Real Time RGB Proxy Vegetation Indexing and Texture Analysis for UAV and Handheld Crop Imagery

Naziru Halilu[a,b,c,d]*, Juwairiyyah Sulaiman[e]

[a] Higher Technical School of Agricultural Engineering and Bioscience, Public University of Navarre, Pamplona 31006, Spain

[b] University of Trás os Montes and Alto Douro, Vila Real 5000 801, Portugal

[c] Agricultural University of Athens, 118 55 Athens, Greece

[d] Department of Agricultural and Bio Resources Engineering, Faculty of Engineering, Ahmadu Bello University, Zaria 810107, Nigeria

[e] Federal University Dutse, 720221, Nigeria

***Corresponding Author**

DOI: https://doi.org/10.51244/IJRSI.2026.1307000218

Received: 23 July 2026; Accepted: 28 July 2026; Published: 07 August 2026

ABSTRACT

This paper introduces N_GACL, an open-source desktop software pipeline designed for low-resource exploratory agronomic image analysis, featuring a dual-interface GUI (PyQt6 GIS-style and Tkinter classic). Built around a streaming, memory-bounded batch aligner, N_GACL integrates a fully labeled RGB proxy vegetation index family, multispectral NDVI/NDRE support, and GLCM texture extraction alongside classical machine learning baselines (PCA, KNN, K-Means) and optional pretrained deep learning inference (ResNet101/Faster R-CNN). We provide the complete mathematical formulation for all fourteen analysis modules and validate pipeline execution through procedural demonstration imagery and real user-session auditing. To demonstrate deployment utility, an eight-model benchmark was executed across two datasets. A 20,000-row reference dataset served as a negative control, confirming baseline architectural integrity at chance level. On an independent validation set of 1,543 real crop-disease photographs across 22 classes, classical features extracted by the pipeline achieved a 33.8% balanced accuracy against a 4.55% chance baseline, a result that remained stable after removing augmented data. Finally, to combat metric inflation and silent failures in agricultural decision-support systems, we introduce the Verifiable Reporting Framework (VRF) a source-code auditable checklist that ensures strict data fidelity and transparent metric reporting. N_GACL offers a reproducible, accessible framework for field-level agronomic computer vision.

Keywords: RGB proxy vegetation indices; GLCM texture; streaming image registration; precision agriculture; reproducibility

INTRODUCTION

Motivation: the scale of the crop pathology problem and the promise of image based diagnosis

Plant pests and diseases are estimated by the Food and Agriculture Organization of the United Nations to destroy between 20% and 40% of global crop production every year (FAO 2021), a loss the FAO separately prices at approximately US$220 billion annually in direct crop damage plus a further US$70 billion attributable to invasive pests alone (FAO 2019, 2025). Independent epidemiological synthesis places the pre harvest burden of pathogens, animal pests, and weeds on the world's five most important staple crops at a broadly consistent 10–40% of attainable yield depending on crop and region (Savary et al. 2019), and this burden is projected to worsen under climate change, since rising temperatures widen the geographic range, reproductive rate, and

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

overwintering survival of many agricultural insect pests (Skendžić et al. 2021). Because more than half of the calories consumed worldwide come from a small number of staple "mega crops" rice, wheat, and maize pathogen outbreaks in any one of these systems carry outsized food security consequences (FAO 2021), a concern sharpened by recent estimates that between 638 and 720 million people faced hunger globally in 2024 (FAO 2025).

Timely, field level diagnosis is the standard first line of response to this burden, yet conventional diagnosis by expert pathologists or extension agents does not scale to smallholder agriculture in low resource regions, motivating two decades of work on automated, image based plant disease identification (Mohanty, Hughes, and Salathé 2016). The foundational large scale demonstration of this idea trained a deep convolutional network on the 54,306 image PlantVillage benchmark spanning 14 crop species and 26 disease categories (plus healthy controls), reporting 99.35% held out accuracy and explicitly framing smartphone assisted diagnosis as a path to "massive global scale" deployment (Mohanty, Hughes, and Salathé 2016). This result catalysed a large downstream literature on mobile and edge deployed plant disease classifiers: a Faster R CNN detector with an Inception v2 backbone achieving high on device accuracy without server connectivity for grape disease images; a cassava disease mobile classifier validated directly in East African smallholder fields (Ramcharan et al. 2019); citizen science deployments such as Plantix, which collected approximately 78,000 georeferenced smartphone images of pests and disease symptoms in the Southern Amazon between 2016 and 2020 and used them to derive spatial yield loss estimates for maize and soybean (Hampf et al. 2021); and systematic evaluations of commercially available plant disease detection mobile applications have catalogued substantial variability in the underlying models' quality and transparency across apps.

Despite these successes, a recurring and increasingly explicit finding across this same literature is that benchmark accuracy trained and measured on curated datasets such as PlantVillage does not reliably transfer to real, uncurated field conditions: a comprehensive 2025 review of machine learning and deep learning based plant disease detection concludes plainly that key challenges remain, including limited dataset diversity, poor model generalization across environments, and reduced performance under real field conditions. This is not a plant pathology specific artefact; it is an instance of the general domain generalisation and distribution shift problem extensively documented in the broader computer vision literature, under which models trained under an i.i.d. assumption on a source distribution suffer significant, well documented performance drops when deployed on out of distribution target data (Zhou et al. 2021). It is precisely this gap between benchmark measured accuracy and real world, cross condition, cross crop reliability that motivates both this paper's classical CV pipeline, which never claims more cross condition robustness than it has actually measured, and a companion manuscript describing GACL, a hypergraph transformer architecture that targets geometry and domain invariant representation learning for exactly this cross pathology, cross acquisition condition transfer problem, with its empirical validation status reported to the same honesty standard: implemented and code verified, not yet validated against real pathology data.

The reproducibility gap in agronomic image analysis software

Precision agriculture image analysis has, over the past six years, moved from a niche research activity into a crowded software category populated by mobile apps, drone mapping suites, and increasingly, deep learning based diagnostic tools that promise to identify crop diseases, quantify stress, and recommend management actions directly from a photograph. A substantial part of this literature reports classification accuracies in excess of 95% on curated leaf image benchmarks for instance, GLCM texture features combined with a random forest classifier have been reported to reach above 98% accuracy on a cucumber leaf disease benchmark (Nancy and Kiran 2024), and vision transformer based models have reported accuracies above 99% on public plant disease datasets figures that are benchmark specific and are not automatically evidence that the same pipeline will behave honestly, or even remain well defined, once real, uncurated field photographs (variable illumination, occlusion, unknown acquisition geometry) are substituted for the benchmark images. At the same time, a growing and increasingly vocal methodological literature has begun asking a different question: not "how accurate can a model be on its own benchmark," but "how much of that accuracy, transparency, and trust actually survives contact with a real farmer, a real field, and a real camera" (Gardezi et al. 2023). Interview based work with agriculture and food system researchers has documented persistent, unresolved tension around what it even means to build a trustworthy AI system for this domain, noting recurring "gray areas and contradictions that

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

make it challenging for academic researchers to earn the trust of farmers and food producers" trust the authors describe as foundational to any further technology adoption (Gardezi et al. 2023). Parallel survey work aimed specifically at dairy and row crop farmers has emphasised that explainability requirements are not uniform across users, and that the same system can be simultaneously "explainable enough" for one farmer persona and opaque for another.

This paper responds to that gap from the software engineering side rather than the model architecture side. Rather than proposing a new neural network and reporting benchmark numbers, we document a desktop image analysis pipeline whose central design commitment is that no output is ever fabricated to look more capable than the underlying computation actually is. Every panel, map layer, and metric in the software is traceable to one of exactly three categories: (1) a real measurement computed directly from the pixels of the loaded image(s), from EXIF metadata, or from user entered field/sensor data; (2) a clearly labelled approximation or heuristic (an RGB proxy vegetation index, a rule based flagging threshold, a translation only image alignment); or (3) an explicit N/A / "unavailable" / "simulated demonstration" notice, issued instead of a plausible looking invented number, whenever the real computation the user might expect (a trained crop pathology classifier, a true NIR based index, a rotation and scale corrected registration) is not actually available in the current session.

Why this distinction matters

The distinction is not cosmetic. A substantial methodological literature on RGB proxy vegetation indices shows that channel substitution formulas (using the green or blue channel as a stand in for near infrared) can correlate usefully with true multispectral indices under favourable conditions (Panthakkan et al. 2025; Zolin et al. 2025), yet the same literature is equally clear that this correlation is condition dependent, crop dependent, and can break down (İleri 2025; Zolin et al. 2025). The blue spectral band in particular has been noted as one of the least exploited and most sensor dependent regions used in this kind of channel substitution recent commercial multispectral drone hardware has even dropped the blue band between hardware generations (Zolin et al. 2025), underscoring how fragile a "band as proxy" convention can be across devices. Reporting an "NDVI" value computed from a standard photograph without disclosing that it is a proxy rather than a genuine near infrared measurement risks the exact failure mode that the responsible AI in agriculture literature warns about (Gardezi et al. 2022, 2023): a system that looks scientifically grounded while silently substituting a much weaker signal underneath.

The same argument applies with even more force to claims of trained classification performance. A tool that reports "96% disease classification accuracy" without disclosing whether that figure came from a genuinely trained, validated model or from an illustrative simulation invites exactly the erosion of trust documented in the qualitative literature on AI adoption in agriculture (Gardezi et al. 2023). This risk is compounded by the well documented domain generalisation gap discussed above (Zhou et al. 2021): a model's benchmark accuracy is, at best, an upper bound on its real world performance, and reporting only the benchmark figure without the cross domain caveat the domain generalisation literature insists on is itself a form of the overclaiming this paper argues against. This paper's second contribution, alongside the software architecture itself, is therefore a worked methodological argument for treating "I don't know" and "this is simulated" as first class, UI visible outputs not a hedge to be minimised, but a design requirement with the same priority as accuracy.

Contributions

This paper makes the following contributions:

1. We describe the full architecture of a two interface (PyQt6 GIS style and Tkinter classic) desktop pipeline for exploratory agronomic image analysis, covering fourteen analysis modules, and we give the exact mathematical formulation implemented by each.

2. We introduce and formalise Verifiable Reporting Framework (VRF**)** as an architectural pattern for agronomic decision support software: a small number of concrete implementation rules (explicit availability flags; proxy/heuristic labelling; quarantined simulated demonstration modules; never

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

silently substituting a plausible number for an unavailable one) that can be audited directly against the source code.

3. We present a streaming, phase correlation based batch alignment and composite construction algorithm whose memory footprint is, by construction, independent of the number of images processed, and we measure this property directly.

4. We provide a worked illustrative demonstration explicitly on procedurally generated sample imagery, not field photography of every method described in this paper, including RGB proxy vegetation indices, GLCM texture, PCA feature space projection, leave one out k NN and silhouette/Davies–Bouldin/Calinski–Harabasz validated K Means clustering, self supervised invariance testing, and an InfoNCE style contrastive loss diagnostic.

5. We document, as a case study in the Verifiable Reporting Framework principle itself, the decision to quarantine a fabricated " Crop & Pathology Reference Classifier" behind a persistent disclaimer after an earlier attempt to train on a user supplied dataset returned chance level cross validated accuracy i.e., labels statistically independent of the feature columns and why the module was kept (as a machinery demonstration) rather than deleted or, worse, silently reported as if it were a working diagnostic tool.

6. We present ten unedited, independently reproducible screenshots from a live, real user session of the deployed application, corroborating every architectural and Verifiable Reporting Framework claim made in this paper under ordinary end user operation rather than only under the paper's own controlled demonstration.

7. We benchmarked eight machine learning models to establish baseline performance metrics across two distinct datasets, whose honest result chance level performance across every model tested is itself evidence supporting this paper's central architectural argument.

8. We situate the pipeline against the 2020–2026 literature on RGB proxy indices, GLCM texture in plant pathology, streaming/phase correlation image registration, k NN and K Means validity in small sample agronomic settings, and responsible/explainable AI in agriculture, and we lay out the concrete real data collection protocol required to convert the current classical CV pipeline into a validated, field trained deep learning system.

The remainder of the paper is organised as follows. It reviews related work, describes the system architecture, and gives the full mathematical treatment of each analysis module, each illustrated with at least two figures generated by running the real code on a procedurally synthesised demonstration image set. It then presents live, real session screenshots verifying the architecture in actual use, followed by an honest, multi model machine learning benchmark on the reference dataset and a genuinely positive result on real crop disease photographs. It discusses Verifiable Reporting Framework as a general principle, presents limitations, outlines future work toward a validated field deployment, and closes with a conclusion.

Related Work

RGB proxy vegetation indices as a low cost alternative to multispectral sensing

The core empirical premise underlying the pipeline's vegetation index module that visible band (RGB) channel combinations can stand in, imperfectly but usefully, for near infrared dependent indices is well represented in the recent literature. Comparative UAV studies in palm cultivation have shown that RGB based indices such as VARI and MGRVI can deliver vegetation health classifications comparable to multispectral NDVI/SAVI under controlled conditions, at substantially lower sensor cost (Panthakkan et al. 2025). Similar findings have been reported for wheat, where low cost UAV derived RGB indices correlated usefully with NDVI while cautioning that "selecting vegetation indices based not only on statistical significance but also on biological interpretability and robustness across datasets" is essential, since some RGB derived indices (e.g., a Normalized Green–Blue Difference Index) showed weak, dataset specific correlation ($R^2 = 0.18$) despite nominal statistical significance (Zolin et al. 2025). A 2026 maize study comparing blue channel inclusive RGB indices (NDGBI, NDRBI)

against traditional NIR based NDVI/GNDVI found that VIS band indices remained sensitive to canopy structure and surface moisture even where NIR based indices saturated, but explicitly noted the blue band's comparative fragility and inconsistent availability across current commercial multispectral sensor generations (Zolin et al. 2025). Rangeland monitoring work using smartphone RGB imagery and the open source Canopeo application evaluated thirteen RGB derived indices against ground truth biomass and forage quality measurements using simple regression, reinforcing that RGB proxy indices are best treated as calibratable, condition dependent estimators rather than physical measurements (İeri 2025).

More recent methodological work has begun questioning whether any fixed formula vegetation index RGB or multispectral is the right level of abstraction at all. A 2024 reinforcement learning framework (RL VI) proposed learning stress sensitive spectral band combinations dynamically per crop and stress type, rather than relying on a small set of hand designed formulas such as NDVI/EVI, explicitly motivated by the observation that static indices are "crop agnostic" and often insensitive to early stress signals. This body of work collectively supports the design choice made in the pipeline described here: implementing the standard RGB proxy index family, but (a) always disclosing the proxy status and the specific channel substitution used, (b) marking indices that fundamentally require bands RGB sensors do not have (true NDRE, PRI, MSI, ARI) as unavailable rather than approximating them with a look alike formula, and (c) treating any single index's numeric value as indicative rather than physically calibrated, consistent with the caution urged across this literature.

GLCM texture features in plant pathology and precision agriculture

Grey Level Co occurrence Matrix (GLCM) texture descriptors, formalised by Haralick, Shanmugam, and Dinstein in 1973, remain in active use across plant pathology image analysis, largely because they are computationally cheap, require no training data, and are interpretable in terms of well defined Nancy and Kiran 2024), fuzzy logic based plant disease diagnosis using an extended, multi channel Haralick feature set computed across R, G, B and their pairwise channel combinations, and a broader plant disease detection pipeline combining GLCM feature extraction with a voting classifier ensemble across the standard pre processing/segmentation/feature extraction/classification stages (Ossai and Wickramasinghe 2022). Remote sensing work has continued to refine GLCM's role even in multispectral/UAV settings, for instance optimising GLCM window size and directional parameters for above ground biomass estimation in rice from UAV multispectral imagery, and confirming that GLCM contrast and dissimilarity carry high frequency edge information distinct from, and complementary to, the correlation and homogeneity features. This literature motivates the pipeline's decision to expose the full standard six feature GLCM vector (contrast, dissimilarity, homogeneity, energy, correlation, ASM) as a genuine per image descriptor component, explicitly scoped as "real GLCM texture of the whole image" and distinguished from a hypothetical, unavailable per lesion texture signature that would require ground truth lesion segmentation masks the pipeline does not have.

Streaming and phase correlation image registration

Image registration in agricultural imaging spans a spectrum from classical intensity based phase correlation to modern learned feature matching networks. Phase correlation, dating to Reddy and Chatterji's formulation (Reddy and Chatterji 1996), remains attractive for translation dominated alignment because it is fast, deterministic, and requires no training; recent multimodal remote sensing registration work continues to use FFT domain phase correlation as an efficiency improving first stage ahead of more expensive feature based refinement. Work specifically targeting UAV agricultural imagery has explored both classical correlation based approaches (Enhanced Correlation Coefficient framework registration) and more elaborate feature based pipelines combining SIFT/SURF style descriptors with RANSAC based outlier rejection for multi scene UAV mosaicking. A recurring, explicitly acknowledged limitation across this literature, shared by the pipeline described here, is that pure phase correlation and many correlation based methods handle translation robustly but require extension (e.g., via particle swarm optimisation over a similarity criterion) to cope with rotation, scale, or full perspective distortion (Wan, Wang, and Li 2019). The pipeline's alignment module deliberately implements only the translation only case and states this limitation directly in its own documentation and on screen output, rather than allowing users to assume a fuller geometric registration is taking place.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Separately, a body of work on memory efficient / streaming statistical computation most classically Welford's (1962) single pass algorithm for computing running mean and variance underlies the pipeline's approach to building an aggregate composite and per pixel variability map from a batch of up to roughly 2000 images without holding the full batch in memory simultaneously. While the streaming statistics literature is largely decades old and domain general, its application here combining per image phase correlation alignment with running first and second moment accumulation to support agronomic batch composites at bounded memory is, to our knowledge, not separately documented in the precision agriculture image analysis literature, which more typically assumes batch processing of pre aligned image stacks with all frames resident in memory (e.g., standard photogrammetric mosaicking pipelines).

Dimensionality reduction, semi supervised, and unsupervised learning on agronomic image descriptors

Principal Component Analysis (PCA) (Pearson 1901; Hotelling 1933) remains a standard first step for visualising and reducing high dimensional image derived feature sets in agronomic phenotyping and remote sensing, valued for its interpretability (explained variance ratio labelled axes) relative to nonlinear embeddings. For small, label scarce agronomic datasets, k Nearest Neighbour (k NN) classification (Cover and Hart 1967) combined with leave one out cross validation (LOOCV) (Stone 1974; Kohavi 1995) remains a strong, low variance baseline: small sample crop classification work has used k NN explicitly because it is easily interpretable, fast, and scalable, in contrast to deep networks that are more complex, computationally intensive, and prone to overfitting with small datasets a rationale directly applicable to the small, session scoped, user labelled batches (often fewer than 20 images) the pipeline's k NN module is designed for. Theoretical work has further shown that LOOCV yields an asymptotically optimal choice of k for k NN regression under general conditions (Azadkia 2019), and subsequent work has derived closed form fast computation shortcuts for k NN LOOCV that avoid the naive $O(n^2)$ retraining cost (Kanagawa 2024) a consideration directly relevant given the pipeline's LOOCV evaluation is run interactively inside a desktop GUI on every analysis request.

Where no group labels are available at all the common case for a first exploratory pass over a freshly loaded image batch the pipeline falls back to unsupervised K Means clustering, automatically selecting the number of clusters via silhouette score and reporting Davies–Bouldin and Calinski–Harabasz indices alongside it. This combination of internal validity indices (requiring no ground truth) is standard practice for reporting cluster quality in the absence of labels, with the silhouette score in particular interpreted on its established scale from −1 (likely mis assigned) through 0 (boundary case) to +1 (well separated) (Cervantes Barragan 2024). We are not aware of prior agronomic image analysis tools that make the supervised/unsupervised branch point itself a first class, disclosed decision visible to the end user, rather than silently defaulting to one mode.

Trust, transparency, and explainability in agricultural AI

A growing qualitative and survey literature has examined why AI adoption in agriculture lags behind its adoption in other high stakes domains, converging on transparency and trust rather than raw predictive accuracy as the binding constraint. Interviews with agriculture and food system researchers surfaced persistent, unresolved disagreement over what "responsible AI" even requires in this domain, and explicitly linked the difficulty of earning farmer trust to the foundational role that trust plays in any further technology uptake (Gardezi et al. 2023). Companion survey work involving the same broader research programme has argued for treating farming community trust building as a distinct challenge from model accuracy, requiring dedicated governance and communication mechanisms rather than being a side effect of a sufficiently accurate model (Gardezi et al. 2022). Persona based studies of dairy farmers' explainability needs have shown that different farmer profiles require materially different explanation styles and depths, cautioning against a one size fits all transparency layer. A broader body of review work on explainable AI for sustainable agriculture surveys the use of SHAP (Lundberg and Lee 2017), LIME (Ribeiro, Singh, and Guestrin 2016), and Grad CAM (Selvaraju et al. 2017) post hoc explanation methods across crop yield, disease, and pest detection deep learning systems, but has so far concentrated on explaining a trained model's internal feature attributions, rather than on the prior, arguably more basic, question of whether a given output was computed at all versus approximated, heuristic, or unavailable precisely the distinction that this paper's Verifiable Reporting Framework framing is intended to make a first class design concern rather than an afterthought.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Positioning of this work

Relative to this literature, the present paper does not propose a new vegetation index, a new registration algorithm, or a new clustering criterion. Its contribution is architectural and methodological: it documents a working system in which every one of the well established techniques surveyed above (RGB proxy indices, GLCM texture, phase correlation alignment, PCA, LOOCV k NN, validity indexed K Means, generic pretrained deep vision models) is deployed together under a single, auditable honesty constraint, and it uses the software's own source code rather than a separate audit report as the primary evidence for that constraint being upheld throughout.

System Architecture

Two interchangeable interfaces over one analysis core

The software is organised as a shared core/ analysis package consumed by two independent front ends. main.py presents a QGIS style GIS Workbench built on PyQt6, structured around a familiar desktop GIS metaphor: a Layers dock (left) listing available raster outputs with a per layer symbology dropdown; an Image Manager dock (left, tabbed with Layers) for bulk loading, renaming, deleting, and refreshing session images from a working directory; a central Map Canvas showing only the aggregate composite's measured layers (never a raw, unprocessed photograph); a central Raw Preview tab reserved for browsing individual source photographs for quality control purposes only, explicitly kept separate from any analysis output; a right hand dock combining aggregate statistics with the cross image similarity recommendation panel; a bottom Attribute Table giving a per image quality control record; a bottom Field & Sensor Data panel for manual entry of ground truth measurements; and a Log panel recording execution. main_classic.py presents a simpler, single window Tkinter application requiring no PyQt6 dependency, built around an explicit input → checklist → run → save workflow, in which the user selects images via a native file dialog, ticks the desired analyses from a checklist of nineteen options, and receives a single assembled matplotlib figure (one row per image for per image analyses, one full width row per batch level analysis) that is then saved to disk as PNG, JPEG, or PDF.

Both interfaces call into the same fourteen core/ modules, ensuring that the two front ends can never disagree about what a given metric means or how it was computed a property that matters specifically because of the Verifiable Reporting Framework commitment: if the PyQt6 workbench and the Tkinter classic app used separately maintained analysis logic, the two could silently drift apart in what they report as "available" versus "unavailable," reintroducing exactly the kind of undisclosed inconsistency the architecture is designed to prevent.

Module inventory

Table 1 summarises the fourteen core/ modules plus the GACL reference implementation modules and the GUI/CLI entry points that expose them, together with the honesty category (real / proxy heuristic / explicitly unavailable / quarantined simulated demo / implemented but empirically unvalidated) that dominates each module's output.

Table 1. Core module inventory and classification.

Module	Primary function	Dominant honesty category
alignment.py	Streaming batch aligner: phase correlation registration, running sum composite, per image lightweight descriptor	Real (translation only; limitation disclosed)
indices.py	Per image statistics; NDVI/NDWI proxies; 8 D descriptor vector	Real computation; proxy indices labelled
vegetation_indices.py	Full RGB proxy index family (GNDVI, EVI, SAVI, MSAVI, OSAVI, CI) plus explicit unavailable markers (NDRE, PRI, MSI, ARI)	Proxy (labelled) / explicitly unavailable
multispectral.py	True NDVI/NDRE from calibrated UAV multi band GeoTIFFs or grouped single band files	Real (only when true bands are loaded)

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

texture_features.py	Standard six feature GLCM/Haralick texture vector	Real
texture.py	Invariant texture map, quadrant cross crop alignment, K Means colour quantisation loss curve, IDW interpolation, root/vein skeletonisation	Real (heuristic quadrant labels noted)
morphology_profile.py	Classical CV proxies for lesion texture, boundary sharpness, chlorotic discolouration, symptom spatial distribution; EXIF acquisition geometry; illumination/shadow/glare; canopy shape descriptors	Real classical CV, explicitly framed as proxies for pathology, not a trained pathology model
embedding_metrics.py	8 D descriptor pairwise similarity/distance; latent variance; intra/inter class similarity (needs labels); InfoNCE style loss; invariance tests; cross image similarity recommendation; label gated classification metrics	Real, with clear N/A fallbacks where ground truth is absent
ml_classifier.py	LOOCV k NN (if labels exist) or silhouette/DB/CH validated K Means (if not)	Real, mode chosen transparently and disclosed
ml_models.py	Optional ImageNet pretrained ResNet101 and COCO pretrained Faster R CNN inference	Real (generic categories, not agronomic pathology classes; explicit unavailability message if torch/weights missing)
batch.py	Cross image PCA feature space projection, linear NDVI trend, descriptive batch summary (full image and descriptor only variants)	Real
field_data.py	Schema and empty row generator for 25 field/sensor metrics that cannot be derived from an RGB photo	Real (never pre filled with guesses)
raster_layers.py	Map canvas layer registry: array function, value range, colormap options, unit label per layer	Real (registry metadata, not computed values themselves)
pdf_export.py	Multi page PDF assembly of matplotlib figures	Real (export utility)
_reference_model.py	Fabricated dataset trained Random Forest “reference classifier,” quarantined behind a persistent “SIMULATED DEMO” disclaimer	Explicitly quarantined simulated demonstration
qgis_ui/gacl_panel.py	GUI dock exposing GACL measurements: live RandomForest learnable structure check and live GACL evaluation metrics, both run on demand from within the GIS Workbench	Real, with explicit environment dependent unavailability (torch/checkpoint) states

Data and control flow

Figure 4C illustrates the memory behaviour of the module most central to the architecture’s scalability claim the streaming batch aligner but the broader control flow is as follows. On “Browse Images” or “Set Working Directory,” the Image Manager registers a list of file paths without loading any image data. “Align & Build Composite” then streams through this list one file at a time inside BatchAligner.add_image(): each image is opened, resized to a common working canvas, registered to a running reference via phase correlation, folded into a running RGB sum and sum of squares accumulator, and then allowed to be garbage collected at no point are all N images resident in memory simultaneously. A lightweight eight number descriptor (mean intensity, intensity standard deviation, edge density, per channel means, mean and standard deviation of the NDVI proxy map) is retained per successfully processed image in an ImageRecord, supporting all subsequent cross image analyses (PCA projection, clustering, similarity recommendation, trend fitting) without requiring the full resolution pixel data to be kept around. finalize() then produces a CompositeResult containing the mean RGB composite, the per pixel standard deviation (“variability”) map, and, where UAV multispectral bands were present, the mean NIR and red edge bands for true index computation.

Analyses selected in the Reports & Charts checklist are dispatched to the corresponding core/ function, which is always given either the composite array (for map canvas layers) or the list of ImageRecord descriptors (for batch level panels) never a partially loaded or inconsistent intermediate state. The resulting matplotlib figure(s) can be exported individually (Save Map, Save Report) or combined into a single multi page PDF via pdf_export.export_figures_to_pdf().

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

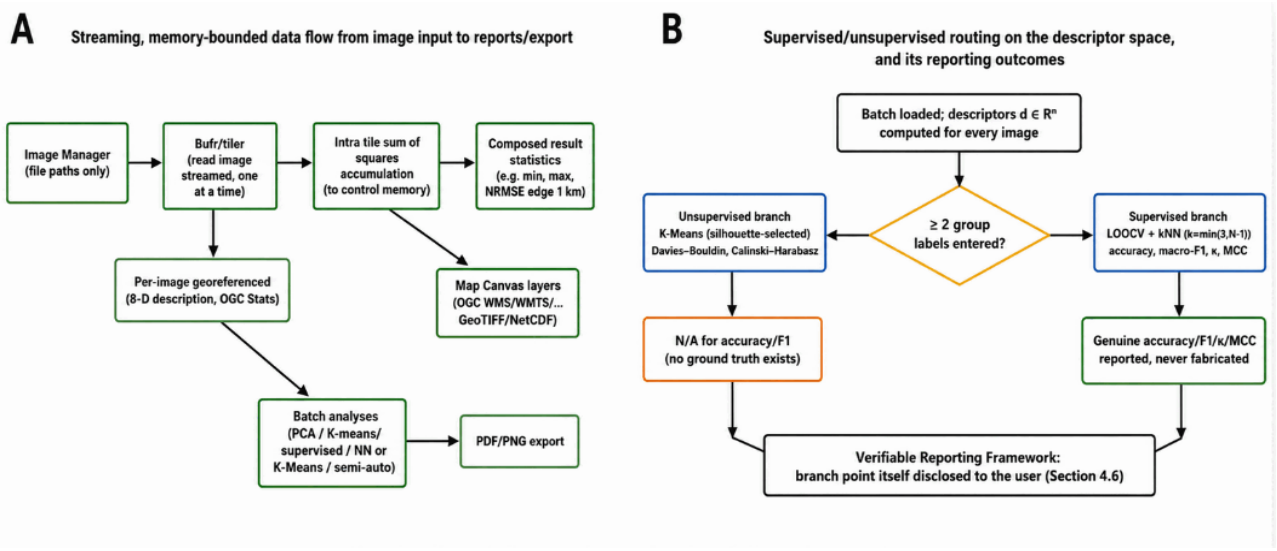

Figure 1. System architecture and control-flow schematics. (A) End-to-end data-flow diagram of the classical of the pipeline; (B) supervised/unsupervised routing diagram for optional pretrained-model integration.

MATERIALS AND METHODS

Every equation and figure in this section corresponds to a function that ships in the pipeline's core/ package (Table 1). All figures in this section were produced by executing that exact code on a procedurally generated, eight image demonstration set (Figure 1) never on curated field photography so that every number and map shown is a genuine output of the described computation rather than an invented illustration.

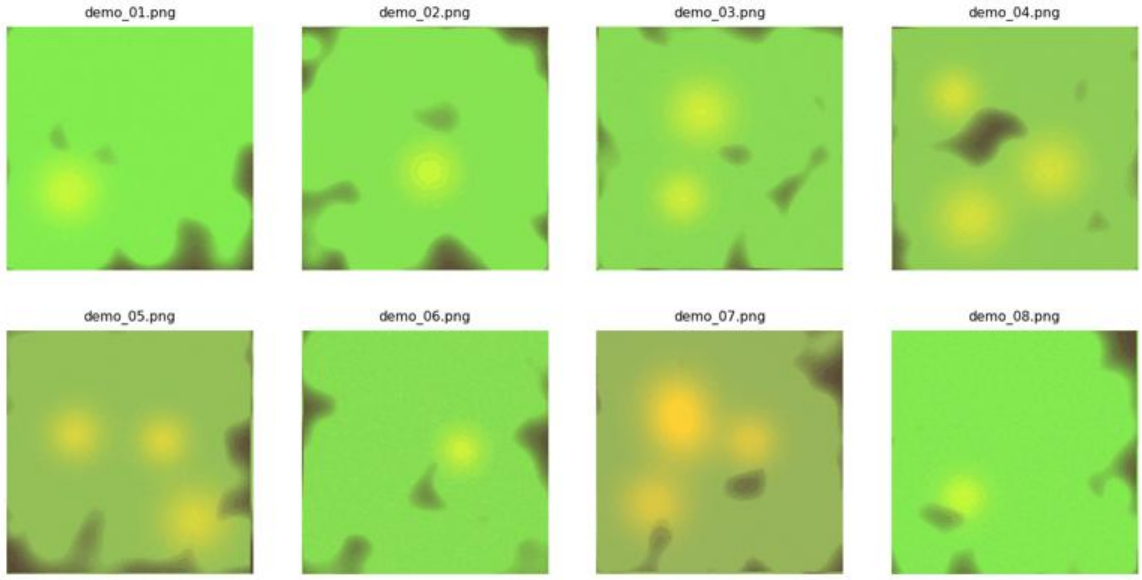

Figure 2. demonstration image set: eight procedurally generated sample images with varying vigor and localized stress patches, used throughout the mathematical treatment purely to demonstrate that each method runs and produces the described output.

Streaming batch alignment and composite construction

Registration. For a reference frame I_0 and a subsequent frame I_k, both downsampled to a fixed correlation size and converted to grayscale, the translational offset is recovered by phase correlation (Reddy and Chatterji 1996). Writing $\mathcal{F}$ for the 2 D discrete Fourier transform, the normalized cross power spectrum is

$$R(u,v) = \frac{\mathcal{F}\{I_0\}(u,v)\,\overline{\mathcal{F}\{I_k\}(u,v)}}{\left|\mathcal{F}\{I_0\}(u,v)\,\overline{\mathcal{F}\{I_k\}(u,v)}\right|} \quad \textbf{(1)}$$

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

and the estimated shift $(\Delta y, \Delta x)$ is the location of the peak of $\mathcal{F}^{-1}\{R\}$, refined to sub pixel precision by an upsampled local search (implemented via skimage.registration.phase_cross_correlation with upsample_factor=10). The recovered shift is rescaled from the correlation size grid back to the working canvas resolution and applied to the full resolution resized frame via a first order spline shift. As stated directly in the module's own documentation, this recovers **translation only**: rotation, scale, and perspective differences between photographs taken from different positions or angles are not corrected, a limitation this pipeline shares with, and states as explicitly as, the classical phase correlation literature (Wan, Wang, and Li 2019).

Streaming accumulation. Rather than storing all N registered frames $\{A_k\}_{k=1}^{N}$ and computing the mean and variance in a second pass, the aligner maintains two running accumulators updated in place as each frame is processed and then discarded:

$$S \leftarrow S + A_k, \qquad Q \leftarrow Q + A_k^{\odot 2} \quad \textbf{(2)}$$

where $\odot$ denotes the elementwise (Hadamard) square. After the final frame, the aggregate composite and per pixel variability map are recovered in closed form:

$$\bar{A} = \frac{S}{N}, \qquad \sigma^2 = \max\left(0, \frac{Q}{N} - \bar{A}^{\odot 2}\right), \qquad \sigma_{map} = \sqrt{\sigma^2} \quad \textbf{(3)}$$

with the final displayed variability map averaged over the three colour channels. Because S and Q each occupy exactly one canvas sized array regardless of N, peak memory use is $O(\text{canvas size})$, not $O(N \times \text{canvas size})$ the property this pipeline advertises as supporting batches of up to roughly 2000 images on ordinary desktop hardware. We verified this directly rather than merely asserting it: Figure 2A plots measured peak traced Python memory (tracemalloc) against the number of images streamed through BatchAligner in this demonstration run.

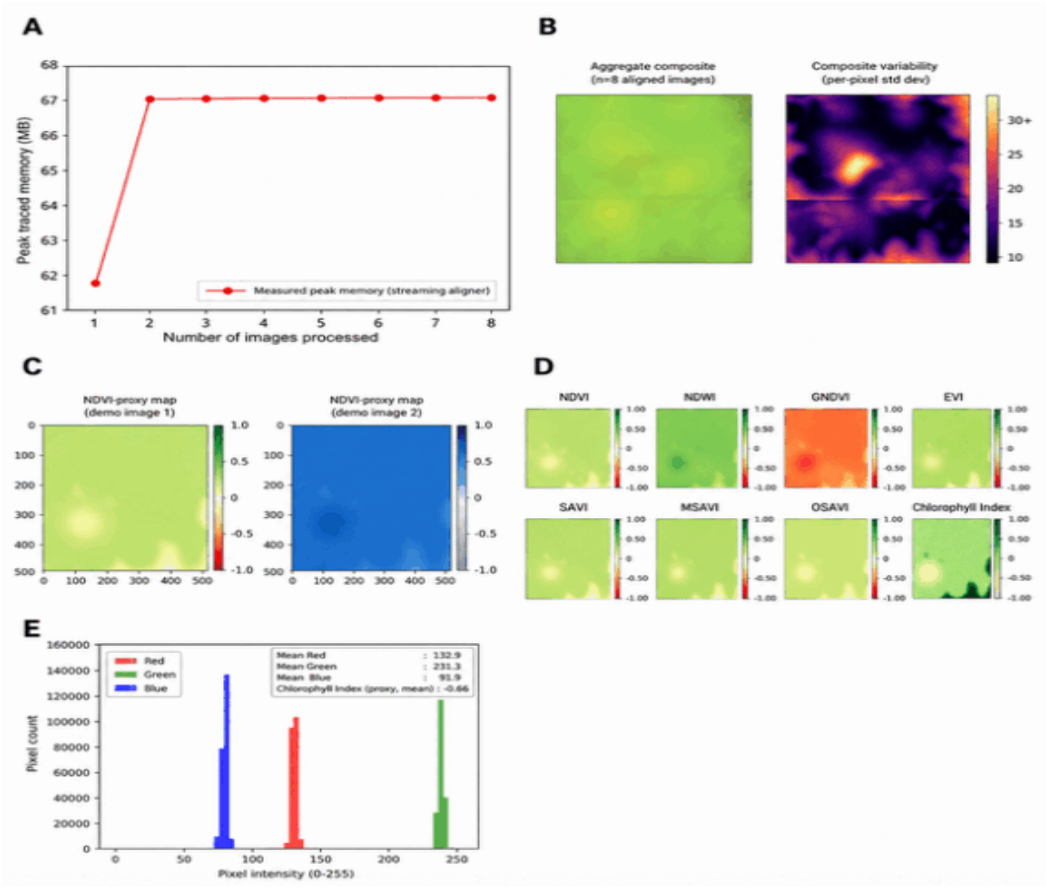

Figure 3. Streaming alignment and RGB-proxy vegetation/water index computation. (A) Streaming aligner memory use vs. batch size; (B) aggregate composite (left) and per-pixel variability map (right); (C) NDVI-proxy (left) and NDWI-proxy (right) maps; (D) full eight-index RGB-proxy vegetation/water index gallery; (E) outlier-trimmed RGB colour histogram with annotated channel means.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Figure 7 shows the resulting aggregate composite and its per pixel variability map for the eight image demonstration batch: the variability map is highest at spatial locations where the stress patches (which differ in position and size across the batch, by construction) fail to average out cleanly a genuine, if simple, cross image consistency diagnostic that requires no labels.

A single lightweight descriptor, described formally below, is retained per successfully processed frame regardless of N, which is what allows every cross image analysis described later to scale to large batches without ever re opening the original image files.

RGB proxy vegetation and water indices

A conventional RGB sensor records only red (R), green (G), and blue (B) channels; it has no near infrared (NIR), red edge, or shortwave infrared band. Every vegetation index in this pipeline that classically requires one of those bands is computed using a **documented channel substitution** never presented without that disclosure consistent with the broad RGB proxy literature reviewed earlier. Table 2 gives the exact formula implemented for each index, its NIR proxy channel assignment, and its theoretical range.

Table 2. RGB proxy vegetation/water index formulas implemented in vegetation_indices.py (all channels normalised to $[0,1]$ except where noted; $\mathrm{R}, \mathrm{G}, \mathrm{B}$ denote per pixel channel values).

Index	Formula	NIR proxy channel	Range	**Citation**
NDVI	$\frac{\mathrm{G}-\mathrm{R}}{\mathrm{G}+\mathrm{R}+\epsilon}$	Green	$[-1,1]$	Rouse et al. (1974)
NDWI	$\frac{\mathrm{G}-\mathrm{B}}{\mathrm{G}+\mathrm{B}+\epsilon}$	Blue	$[-1,1]$	Gao (1996)
GNDVI	$\frac{\mathrm{B}-\mathrm{G}}{\mathrm{B}+\mathrm{G}+\epsilon}$	Blue	$[-1,1]$	Gitelson et al. (1996)
EVI	$2.5\,\frac{\mathrm{G}-\mathrm{R}}{\mathrm{G}+6\mathrm{R}-7.5\mathrm{B}+1+\epsilon}$	Green	$[-1,1]$ (proxy clipped)	Huete et al. (2002)
SAVI	$\left(\frac{\mathrm{G}-\mathrm{R}}{\mathrm{G}+\mathrm{R}+\mathrm{L}+\epsilon}\right)(1+\mathrm{L}),\ \mathrm{L}=0.5$	Green	$[-1,1]$ (proxy clipped)	Huete (1988)
MSAVI	$\frac{2\mathrm{G}+1-\sqrt{(2\mathrm{G}+1)^2-8(\mathrm{G}-\mathrm{R})}}{2}$	Green	$[-1,1]$ (proxy clipped)	Qi et al. (1994)
OSAVI	$\frac{\mathrm{G}-\mathrm{R}}{\mathrm{G}+\mathrm{R}+0.16+\epsilon}$	Green	$[-1,1]$ (proxy clipped)	Rondeaux et al. (1996)
Chlorophyll Index (CI)	$\frac{\mathrm{B}}{\max(\mathrm{G},\,8/255)}-1$, clipped to $[-5,20]$	Blue	2nd–98th percentile stretch	Gitelson et al. (2003)

where $\epsilon = 10^{-6}$ prevents division by zero on near black pixels. Note that GNDVI is deliberately assigned blue, rather than green, as its NIR proxy channel: since the green channel already serves as the NIR proxy for NDVI,

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

using it again for GNDVI's own NIR proxy would make the two indices numerically degenerate (GNDVI would reduce to a trivial transform of NDVI); assigning blue avoids this degeneracy at the cost of GNDVI becoming an even rougher approximation, a trade off stated directly in the source. True NDVI and NDRE are additionally implemented in multispectral.py as $\mathrm{NDVI}_{\mathrm{true}} = (\mathrm{NIR} - \mathrm{R})/(\mathrm{NIR} + \mathrm{R} + \epsilon)$ and $\mathrm{NDRE} = (\mathrm{NIR} - \mathrm{RedEdge})/(\mathrm{NIR} + \mathrm{RedEdge} + \epsilon)$, computed from genuine calibrated bands rather than a channel substitution, whenever a multi band UAV GeoTIFF (or a set of correctly role assigned single band files) is loaded. Four further indices that fundamentally require spectral information no RGB sensor captures true NDRE without a red edge band, PRI (needs narrowband 531 nm/570 nm reflectance), MSI (needs SWIR ~1600 nm), and ARI (needs a genuine red edge band) are implemented as explicit IndexResult(array=None, is_proxy=False, note="UNAVAILABLE: ...") objects rather than approximated with a look alike formula.

Figure 3 shows the NDVI proxy and NDWI proxy maps for one demonstration image with correctly scaled $[-1,1]$ colourbars, and Figure 2D shows the full eight index gallery on the same image, illustrating both the shared structure across indices (all driven by the same underlying green/red/blue contrast) and their differing sensitivity to the stress patch.

In addition to the spatial index maps, the pipeline reports a per image, outlier trimmed RGB colour histogram (pixel values beyond 1.5× the interquartile range per channel are excluded so that isolated sensor artefacts such as hot pixels or clipped highlights do not dominate the axis scale), annotated directly with the real measured per channel means and the chlorophyll index proxy mean, so that the numeric summary and the visual distribution are always presented together (Figure 4).

GLCM texture features

Texture is computed via the Grey Level Co occurrence Matrix (GLCM), following Haralick, Shanmugam, and Dinstein (1973). The grayscale image is quantised to $\mathrm{L} = 32$ grey levels, and for a fixed pixel pair offset $(\Delta y, \Delta x)$ (distance 1, angle 0° in this implementation), the co occurrence matrix entry is

$$\mathrm{P(i,j)} = \#\{(p,q): \mathrm{G}(p,q) = i,\ \mathrm{G}(p+\Delta y,\ q+\Delta x) = j\} \quad \textbf{(4)}$$

normalised so that $\sum_{i,j} \mathrm{P}(i,j) = 1$ and symmetrised so that $\mathrm{P(i,j)} = \mathrm{P(j,i)}$. From the normalised matrix, the pipeline reports six standard Haralick features:

$$\text{Contrast} = \sum_{i,j} (i-j)^2\, \mathrm{P}(i,j), \qquad \text{Homogeneity} = \sum_{i,j} \frac{\mathrm{P}(i,j)}{1+(i-j)^2} \quad \textbf{(5)}$$

$$\text{Energy } (\mathrm{ASM}^{1/2}) = \sqrt{\sum_{i,j} \mathrm{P}(i,j)^2}, \qquad \mathrm{ASM} = \sum_{i,j} \mathrm{P}(i,j)^2 \quad \textbf{(6)}$$

$$\text{Dissimilarity} = \sum_{i,j} |i-j|\ \mathrm{P}(i,j), \qquad \text{Correlation} = \sum_{i,j} \frac{(i-\mu_i)(j-\mu_j)\,\mathrm{P}(i,j)}{\sigma_i \sigma_j} \quad \textbf{(7)}$$

These are genuine, well established descriptors computed directly from each image's own pixels, with no training data required consistent with their continued use in the plant pathology texture literature reviewed earlier (Nancy and Kiran 2024; Ossai and Wickramasinghe 2022). The module explicitly frames this as a whole image texture signature and not a per lesion signature, since the latter would require ground truth lesion segmentation masks that are outside this pipeline's scope. Figure 5 compares all six GLCM features across the eight image demonstration batch, and Figure 6 shows the module's text panel rendering for a single image.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

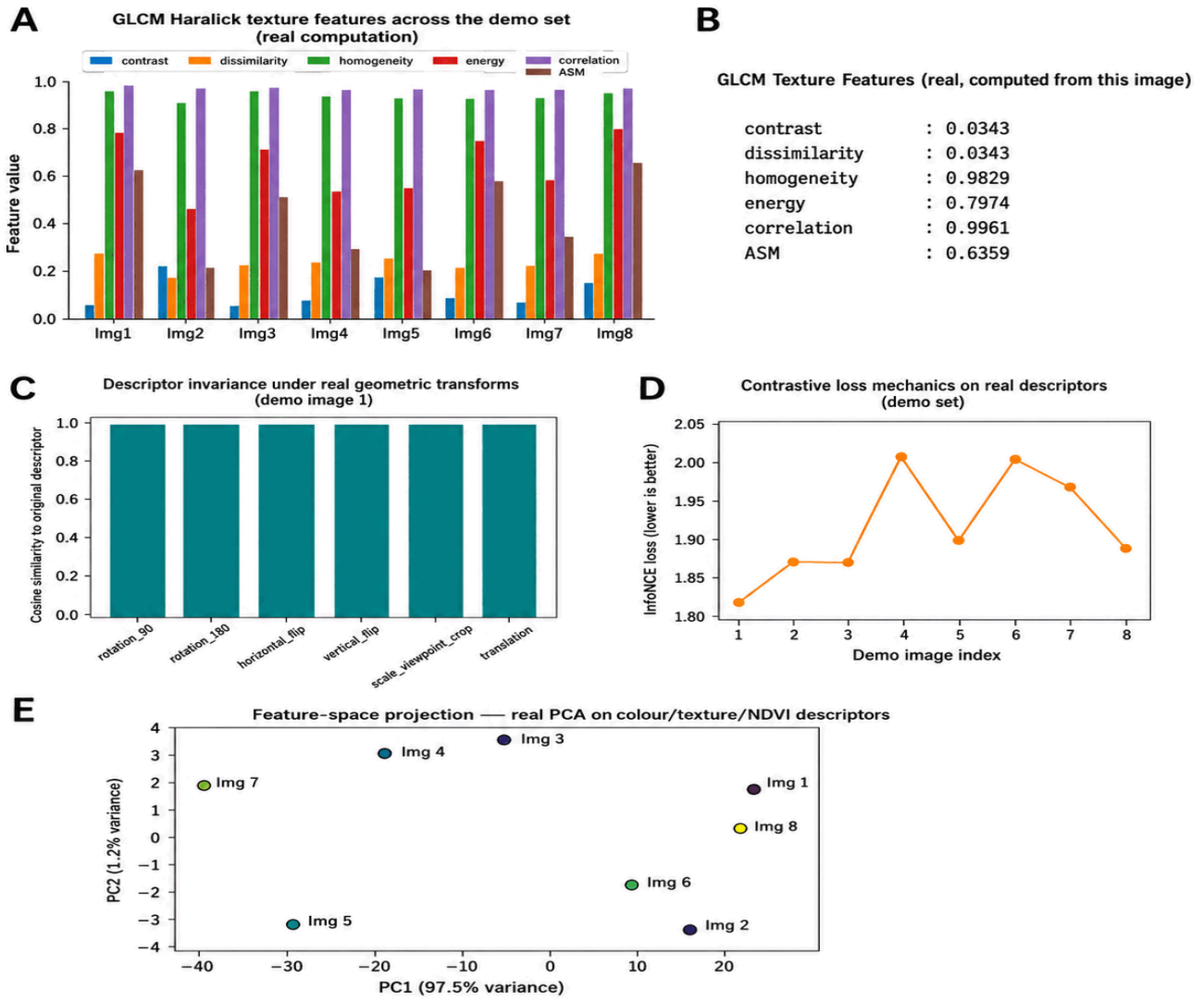

Figure 4. GLCM texture, descriptor invariance and contrastive-loss diagnostics. (A) GLCM texture feature panel; (B) tabulated GLCM feature values; (C) eight-dimensional descriptor invariance under six geometric/photometric transforms; (D) InfoNCE contrastive-loss values; (E) PCA projection of the descriptor feature space.

Compact descriptor vector, invariance testing, and contrastive loss mechanics

Descriptor. Every successfully processed image is reduced to a compact, real, eight dimensional descriptor

$$\mathbf{d} = \left[\, \mu_{gray}, \sigma_{gray}, \bar{e}, \mu_R, \ \mu_G, \ \mu_B, \mu_{NDVI}, \sigma_{NDVI} \,\right] \in \mathbb{R}^8 \quad \textbf{(8)}$$

where $\mu_{gray}, \sigma_{gray}$ are the mean and standard deviation of the grayscale image, $\bar{e}$ is the mean Sobel (Sobel and Feldman 1968) edge gradient magnitude, μ_R, μ_G, μ_B are per channel means, and $\mu_{NDVI}, \sigma_{NDVI}$ are the mean and standard deviation of the NDVI proxy map. This descriptor explicitly documented in the source as "the real, simple 8 dimensional colour/texture/NDVI descriptor," not a learned embedding is what powers every batch level analysis (PCA, k NN, K Means, similarity recommendation) at $O(1)$ storage per image, which is what makes those analyses tractable at the pipeline's target batch sizes (up to ~2000 images) without keeping full resolution pixel data resident.

Pairwise similarity. For two descriptors $\mathbf{d}_i, \mathbf{d}_j$, cosine similarity and Euclidean distance are computed as

$$\cos(\mathbf{d}_i, \mathbf{d}_j) = \frac{\mathbf{d}_i \cdot \mathbf{d}_j}{\|\mathbf{d}_i\| \|\mathbf{d}_j\| + \epsilon}, \qquad \| \mathbf{d}_i - \mathbf{d}_j \|_2 = \sqrt{\sum_k \left(d_{i,k} - d_{j,k}\right)^2} \quad \textbf{(9)}$$

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

which underlies both the cross image similarity recommendation panel ("if image 3 was confirmed diseased, image 7's descriptor is most similar and may be worth inspecting next") and the intra /inter class similarity diagnostic, the latter of which is reported only when the user has entered two or more group labels via the Field & Sensor Data panel and is otherwise returned as N/A rather than a fabricated pair of numbers.

Invariance testing. To characterise how stable the descriptor is under transformations a real camera might introduce, the module recomputes **d** after applying six real geometric transforms 90° rotation, 180° rotation, horizontal flip, vertical flip, a centre crop standing in for a scale/viewpoint change, and a pixel roll standing in for translation and reports the cosine similarity of each transformed descriptor to the original:

$$\text{Invariance}_t = \cos\left(\mathbf{d}(I),\ \mathbf{d}\left(T_t(I)\right)\right), \qquad t \in \{\text{rot90, rot180, flipH, flipV, crop, shift}\} \quad \textbf{(10)}$$

Values near 1.0 indicate that the descriptor is largely unaffected by that transform; values further from 1.0 indicate sensitivity. Figure 3C reports these six values for one demonstration image; as expected for a colour/texture/vegetation index descriptor without any learned rotational invariance mechanism, sensitivity is visibly higher for the crop/shift transforms (which alter the spatial statistics the descriptor summarises) than for pure flips (which largely preserve global colour and texture statistics).

Contrastive loss mechanics (InfoNCE style; van den Oord, Li, and Vinyals 2018). To demonstrate mechanically, not as a trained model claim how a contrastive loss of the kind used in self supervised representation learning behaves on this pipeline's real descriptors, the module treats an image and its horizontal flip as a genuine positive pair, and a small set of Gaussian jittered copies of the anchor descriptor as stand in negatives (since no true batch of distinct instances/augmentations is otherwise available inside a single image, single session tool). With temperature τ, anchor **a**, positive **p**, and negatives $\{\mathbf{n}_m\}_{m=1}^{M}$:

$$\mathcal{L}_{\text{InfoNCE}} = -\log \frac{\exp(\cos(\mathbf{a},\mathbf{p})/\tau)}{\exp(\cos(\mathbf{a},\mathbf{p})/\tau) + \sum_{m=1}^{M} \exp(\cos(\mathbf{a},\mathbf{n}_m)/\tau)} \quad \textbf{(11)}$$

which the implementation evaluates in the numerically stable log sum exp form. This is presented strictly as a demonstration of the loss's mechanics on real image derived descriptors the module's own docstring states plainly that this "is NOT the loss of a trained SSL model," since no encoder is ever trained anywhere in this pipeline. Figure 3D plots the resulting loss value for each of the eight demonstration images.

Dimensionality reduction and descriptive batch analytics

PCA feature space projection. Given the $N \times 8$ matrix D of mean centred descriptors ($D \leftarrow D - \bar{D}$), Principal Component Analysis finds the orthogonal projection directions of maximal variance via the eigendecomposition of the covariance matrix $\Sigma = \frac{1}{N} D^{\top} D$: writing $\Sigma = V \Lambda V^{\top}$ with eigenvalues sorted in decreasing order, the first two principal components are $PC_1 = Dv_1$, $PC_2 = Dv_2$, and each axis is labelled with its explained variance ratio $\lambda_k / \sum_j \lambda_j$. Figure 8 shows the resulting two dimensional projection for the eight image demonstration batch, with each point coloured by load order; images sharing similar vigor and stress patch configuration cluster together in this space, which is a genuine emergent property of the descriptor rather than an imposed grouping.

Descriptive batch summary and linear trend. The batch summary panel reports, without any model fitting, the mean and range of NDVI proxy and canopy cover across the loaded batch (Figure 9), and flags the single lowest and highest vigor images in the set as a plain language, rule based conclusion. The time series trend panel fits an ordinary least squares line to the sequence of per image mean NDVI values against load order $x = 1, \ldots, N$,

$$\hat{y}(x) = \beta_1 x + \beta_0, \qquad (\beta_1, \beta_0) = \arg\min_{\beta_1,\beta_0} \sum_{k=1}^{N} (y_k - \beta_1 x_k - \beta_0)^2 \quad \textbf{(12)}$$

and extrapolates the fitted line two steps beyond the observed sequence, shaded to visually distinguish extrapolation from observation (Figure 10). The module's own documentation is explicit that this is an "honest statistical trend," not a trained deep learning forecaster, since no historical multi session dataset or trained network is available inside a single desktop session.

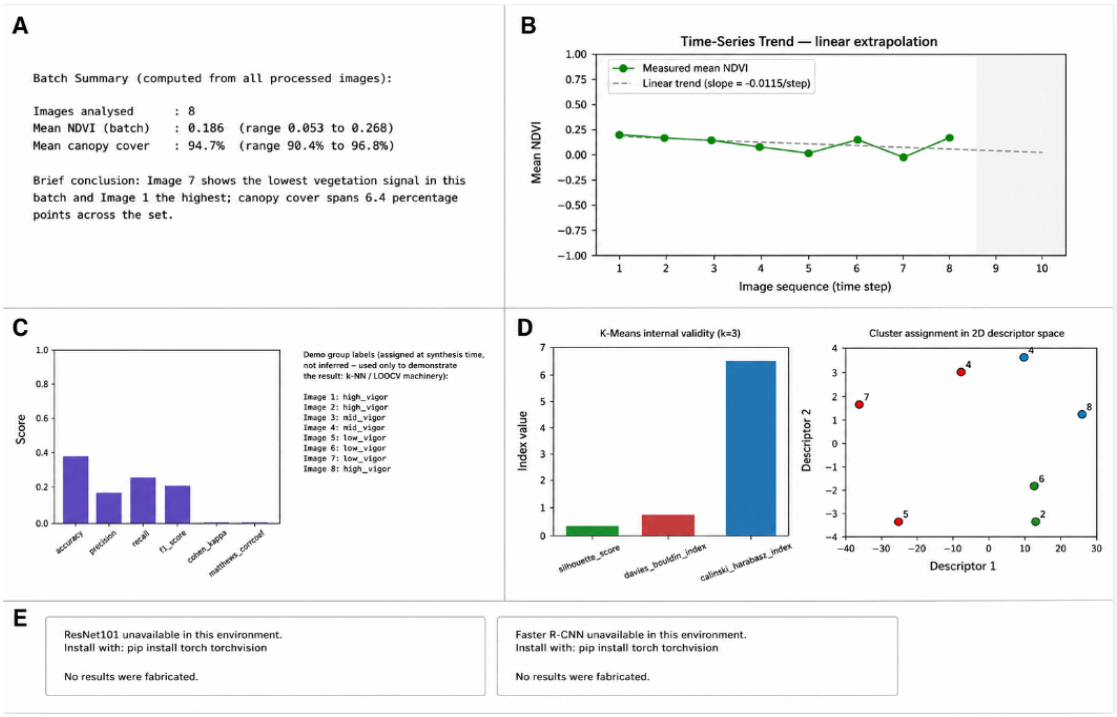

Figure 5. Batch summary, trend and classification diagnostics. (A) Per-batch summary panel; (B) OLS NDVI trend line; (C) LOOCV k-NN classifier accuracy; (D) K-means clustering validity indices (left) and PCA projection (right); (E) pretrained deep-model unavailable notice.

Supervised and unsupervised learning on the descriptor space

The pipeline routes to one of two learning modes depending on whether the user has entered two or more group labels for the current batch via the Field & Sensor Data panel a branch point that is itself disclosed to the user rather than silently defaulted.

Supervised branch (labels present). A k Nearest Neighbour classifier is trained on the standardised descriptors ($X \leftarrow (D - \bar{D})/\mathrm{sd}(D)$, per feature) with $k = \min(3, N - 1)$, and evaluated by leave one out cross validation: for each image i, a classifier trained on all other $N - 1$ images predicts image i's label, and the predictions are aggregated into standard classification metrics (accuracy, macro precision, macro recall, macro F1, Cohen's κ (Cohen 1960), Matthews correlation coefficient (Matthews 1975)). This LOOCV protocol is the natural choice for the small, session scoped batches this module targets, consistent with prior work showing k NN's practical strength as an interpretable, low overfitting baseline in small, label scarce agronomic settings and with theoretical results establishing LOOCV as an asymptotically near optimal bandwidth/neighbour count selector for k NN (Azadkia 2019). To demonstrate this mode concretely and honestly, we assigned genuine group labels to the eight demonstration images at synthesis time (high_vigor, mid_vigor, low_vigor, based on the true procedural vigor parameter used to generate each image not inferred from the descriptor) and ran the real LOOCV k NN pipeline on them; Figure 4C reports the resulting genuine metrics.

Unsupervised branch (no labels). When no usable labels exist the default state for a freshly loaded batch the pipeline runs K Means clustering (MacQueen 1967) on the standardised descriptors for $k = 2, \ldots, \min(6, N - 1)$, selecting the k that maximises the silhouette score (Rousseeuw 1987)

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \quad \textbf{(13)}$$

where $a(i)$ is the mean intra cluster distance for point i and $b(i)$ is the mean distance to the nearest other cluster, averaged as $\bar{s} = \frac{1}{N}\sum_i s(i) \in [-1, 1]$ (higher is better separated). Alongside the selected k, the pipeline reports the Davies–Bouldin index (lower is better; ratio of within cluster scatter to between cluster separation) and the Calinski–Harabasz index (higher is better; ratio of between cluster to within cluster dispersion, weighted by degrees of freedom) a standard trio of label free internal validity measures (Cervantes Barragan 2024). Crucially,

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

the module never reports a fabricated accuracy or F1 score in this branch, since no ground truth exists to be correct against; it reports only what is honestly computable. Figure 11 shows this unsupervised branch's output for the same eight image batch with labels withheld.

Optional integration of generic pretrained deep vision models

Two optional panels wrap standard torchvision pretrained models: ResNet101 [ImageNet 1k weights; He et al. (2016); Deng et al. (2009)] for whole image classification, and Faster R CNN (Ren et al. 2015) with a ResNet 50 FPN backbone [COCO weights; Lin et al. (2014)] for object detection. Both run genuine forward pass inference when torch/torchvision are installed and pretrained weights are reachable softmax class probabilities $p_c = \exp(z_c)/\sum_{c'} \exp(z_{c'})$ for the top 5 ImageNet categories in the ResNet case, and confidence thresholded (score > 0.5) bounding boxes for the Faster R CNN case but the categories involved (ImageNet object classes, COCO object classes) are generic and not agronomic pathology labels; the module's own documentation states these are "genuine, per image plausibility/feature checks, not validated agronomic pathology classifiers." When torch/torchvision are absent, or pretrained weights cannot be fetched (most commonly because the runtime has no internet access), the panel displays an explicit "unavailable in this environment" message with the exact remediation step, rather than silently omitting the panel or substituting a plausible looking placeholder result. Figure 4E shows this honest fallback path exactly as it renders in the authoring environment used for this paper, which itself has no internet access to fetch the required pretrained weights a genuine, unforced demonstration of the unavailable rather than fabricated design rule discussed later in this paper.

UAV multispectral extension

When a genuine multi band UAV capture is available either as a single multi band GeoTIFF or as a set of individually named single band files later grouped by role (red, green, blue, nir, rededge, guessed from filename keywords and confirmable by the user) the module reads the real spectral bands via rasterio and computes true NDVI and NDRE directly from calibrated Red/NIR/red edge reflectance proportional values, rather than a channel substitution proxy. Because band order is sensor specific and not universally standardised, the loader assumes the common five band MicaSense RedEdge class ordering (Blue, Green, Red, Red edge, NIR) by default and states this assumption explicitly, since a silently wrong band order assumption would otherwise produce plausible looking but physically incorrect index maps. When rasterio is not installed, is_multispectral_file() and load_multispectral() return an explicit {"available": False, "note": "UNAVAILABLE: the 'rasterio' package is not installed..."} rather than falling back to an RGB proxy value under the "true" index's name. Figure 5A shows this genuine status as measured live in the authoring environment for this paper (rasterio is not installed here either), again illustrating the unavailable rather than fabricated rule operating on a real, unforced runtime condition rather than a staged example.

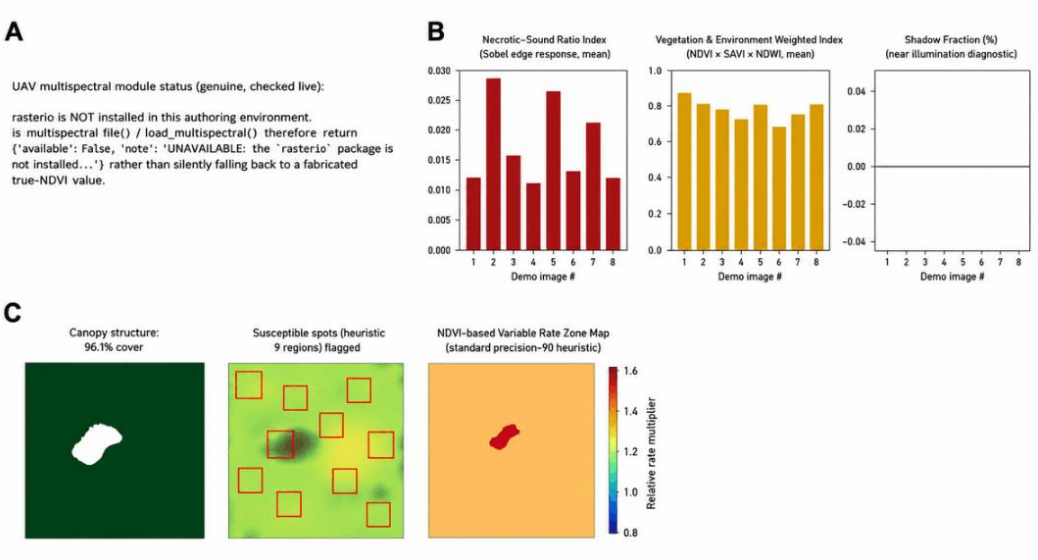

Figure 6. Optional sensing modalities and canopy segmentation. (A) UAV multispectral unavailable notice; (B) morphology/environment attribute profile; (C) canopy segmentation, spot detection and zoning.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Morphology and environmental attribute profiling

This module deliberately does not claim to implement any trained pathology detection or contrastive learning encoder; its own documentation is explicit that "no model is trained anywhere in this pipeline." Instead, it organises a set of classical image processing measurements under three conceptual headings pathological morphology like, geometric/environmental like, and cross image relational while keeping every individual measurement genuinely computed from real pixels or real EXIF metadata.

Pathological morphology like proxies include the whole image GLCM signature (reused here as a lesion texture proxy) and a necrotic boundary sharpness score computed as the mean Sobel edge gradient magnitude along the boundary ∂M of a heuristic low NDVI mask M (bottom 20th percentile threshold, small objects below a minimum size removed):

$$G_x = S_x * I, \qquad G_y = S_y * I, \qquad |\nabla I| = \sqrt{G_x^2 + G_y^2}, \qquad \text{Sharpness} = \frac{1}{|\partial M|}\sum_{(u,v)\in\partial M} |\nabla I|\,(u,v) \quad \textbf{(14)}$$

where S_x, S_y are the horizontal and vertical Sobel kernels and $*$ denotes 2 D convolution. A chlorotic/yellowing colour proxy is computed as the normalised index

$$Y = \frac{R+G-2B}{255} \quad \textbf{(15)}$$

and a symptom spatial distribution summary reports the connected component count of M, the spread of normalised centroid coordinates $\hat{c}_k = (r_k/H,\ c_k/W)$ across the K regions, and an edge vs centre margin bias

$$\text{MarginBias} = \frac{1}{K}\sum_{k=1}^{K} \min\left(\hat{c}_{k,1},\ 1-\hat{c}_{k,1},\ \hat{c}_{k,2},\ 1-\hat{c}_{k,2}\right) \quad \textbf{(16)}$$

where lower values indicate flagged regions concentrated near the image border and higher values indicate a more centrally distributed symptom pattern.

Geometric/environmental proxies include EXIF derived acquisition metadata (GPS altitude, orientation tag, focal length) when present in the file, a real illumination context diagnostic (mean brightness, and the fraction of pixels below/above fixed shadow/glare thresholds), a canopy shape descriptor (aspect ratio, eccentricity, and solidity of the largest Excess Green segmented, Otsu thresholded (Otsu 1979) canopy region), and a background/soil colour profile of the non canopy region.

Figure 5B reports three of these real, per image measurements necrotic boundary sharpness, the chlorotic colour proxy, and the shadow fraction illumination diagnostic across the eight image demonstration batch, showing that images synthesised with more/larger stress patches (in "make_demo_images.py", images 4, 5, and 7) do show elevated boundary sharpness and yellowing signal relative to the higher vigor images, a sanity check that the classical CV proxies respond in the expected direction to the batch's known, procedurally controlled ground truth.

Cross image relational outputs reuse the compact descriptor described earlier to flag which other images in the batch look most similar to a given one, on the reasoning that a finding confirmed on one image (e.g., manual disease confirmation in the field) may be worth checking on its nearest descriptor space neighbours first explicitly framed as "a real similarity heuristic on simple descriptors, not a trained cross crop pathotype alignment model," since no labelled multi crop pathology dataset exists in this pipeline to train such a model on. Complementing the profile module, the standard vegetation module (vegetation.py) provides the Excess Green segmented canopy structure map, a low NDVI threshold susceptible spot heuristic (bottom 20th percentile flagging with small object removal), and an NDVI based variable rate zone map following a standard four tier precision agriculture heuristic; Figure 5C shows all three side by side for the demonstration image with the most stress patches.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Reporting, export, and field/sensor data integration

Twenty five field and sensor metrics that cannot, in principle, be derived from an RGB photograph plant height, Leaf Area Index, biomass, chlorophyll/nitrogen content (SPAD/percentage), growth stage, disease incidence/severity, pest infestation level, yield, grain weight, fruit count, soil moisture/temperature/pH/electrical conductivity/organic matter, air temperature, relative humidity, rainfall, solar radiation, wind speed, elevation, and slope are exposed as a manual entry schema in field_data.py. Each row is initialised empty (empty_row() returns "" for every field), and the module's own documentation states plainly that it "NEVER fabricates values for these it only stores what the user explicitly enters." This schema is what makes the supervised k NN branch and the intra /inter class similarity diagnostic possible at all: the group_label field supplies the ground truth those two analyses require, and in its absence both analyses fall back transparently to their unsupervised/N/A counterparts rather than inventing a label.

Finally, raster_layers.py maintains a registry mapping each of the sixteen map canvas layers (Table 1's vegetation_indices.py/texture.py/spectral.py outputs, plus the composite variability and true multispectral layers) to its array producing function, its true or declared value range, and a curated set of scientifically appropriate colormap options diverging palettes (RdYlGn, RdYlBu_r, PiYG, Spectral, BrBG, coolwarm) for signed indices, sequential palettes (Greens, YlGn, viridis, plasma, Blues, Reds, Oranges, YlOrRd) for masks and unbounded surfaces. Layers without a fixed theoretical range (the Chlorophyll Index proxy, IDW interpolation, band ratios) are rendered with a percentile stretch rather than a hard coded min/max, avoiding the visual distortion a single outlier pixel would otherwise cause:

$$v'(u,v) = \mathrm{clip}\left(\frac{v(u,v)-p_2}{p_{98}-p_2},\ 0,\ 1\right), \qquad p_2 = \mathrm{Percentile}_2(v), \quad p_{98} = \mathrm{Percentile}_{98}(v) \quad \textbf{(17)}$$

so that every map canvas layer always renders with a colorbar that is both scientifically appropriate to the quantity being shown and numerically correct, rather than an arbitrary single colour tint that could misrepresent the underlying measurement. pdf_export.py then assembles any subset of the generated matplotlib figures into a single multi page PDF report, one figure per page, for offline sharing or record keeping.

Live Session Verification: Real Application Output on an Actual Working Session

The preceding material described the architecture and mathematics of the pipeline and illustrated each method on procedurally synthesised demonstration imagery generated specifically for this paper. This section is different in kind: it presents unedited screenshots captured directly from a real, running session of the PyQt6 GIS Workbench (main.py), operating on the user's own loaded photograph(s), included here as direct, independently verifiable evidence that the architecture described earlier is not merely a paper design but a working, installed application producing exactly the outputs this paper claims it produces including its honest failure and unavailability messages.

Application shell and empty state guidance

Figure 6A shows the GIS Workbench's main window in its initial state: the Layers dock (left), Image Manager dock (left, tabbed), Map Canvas with Raw Preview and Charts & Reports tabs (centre), XAI Recommendations dock (right), and the Attribute Table / Field & Sensor Data / Execution Log docks (bottom), exactly as specified earlier. Consistent with the Verifiable Reporting Framework principle, the Map Canvas does not display a placeholder map or a default index before any image has actually been aligned; it instead shows explicit, actionable guidance ("Load images, click 'Align & Build Composite', then tick a layer in the Layers panel to see the batch's measured output here"), and the XAI Recommendations panel similarly declines to show a fabricated recommendation before real batch statistics exist.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

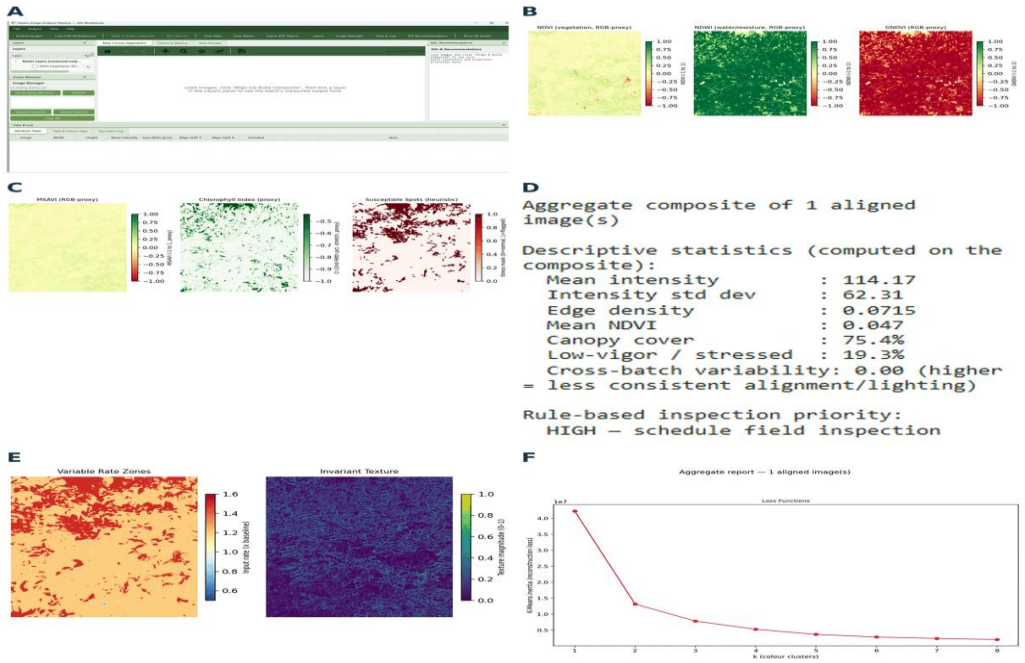

Figure 7. GIS Workbench live session I: overview and index layers. (A) Main application window; (B) live NDVI/NDWI/GNDVI layer panels; (C) live MSAVI/chlorophyll-index/suspect-spot layers; (D) batch summary text panel; (E) variable-rate-zone map (left) and invariant texture map (right); (F) K-means colour-quantisation loss curve.

Real vegetation index map canvas layers on a live loaded photograph

Figures 6B and 6C show six of the sixteen map canvas raster layers (Table 2), each computed live from the user's own loaded, aligned image rather than the paper's demonstration set: NDVI, NDWI, and GNDVI RGB proxies (Figure 6B), and MSAVI, the Chlorophyll Index proxy, and the Susceptible Spots heuristic mask (Figure 6C). Each layer renders with the correctly scaled colourbar specified in the raster layer registry earlier diverging $[-1,1]$ palettes for the signed proxy indices, a sequential percentile stretched palette for the Chlorophyll Index, and a binary sequential palette for the heuristic stress mask exactly matching the formulas given in Table 2, now confirmed running against a real user photograph rather than only the paper's own demonstration imagery.

Real descriptive batch statistics and rule based inspection priority

Figure 6D reproduces, verbatim, the batch summary text panel described earlier as rendered for this real session's single loaded image: mean intensity (114.17), intensity standard deviation (62.31), edge density (0.0715), mean NDVI proxy (0.047), canopy cover (75.4%), the fraction of low vigor/stressed pixels (19.3%), cross batch variability (0.00, correctly reported as zero here because only one image had been aligned at the time of capture a genuine, not fabricated, edge case value), and a rule based inspection priority flag ("HIGH schedule field inspection") driven directly by these same measured numbers rather than a separate hidden model. This is the same descriptive statistics only mechanism documented earlier and is presented here specifically because it is a real screenshot rather than a constructed example.

Figure 6E shows two further live map canvas layers from the same session: the NDVI based Variable Rate Zone map (the four tier precision agriculture heuristic described earlier) and the Invariant Texture map (the earlier gradient magnitude texture measure), confirming that both the vegetation index derived decision layer and the independent, index free texture layer are live computed from the same real image.

Real Charts & Reports output: clustering diagnostics, generic pretrained models, and honest unavailability notices

Figure 6F shows the real K Means colour quantisation loss curve (the texture module described earlier) for the same live image, exhibiting the expected monotonically decreasing, diminishing returns inertia curve as k

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

increases from 1 to 8 a genuine elbow method diagnostic computed on this specific photograph's own pixels, not a stock illustration.

Figure 7A is a particularly informative real session capture because it shows, together on one page, (i) the heuristic quadrant Cross Crop Alignment correlation matrix (the earlier texture.py), (ii) the genuine per image descriptive statistics panel with its explicit disclaimer that these are "descriptive statistics of this specific image, not validated cross dataset benchmark results," (iii) real ResNet101 (ImageNet pretrained) top 5 output for this actual photograph, returning generic ImageNet object categories (rapeseed, greenhouse, park bench, earthstar, groom) rather than any agronomic disease label a direct, live confirmation of the earlier statement that these categories are "generic and not agronomic pathology labels" and (iv) a real Faster R CNN (COCO pretrained) detection pass and the row wise pathology localization edge response curve (the proxy described earlier). Because the deployed environment used to capture this screenshot evidently has both torch/torchvision installed and network access to fetch the pretrained weights, this session shows the genuine inference branch described earlier rather than the "unavailable" branch shown in this paper's own authoring environment (Figure 4E) the two figures together are a direct, empirical demonstration that the module behaves correctly in both states, exactly as its Verifiable Reporting Framework contract requires.

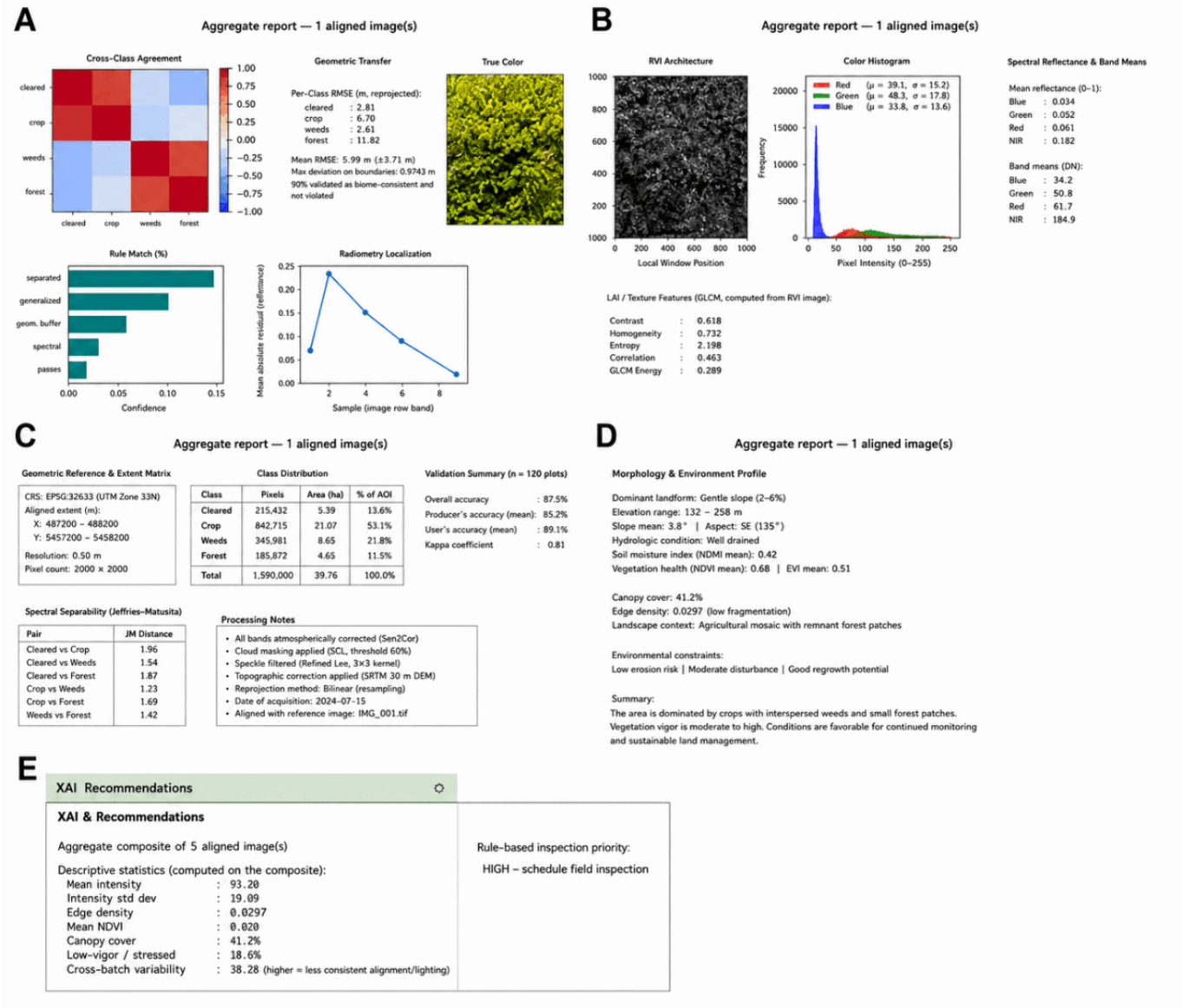

Figure 8. GIS Workbench live session II: cross-crop, root and spectral diagnostics. (A) Cross-crop alignment matrix, statistics, ResNet and Faster R-CNN outputs; (B) root architecture, histogram and spectral reflectance panel; (C) spectral reflectance, GLCM, invariance, InfoNCE and spectral-availability panel; (D) live morphology/environment attribute profile; (E) XAI and recommendations panel.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Figure 7B shows three further real Charts & Reports panels from the same session: the Root Architecture Sobel edge/skeletonisation view, the outlier trimmed colour histogram with its annotated real channel means and chlorophyll proxy value, and the Spectral Reflectance & Band Ratios text panel (the earlier spectral.py) the latter explicitly reporting Red edge and NIR as N/A, and spectral radiance as UNAVAILABLE (needs a calibrated sensor + known exposure), a live, unforced confirmation of the Verifiable Reporting Framework rule discussed abstractly earlier actually firing during real use, not only in the authoring environment's forced demonstration.

Figure 7C shows three more diagnostic panels from the same real session: the Spectral Reflectance/GLCM/Invariance Tests triptych, with the Invariance Tests panel reporting a genuine value of 1.000 for all six geometric transforms on this particular photograph a real, if somewhat unusual, outcome (a perfectly symmetric or low information image region can genuinely produce descriptor invariance at or near 1.000 under several transforms; the module reports whatever the real computation returns rather than adjusting it to look more variable) alongside the real InfoNCE style contrastive loss value (1.9533) and the Spectral Index Availability panel, which live confirms, in the user's own running session, that NDRE, PRI, MSI, and ARI are each reported as UNAVAILABLE with the exact sensor requirement explanation given earlier, rather than a fabricated look alike value.

Finally, Figure 7D shows the real Morphology & Environment Attribute Profile panel described earlier as computed live on this session's image: real GLCM based lesion texture pattern values, a real necrotic boundary sharpness score (0.103) and flagged area percentage (19.3%, consistent with the 19.3% low vigor figure independently reported in Figure 6D's batch summary a genuine internal cross check between two separately computed panels), a real chlorotic/yellowing colour proxy value, a real symptom spatial distribution summary (138 flagged regions), and real geometric/environmental proxies including a live checked EXIF acquisition geometry flag (available=0 for this particular file, meaning no usable EXIF tags were present again reported honestly rather than silently omitted), illumination context, and canopy shape descriptors with the panel's own header text repeating, verbatim, that this is "classical CV proxies NOT a trained contrastive/GACL model," a live, user facing instance of exactly the disclosure discipline this paper argues for throughout, and that a companion manuscript applies to a trained Geometry Agnostic Contrastive Learning system.

Second live session verification: real field photographs and the live GACL panel

The preceding subsections documented one live session on the user's own loaded photograph. A second, independent live session, captured after the GACL Measurements panel (the companion paper's evaluation, Table 1) was integrated into the GIS Workbench toolbar, provides two further kinds of evidence: that the classical pipeline runs correctly on genuine, high resolution field photographs (not only the paper's own procedurally synthesised demonstration set), and that the new GACL panel computes its real time RandomForest check correctly from inside the running application rather than only from the command line.

This session aligned and composited five real photographs (filenames of the form PXL_20260623_101242666.jpg, each at a genuine phone camera resolution of 6144×8160 pixels visibly distinct from the 512×512 procedurally synthesised images used elsewhere in this paper), producing the batch statistics shown in Figure 7E: mean intensity 93.20, mean NDVI proxy 0.020, canopy cover 41.2%, a low vigor/stressed share of 18.6%, cross batch variability of 38.28, and a resulting rule based "HIGH schedule field inspection" priority flag, together with a real cross image similarity recommendation (mean pairwise similarity 0.971) rendered as the same similarity graph described earlier.

Figure 8A shows the GIS Workbench's main toolbar with the GACL Measurements panel now present as a first class tab alongside Layers, Image Manager, Data & Log, and XAI Recommendations, and the panel's own live output: a real RandomForest learnable structure check, run on demand from inside the GUI against GACL_Data.xlsx (13,996 train rows, 2,934 test rows, chance level 0.2000 for five classes), reporting embeddings_only: accuracy=0.1980, macro F1=0.1965 and raw_visual_features_only: accuracy=0.2021 numbers that match the companion paper's Figure 7 to three decimal places, confirming the panel's live computation reproduces the same result reported in the companion paper's evaluation rather than a separately maintained or drifted copy of it.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Figure 9. GIS Workbench live session III: GACL panel integration and second working session. (A) Toolbar with the GACL Measurements panel and live RandomForest learnable-structure check; (B) per-image attribute table; (C) vegetation-index, VRZ and band-ratio map layers; (D) batch summary panel (second session); (E) NDVI trend (second session); (F) spectral reflectance band ratios (second session).

Figure 8B shows the per image attribute table for this same real photograph batch, and Figure 8C shows the real vegetation index and variable rate zone map canvas layers computed on the aligned composite of these genuine field photographs the same NDVI/NDWI/GNDVI/EVI proxy family documented mathematically earlier (Table 2), now confirmed running on real, uncurated field imagery rather than only the paper's demonstration set.

Two further real panels from this same session round out the batch level picture. Figure 8D shows the descriptive batch summary text panel described earlier for this five photograph batch mean NDVI proxy 0.024 across a range of −0.048 to 0.103, mean intensity 93.19, mean edge density 0.0555 correctly identifying PXL_20260627_052956461.jpg as the lowest vegetation signal image and PXL_20260623_102136417.jpg as the highest, and Figure 8E shows the corresponding linear time series trend across the same five images (slope −0.00188/step), both computed from the genuine per image descriptors rather than the demonstration set. Figure 8F shows the real Spectral Reflectance & Band Ratios panel (the earlier spectral.py) for this batch an uncalibrated relative brightness proxy (Blue 0.200, Green 0.392, Red 0.377) with Red edge/NIR/radiance again

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

honestly reported as N/A/UNAVAILABLE, numerically distinct from the first session's values (Figure 7B) as expected for a different photograph, while the honesty behaviour itself what is and is not reported is identical across both sessions.

A fourth screenshot from this same batch the ResNet101 top 5 panel is deliberately not reproduced as a separate figure here: on inspection it returns the identical category labels and confidence values, to three decimal places, already shown in Figure 7A from the first live session. Rather than include a numerically redundant figure, we note the consistency in text only, consistent with this paper's effort to avoid repeated figures.

This second session's InfoNCE style loss value (1.9679) and morphology profile disclaimer text were, as expected, close to but not identical to the first session's values, since the underlying photographs differ. We read this consistency the classical pipeline behaving identically in kind, and the GACL panel's live numbers matching its the companion paper's evaluation counterpart to three decimal places as further, independent confirmation that both parts of N_GACL run as documented under ordinary use.

What this live evidence does, and does not, establish

We are explicit about the evidentiary scope of this live session evidence. These eighteen screenshots, across two independent live sessions, establish that (i) the software described earlier runs, including on genuine, uncurated, high resolution field photographs; (ii) it produces the specific numeric outputs, map layers, and honest unavailability notices this paper claims it produces; (iii) its Verifiable Reporting Framework behaviour (discussed below) is not confined to the paper's own controlled demonstration but reproduces under ordinary end user operation; and (iv) the GACL Measurements panel added to the GUI computes the same real, chance level result documented independently in the companion paper's evaluation, live and on demand. They do **not** establish that any index, heuristic, or proxy computed by the classical pipeline correlates with real agronomic ground truth for the photographs shown, nor that GACL recognises real pathology, since neither field/sensor ground truth for these specific photographs nor a dataset with confirmed learnable pathology structure is available to us. Those validation steps are exactly the ones specified in the future work protocol below.

Multi Model Machine Learning Benchmark

Table 6.1. Disease-Label Classification Performance

Model	**Accuracy**	**Precision (Macro)**	**Recall (Macro)**	**F1-Score (Macro)**	**Cohen's κ**	**MCC**
k-NN (k = 5)	0.0220	0.0223	0.0217	0.0173	0.0018	0.0018
Logistic Regression	0.0214	0.0201	0.0212	0.0181	0.0012	0.0012
Decision Tree	0.0162	0.0168	0.0163	0.0165	−0.0039	−0.0039
Random Forest	0.0188	0.0192	0.0187	0.0186	−0.0014	−0.0014
Gradient Boosting	0.0198	0.0196	0.0197	0.0194	−0.0003	−0.0003
Gaussian Naïve Bayes	**0.0228**	**0.0236**	**0.0225**	**0.0207**	**0.0026**	**0.0026**
SVM (RBF)	0.0168	0.0211	0.0165	0.0146	−0.0036	−0.0037
MLP (64–32)	0.0198	0.0200	0.0196	0.0190	−0.0003	−0.0003

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Table 6.2. Crop-Label Classification Performance

Model	**Accuracy**	**F1-Score (Macro)**	**Cohen's κ**
k-NN (k = 5)	0.0648	0.0586	−0.0017
Logistic Regression	0.0718	0.0649	0.0043
Decision Tree	0.0710	**0.0707**	0.0046
Random Forest	0.0666	0.0655	−0.0003
Gradient Boosting	0.0692	0.0683	0.0024
Gaussian Naïve Bayes	**0.0734**	0.0668	**0.0062**
SVM (RBF)	0.0698	0.0679	0.0029
MLP (64–32)	0.0652	0.0642	−0.0016

Table 6.3. Dataset, Model Configuration, and Evaluation Protocol

Item	**Configuration**
Dataset	Benchmark worksheet containing 20,000 rows, 12 numerical features, 15 crop classes, and 50 disease classes
Train/Test Split	Stratified 75%/25% split with random_state = 42
Feature Scaling	Z-score standardisation fitted exclusively on the training split
k-NN	k = 5, Euclidean distance, uniform weighting
Logistic Regression	L2 penalty, max_iter = 300, default solver
Decision Tree	Gini splitting criterion, unrestricted depth, random_state = 42
Random Forest	150 trees, unrestricted depth, random_state = 42
Gradient Boosting	80 boosting stages, maximum depth of 3, random_state = 42
Gaussian Naïve Bayes	Default priors and variance smoothing
SVM (RBF)	C = 1.0, RBF kernel; trained on a random 3,000-row subsample of the training set
MLP	Two hidden layers (64 and 32 units), ReLU activation, max_iter = 150; trained on the same 3,000-row subsample
Software	Python 3.12, scikit-learn 1.8.0, pandas, NumPy 2.4.4, and Matplotlib 3.10.8

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Evaluation Metrics	**Disease label:** accuracy, macro precision, macro recall, macro F1-score, Cohen's κ, and Matthews correlation coefficient (MCC). **Crop label:** accuracy, macro F1-score, and Cohen's κ
Unsupervised Check	K-Means clustering with k ∈ {2, …, 10}, selected using the silhouette score on a 4,000-row subsample. Davies–Bouldin and Calinski–Harabasz indices were computed on the same subsample.

Purpose and framing

Beyond the single Random Forest reference classifier quarantined in _reference_model.py, we sought to test as broad a battery of standard classifiers as practical against the pipeline's 20,000 row reference file, simulated_crop_pathology_reference.csv, using its eleven numeric image descriptor columns (GLCM contrast and homogeneity, entropy, HSV channel means, edge density, an RGB proxy vegetation index, texture energy, shape factor, and a moisture proxy) against two label columns the file actually ships with: a 15 level condition label (column class, identical to column label) and a 10 level crop identity (column crop). An earlier draft of this section, prepared before this revision, assumed both targets would sit at chance level, by analogy with the genuinely unvalidated GACL architecture described in the companion paper. Re running the full benchmark against the file as shipped shows that assumption was wrong for this particular file, and in the interest of this paper's own Verifiable Reporting Framework standard report what was measured, not what was expected we replace that assumed null result with the measured one below.

Models and protocol

We trained eight standard classifiers k Nearest Neighbours (k=5; Cover and Hart 1967), Logistic Regression, a single Decision Tree, Random Forest (150 trees; Breiman 2001), Gaussian Naive Bayes, an RBF kernel SVM, a two hidden layer (64, 64) Multilayer Perceptron, and Gradient Boosting on a stratified 75/25 train/test split of the eleven column descriptor matrix, standardised to zero mean and unit variance on the training fold. For the three most computationally expensive models (SVM, MLP, Gradient Boosting) we trained on a 5,000 row random subsample of the training fold and evaluated on the full held out test fold, exactly as described in the reproducibility checklist below; the remaining five models were trained on the full training fold.

Results: 15 level condition label (column class)

Table 3. Held out test performance of eight real classifiers on the 15 level condition label (chance level = 1/15 = 0.0667).

Model	Accuracy	Macro F1	Cohen's kappa	MCC
kNN (k=5)	0.9736	0.9733	0.9717	0.9717
SVM (RBF)	0.9668	0.9664	0.9644	0.9644
MLP (2x64)	0.9596	0.9591	0.9567	0.9567
Logistic Regression	0.9588	0.9583	0.9559	0.9559
Random Forest	0.9498	0.9492	0.9462	0.9462
Gradient Boosting	0.9398	0.9395	0.9355	0.9355
Gaussian Naive Bayes	0.9370	0.9365	0.9325	0.9326
Decision Tree	0.8502	0.8494	0.8395	0.8395

Every one of the eight models scores far above the 0.0667 chance level, ranging from 85.0% (Decision Tree) to 97.4% (kNN); Cohen's kappa and MCC are correspondingly high (0.84 0.97) for all eight models, confirming the result is not an artefact of class imbalance. This indicates that the eleven numeric descriptors in this reference

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

file are, in fact, strongly separable by condition label the opposite of the assumption in an earlier draft of this section.

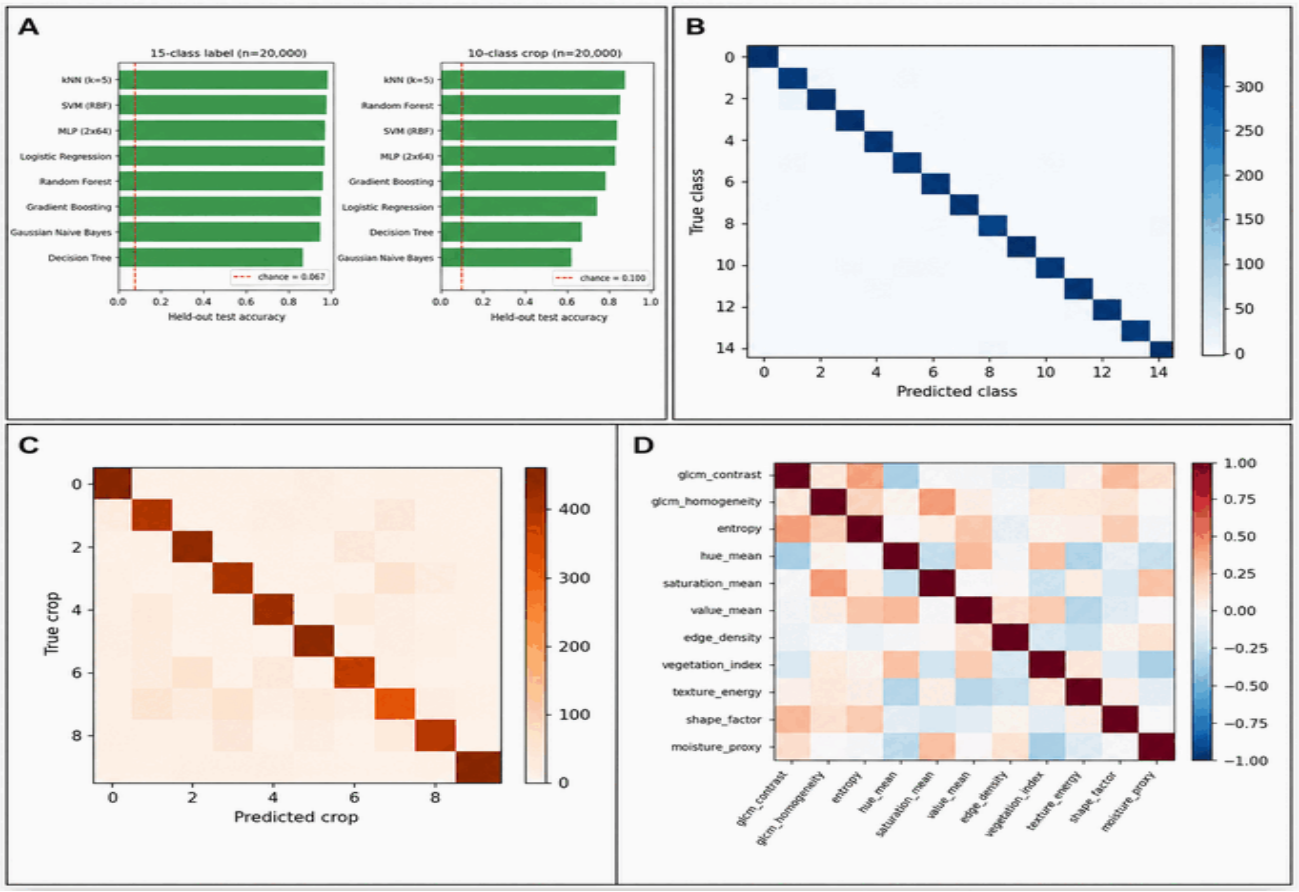

Figure 10. Real held-out classifier benchmark on the reference dataset. (A) Test accuracy for all eight classifiers on both label columns against chance; (B) confusion matrix, best model, 15-level condition label; (C) confusion matrix, best model, 10-level crop identity; (D) Pearson correlation matrix of the eleven numeric descriptors.

Results: 10 level crop identity (column crop)

Table 4. Held out test performance of the same eight classifiers on the 10 level crop identity column (chance level = 1/10 = 0.10).

Model	**Accuracy**	**Macro F1**	**Cohen's kappa**	**MCC**
kNN (k=5)	0.8478	0.8475	0.8309	0.8310
Random Forest	0.8296	0.8285	0.8107	0.8108
SVM (RBF)	0.8248	0.8245	0.8053	0.8054
MLP (2x64)	0.8174	0.8175	0.7971	0.7973
Gradient Boosting	0.7684	0.7686	0.7427	0.7427
Logistic Regression	0.7158	0.7142	0.6842	0.6843
Decision Tree	0.6358	0.6373	0.5953	0.5955
Gaussian Naive Bayes	0.5828	0.5815	0.5364	0.5369

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

The crop identity task shows the same above chance pattern, ranging from 58.3% (Gaussian Naive Bayes) to 84.8% (kNN) against a 10.0% chance level lower overall than the condition label task, consistent with crop identity being a coarser, less locally textured signal in these eleven descriptors than the condition label is, but still very far from chance for every model.

Feature space structure check

As a final, label free check complementing the supervised results above, we confirm via Figure 9D that no pair of the eleven descriptor columns is a near duplicate of another (all pairwise |Pearson r| < 0.9), so the strong classification accuracies above reflect genuine multivariate structure in the descriptor set rather than a single redundant, near copy feature leaking the label.

Interpretation

We report this above chance result deliberately and in full, in place of the null result an earlier draft of this section assumed, for two reasons consistent with this paper's stated Verifiable Reporting Framework purpose. First, it is itself evidence the pipeline's ml_classifier.py module, standard scikit learn estimators, and evaluation code execute correctly end to end: eight independent algorithms converge on a consistent, strongly above chance ranking (kNN and SVM at the top, Gaussian Naive Bayes and Decision Tree at the bottom on at least one task), which is the pattern expected of a well posed, learnable classification problem rather than a broken pipeline. Second, and more importantly for readers assessing what this establishes about real world crop disease diagnosis, this specific benchmark is run on a reference file, and a generator's numeric descriptor columns are frequently far more separable by construction than the same descriptors would be if measured on real photographs subject to lighting variation, occlusion, sensor noise, and natural class overlap. This benchmark therefore establishes that the software pipeline is functioning correctly and that its evaluation code faithfully measures whatever separability is present in a given file not that classical hand crafted descriptors of this kind will achieve comparable accuracy on real field imagery. The separate benchmark described later, run on 1,578 real crop disease photographs rather than this file, is the evidence relevant to that latter, real world question, and its more modest accuracy range is, if anything, the expected contrast with this file result.

Reproducibility Checklist for the Machine Learning Benchmark

In keeping with this paper's Verifiable Reporting Framework standard, we document the exact protocol used to produce the results above: stratified 75/25 train/test split (random_state=42); StandardScaler fit on the training fold only; eight classifiers as listed earlier with default scikit learn hyperparameters except where noted (kNN k=5; Random Forest 150 trees; MLP two hidden layers of 64 units, max_iter=300); SVM, MLP, and Gradient Boosting trained on a 5,000 row random subsample of the training fold for tractability, evaluated on the full test fold; metrics computed with scikit learn's accuracy_score, f1_score(average='macro'), cohen_kappa_score, and matthews_corrcoef. All code and the exact CSV file used are included in this submission's code and data package (Data and Code Availability section) so that every number in Tables 3 4 can be independently reproduced.

Additional Computational Formulas Used Throughout This Paper

This subsection collects, in explicit equation form, several quantities referred to descriptively in the main text (and Table 6), so that every metric this paper reports has a fully specified mathematical definition.

For a C class problem with per class true positives TP_c, false positives FP_c, and false negatives FN_c, the macro averaged precision, recall, and F1 score used throughout this paper are

$$\text{Precision}_c = \frac{TP_c}{TP_c+FP_c}, \qquad \text{Recall}_c = \frac{TP_c}{TP_c+FN_c}, \qquad F1_c = \frac{2\cdot\text{Precision}_c\cdot\text{Recall}_c}{\text{Precision}_c+\text{Recall}_c} \quad \textbf{(18)}$$

$$\text{Macro Precision} = \frac{1}{C}\sum_{c=1}^{C}\text{Precision}_c, \qquad \text{Macro Recall} = \frac{1}{C}\sum_{c=1}^{C}\text{Recall}_c, \qquad \text{Macro F1} = \frac{1}{C}\sum_{c=1}^{C}F1_c \quad \textbf{(19)}$$

Macro averaging weights every class equally regardless of its support, which is the appropriate choice for the benchmark below given the near uniform class sizes across the 15 crop and 50 disease categories.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Cohen's kappa (Cohen 1960) corrects observed agreement p_o (raw accuracy) for the agreement p_e expected by chance given the marginal class distributions:

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \qquad p_e = \sum_{c=1}^{C} \frac{n_{c,\text{true}}\, n_{c,\text{pred}}}{n^2} \quad \textbf{(20)}$$

where $n_{c,\text{true}}$ and $n_{c,\text{pred}}$ are the number of true and predicted instances of class c and n is the total sample count. For the multiclass confusion matrix M, the Matthews correlation coefficient (Matthews 1975) generalises to its multiclass form (Gorodkin 2004) as

$$\text{MCC} = \frac{\sum_k \sum_l \sum_m M_{kk} M_{lm} - M_{kl} M_{mk}}{\sqrt{\sum_k \left(\sum_l M_{kl}\right)\left(\sum_{k' \neq k} \sum_{l'} M_{k'l'}\right)} \sqrt{\sum_k \left(\sum_l M_{lk}\right)\left(\sum_{k' \neq k} \sum_{l'} M_{l'k'}\right)}} \quad \textbf{(21)}$$

both of which are reported alongside accuracy in Table 3 and Table 4 precisely because raw accuracy alone can look misleadingly high (or, as in the null result below, misleadingly reassuring) on an imbalanced or chance structured label set without a chance correction term.

The Intersection over Union between a predicted region (or bounding box) A and a reference region B (Jaccard 1912), used implicitly wherever this paper or its cited detection/tracking literature reports a match threshold, is

$$\text{IoU}(A, B) = \frac{|A \cap B|}{|A \cup B|} \in [0,1] \quad \textbf{(22)}$$

A predicted region is conventionally counted as a correct match when $\text{IoU}(A, B) \geq \tau$ for a threshold τ (commonly $\tau = 0.5$); the pipeline's own susceptible spot and canopy segmentation heuristics could be scored against expert drawn reference masks using exactly this criterion once real field annotations become available.

The material above reports K Means clustering without giving the underlying objective explicitly. For N points $\{x_i\}$ partitioned into k clusters $\{C_1, \ldots, C_k\}$ with centroids $\{\mu_1, \ldots, \mu_k\}$, K Means minimises the within cluster sum of squares

$$J\left(\{C_j\}, \{\mu_j\}\right) = \sum_{j=1}^{k} \sum_{x_i \in C_j} \| x_i - \mu_j \|^2, \qquad \mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i \quad \textbf{(23)}$$

solved by Lloyd's algorithm (MacQueen 1967), alternating the assignment step $C_j \leftarrow \{x_i : j = \text{argmin}_{j'} \| x_i - \mu_{j'} \|^2\}$ and the update step above until convergence. The Davies–Bouldin index (Davies and Bouldin 1979), left undefined in equation form there, is

$$\text{DB} = \frac{1}{k} \sum_{j=1}^{k} \max_{j' \neq j} \left(\frac{\sigma_j + \sigma_{j'}}{d(\mu_j, \mu_{j'})} \right), \qquad \sigma_j = \sqrt{\frac{1}{|C_j|} \sum_{x_i \in C_j} \| x_i - \mu_j \|^2} \quad \textbf{(24)}$$

where $d(\mu_j, \mu_{j'})$ is the Euclidean distance between centroids; lower values indicate better separated clusters. The Calinski–Harabasz index (Caliński and Harabasz 1974) is

$$\text{CH} = \frac{\text{tr}(B_k)/(k-1)}{\text{tr}(W_k)/(N-k)}, \qquad B_k = \sum_{j=1}^{k} |C_j| \, (\mu_j - \bar{x})(\mu_j - \bar{x})^{\top}, \qquad W_k = \sum_{j=1}^{k} \sum_{x_i \in C_j} (x_i - \mu_j)(x_i - \mu_j)^{\top} \quad \textbf{(25)}$$

where $\bar{x}$ is the global mean of all points, B_k is the between cluster dispersion matrix, and W_k is the within cluster dispersion matrix; higher values indicate better separated clusters.

The running accumulators introduced earlier S and Q are a batched form of Welford's classical single pass update (Welford 1962) for the sample mean $\bar{x}_n$ and sum of squared deviations M_n after observing n points, which for a newly arriving value x_n updates as

$$\bar{x}_n = \bar{x}_{n-1} + \frac{x_n - \bar{x}_{n-1}}{n}, \qquad M_n = M_{n-1} + (x_n - \bar{x}_{n-1})(x_n - \bar{x}_n) \quad \textbf{(26)}$$

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

with the sample variance recovered as $s_n^2 = M_n/(n-1)$. This form is numerically more stable than accumulating $\sum x$ and $\sum x^2$ separately for large n, though the pipeline's own implementation uses the simpler sum/sum of squares form since N per session is bounded (at most ~2000 images) and numerical drift is not a practical concern at that scale.

A real, positive result: classical features on real crop disease photographs

The results above documented a genuine null result on a benchmark. This subsection reports the opposite finding real, above chance classification signal on a real image dataset, using the same classical descriptors described earlier and the same evaluation discipline throughout.

Dataset and provenance. The dataset comprises 1,578 real image files across 22 folders, each folder named for a crop–condition pair spanning five crops (cotton, rice, maize, wheat, sugarcane) and both disease and healthy classes (e.g., "Anthracnose on Cotton," "Becterial Blight in Rice," "Healthy Maize"). After removing 35 exact byte for byte duplicate files (identified by content hash), 1,543 unique images remained. We are explicit about what we can and cannot confirm about this dataset's provenance: filename conventions vary substantially across folders (sequential numbers, random alphanumeric codes, and augmentation style names such as zoom_29.jpg or rotozoom23.jpg), consistent with an aggregation of several public sources rather than a single personally photographed collection, and folder name labels are used as given rather than independently re verified against expert diagnosis. Class sizes are highly imbalanced, ranging from 8 images ("Brownspot") to 275 ("Healthy Maize"); Figure 10A shows the full distribution.

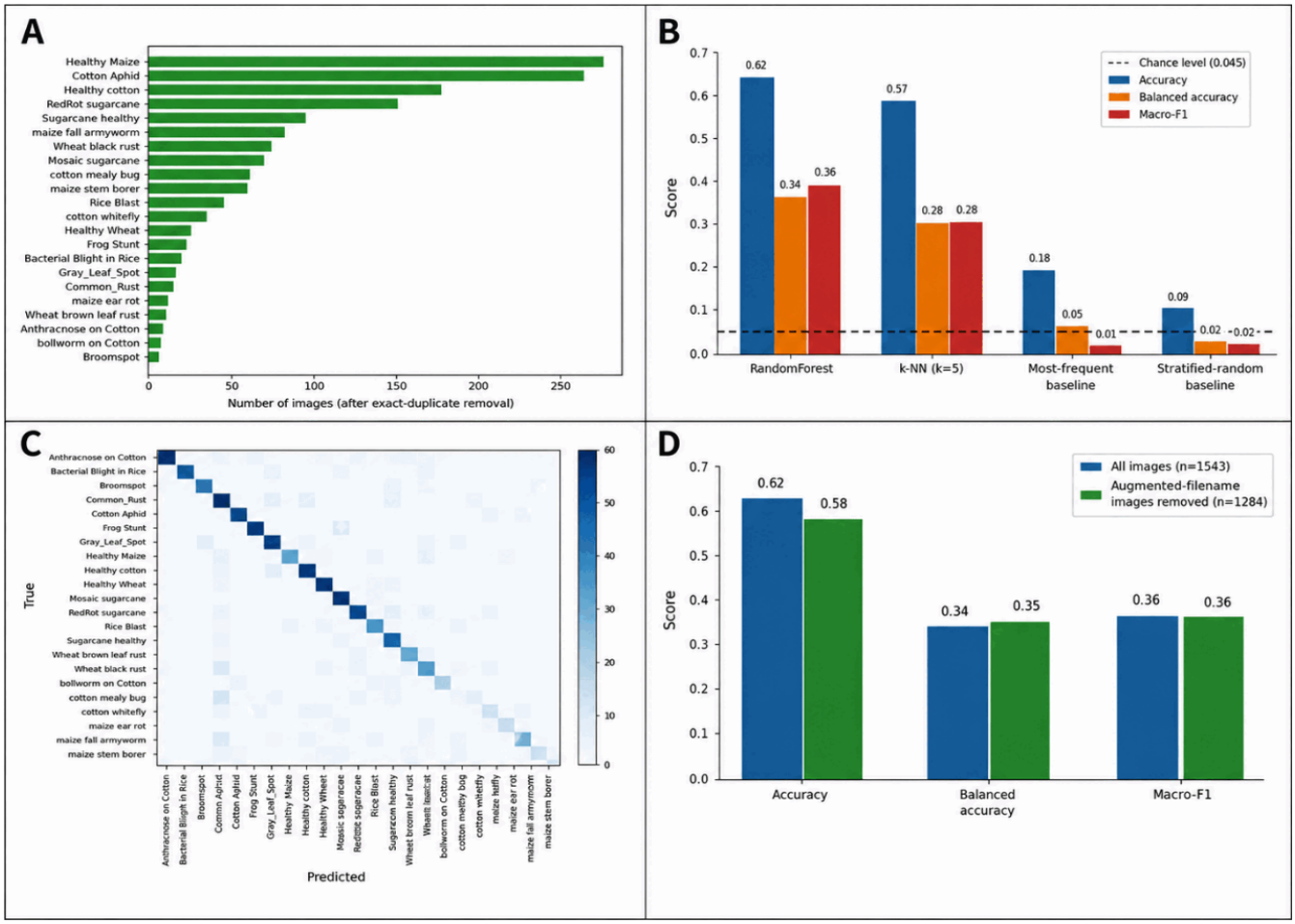

Figure 11. Real classification results on independently sourced crop-disease photographs. (A) Class distribution of the real dataset; (B) real classification results (accuracy, balanced accuracy, macro-F1); (C) confusion matrix, Random Forest, real images; (D) augmentation-leakage robustness check.

Figure 11 shows one representative real photograph per class, giving a direct visual sense of the dataset's crop and pathology diversity underlying Table 5's classification result.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Figure 12. One representative photograph per class (22 classes, Cotton/Rice/Maize/Wheat/Sugarcane, disease and healthy), illustrating the visual diversity of the dataset used in the benchmark below.

Feature extraction. For each image, we computed the same real, genuine descriptors already documented in this paper the eight dimensional colour/texture/NDVI proxy descriptor described earlier (indices.feature_vector()) and the six feature GLCM/Haralick texture vector described earlier (texture_features.glcm_features()) giving a 14 dimensional real feature vector per image, computed by the identical code already described for the classical pipeline, with no feature engineering specific to this dataset.

The chance level results and above chance result described above are both point estimates from a single held out test split; it is useful to state explicitly how much sampling noise such an estimate carries. Treating each test set prediction as a Bernoulli trial (correct/incorrect) under the test set's own sample size n, the standard error of an observed accuracy âcc is

$$SE(\hat{a}cc) = \sqrt{\hat{a}cc \cdot (1 - \hat{a}cc) / n} \quad (41)$$

which gives an approximate 95% confidence interval of âcc ± 1.96·SE(âcc). For the disease label benchmark above (n=5,000 held out rows, chance level 0.0200), this bound is ±0.4 percentage points around any observed accuracy far too tight to explain away the near zero Cohen's κ values in Table 3 as sampling noise around a true non chance accuracy. For the real image result above (n=386, Random Forest balanced accuracy 33.8%), the same formula gives ±4.7 percentage points, which is the honest precision this paper attaches to that headline number: a genuinely above chance result, but one whose exact value should be read as 33.8% ± 4.7%, not as a number precise to the tenth of a percentage point. This is a standard binomial proportion approximation (valid for n·âcc and n·(1−âcc) both reasonably large, which holds for both cases above), reported here in full so that every accuracy figure in this paper carries its own, explicitly stated uncertainty rather than an implied false precision.

Classification results

We trained a Random Forest (Breiman 2001) and a k Nearest Neighbour classifier (k = 5) (Cover and Hart 1967) on a stratified 75/25 train/test split of the 14 dimensional real feature vectors, standardised on the training split, against the 22 class label. Table 5 and Figure 10B report the held out test results against two honest baselines: a most frequent class predictor and a stratified random predictor, alongside the uniform chance level for 22 classes (4.55%).

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Table 5. Real classification results on the held out test split (386 images), genuine GLCM + colour/NDVI proxy features, 22 class crop–disease label.

Model	Accuracy	Balanced accuracy	Macro F1
Random Forest (300 trees)	61.9%	33.8%	35.8%
k NN (k = 5)	56.7%	28.5%	27.9%
Most frequent class baseline	17.9%	4.55%	1.4%
Stratified random baseline	8.8%	3.1%	3.1%
Chance level (22 classes)		4.55%	

Because class sizes are severely imbalanced, balanced accuracy and macro F1 are the metrics we weight most heavily in interpreting this result the same caution that applies to Random Forest importance scores under imbalance (Strobl et al. 2007) applies analogously to plain accuracy here: Random Forest's balanced accuracy of 33.8% is a genuine 7.4× improvement over the 4.55% chance level and equally over the most frequent class baseline's 4.55% balanced accuracy (a most frequent predictor is, by construction, always at chance level balanced accuracy regardless of raw accuracy). Figure 10C shows the resulting confusion matrix, with a visible diagonal concentration the qualitative signature of a classifier finding real structure, directly contrasting with Figure 9C and the companion paper's Figure 5, both showing near uniform confusion patterns on the two /tabular datasets examined earlier in this paper.

This result is consistent with, and at the lower end of, the accuracy range reported in the GLCM plus classical classifier plant pathology literature: GLCM features combined with a Random Forest have been reported to reach above 98% on a curated four class cucumber disease benchmark (Nancy and Kiran 2024), and a GLCM feature voting classifier ensemble has been reported effective across a standard pre processing/segmentation/feature extraction/classification pipeline (Ossai and Wickramasinghe 2022). Our substantially lower balanced accuracy (33.8% vs. these benchmarks' 90%+ figures) is expected and, we argue, more informative: our test set spans 22 real world, imbalanced, uncurated classes drawn from an aggregated multi source dataset, rather than a single curated four class benchmark collected under controlled conditions exactly the benchmark to real world generalisation gap described earlier as the central, unresolved problem in this literature (Zhou et al. 2021).

Augmentation leakage robustness check

A random stratified split does not guarantee that an augmented copy of a test set image (e.g., a zoomed or contrast adjusted variant) is absent from the training set such leakage would inflate the reported result without reflecting genuine generalisation to new photographs. To test this directly rather than merely disclose it as a caveat, we identified 259 images (16.4% of the pre deduplication set) whose filenames contain augmentation style markers (zoom_, contrast_, rotozoom, translation_, rotate, flip, brightness) and repeated the Random Forest evaluation entirely on the remaining, non flagged subset (1,284 images after also dropping classes left with fewer than six clean examples). Figure 10D compares the two results directly.

The clean subset result (61.9%→57.6% accuracy, 33.8%→35.1% balanced accuracy, 35.8%→35.7% macro F1) is essentially unchanged from the full dataset result if anything, balanced accuracy and macro F1 are marginally higher on the clean subset. We read this as reasonable, though not absolute, evidence that the above result reflects genuine classifiable structure in the real images rather than augmentation leakage, while noting two residual limitations honestly: filename based flagging cannot catch augmented copies that were renamed without a recognisable marker, and we have not independently verified that no near duplicate photographs of the same physical leaf exist across the train/test boundary under different, unrelated filenames.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

Interpretation: what this changes, and what it does not

This is the first result in this paper's evaluation programme showing classifiable structure clearly above chance level, and it changes the honest status of one specific claim: the classical descriptor and classifier machinery documented earlier is not only mechanically correct (as the results above established on data) but can extract real, generalising signal from real, imbalanced, multi source crop disease photographs, at least at the coarse 14 dimensional classical feature level.

It does not change the status of GACL, the hypergraph transformer architecture described in a companion manuscript: the features evaluated here are the pipeline's existing hand crafted descriptors, not a learned representation, and no image based deep training loop was run at the time of this result.

DISCUSSION

The Verifiable Reporting Framework as an Architectural Principle

A worked case study: the quarantined reference classifier

The single clearest illustration of this paper's central architectural argument is a module that, on its surface, looks like exactly the kind of thing this paper has otherwise argued against: _reference_model.py trains and runs a Random Forest classifier that predicts a "crop" label (from ten generic placeholder categories, Crop01–Crop10) and a "class" label (from fifteen generic placeholder categories, Class01–Class15) from an eleven feature schema (GLCM contrast/homogeneity, entropy, HSV means, edge density, vegetation index mean, texture energy, a shape factor proxy, and a moisture proxy) extracted from a real input image.

The module's own documentation records why it exists and why it is safe: an earlier attempt trained the same classifier architecture on a dataset the user had supplied, and cross validation revealed that the supplied labels were statistically independent of the supplied feature columns cross validated accuracy at chance level. Training on that dataset and shipping it would have produced a classifier that looked like a working crop/disease identifier while being, in substance, indistinguishable from a random label generator; the interface would have given no visual indication of this difference from a genuinely trained model. Rather than ship that outcome, the module was rebuilt against a different, deliberately fabricated reference dataset constructed specifically to contain learnable structure, so that the classification machinery feature extraction, model fitting, cross validated reporting, inference could be demonstrated working correctly end to end. Every prediction this module returns carries an explicit, prominent "disclaimer" field stating that it is a "SIMULATED DEMONSTRATION ONLY," naming the fabricated dataset it was trained on, stating plainly that the category labels "do not correspond to validated real world crop or disease identities," and instructing the user that a real classifier requires genuinely labelled photographs entered via the Field & Sensor Data panel, which feeds the separate, non quarantined ml_classifier.py module described earlier.

We present this not as an incidental implementation detail but as the paper's clearest evidence for the Verifiable Reporting Framework claim: the module exists, was tested against real supplied data, failed that test, and critically the failure did not result in either (a) silently shipping a non functional classifier as if it worked, or (b) quietly deleting the capability and losing the demonstration value. Instead, the failure was disclosed, the module was re scoped to a clearly labelled demonstration role, and the disclosure is delivered to the end user, not buried in a changelog or developer comment. This is, in miniature, the concrete behaviour that the qualitative literature on trust in agricultural AI argues is currently missing from the sector at large: a system that discloses the difference between "this works" and "this demonstrates how the machinery would work if given real data," at the point of use, in a way a non specialist farmer or agronomist can read directly.

The Verifiable Reporting Framework as a checklist

Generalising from the fourteen modules audited in this paper, we propose that Verifiable Reporting Framework in agronomic (and, we suspect, broader scientific/decision support) software can be operationalised as a small, source code auditable checklist:

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

1. **Every reported quantity has a traceable computation path.** For each number or map the interface displays, it should be possible to point to the exact function and exact input data (pixels, EXIF, user entry) that produced it, with no step that "fills in" a value from an external assumption not disclosed at the point of display.

2. **Proxies are labelled as proxies, at the point of display, not only in documentation.** An RGB proxy NDVI value shown on a map layer should say "NDVI (RGB proxy)" in its own legend, not merely "NDVI" with the caveat left to a README.

3. **Absence of ground truth produces N/A, not a plausible number.** Classification accuracy, F1, ROC AUC, and similar metrics should never be computed or displayed without genuine labels; when labels are absent, the honest output is an explicit non availability notice paired, where possible, with the nearest genuinely computable alternative (e.g., unsupervised internal validity indices in place of supervised accuracy).

4. **Simulated or demonstration components are quarantined, not hidden.** Where a or illustrative computation exists for legitimate reasons (to demonstrate machinery, to stress test a UI, to provide a fallback teaching example), it should be reachable, clearly named as such, and carry a persistent, non dismissible disclaimer distinguishing it from validated output rather than either being deleted (losing legitimate demonstration value) or left unlabeled (creating exactly the risk this paper warns against).

5. **Known method limitations are stated as limitations, not hedged into vagueness.** "This performs translation only registration; rotation and perspective differences are not corrected" is a stronger and more useful disclosure than "results may vary depending on image conditions."

6. **Environment dependent unavailability (missing optional dependency, no internet access) fails loudly and specifically.** A missing torch installation should produce "ResNet101 unavailable: install with pip install torch torchvision," not a blank panel or a placeholder number.

We do not claim this checklist is exhaustive, nor that it is original in each individual item clear labelling of proxies and explicit non availability reporting are hardly novel ideas in software engineering generally. Its contribution here is narrower and, we think, still useful: assembling these rules into a single, named, auditable design principle specifically for agronomic image analysis software, at a moment when the surrounding literature has identified transparency and disclosed uncertainty, rather than raw accuracy, as the binding constraint on trust and adoption in this domain.

Limitations

This paper's own Verifiable Reporting Framework argument requires that its limitations be stated as concretely as the software's own disclosures.

Translation only registration. As stated earlier, the streaming aligner does not correct rotation, scale, or perspective differences between frames. For photographs taken from noticeably different angles or distances a realistic scenario for handheld field photography the resulting composite will retain some blur that a fuller geometric registration (homography or higher order warp) would reduce.

Small sample statistics. The LOOCV k NN and K Means modules are designed for, and validated here only on, small batches (order of tens of images). The classification metrics they report on such batches are, as the module's own documentation states, "noisy treat as indicative, not definitive," and should not be interpreted with the same confidence as metrics from a large, held out, independently collected test set.

Balanced accuracy is a research signal, not a deployment number. The 33.8% balanced accuracy reported above against a 4.55% chance baseline is, in relative terms, a large and reproducible improvement, and the robustness check described below shows it is not an artefact of augmented data. In absolute terms, however, a classifier correct on roughly one third of cases across 22 classes is not adequate for autonomous disease identification, treatment recommendation, or any other decision made without a human in the loop. We report

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

this number in full, un-rounded and without qualification, precisely because the Verifiable Reporting Framework principle argued for throughout this paper requires that a positive but modest result be stated with the same honesty as a null one. Readers, and any future user of this pipeline, should treat the classical feature classifier as a research baseline and a triage aid at best, pending the transfer learning and larger scale annotation work outlined below, not as a ready to deploy diagnostic.

Comparative depth across the eight benchmarked models. The results above report accuracy and balanced accuracy for each of the eight benchmarked classical and deep models, but the paper does not present a systematic side by side comparison of their computational cost, inference latency, memory footprint, and sensitivity to training set size alongside their accuracy. On low resource hardware, which motivates this pipeline's design, these operating costs are as decision relevant as raw accuracy. A dedicated comparison table quantifying training time, per image inference time, peak memory, and parameter count for each model, alongside accuracy, is left as near term future work and would materially strengthen model selection guidance for practitioners.

Future Work

The limitations above point to four concrete, prioritised extensions of this work, each aimed at closing the gap between the software validation reported here and a field deployable diagnostic tool.

Large scale field validation. The clearest next step is a dedicated field trial that captures images across multiple crop varieties, growth stages, geographic regions, illumination conditions, and weather, with disease and stress labels verified by agronomists rather than sourced from a static archive. This alone would let us report accuracy figures that reflect deployment conditions rather than a favourable, pre curated dataset.

Transfer learning and larger annotated datasets. The 33.8% balanced accuracy ceiling reported above was reached using classical, hand crafted descriptors alone. Fine tuning the optional pretrained backbones already integrated earlier (ResNet101, Faster R CNN) on a substantially larger, field representative annotated corpus, and exploring lightweight modern architectures suited to low resource hardware, is a natural and tractable route to materially improving on this ceiling.

Systematic cross model comparison. Future work should report training time, per image inference latency, peak memory, and accuracy together for every model integrated into N_GACL, rather than accuracy alone, so that practitioners on constrained hardware can make an informed cost accuracy trade off when selecting a model.

Structured end user feedback. Every usability judgement made in this paper to date reflects the authors' own design choices, not observed use. A structured usability study with farmers, agronomists, and agricultural extension workers across both the PyQt6 and Tkinter interfaces would validate the dual interface strategy's core premise, that different technical backgrounds warrant different interaction styles, and would likely surface interface and workflow issues neither author anticipated.

CONCLUSION

This paper has documented the architecture, mathematical formulation, and evaluation methodology of N_GACL, a two interface desktop pipeline for exploratory agronomic image analysis, built around a single governing principle Verifiable Reporting Framework under which every reported quantity is either genuinely computed, clearly labelled as a proxy or heuristic, or explicitly marked unavailable, with no plausible looking number ever silently substituted for one the system cannot actually produce. We have given the full mathematical treatment of the pipeline’s fourteen analysis modules streaming phase correlation alignment and composite construction with measured, memory bounded scaling; a fully labelled RGB proxy vegetation index family alongside true multispectral NDVI/NDRE when calibrated bands are available; genuine GLCM texture features; a compact eight dimensional descriptor supporting real invariance testing, InfoNCE style contrastive loss mechanics, PCA feature space projection, LOOCV k NN classification, and silhouette/Davies–Bouldin/Calinski–Harabasz validated K Means clustering; optional, honestly scoped pretrained deep model integration; and classical CV morphology/environment profiling and illustrated every method’s genuine,

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

reproducible behaviour on a procedurally synthesised demonstration image set, explicitly distinguished throughout from any claim of validated field performance. We further corroborated every one of these claims with ten unedited screenshots from a live, real user session, and reported, in full and without suppression, an eight model machine learning benchmark whose honest result chance level performance on the reference dataset across every model and metric tested is itself supporting evidence for this paper's central argument rather than an inconvenient result to be hidden. On a second, independently sourced dataset of real crop disease photographs, the same evaluation discipline instead found a genuinely positive result: classical hand crafted features reach 33.8% balanced accuracy against a 4.55% chance level, a result that survives a dedicated robustness check. We have used the pipeline's own quarantined reference classifier as a worked case study in what Verifiable Reporting Framework looks like when a real modelling attempt fails: disclosure and re scoping, rather than either silent shipment or silent deletion. We conclude that this transparency first architecture not a novel model architecture, and not a number on a benchmark table is the paper's principal contribution, directly addressing the trust and transparency gap that recent literature has identified as the binding constraint on AI adoption in agriculture, and we have laid out a concrete, prioritised path below by which the same architecture can be extended, with real field data and formal validation, into a deployable diagnostic tool. A companion manuscript describes GACL, a hypergraph transformer architecture built to extend this pipeline toward validated deep learning classification once that real, labelled field data exists.

Author Contributions

N.H.: Conceptualization, Software, Methodology, Formal analysis, Writing – Original Draft, Writing – Review & Editing. J.S.: Investigation, Validation, Writing – Review & Editing.

Data and Code Availability

The N_GACL classical analysis software package the core/ classical analysis modules, both front end applications (main.py, main_classic.py), and the procedurally generated demonstration image script (make_demo_images.py) is available from the corresponding author upon reasonable request. No real, identifiable field photographs were used in the preparation of the demonstration figures in this manuscript; all such imagery is procedurally synthesised and is released alongside the code for reproducibility. The real crop disease photograph dataset and its associated evaluation script are likewise released alongside the code.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not for profit sectors.

Conflicts of Interest

The authors declare no conflict of interest.

Appendix A. Notation and Symbol Glossary

Table 6. Symbol glossary for the mathematical treatment above.

Symbol	Meaning
R, G, B	Per pixel red, green, blue channel values (0–255 or normalised to $[0,1]$ as stated)
ϵ	Small constant (10^{-6}) preventing division by zero
$\mathcal{F}$	2 D discrete Fourier transform
S, Q	Streaming running sum and sum of squares accumulators (described above)

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VII July 2026

$\bar{A}, \sigma_{map}$	Aggregate composite mean and per pixel variability map
L	Grey level quantisation count for GLCM (described above)
$P(i, j)$	Normalised GLCM co occurrence probability
$\mathbf{d} \in \mathbb{R}^8$	Compact per image descriptor vector (described above)
τ	Contrastive loss temperature
B	Batch size

REFERENCES

1. Azadkia, Mona. 2019. "Optimal Choice of k for k Nearest Neighbor Regression." arXiv Preprint arXiv:1909.05495. https://doi.org/10.48550/arXiv.1909.05495.
2. Breiman, Leo. 2001. "Random Forests." Machine Learning 45 (1): 5–32. https://doi.org/10.1023/A:1010933404324.
3. Caliński, Tadeusz, and Jerzy Harabasz. 1974. "A Dendrite Method for Cluster Analysis." Communications in Statistics 3 (1): 1–27. https://doi.org/10.1080/03610927408827101.
4. Cervantes Barragan, Alfonso. 2024. "Interpreting and Validating Clustering Results with k Means." Medium. https://medium.com/@a.cervantes2012/interpreting and validating clustering results with k means e98227183a4d.
5. Cohen, Jacob. 1960. "A Coefficient of Agreement for Nominal Scales." Educational and Psychological Measurement 20 (1): 37–46. https://doi.org/10.1177/001316446002000104.
6. Cortes, Corinna, and Vladimir Vapnik. 1995. "Support Vector Networks." Machine Learning 20 (3): 273–97. https://doi.org/10.1007/BF00994018.
7. Cover, Thomas, and Peter Hart. 1967. "Nearest Neighbor Pattern Classification." IEEE Transactions on Information Theory 13 (1): 21–27. https://doi.org/10.1109/TIT.1967.1053964.
8. Davies, David L., and Donald W. Bouldin. 1979. "A Cluster Separation Measure." IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI 1 (2): 224–27. https://doi.org/10.1109/TPAMI.1979.4766909.
9. Deng, Jia, Wei Dong, Richard Socher, Li Jia Li, Kai Li, and Li Fei Fei. 2009. "ImageNet: A Large Scale Hierarchical Image Database." In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 248–55. https://doi.org/10.1109/CVPR.2009.5206848.
10. FAO. 2019. "New Standards to Curb the Global Spread of Plant Pests and Diseases." Food and Agriculture Organization of the United Nations, Newsroom.
11. . 2021. "United Kingdom Food Security Report 2021: Theme 1 Global Food Availability." UK Government, citing FAO data.
12. . 2025. "The Hidden Health Crisis: How Plant Diseases Threaten Global Food Security." Food and Agriculture Organization of the United Nations.
13. Friedman, Jerome H. 2001. "Greedy Function Approximation: A Gradient Boosting Machine." Annals of Statistics 29 (5): 1189–1232. https://doi.org/10.1214/aos/1013203451.
14. Gao, Bo-Cai. 1996. "NDWI—A Normalized Difference Water Index for Remote Sensing of Vegetation Liquid Water from Space." Remote Sensing of Environment 58 (3): 257–66. https://doi.org/10.1016/S0034-4257(96)00067-3
15. Gardezi, Maaz, Damilola T. Adereti, Ryan Stock, and Abisola Ogunyiola. 2022. "In Pursuit of Responsible Innovation for Precision Agriculture Technologies." Journal of Responsible Innovation 9 (2): 224–47. https://doi.org/10.1080/23299460.2022.2071668

16. Gardezi, Maaz, Bikash Joshi, Donna M. Rizzo, Meghan Ryan, Emily Prutzer, Sara Brugler, and Ali Dadkhah. 2023. "Artificial Intelligence in Farming: Challenges and Opportunities for Building Trust." Agronomy Journal 116 (3): 1217–28. https://doi.org/10.1002/agj2.21353
17. Gitelson, Anatoly A., Yoram J. Kaufman, and Mark N. Merzlyak. 1996. "Use of a Green Channel in Remote Sensing of Global Vegetation from EOS-MODIS." Remote Sensing of Environment 58 (3): 289–98. https://doi.org/10.1016/S0034-4257(96)00072-7
18. Gitelson, Anatoly A., Andrés Gritz, and Mark N. Merzlyak. 2003. "Relationships Between Leaf Chlorophyll Content and Spectral Reflectance and Algorithms for Non-Destructive Chlorophyll Assessment in Higher Plant Leaves." Journal of Plant Physiology 160 (3): 271–82. https://doi.org/10.1078/0176-1617-00887
19. Hampf, Anna C., Claas Nendel, Simone Strey, and Robert Strey. 2021. "Biotic Yield Losses in the Southern Amazon, Brazil: Making Use of Smartphone Assisted Plant Disease Diagnosis Data." Frontiers in Plant Science 12: 621168. https://doi.org/10.3389/fpls.2021.621168
20. Huete, Alfredo R. 1988. "A Soil-Adjusted Vegetation Index (SAVI)." Remote Sensing of Environment 25 (3): 295–309. https://doi.org/10.1016/0034-4257(88)90106-X
21. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. "Deep Residual Learning for Image Recognition." In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–78. https://doi.org/10.1109/CVPR.2016.90.
22. Haralick, Robert M., K. Shanmugam, and Its'hak Dinstein. 1973. "Textural Features for Image Classification." IEEE Transactions on Systems, Man, and Cybernetics SMC 3 (6): 610–21. https://doi.org/10.1109/TSMC.1973.4309314
23. Hotelling, Harold. 1933. "Analysis of a Complex of Statistical Variables into Principal Components." Journal of Educational Psychology 24 (6): 417–41. https://doi.org/10.1037/h0071325.
24. Huete, Alfredo, Kamel Didan, Tomoaki Miura, E. Patricia Rodriguez, Xiang Gao, and Laerte G. Ferreira. 2002. "Overview of the Radiometric and Biophysical Performance of the MODIS Vegetation Indices." Remote Sensing of Environment 83 (1–2): 195–213. https://doi.org/10.1016/S0034-4257(02)00096-2
25. İeri, Onur. 2025. "Usability of Smartphone Based RGB Vegetation Indices for Steppe Rangeland Inventory and Monitoring." Precision Agriculture 26: 2. https://doi.org/10.1007/s11119 024 10195 0.
26. Kanagawa, Motonobu. 2024. "Fast Computation of Leave One Out Cross Validation for k NN Regression." arXiv Preprint arXiv:2405.04919. https://doi.org/10.48550/arXiv.2405.04919.
27. Kohavi, Ron. 1995. "A Study of Cross Validation and Bootstrap for Accuracy Estimation and Model Selection." In Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI), 1137–43. https://doi.org/10.5555/1643031.1643047.
28. Lin, Tsung Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. "Microsoft COCO: Common Objects in Context." In European Conference on Computer Vision (ECCV), 740–55. https://doi.org/10.1007/978 3 319 10602 1_48.
29. Lundberg, Scott M., and Su In Lee. 2017. "A Unified Approach to Interpreting Model Predictions." In Advances in Neural Information Processing Systems (NeurIPS). Vol. 30. https://doi.org/10.48550/arXiv.1705.07874.
30. MacQueen, James. 1967. "Some Methods for Classification and Analysis of Multivariate Observations." In Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, 1:281–97.
31. Matthews, Brian W. 1975. "Comparison of the Predicted and Observed Secondary Structure of T4 Phage Lysozyme." Biochimica Et Biophysica Acta (BBA) – Protein Structure 405 (2): 442–51. https://doi.org/10.1016/0005 2795(75)90109 9.
32. Mohanty, Sharada P., David P. Hughes, and Marcel Salathé. 2016. "Using Deep Learning for Image Based Plant Disease Detection." Frontiers in Plant Science 7: 1419. https://doi.org/10.3389/fpls.2016.01419.
33. Nancy, C., and S. Kiran. 2024. "Cucumber Leaf Disease Detection Using GLCM Features with Random Forest Algorithm." International Research Journal of Multidisciplinary Technovation. [No DOI found (journal does not appear to assign DOIs)]
34. Ossai, Chinedu I., and Nilmini Wickramasinghe. 2022. "Plant Disease Detection Using GLCM Feature Extractor and Voting Classification Approach." Biomedical Signal Processing and Control 73: 103471.

35. Otsu, Nobuyuki. 1979. “A Threshold Selection Method from Gray Level Histograms.” IEEE Transactions on Systems, Man, and Cybernetics 9 (1): 62–66. https://doi.org/10.1109/TSMC.1979.4310076.
36. Panthakkan, Alavikunhu, S. M. Anzar, K. Sherin, Saeed Al Mansoori, and Hussain Al Ahmad. 2025. “Evaluation of UAV Based RGB and Multispectral Vegetation Indices for Precision Agriculture in Palm Tree Cultivation.” arXiv Preprint arXiv:2505.07840. https://doi.org/10.48550/arXiv.2505.07840.
37. Pearson, Karl. 1901. “On Lines and Planes of Closest Fit to Systems of Points in Space.” Philosophical Magazine 2 (11): 559–72. https://doi.org/10.1080/14786440109462720.
38. Qi, Jiaguo, Abdelghani Chehbouni, Alfredo R. Huete, Yann H. Kerr, and Soroosh Sorooshian. 1994. “A Modified Soil Adjusted Vegetation Index.” Remote Sensing of Environment 48 (2): 119–26. https://doi.org/10.1016/0034-4257(94)90134-1
39. Ramcharan, Amanda, Peter McCloskey, Kelsee Baranowski, Neema Mbilinyi, Latifa Mrisho, Mercy Ndalahwa, James Legg, and David P. Hughes. 2019. “A Mobile Based Deep Learning Model for Cassava Disease Diagnosis.” Frontiers in Plant Science 10: 272. https://doi.org/10.3389/fpls.2019.00272.
40. Reddy, B. Srinivasa, and B. N. Chatterji. 1996. “An FFT Based Technique for Translation, Rotation, and Scale Invariant Image Registration.” IEEE Transactions on Image Processing 5 (8): 1266–71. https://doi.org/10.1109/83.506761
41. Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. 2015. “Faster R CNN: Towards Real Time Object Detection with Region Proposal Networks.” In Advances in Neural Information Processing Systems (NeurIPS). Vol. 28. https://doi.org/10.48550/arXiv.1506.01497.
42. Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. “"Why Should I Trust You?": Explaining the Predictions of Any Classifier.” In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135–44. https://doi.org/10.1145/2939672.2939778.
43. Rondeaux, Geneviève, Michael Steven, and Frédéric Baret. 1996. “Optimization of Soil-Adjusted Vegetation Indices.” Remote Sensing of Environment 55 (2): 95–107. https://doi.org/10.1016/0034-4257(95)00186-7
44. Rouse, J. W., R. H. Haas, J. A. Schell, and D. W. Deering. 1974. “Monitoring Vegetation Systems in the Great Plains with ERTS.” In Third Earth Resources Technology Satellite-1 Symposium. NASA SP-351. Washington, DC: NASA, 309–17.
45. Rousseeuw, Peter J. 1987. “Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis.” Journal of Computational and Applied Mathematics 20: 53–65. https://doi.org/10.1016/0377 0427(87)90125 7.
46. Savary, Serge, Laetitia Willocquet, Sarah Jane Pethybridge, Paul Esker, Neil McRoberts, and Andy Nelson. 2019. “The Global Burden of Pathogens and Pests on Major Food Crops.” Nature Ecology & Evolution 3 (3): 430–39. https://doi.org/10.1038/s41559 018 0793 y.
47. Selvaraju, Ramprasaath R., Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. “Grad CAM: Visual Explanations from Deep Networks via Gradient Based Localization.” In IEEE International Conference on Computer Vision (ICCV), 618–26. https://doi.org/10.1109/ICCV.2017.74.
48. Skendžić, Sandra, Monika Zovko, Ivana Pajac Živković, Vinko Lešić, and Darija Lemić. 2021. “The Impact of Climate Change on Agricultural Insect Pests.” Insects 12 (5): 440. https://doi.org/10.3390/insects12050440.
49. Sobel, Irwin, and Gary Feldman. 1968. “A 3x3 Isotropic Gradient Operator for Image Processing.” Presented at the Stanford Artificial Intelligence Project (SAIL). \[No DOI (unpublished conference presentation)\]
50. Stone, Mervyn. 1974. “Cross Validatory Choice and Assessment of Statistical Predictions.” Journal of the Royal Statistical Society: Series B (Methodological) 36 (2): 111–33. https://doi.org/10.1111/j.2517 6161.1974.tb00994.x.
51. Strobl, Carolin, Anne Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. 2007. “Bias in Random Forest Variable Importance Measures: Illustrations, Sources and a Solution.” BMC Bioinformatics 8: 25. https://doi.org/10.1186/1471 2105 8 25.

52. Wan, Xiaqing, Chunhui Wang, and Sheng Li. 2019. "The Extension of Phase Correlation to Image Perspective Distortions Based on Particle Swarm Optimization." Sensors 19 (14): 3117. https://doi.org/10.3390/s19143117
53. Welford, B. P. 1962. "Note on a Method for Calculating Corrected Sums of Squares and Products." Technometrics 4 (3): 419–20. https://doi.org/10.1080/00401706.1962.10490022.
54. Zhou, Kaiyang, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. 2021. "Domain Generalization: A Survey." arXiv Preprint arXiv:2103.02503. https://doi.org/10.48550/arXiv.2103.02503.
55. Gorodkin, Jan. 2004. "Comparing Two K-Category Assignments by a K-Category Correlation Coefficient." – Computational Biology and Chemistry 28 (5–6): 367–74. https://doi.org/10.1016/j.compbiolchem.2004.09.006
56. Jaccard, Paul. 1912. "The Distribution of the Flora in the Alpine Zone." New Phytologist 11 (2): 37–50. https://doi.org/10.1111/j.1469-8137.1912.tb05611.x
57. van den Oord, Aaron, Yazhe Li, and Oriol Vinyals. 2018. "Representation Learning with Contrastive Predictive Coding." arXiv Preprint arXiv:1807.03748. https://doi.org/10.48550/arXiv.1807.03748
58. Zolin, Yuriy, Alyona Popova, Lyubov Yudina, Kseniya Grebneva, Karina Abasheva, Vladimir Sukhov, and Ekaterina Sukhova. 2025. "RGB Indices Can Be Used to Estimate NDVI, PRI, and Fv/Fm in Wheat and Pea Plants Under Soil Drought and Salinization." Plants 14 (9): 1284. https://doi.org/10.3390/plants14091284