KH9 land cover classification workflow guideline

Author: Roman Sitko

This guide explains how to configure, run and interpret the supplied seven-notebook workflow for supervised land cover classification of orthorectified KH-9 grayscale imagery. The workflow trains a Random Forest on two to four represented classes, validates it with spatial cross-validation, and exports classified maps with QC exclusions. It is intended for GIS users and researchers, and for developers integrating the notebook steps into an application.

Contents

1 Use case overview 2

2 Inputs and the classification legend 2

3 Setup and coordinate reference systems 3

4 Image and terrain features 4

5 References scene selection and spatial validation 5

6 Random Forest training and model use 6

7 Evaluation and scientific interpretation 7

8 Outputs and reproducibility 9

9 Operation checks and troubleshooting 10

10 Demonstration and future extensions 10

1 Use case overview

Item	Specification
Use case	Static historical land cover classification with spatial validation
Demonstration region	TANAP / High Tatras, Slovakia; Hrebienok, Bielovodska and Uplaz scenes
Scientific question	How reliably can grayscale tone, texture and optional terrain distinguish the selected land cover classes within and across scenes?
Workflow form	Seven standalone Jupyter notebooks, numbered 00-06
Primary indicators	Per-class precision, recall and F1; scene Macro F1, Cohen kappa and accuracy
Required source metadata	Record image IDs, camera/product type, acquisition dates, pixel size, extent and orthorectification quality before interpreting results. These are not established by the default filenames.

The original project context uses three scenes of approximately 5 × 5 km at 0.5 m pixel spacing. Read the actual grid and CRS from each input; do not assume that these dimensions apply to a new project. The workflow expects already orthorectified data and does not perform image orthorectification or two-date co-registration.

Three levels of reproducibility

At the computation level, functions define inputs, parameters, calculations and exports. At the notebook level, settings, implementation, visual checks and interpretation are combined. At the use case level, the seven steps form a documented demonstrator with a shared legend, provenance, validation and output products. This follows the structure requested in Use_cases_guideline.odt.

The ODT also proposes temporal change, landscape metrics, edge/core analysis and ecological continuity. Those are extension ideas, not implemented products of this classification package. Section 10 explains what is required before adding them.

2 Inputs and the classification legend

For each scene, configure an image and the associated reference and QC layers in SCENES in notebook 00. A DEM and AOI are optional. Keep all shapefile components together. Use stable scene_id values throughout the project.

Input	Requirements and purpose
KH-9 image	Single-band, north-up orthorectified raster; valid integer DN 0-255. Preserve original radiometry. A wider integer storage type is acceptable if valid values meet this range.
Training/reference polygons	Valid polygons with poly_id, subclass, main_cls, site_id, conf and train_role. poly_id must be non-empty and unique within a scene; Id is an accepted alias. confidence is an alias for conf.
QC polygons	QC_ID identifies mask classes. The default excludes codes 1-4 and allows 0/unmasked centres. Interpret each QC code consistently in the project.
DEM	Required only for terrain features; supply a verifiable CRS, NoData and the vertical-unit conversion to metres.

Input	Requirements and purpose
AOI	Optional polygon layer restricting sample centres and output map coverage.
External evaluation data	Optional classified rasters with explicit code mappings, or independent reference points/polygons. Configure in 06.

main_cls	Class	subclass codes
1	Forest	10, 11, 12
2	Dwarf pine and shrubs	20, 21
3	Other vegetated land cover	30, 31
4	Non-vegetated land and water	40, 41, 42, 43, 44

Forest means image-recognizable tree vegetation cover in this project. Apply the project annotation definitions consistently; this guide does not invent new subclass definitions. Subclass 12 must retain train_role=CHECK. Training uses CORE polygons with conf=1; other roles and confidence levels remain audited. Invalid reference geometries and overlapping training polygons require correction before proceeding.

Assign the same site_id to polygons belonging to the same physical site when they must stay together in validation. Blank site_id values are not treated as one shared site. Use SITE_SCOPE="global" for physical sites shared across scenes; otherwise the default scope is scene-specific.

Before running 04, inspect image/reference alignment, class attribution, polygon boundaries and QC coverage. Reference absence alone does not establish real landscape absence; the project owner must confirm that the annotations cover the intended class set.

3 Setup and coordinate reference systems

Start the project

Keep the seven IPYNB files in one folder. Create the environment from environment.yml or use an existing compatible JupyterLab environment. requirements_observed.txt records an earlier observed environment; it is not a guarantee that every later package combination has been tested.

```
conda env create -f environment.yml
conda activate kh9-user-workflow
jupyter lab
```

In 00, edit PROJECT_ROOT, OUTPUT_ROOT and the SCENES entries. The supplied default remains C:\UC1. All executable paths and defaults are unchanged in the English twin. Run all cells in each notebook. The implementation cell can be collapsed but must execute. No additional local Python module is needed.

00 creates KH9_project.json in the kernel working folder. Other notebooks discover it automatically. If working elsewhere, set MANIFEST_PATH to the generated project_manifest.json. The notebook prints the selected source run. On another computer, run 00 with the new paths.

Local names and verified assignments

A local CRS name is acceptable when the full geographic/projected definition is valid and can be verified. Proven equivalent definitions are canonicalized. A name alone, a missing CRS or an incomplete LOCAL_CS does not establish a usable coordinate system.

USE_PREVIOUS_KH9_CRS=True preserves the narrowly defined compatibility rule for the three known original KH9 images/DEMs and their verified EPSG:5514 assignment. It checks filenames, units, grid orientation, geographic plausibility and overlap. For a new project, set it to False and provide a verified override only where needed. The rule does not overwrite another valid CRS.

WRITE_CORRECTED_COPIES=True creates copies with corrected CRS metadata; source pixels and files remain unchanged. Assigning CRS metadata is different from reprojecting coordinates. Consult crs_resolution_report.json/csv to see which method was used.

Working with WGS84

Geographic inputs such as EPSG:4326 are supported. Distances, blocks, erosion and DEM derivatives use a metric analysis CRS rather than degrees. For a small local geographic project, 00 can select local UTM; ANALYSIS_CRS permits an explicit suitable metric CRS. Large extents, polar locations outside UTM coverage and antimeridian crossings require separate preparation.

Maps retain the image CRS by default. For additional WGS84 exports, set MAP_CRS="EPSG:4326" and MAP_MODE="create" in 06. This output choice does not require retraining. Terrain aspect is relative to north of the metric DEM grid, not a stated true geographic azimuth.

4 Image and terrain features

For new image features, run 00 → 01 → 02 → 03. If compatible image features already exist, run 00 → 03. Existing layers are reused by default. OVERWRITE_FEATURES=True requests deliberate regeneration in the relevant step.

Step	Inputs and computation	Checks and exports
00 Input QC	Image, references, QC, optional AOI/DEM; resolve CRS and inspect valid DN	DN histograms, QC rasters, geometry diagnostics, effective CRS, project manifest
01 Basic features	Original DN; mean, standard deviation and entropy in 5, 11 and 21 px windows	Feature GeoTIFFs and generation records; DN remains the original raster
02 GLCM and pattern	Contrast, homogeneity and ASM in 5, 11 and 21 px windows; optional NDC/CVN for profile 43	Feature GeoTIFFs, cached-layer checks and processing records
03 Catalogue and DEM	Verify required features and grid/CRS; generate extended features for profile 43 and optional DEM derivatives	Feature catalogue/signatures, feature-support erosion check, terrain rasters and availability records

The default 19 features

FEATURE_COUNT=19 selects DN plus mean, STD and entropy at three window sizes, and GLCM contrast, homogeneity and ASM at the same sizes. Mean/STD use original DN. Entropy uses DN//16, 16 levels and nats. GLCM uses distance 1 pixel, four directions, non-symmetric matrices and the mean of directional properties. No Gabor or other unused layers are required for this profile.

FEATURE_COUNT=43 adds six NDC/CVN, four Gabor, four top-hat and ten LBP features. Windows remain in pixels; changing image resolution changes their physical interpretation. Tile halos preserve neighbourhood calculations. Raster NoData is respected across filter support; thematic QC excludes sample/map centres and is not applied to image DN before filtering.

Optional terrain features

Set CREATE_DEM_FEATURES=True in 03 to generate elevation, slope in degrees, and sine/cosine of aspect. DEM_Z_SCALE_TO_M converts vertical units; DEM_RESOLUTION_M optionally sets metric resolution. Elevations are first prepared on a metric north-up grid, Horn derivatives are calculated there, and features are aligned to the image grid. Flat terrain below 0.1° slope has aspect sine/cosine equal to zero.

Set USE_DEM=True in 05 to train with terrain. COMPARE_DEM performs a paired comparison on common valid samples. COMPARE_ASM compares predictive benefit with and without ASM; it does not recreate ASM or replace its existing generation-cost report.

Inspect the catalogue, missing-layer diagnostics and effective CORE erosion before proceeding. Feature generation and mapping are tiled, but exact ASM and the full profile may still be expensive. No processing-time estimate is implied by the synthetic verification.

5 References scene selection and spatial validation

Notebook 04 reads the successful 03 output, selects development scenes, audits annotations, applies CORE erosion and masks, creates spatial groups/folds, samples pixel centres and exports the feature matrix with fixed CV indices. It does not train the RF.

Choose training scenes

```
TRAIN_SCENES = None                # all scenes
# Or use only Scene 2
TRAIN_SCENES = ["bielovodska"]
```

Use one of these settings, not both. A single string is also supported. Exact scene_id values take priority; aliases must be unique in metadata or filenames. Numbers identify explicitly numbered scenes, not their positions in the list. Check the printed scene table and scene_selection.json before proceeding. Unknown, duplicate and ambiguous selections are errors.

Two to four training classes

Active classes are derived from annotations in the selected training scenes before erosion, QC and feature filtering. The union across selected scenes must contain at least two classes. Original IDs remain unchanged: Scene 2 references containing only 1, 3 and 4 yield a three-class training set. With all scenes selected, class 2 remains active if present elsewhere.

A class annotated only as HARD/CHECK, or a class whose valid samples disappear during filtering, is not silently declared absent. Supply appropriate CORE/conf=1 data or resolve the diagnosed sampling issue. training_classes.json and training_class_availability.csv document the active and absent classes.

Parameter	Supplied default	Meaning
BLOCK_SIZE_M	1000 m	Spatial block size
CV_BUFFER_M	50 m	Training exclusion around validation blocks
N_FOLDS	3	Spatial validation folds
MAX_SAMPLES_PER_POLYGON	250	Maximum sampled centres per polygon
MIN_SAMPLE_SPACING_M	2 m	Minimum within-polygon sample spacing
CORE_EROSION_M	None	Inherit the safe feature-support erosion from 03
SITE_SCOPE	scene	Use global when site IDs refer to shared physical sites
SEED	20261003	Reproducible sampling and fold design

Entire polygons, shared sites and connected blocks remain together. Training is purged near validation blocks. Active main classes must be represented in training and validation; too few independent groups can still stop the run. Do not bypass this with random pixel splitting. Inspect fold maps, polygon_eligibility.csv, subclass_fold_coverage.csv and fold summaries before training. Block and buffer sizes are design choices, not measured autocorrelation ranges.

6 Random Forest training and model use

Notebook 05 loads the successful 04 samples and exact saved splits. It fits models for spatial CV, writes out-of-fold predictions and validation results, then trains and saves the selected production model on all eligible training samples. It does not choose its model by examining later transfer results.

Setting	Supplied value	Effect
USE_DEM	False	Image-only production model by default
COMPARE_ASM	False	Optional comparison without ASM
COMPARE_DEM	False	Optional paired image/terrain comparison
n_estimators	350	Same tree count for CV and final fit
min_samples_leaf	5	Minimum leaf sample count
max_features	sqrt	Features considered at each split
class_weight	balanced_subsample	Class weights per bootstrap sample
n_jobs	-1	Use available parallel processing

Outputs include fold/class metrics, confusion matrices, OOF predictions, feature importance and a model bundle carrying the feature order, class legend, active labels, configuration and provenance. Feature importance indicates model usage; it is not evidence of a causal relationship.

Training coverage limits prediction

A model trained without class 2 cannot predict class 2. This is appropriate for a scene where that class is genuinely absent, but it limits transfer to a scene containing it. Reducing the class set does not guarantee better scores for the remaining classes. Validate the new model with its own predictions.

Notebook 06 loads the saved model

The final model from 05 is reused for production maps and direct transfer evaluation. No additional model is trained in 06. Training-scene evaluation uses OOF predictions from 05 to avoid presenting training-fit performance as validation.

06 control	Choices and meaning
SCENES_TO_PROCESS	training (supplied default), all, other, or an exact list of scene_id values
MAP_MODE	auto: reuse compatible maps or create; reuse: require compatible maps; create: always create selected maps
EVALUATE_TRANSFER	True by default; evaluate selected non-training scenes when references exist
WRITE_PROBABILITY_MAP	True by default; export maximum predicted class probability
MAP_CRS	None preserves source CRS; an explicit CRS adds exports and requires MAP_MODE=create

For maps of all scenes, set SCENES_TO_PROCESS="all". The notebook UI default is training, although the reusable run06 function's omitted-argument default remains all. Application developers should pass this argument explicitly. Scene selection does not change the trained model or its class count.

7 Evaluation and scientific interpretation

Read tables/scene_metrics.csv together with tables/scene_class_metrics.csv. Both identify the scene explicitly. Inspect confusion matrices, support counts and map previews; a single summary score cannot identify which classes are confused.

Indicator	Interpretation
Precision	For a class, the proportion of its predictions that are correct; comparable to user accuracy on the evaluated samples.
Recall	For a class, the proportion of its reference samples recovered; comparable to producer accuracy on those samples.
F1	Harmonic mean of class precision and recall. It balances omission and commission errors.
macro_f1	Unweighted mean F1 over classes with positive reference support in this evaluation set.
macro_f1_fixed_legend	Additional historical comparison over all four legend IDs, with zero for undefined F1.
kappa	Cohen agreement index for the scene, accounting for agreement expected from the observed marginals.

Indicator	Interpretation
kappa_ovr	Binary agreement for one class versus all others; different from scene-level multiclass kappa.
accuracy	Fraction of correctly classified evaluated samples; influenced by their class distribution.

Check macro_labels and n_macro_classes when comparing scenes. Bielovodska references without class 2 are averaged over 1, 3 and 4. A class present in references but never predicted remains in Macro F1 with F1=0. Incorrect predictions of an absent-reference class remain errors for the true classes; they are not discarded.

Recall is N/A with no reference support; precision is N/A with no predicted support. F1 is N/A only when both supports are zero. Class kappa is N/A without positive or negative reference support. Empty CSV cells represent these undefined values. train_labels and learned_by_model distinguish reference coverage from model knowledge.

External and independent evaluation

For an external classified raster, specify its scene, path, class_mapping, nodata_codes and provenance in EXTERNAL_CLASSIFICATIONS. Add reference_mapping for a coarser common legend. Compare both models on common valid reference centres and inspect coverage. An external map is not automatically reference truth; different dates and legends affect interpretation.

INDEPENDENT_REFERENCE accepts separately supplied points/polygons with main_cls. They must be geometrically separate from annotations and their CV buffers; QC, AOI and feature validity still apply. Targeted CORE references and a convenience independent sample do not establish design-based whole-map accuracy.

Maximum predicted class probability is not calibrated probability of correctness. Differences in illumination, shadows, terrain, season, reference interpretation and spatial alignment remain possible sources of uncertainty. Report these with the validation scope and sample support.

8 Outputs and reproducibility

Location or file	Purpose
Output/00_QC	Input checks, histograms, QC rasters and corrected CRS copies when requested; reports/project_manifest.json
Output/01_basic_local_features and 02_pattern_glcm_features	Default new basic and GLCM feature rasters, organized by scene
Output/03_extended_features	Optional new Gabor, top-hat and LBP feature rasters
Output/FEATURES/04_terrain_features_v2	Terrain feature rasters and availability metadata
Output/MODELS/04_samples_spatial_cv	Time-stamped sample runs, CV indices, annotation/fold audits and maps
Output/MODELS/05_rf_ablation	Training runs, OOF tables, metrics, importance and model bundles
Output/MODELS/06_scene_evaluation	Evaluation runs, tables, figures, scene outputs and maps_manifest.json
<scene_id>_classes.tif	Classified raster with original class IDs and 0 as NoData
<scene_id>_max_probability.tif	Optional maximum predicted class probability
<scene_id>_exclusion_reason.tif	Bit-coded reasons why a pixel was excluded
review_bundle.zip	Portable review reports, tables and figures; excludes large map TIFFs and RF models

These examples use the default project Output folder. OUTPUT_ROOT can select a different folder. The catalogue searches existing image layers first under Output/FEATURES and uses the corresponding older direct Output folders as fallbacks; newly generated image layers use the direct folders shown above. Use the catalogue and maps_manifest.json for authoritative paths rather than guessing run folder names.

Exclusion bits are 1 raw NoData, 2 invalid image feature, 4 QC, 8 outside AOI and 16 DEM. Multiple reasons can be combined. Excluded class-map pixels are 0. QC repairs are audited; original source shapefiles are preserved.

Keep a reproducible record

Retain input files and shapefile components, image/date/product metadata, notebook settings, the environment record, project manifest, configuration snapshots, fold indices, model bundle and evaluation scope. Automatic IDs select the latest successful compatible run; no manual run_id is needed. Failed runs do not replace the successful pointer.

Changing 04 requires new 05 and 06. Changing 03 requires downstream 04-06. Changed inputs require a new 00 and subsequent steps; existing compatible image features can still be reused. Starting 00 begins a new chain. In 06, map reuse is allowed only for the compatible saved model and recorded dependencies. Preserve the earlier map directory if a reused-map manifest points to it.

9 Operation checks and troubleshooting

Situation	Action
Unknown training scene	Read the printed scene table; use an exact scene_id or an unambiguous metadata alias. Do not interpret a number as list order.
Only one annotated class	Provide at least two genuinely represented classes or select a suitable set of training scenes. Do not fabricate a missing class.
Class lost after masks or in folds	Inspect polygon eligibility, group counts and fold summaries. Resolve reference coverage or spatial design; do not silently drop an annotated class.
Missing or incomplete CRS	Check the CRS report. Verify the source CRS independently and supply an override if appropriate; a local name alone is insufficient.
NoData inside DN 0-255	Verify how the input encodes validity before choosing RAW_NODATA_POLICY. Valid DN must not be discarded accidentally.
Stale features or outputs	Repeat the indicated upstream step and its dependants. Use OVERWRITE_FEATURES only when regeneration is intended.
06 maps only training scenes	Set SCENES_TO_PROCESS="all". MAP_MODE alone does not select scenes.
reuse rejects existing maps	The model, dependencies or output settings differ. Use auto/create when new maps are intended.
Warning during execution	Read its text and subsequent progress/status. A warning alone does not establish either success or failure; inspect the final status and traceback.

Notebook and application interfaces

Each notebook follows settings → reusable functions → execution and display → interpretation. run00 through run06 are the public entry points. notebook_interfaces.json lists signatures extracted from this supplied code, including scenes_to_process in run06. An application should explicitly pass user choices and show the resolved project, scenes, active classes and output paths.

The English twin preserves calculations, variable/function names, feature identifiers, class codes, configuration keys, machine-readable states, file paths and executable defaults. It translates explanatory markdown, comments, diagnostics, report text and figure labels. Place names remain proper names. The original Slovak archive is not overwritten.

The operating guide is supplied as KH9_Use_Case_Guideline_EN.pdf. TRANSLATION_MANIFEST.json records the source archive and renamed documentation. ENGLISH_TWIN_VERIFICATION.md distinguishes checks for this edition from the historical TEST_REPORT.md. No new real-image accuracy result is claimed by translation or by synthetic tests.

10 Demonstration and future extensions

A concise demonstration sequence

Begin with the historical image, its acquisition metadata and the shared legend. Show the QC mask and reference polygons. Explain the 19-feature profile and the reason for spatial validation. Display the selected scenes, active classes and fold map. Finish with scene/class metrics, a classified map and exclusion mask, then discuss limitations and transfer coverage.

Suggested presentation material

Slide	Evidence to show
1 Question and region	Scene locations, acquisition dates and the mapping objective
2 Inputs and annotation	Original image, legend, training polygons and QC coverage
3 Reproducible processing	00-06 sequence, feature profile, optional terrain and saved manifests
4 Validation design	Spatial fold map, site grouping and buffer explanation
5 Classification outputs	Map preview, exclusion mask, class precision/recall/F1 and scene metrics
6 Limits and reuse	Absent-class policy, transfer limitations, provenance and extension requirements

This is a presentation outline, not a supplied slide deck. Use figures produced by the actual project run and identify their scene and evaluation scope. Do not substitute historical or synthetic test scores for current results.

Extensions proposed by the ODT

Forest/non-forest maps could be derived from the agreed main-class legend, retaining NoData. Landscape metrics, minimum mapping units, speckle filtering, patch vectorization, edge/core areas and connectivity would require additional modules and explicit parameter choices. They are not automatically produced by these seven notebooks.

Two-date forest loss/gain and persistence would additionally require acquisition-date records, co-registration quality assessment, aligned grids, consistent legends and comparable classification rules. Apparent small boundary changes may reflect misregistration rather than land cover change. Ecological continuity or disturbance causes require evidence beyond a categorical map.

Cross-region replication should preserve feature definitions and document differences in resolution, annotation effort, class coverage and validation design. Whole-area accuracy estimates, uncertainty intervals and ecological indicators require their own sampling and analytical methods. The current outputs provide inputs and provenance for such extensions, not proof that the extensions have been implemented.

Source basis

This guideline follows the three-level structure and use case headings in the supplied Use_cases_guideline.odt. Operational details are grounded in the seven notebooks and accompanying text files in KH9_Final_workflow_v1.0.zip. Historical notes were retained as historical material; the current 04 alias resolver and 06 scene-selection defaults take precedence over older explanatory text. Acquisition dates and real-data classification results were not supplied in this archive and must be completed from the project records.