

Supplementary Methods

p53 immunohistochemistry quantification in gastric adenocarcinoma: digital image analysis pipeline (QuPath v0.7.0)

Quantification of p53 immunohistochemistry proceeded in four stages: (1) three-tier interpretation of the p53 expression pattern by pathologists, (2) manual H-score assessment, (3) digital cell detection and tumor-cell classification with formal validation, and (4) calibration of the DAB intensity thresholds and computation of the digital H-score. All digital analyses were performed in QuPath v0.7.0. The in-house Groovy tools used at each stage are provided as Supplementary Software and are illustrated in Supplementary Fig. S2.

The pipeline was not executed as a single batch process. Cell detection, classifier training, classifier application, performance validation, threshold calibration, and H-score computation were each carried out sequentially in a separate QuPath project. In particular, the legacy watershed and InstanSeg detection routes were built as two entirely independent pipelines, each with its own classifier and its own threshold calibration; the two are not interchangeable.

S1. Three-tier interpretation of the p53 expression pattern

p53 immunostaining was classified into three patterns according to standard criteria: wild-type, null (complete loss), and overexpression (diffuse strong nuclear positivity). Two pathologists (Reader 1, S.W.C.; Reader 2, K.K.), blinded to all clinicopathologic data and to each other's assessment, interpreted every core independently using an in-house scoring tool (Manual p53 Pattern Scorer) (Supplementary Fig. S2A). Interobserver agreement between these two independent readings is given in Supplementary Table S1 (500 of 545 cores concordant, 91.7%; Cohen's $\kappa = 0.845$; linear-weighted $\kappa = 0.861$). Discordant cores were subsequently reconciled in a joint review by three pathologists (S.W.C., K.K., and J.A.), and this consensus classification was used as the reference standard in all downstream analyses.

S2. Manual H-score assessment

The manual H-score of the tumor cells was recorded with an in-house tool (Manual H-Score Scorer) (Supplementary Fig. S2B). The percentage of positive cells at each intensity grade was estimated in 5% increments and the H-score computed as $1 \times (\%1+) + 2 \times (\%2+) + 3 \times (\%3+)$, range 0 to 300, on the principle of one core = one case = one H-score. A single pathologist (S.W.C.) read every case twice, 2 weeks apart and blinded to the first reading, and the arithmetic mean of the two readings was used as the final manual H-score in all analyses. Intraobserver (test-retest) reproducibility is reported in Supplementary Table S5.

S3. Digital cell detection, tumor-cell classification, and classifier validation

S3.1 Workflow organization

Each stage of the pipeline was executed in its own QuPath project, and the output of each stage (classifier, threshold set) served as the input to the next.

Stage	Tools / scripts	Output
1. Cell detection + classifier training (legacy)	Legacy detection script (Listing S1); interactive Random Trees training	Classifier p53_Legacy_v01 (feature importance, Supplementary Table S3B)
2. Cell detection + classifier training (InstanSeg)	InstanSeg detection script (Listing S2); interactive Random Trees training	Classifier p53_InstanSeg_v01 (feature importance, Supplementary Table S3A)
3. Classifier application	runObjectClassifier (Listing S3)	Cells classified as Tumor or Other
4. Classifier validation	Validation GT Sampler, Consensus GT Merger, Object Classifier Validator (Supplementary Fig. S2C-F)	Confusion matrix and performance metrics (Supplementary Table S2)
5. Intensity-threshold calibration	Cell Grade Labeler, IHC Threshold Calibrator (Supplementary Fig. S2G, S2H)	DAB optical-density cutoffs (Supplementary Table S4)
6. Digital H-score computation	setCellIntensityClassifications setDetectionIntensityClassifications (Listing S4)	/ Per-case H-score

S3.2 Cell detection and feature computation

Within each build project, cells inside the annotated Tumor regions were detected and their features computed in a single continuous script. The legacy and InstanSeg routes differ only in the detection algorithm; both use the same H-DAB color deconvolution and the same intensity and Haralick texture features, computed on the nuclear region at 1 and 2 micrometers per pixel. A single H-DAB stain vector set was defined for this cohort and applied uniformly to every slide.

Listing S1. Legacy detection (watershed).

```
setColorDeconvolutionStains({'Name': "H-DAB p53",
    "Stain 1": "Hematoxylin", "Values 1": "0.69466 0.58636 0.41669",
    "Stain 2": "DAB", "Values 2": "0.38793 0.56464 0.72848",
    "Background": " 249 249 249"});
```

```
selectObjectsByClassification("Tumor");
runPlugin('qupath.imagej.detect.cells.WatershedCellDetection', '{
    "detectionImageBrightfield": "Optical density sum",
    "requestedPixelSizeMicrons": 0.5,
    "backgroundRadiusMicrons": 8.0, "backgroundByReconstruction": true,
    "medianRadiusMicrons": 0.0, "sigmaMicrons": 1.5,
    "minAreaMicrons": 10.0, "maxAreaMicrons": 400.0,
    "threshold": 0.1, "maxBackground": 2.0,
    "watershedPostProcess": true, "excludeDAB": false,
    "cellExpansionMicrons": 3.0, "includeNuclei": true,
    "smoothBoundaries": true, "makeMeasurements": true}');
```

```
selectDetections();
```

```
// intensity + Haralick features at 1.0 and 2.0 micrometers per pixel
runPlugin('qupath.lib.algorithms.IntensityFeaturesPlugin', '{
    "pixelSizeMicrons":1.0, "region":"NUCLEUS", "tileSizeMicrons":25.0,
    "colorOD":true, "colorStain1":true, "colorStain2":true,
    "doMean":true, "doStdDev":true, "doMinMax":true, "doMedian":true,
    "doHaralick":true, "haralickDistance":1, "haralickBins":32}');
runPlugin('qupath.lib.algorithms.IntensityFeaturesPlugin', '{
    "pixelSizeMicrons":2.0, "region":"NUCLEUS", ... "haralickBins":32}');
```

Listing S2. InstanSeg detection (deep-learning nuclear segmentation).

```
setColorDeconvolutionStains({'Name' : "H-DAB p53",
    "Stain 1" : "Hematoxylin", "Values 1" : "0.69466 0.58636 0.41669",
    "Stain 2" : "DAB",           "Values 2" : "0.38793 0.56464 0.72848",
    "Background" : " 249 249 249"});

selectObjectsByClassification("Tumor");
qupath.ext.instanseg.core.InstanSeg.builder()
    .modelPath("../instanseg/downloaded/brightfield_nuclei-0.1.1")
    .device("gpu0")
    .inputChannels([ColorTransforms.createChannelExtractor("Red"),
        ColorTransforms.createChannelExtractor("Green"),
        ColorTransforms.createChannelExtractor("Blue")])
    .outputChannels().tileDims(512).interTilePadding(32)
    .nThreads(4).makeMeasurements(true).randomColors(false)
    .outputType("Default").build().detectObjects();

selectDetections();

// same intensity + Haralick features as Listing S1, at 1.0 and 2.0 micrometers per pixel
```

S3.3 Tumor-cell classification (Random Trees)

Within each build project, a Random Trees object classifier was trained to separate tumor cells from other cells. Training was interactive (expert-in-the-loop): a pathologist labeled cells on the training images and reviewed the live classification, adjusting the labels until the result was satisfactory. No held-out training and validation split was used; classifier accuracy was instead assessed afterwards against an independent, blinded ground truth (S3.4). To keep tumor-cell identification independent of p53 staining intensity, all DAB-derived features were excluded from the feature set, which comprised hematoxylin optical density, nuclear and cell morphometry, and Haralick texture features. The InstanSeg classifier was trained with 539 labeled cells and 83 features; the legacy classifier used 95 features. Complete feature-importance rankings for both classifiers are given in Supplementary Table S3.

Listing S3. Application of a saved classifier (a separate execution from training).

```
runObjectClassifier("p53_Legacy_v01");    // legacy detections
runObjectClassifier("p53_InstanSeg_v01"); // InstanSeg detections
```

S3.4 Classifier validation

Classification accuracy was validated independently of training, in a dedicated QuPath project. Cells were sampled at random with a fixed seed from cores not used for training (Validation GT Sampler) (Supplementary Fig. S2C, S2D). Two pathologists labeled the identical sampled cells while blinded to the classifier output, and their labels were matched by cell object identifier to build a consensus ground truth, concordant labels being accepted automatically and discordant labels adjudicated (Consensus GT Merger) (Supplementary Fig. S2E). The classifier prediction was then matched to the consensus ground truth cell by cell, and the confusion matrix, accuracy, sensitivity, specificity, predictive values, balanced accuracy, F1, Cohen's kappa and the Matthews correlation coefficient were computed, with 95% confidence intervals for proportions by the Wilson score method (Object Classifier Validator) (Supplementary Fig. S2F). Results are given in Supplementary Table S2.

S4. DAB intensity-threshold calibration and the digital H-score

To grade nuclear p53 intensity as negative, 1+, 2+ or 3+, thresholds on the mean nuclear DAB optical density were calibrated from the pathologist's cell-level reading in a dedicated calibration project. Calibration was performed after the classifier had been applied, so that only cells classified as tumor cells were used.

Reference labeling (Cell Grade Labeler) (Supplementary Fig. S2G). One pathologist (S.W.C.) selected representative tumor cells of each intensity grade, at least 50 per grade and typically 50 to 70, by marking them with unclassified brush or point annotations; cells whose centroid fell inside the annotation were labeled with the specified grade. The target class was restricted to Tumor cells.

Threshold derivation (IHC Threshold Calibrator) (Supplementary Fig. S2H). From the mean nuclear DAB optical density of the labeled reference cells (legacy: Nucleus: DAB OD mean; InstanSeg: DAB Mean), the threshold between two adjacent grades was placed at the midpoint between the mean optical densities of those two grades, giving t_1 (negative | 1+), t_2 (1+ | 2+) and t_3 (2+ | 3+). Calibration precision was assessed by bootstrap 95% confidence intervals, and label quality was checked for monotonicity ($t_1 < t_2 < t_3$), adjacent-grade AUC, and quadratic-weighted kappa.

Repeated calibration. The same reader repeated the labeling and threshold derivation independently three times, on different tissue-microarray images, yielding three threshold sets (a, b, c). For each detector, the component-wise mean of the three sets was applied as the final threshold set and was used to compute the H-scores in all analyses reported here; the three individual sets were retained for sensitivity analysis. The three threshold sets yielded identical three-tier assignments in 85.4% (InstanSeg) and 84.3% (Legacy) of cores (Fleiss' κ 0.82 and 0.80; pairwise quadratic-weighted κ 0.81–0.96 and 0.78–0.90, respectively). Threshold values and their reproducibility are given in Supplementary Table S4.

Listing S4. Application of the final thresholds to all cores. The two pipelines use different measurements and different application functions; thresholds are not interchangeable between them.

```
// Legacy
```

```
setCellIntensityClassifications("Nucleus: DAB OD mean", 0.110, 0.310, 0.500);
```

```
// InstanSeg
```

```
setDetectionIntensityClassifications("DAB: Mean", 0.117, 0.337, 0.527);
```

// Digital H-score. With the final thresholds applied, every tumor cell was graded 0 / 1+ / 2+ / 3+ and the per-case digital H-score computed as $1 \times (\%1+) + 2 \times (\%2+) + 3 \times (\%3+)$, range 0 to 300.

Supplementary Software

All tools are in-house Groovy scripts written for QuPath v0.7.0 and are provided with the article. Their interfaces are shown in Supplementary Fig. S2.

Stage	Tool
Three-tier p53 pattern interpretation	Manual_p53_Pattern_Scorer_v1.0.groovy
Manual H-score	Manual_HScore_Scorer_v1.0.groovy
Validation cell sampling	Validation_GT_Sampler_v1.0.groovy
Validation consensus	Consensus_GT_Merger_v0.1.groovy
Classifier validation	Object_Classifier_Validator_v1.0.groovy
Reference cell grading	Cell_Grade_Labeler_v1.0.groovy
Threshold calibration	IHC_Threshold_Calibrator_v1.0.groovy