the Site Configuration

If an RC8 Site Configuration already exists, load it.

For a new site, author the ROIs and physical calibration values, then use Create.

15.5 Step 4: visually verify active geometry

Turn on diagnostic overlays and confirm:

- Found Octagon follows the physical target;
- the waterline ROI projects correctly;
- Grid and Staff Gauge are plausible;
- Active projection information is credible.

15.6 Step 5: perform the large-image check

For a high-resolution field image:

1. confirm that the octagon ROI contains the complete physical target;
2. enable Found Octagon;
3. verify that the detected perimeter follows the inner boundary between the target face and black border;
4. do not use the outer black-border edge as the intended calibration boundary;
5. inspect Detection path and Adaptive large-ROI eligible;
6. for a successful adaptive case, expect `adaptive_large_roi_grayscale_inner_facets`;
7. enable Grid and verify that one yellow horizontal line passes through the bottom octagon facet;
8. verify that the blue `0.0 water level` line agrees with the virtual staff-gauge scale;
9. retain the complete diagnostic message if calibration is rejected.

15.7 Step 6: measure one image

Do not start the large run until the representative result is credible.

15.8 Step 7: run a small sample

Use a small sample containing easy and difficult images.

Inspect both Accepted and Not Accepted results.

15.9 Step 8: run the full dataset

After the small sample is satisfactory, run the complete folder or folder tree.

15.10 Step 9: review questionable measurements

Review saved decision overlays and completed result rows.

Use Found Waterline Points when the fitted line needs explanation.

15.11 Step 10: keep the audit trail

Preserve:

- source images;
- exact Site Configuration JSON;
- CSV;
- JSON;
- decision overlays;
- `run_manifest.json`;
- GRIME3 build identity.

16 Troubleshooting

16.1 The program will not start

Check:

1. the ZIP was completely extracted;
2. `GRIME3`.exe still has its `_internal` directory;
3. files from different releases were not mixed;
4. Windows security did not quarantine a required file;
5. the checksum matches this guide.

16.2 The browser does not open

Leave the GRIME3 console running and manually open:

`http://127.0.0.1:8000/`

16.3 The image folder does not load

Check:

- the path exists;
- the folder is readable;
- the expected image types are present;
- recursive mode is enabled if needed.

16.4 The Site Configuration does not load

Check:

- the path is correct;

- the file exists;
- it is a valid RC8 Site Configuration JSON;
- the file belongs to the intended site;
- the browser status/message text for the exact error.

16.5 The octagon overlay is wrong

Check:

1. the octagon ROI contains the complete physical target;
2. the correct Site Configuration is loaded;
3. the octagon is visible;
4. Detection path;
5. Candidate count;
6. Selected score;
7. Octagon ROI size;
8. Adaptive large-ROI eligible.

For an eligible large-image adaptive failure, retain the complete error text rather than trying to force a small substitute target.

16.6 The grid or staff gauge is wrong

Verify the Found Octagon first. Then verify:

- Active projection;
- facet length;
- units;
- bottom-left octagon-corner waterlevel;
- grid divisions per octagon.

16.7 The waterline is wrong

Inspect:

- waterline ROI;
- Found Waterline;
- Found Waterline Points;
- rejected/alternate line diagnostics;
- reflections and image artifacts.

16.8 A folder run encounters a corrupt image

The intended batch behavior is to record the problem where possible and continue rather than losing the entire run.

Keep the Error row and exact message.

16.9 Output files cannot be written

Verify:

- output directory exists or can be created;
- Windows permissions allow writing;
- the directory is not read-only;
- the destination is not being blocked by another application.

16.10 The browser looks stale

RC8 serves the browser shell and static resources with no-store cache headers, reducing stale-resource problems.

If something still looks wrong after replacing a release, close the old GRIME3 process and browser tab, then start the intended release again.

17 About, Help, Release notes, and Licenses

17.1 The About area

The top-level About tab contains three practical information areas:

- About/release identity;
- Help;
- Licenses.

Release notes are also presented in the About area.

17.2 Build identity and provenance

Every official RC8 package carries machine-readable build provenance.

Depending on the distribution, it records:

- GRIME3 version and build identifier;
- exact source Git commit;
- immutable release tag;
- clean source state;
- build timestamp;

- release channel;
- distribution type;
- platform and architecture.

Use the packaged `BUILD_INFO.json` and the distributed checksum rather than copying artifact-specific values from an older guide.

17.3 Help

The built-in Help summarizes the practical browser workflow:

- loading image folders;
- loading/creating Site Configuration;
- overlays;
- measurements;
- timestamps;
- folder runs;
- outputs;
- troubleshooting information.

17.4 Licenses

GRIME3 is licensed under the Apache License, Version 2.0.

The Windows package includes:

- `LICENSE`;
- `THIRD_PARTY_NOTICES.md`;
- `THIRD_PARTY_LICENSES/`;
- `THIRD_PARTY_LICENSE_INDEX.json`;
- native runtime license information for CPython, OpenSSL, libffi, and Microsoft runtime components.

The RC8 packaging workflow audits this material against the actual Windows build environment and packaged native binaries.

18 Known RC8 limits and expectations

18.1 This is a release candidate

RC8 means Release Candidate 6. It is evaluation software intended to be close enough to a production release for serious testing.

Testing should therefore focus on both scientific behavior and ordinary user workflow.

18.2 Unsigned Windows executable

The Windows executable is not digitally code-signed. SmartScreen may warn about it.

Use the SHA-256 checksum to verify release identity.

18.3 PyTorch/AI model is not bundled

The official RC8 Windows application contains the classical deterministic GRIME3 measurement workflow.

Future learned waterline research is outside the official RC8 package.

18.4 The console must stay open

The browser interface depends on the local GRIME3 server process. Closing the console stops that process.

18.5 Automated versus human testing

The exact RC8 executable was:

- built on a GitHub-hosted Windows x64 runner;
- constructed from immutable RC8 source tag `grime3-0.1.0-rc8`;
- source-tested before packaging;
- structurally audited;
- launched as the actual packaged `GRIME3.exe`;
- smoke-tested;
- checksum verified;
- license audited.

Human field-data testing is still necessary because an automated build cannot judge every real scene.

19 How to report a problem well

19.1 For any problem

Include:

- what you were trying to do;
- what happened;
- what you expected;
- exact error/status text;
- version;

- Build ID;
- Git commit;
- operating system.

19.2 For a measurement problem

Also include:

- source image filename;
- Site Configuration JSON;
- result status;
- saved decision overlay if available;
- screenshot with useful diagnostic overlays enabled;
- relevant calibration diagnostics.

19.3 For a startup problem

Include:

- screenshot or text from the console window;
- Windows version;
- exact ZIP filename;
- whether the SHA-256 matched;
- whether all files were extracted together.

19.4 Why this helps

A reproducible problem can be investigated efficiently.

The goal is to recreate the same input, configuration, software build, and failure.

20 Glossary

Accepted	A result whose selected waterline passed current acceptance criteria.
Active projection	The loaded calibration geometry after it has been projected onto the currently selected image.
Calibration	The mapping between image coordinates and physical world coordinates.
Decision overlay	A saved image showing diagnostic measurement information.
Facet	One side of the physical octagon.
Found Octagon	The detected/refined octagon geometry for the current image.

Found Waterline	The selected waterline result displayed on the image.
Found Waterline Points	Diagnostic candidate/inlier/outlier points associated with waterline fitting.
Grid divisions per octagon	A physical display-density setting for the calibrated grid.
Not Accepted	Processing completed, but no candidate waterline passed current acceptance criteria.
Octagon ROI	The region where GRIME3 searches for the calibration octagon.
ROI	Region of Interest; a bounded region telling GRIME3 where to search.
run_manifest.json	Run-level provenance file containing software/configuration/run settings.
Site Configuration	The canonical RC8 JSON containing persistent site identity, ROIs, calibration, physical units, datum, and related provenance.
Stage	The physical water-level result in the Site Configuration's units.
Waterline ROI	The quadrilateral region where GRIME3 searches for waterline evidence.

21 Quick-reference checklists

21.1 First installation checklist

- ☐ Correct RC8 ZIP downloaded.
- ☐ SHA-256 matches the release checksum file and Appendix A.
- ☐ ZIP fully extracted.
- ☐ `GRIME3.exe` and `_internal` remain together.
- ☐ GRIME3 starts.
- ☐ Browser opens at `127.0.0.1:8000`.
- ☐ About identifies build See `BUILD_INFO.json`.

21.2 Before measuring one image

- ☐ Correct image folder loaded.
- ☐ Representative image selected.
- ☐ Correct Site Configuration loaded or newly created.
- ☐ Found Octagon matches the physical target.
- ☐ Active projection looks plausible.
- ☐ Waterline ROI is correctly placed.
- ☐ Units and datum are correct.

- ☐ Found Waterline will be visually inspected.

21.3 Before a full folder run

- ☐ Representative single-image result verified.
- ☐ Small sample run completed.
- ☐ Timestamp source selected.
- ☐ Filename timestamp format verified if used.
- ☐ Output directory selected.
- ☐ CSV/JSON/overlay options selected.
- ☐ Decision-overlay scale selected.
- ☐ Site Configuration preserved.

21.4 After a folder run

- ☐ Accepted and Not Accepted rows reviewed.
- ☐ Error rows reviewed.
- ☐ CSV preserved.
- ☐ JSON preserved.
- ☐ Decision overlays preserved.
- ☐ `run_manifest.json` preserved.
- ☐ Exact Site Configuration preserved.

A RC8 release identity and distribution matrix

The immutable source tag is:

`grime3-0.1.0-rc8`

Final release artifacts are rebuilt from that tag.

Distribution	Entry point	RC8 format
Windows x64	`GRIME3.exe`	PyInstaller one-folder ZIP
Linux x86_64	`./grime3`	PyInstaller one-folder `.tar.gz`
Conda	`grime3`	Noarch GRIME3 package with Conda-resolved runtime dependencies
Docker/OCI	`/opt/grime3/grime3`	Minimal Linux/amd64 container baseline

Exact final checksums, build identifiers, and commits are generated by the immutable tag builds and travel with their corresponding artifacts.

A.1 What was validated before tagging

The final pre-tag candidate passed:

- the complete source test suite;
- the permanent 22-field scientific-reference comparison;
- fresh Conda build and fresh dependency-solved installation;
- fresh clean-environment Linux standalone build;
- fresh Docker image build;
- native Windows hosted validation;
- five-way Source/Conda/Linux/Docker/Windows scientific equivalence;
- provenance and licensing checks;
- runtime exclusion audits for Qt, PyTorch, SAM2, and GNU Readline where prohibited.

B One-page field workflow

1. Start GRIME3.
2. Load the image folder.
3. Select a representative image.
4. Load or create the site's Site Configuration JSON.
5. Turn on Found Octagon and verify the physical target.
6. Inspect Active projection and calibrated overlays.
7. Measure one image.
8. Inspect Found Waterline and Found Waterline Points.
9. If credible, configure timestamp source and output directory.
10. Run a small sample.
11. Review decision overlays and result statuses.
12. Run the full folder or folder tree.
13. Preserve the Site Configuration, CSV, JSON, overlays, and `run_manifest.json`.

C Credits and licensing note

GRIME3 is part of the GaugeCam / GRIME research and software effort.

Software: GaugeCam GRIME3.

Author of this guide: Dr. Kenneth W. Chapman, Adjunct Researcher, GRIME Lab, University of Nebraska-Lincoln.

GRIME3 is distributed under the Apache License, Version 2.0.

Third-party software remains under its respective licenses. The Windows release contains package and native-runtime license indexes describing the components included in the validated build.

D RC8 large-image calibration diagnostics

RC8 replaces the earlier color-dependent large-image calibration approach with a grayscale inner-facet method.

The physical boundary used for calibration is the inner octagon boundary between the calibration-target face and its black border. The outside edge of the black border is deliberately excluded because border width is not a calibration dimension and the surrounding scene is uncontrolled.

Successful diagnostic path

A successful adaptive large-image calibration normally reports:

```
adaptive_large_roi_grayscale_inner_facets
```

Acquisition and precision

Reduced-resolution grayscale detection establishes only an approximate center, scale, and orientation. Precision geometry is then recovered from the original full-resolution image.

For each expected inner facet, RC8 searches a broad full-resolution grayscale band and can retain multiple polarity-correct line hypotheses. Distributed supporting pixels are robustly fitted rather than requiring one unbroken contour.

The eight facet hypotheses are selected jointly under one projective model of the physical octagon. This rejects locally plausible distractor edges that are inconsistent with the complete target geometry.

The homography-based projective fit is used for global consistency and acceptance. Final precision corners remain intersections of the measured facet lines.

Interrupted facets and distractors

A facet can contain an interruption, partial occlusion, or locally missing boundary evidence and still be usable when sufficient distributed support remains.

Polarity alone is not sufficient for acceptance. Same-polarity distractor edges can exist in the scene, so the complete eight-facet projective geometry participates in selecting the physical target.

Failure behavior

If the large-image path cannot establish a credible inner octagon, calibration fails safely. It does not silently fall back to a smaller octagon-like feature inside the ROI.

One protected full-resolution failure path is:

`adaptive_large_roi_failed_inner_facet_acquisition`

The browser diagnostics also distinguish failures associated with reduced-resolution coarse localization from failures during full-resolution inner-facet acquisition.

Resolution rule

The final detected octagon must be at least 100 pixels wide and 100 pixels high in the original source image.

Below the hard minimum, the browser reports a red error. Between the hard minimum and 125 pixels in either dimension, it reports a yellow advisory. When both dimensions are at least 125 pixels, no routine resolution warning is displayed.

Calibration-grid and water-level datums

The yellow horizontal calibration-grid family is anchored to the physical bottom octagon facet.

The blue `0.0 water level` line represents actual hydrologic stage zero derived from the Bottom left octagon corner waterlevel. It is not calibration-world $Y = 0$.

These two datums are intentionally displayed separately so that geometric calibration and water-level stage remain visually distinguishable.