EXPAND: User Guide

Overview

EXPAND (**EX**plainable **P**athologist **A**ligned **N**uclear **D**iscriminator) is a fully automated, interpretable AI pipeline for:

- **Breast cancer subtype classification** (HER2+, HR+, TNBC, and 4-class TPBC/HER2+/HR+/TNBC)
- **Survival risk stratification**

  It operates on **12 nuclear pathologist-interpretable features (NPIFs)** extracted from H&E whole-slide images (WSIs) or from pre-computed feature tables.

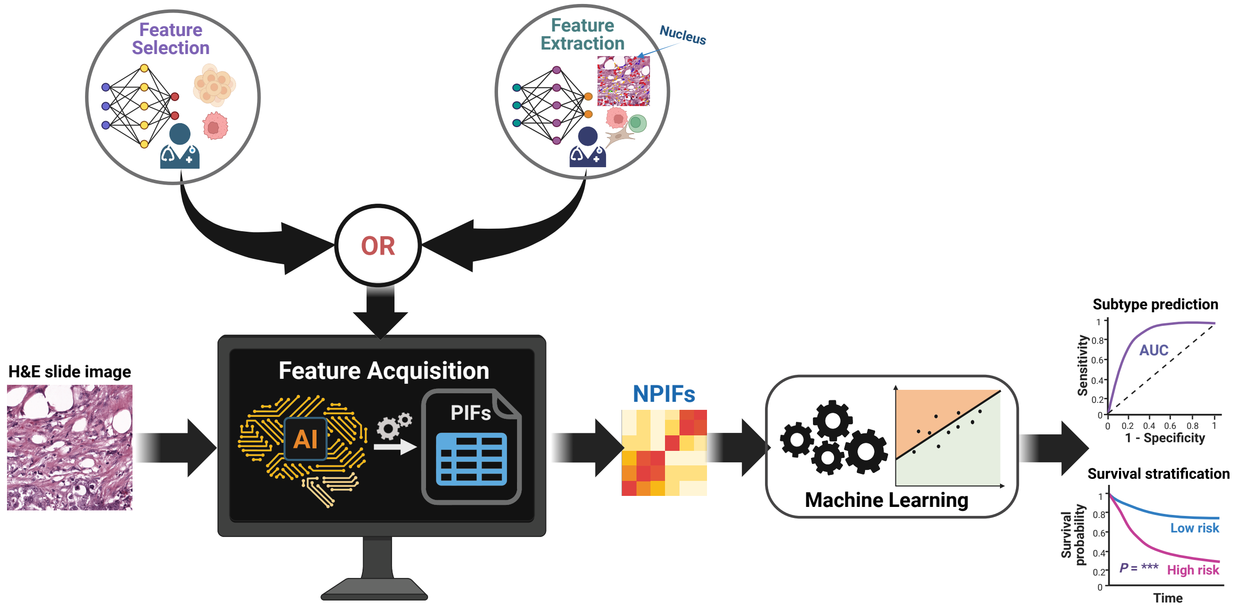

Figure 1: Overview of the methodology

Overview of analysis pipeline, **EXPAND**. *First*, the nuclear pathologist-interpretable features (NPIFs) from hematoxylin and eosin (H&E)-stained whole-slide images are obtained through the feature acquisition module (either two-stage selection from already-extracted HIFs and nuHIFs, or direct extraction from nuclear morphology *via* segmentation); *Second*, we train machine learning pipelines with NPIFs to predict tumor subtypes or patient survival and evaluate performance by using the area under the receiver operating characteristics curve (AUC) metric or Kaplan-Meier analysis and Log-rank test, respectively.

1. Subtype prediction from PathAI-derived feature sets

This subsection maps PathAI-derived features (HIFs, nuHIFs, PIFs, NPIFs) to breast cancer subtypes and trains binary classifiers for each subtype.

PIFs and NPIFs were developed with expert pathologist input using the Nottingham Histologic Grade (NHG) criteria. NPIFs are a compact set of 12 nuclear pleomorphism features, the most interpretable to pathologists, chosen to maximize clarity and generalizability.

1.1 Map features to subtypes

Purpose: Merge feature tables with HER2/ER/PR status to assign:

- **3-class:** HER2+, HR+, TNBC
- **4-class:** TPBC, HER2+, HR+, TNBC

Scripts:

- HIFs – `1_01_01_mapped_tcga_biomarker_status_to_original_hifs_with_comments.py / .ipynb`
- nuHIFs – `2_01_01_PathAI_Metadata_Original_nuHIFs_And_TCGA_BiomarkerStatus.py / .ipynb`
- PIFs –

`3_01_01_PathAI_Metadata_Original_PIFs_And_TCGA_BiomarkerStatus.py / .ipynb`

Output: <feature_set>_with_subtypes.csv containing subtype labels and features.

1.2 Train binary classifiers

Purpose: Train **L1-regularized logistic regression** (one-vs-all) with nested cross-validation.

Scripts:

- HIFs –

  `1_01_04_103_04_103_BRCA_Clinical_Subtype_Prediction_Using_All_PathAI_HIFs_Binary_Subtype_Classification.py / .ipynb`

- nuHIFs –

  `2_01_04_103_04_103_BRCA_Clinical_Subtype_Prediction_Using_All_PathAI_nuHIFs_Binary_Subtype_Classification.py / .ipynb`

- PIFs –

  `3_01_04_103_04_103_BRCA_Clinical_Subtype_Prediction_Using_All_PathAI_PIFs_Binary_Subtype_Classification.py / .ipynb`

- NPIFs –

  `3_01_04_103_04_103_01_BRCA_Clinical_Subtype_Prediction_Using_All_PathAI_NPIFs_Binary_Subtype_Classification.py / .ipynb`

Outputs:

- Metrics CSV (AUC, accuracy)
- ROC curves per subtype

- Feature coefficient CSV

1A. Direct Feature Extraction from H&E WSIs with ResNet50

1A.1 What this does

Provides a **baseline comparison** against EXPAND by extracting **non-interpretable deep features** directly from H&E slides using a ResNet50 model. These features are pooled at the slide level and used to train subtype classifiers. Unlike NPIFs or PathAI-derived features, this step bypasses segmentation and morphology computation, relying solely on black-box CNN embeddings.

1A.2 Scripts

- `3_01_01_02_TCGA_BRCASubtypes_to_DirectHnE_Features_Resnet50.py / .ipynb`
  - Already extracted ResNet50 embeddings from TCGA-BRCA slide and maps them to BRCA subtypes (HER2, ER, PR).
- `3_01_04_103_04_103_02_BRCA_Clinical_Subtype_Prediction_Using_All_Direct_Features_Binary_Subtype_Classification.py / .ipynb`
  - Uses the extracted ResNet50 features to train binary classifiers for each subtype with nested cross-validation.

1A.3 Inputs

- Raw H&E whole-slide images (WSIs) or tiles at 20× magnification.
- BRCA biomarker status metadata (HER2, ER, PR).

1A.4 Outputs

- **Feature embeddings** from ResNet50 per tile, aggregated to slide level.
- **Subtype classification results** (ROC curves, AUC scores, confusion matrices, accuracy).
- Saved models for reproducibility and comparison with interpretable pipelines.

2. Tile generation from H&E WSIs (TCGA-BRCA)

2.1 What this does

- Divides each WSI into non-overlapping 512×512 tiles at 20× magnification.
- Applies skip blank/background and stain normalization.
- Writes tiles to disk and records a simple tile manifest (slide ID, tile Number).

2.2 Scripts

- `1_01_get_tiles_from_slide.py` — core tiling for a single slide.
- `1_11_jobs_to_get_tiles.py` — batch/HPC launcher to tile a whole cohort.
- Utilities: `utils_preprocessing.py` (tissue masking, I/O), `utils_color_norm.py` (optional stain normalization).
- Notes/usage: how_to_run.md.

2.3 Inputs

- Directory of TCGA-BRCA WSIs (e.g., .svs).

- Output root folder for tiles.
- Dataset label (e.g., TCGA-BRCA-FFPE, CPTAC-BRCA, POST-NAT-BRCA) and desired magnification (20×).

2.4 Key parameters (typical)

- tile_size=512, stride=512 (non-overlapping).
- mag=20 (or nearest level to 20× in the WSI).
- Optional: tissue threshold / min tissue area, enable stain normalization, output format (png/jpg).

2.5 Outputs

- Folder per slide containing PNG/JPG tiles named by tile coordinates.
- Tile manifest CSV with slide ID, tile number.

3. Tile-level nucleus segmentation with Hover-Net (TCGA-BRCA tiles)

Scope: These scripts run Hover-Net inference on the pre-prepared TCGA-BRCA tiles from step 2 to segment all nuclei per tile.

3.1 What this does

- Reads pre-generated H&E tile images for TCGA-BRCA.
- Uses the official **Hover-Net** implementation to segment and classify nuclei in each tile.

For environment setup and package dependencies, refer to the original Hover-Net repository: https://github.com/vqdang/hover_net

- Saves per-tile predictions, including instance masks and per-nucleus classification data.

3.2 Scripts

- `2_01_22_ExtractMorphologicalFeaturesFromHnE.py / .ipynb`
- `2_01_100_01_JobSubmissionCode.py / .ipynb`

  (Batch/SLURM launcher to process many tiles/slides.)

3.3 Inputs / Outputs

Input

- Directory of tile images (e.g., PNG/JPG) organized by slide or sample.

Outputs (per tile)

- Instance masks / overlay images with nuclei boundaries.
- Prediction files (JSON/CSV as configured) containing nucleus instances, polygons, and predicted class (cancer, immune, fibroblast, epithelial, dead).
- Logs for QC.

3.4 Dependencies & reference

Uses **Hover-Net** (Graham et al., 2019, *Med Image Anal.*), official GitHub source code at https://github.com/vqdang/hover_net (pin the commit/model weights used in the manuscript for reproducible.

4. TCGA-BRCA: Morphology Computation from Hover-Net Predictions

4.1 What this does

Reads per-tile Hover-Net outputs (instance polygons + nucleus type) for TCGA-BRCA. Computes morphology for each individual tumor nucleus (Area, Perimeter, Major/Minor Axis, Eccentricity, Circularity, etc.). Writes per-nucleus records at the tile level.

4.2 Scripts

```
2_02_03_MorphologyCalculation_All_Slides.py / .ipynb
2_02_13_Job_Submission_MorphologyCalculation_All_Slides.py / .ipynb
```

4.3 Inputs

Hover-Net prediction files (JSON per tile) for TCGA-BRCA.

Tile → slide mapping via folder structure or a manifest.

4.4 Outputs

Per-nucleus morphology CSVs per tile (one row = one cancer nucleus in that tile).

5. NPIF Calculation from Hover-Net Outputs

5.1 What this does

Computes **12 nuclear pathologist-interpretable features (NPIFs)** at the slide level from predicted cancer nuclei in per-nucleus morphology files (Step 4). Two approaches are used:

- **All tiles** – includes every tile containing cancer nuclei.
- **Top 25% cancer-enriched tiles** – ranks tiles within each slide by cancer nuclei count and keeps only the top quartile.

For each approach, cancer nuclei are aggregated per slide to produce one NPIF row with summary statistics for the 12 features.

5.2 Scripts

All tiles – `2_03_01_01_NPIFs_Calculation_HoverNet_V0.py / .ipynb`

Top 25% tiles – `2_03_01_01_NPIFs_Calculation_HoverNet_V1.py / .ipynb`

5.3 Inputs

Per-nucleus morphology CSVs from Step 3, organized by slide folder.

5.4 Output

<dataset>_HoverNet_NPIFs_All_Tiles.csv or <dataset>_HoverNet_NPIFs_25Q.csv, sample_id and 12 NPIF columns (summary statistics per slide)

6. Mapping NPIFs to BRCA Biomarker Status

6.1 What this does

Reads NPIF feature tables generated from Hover-Net outputs (either all tiles or the top 25% cancer-enriched tiles).

Merges NPIF features with TCGA-BRCA biomarker status metadata (HER2, ER, PR).

Produces a mapped dataset linking each sample's NPIFs to its biomarker status, ready for downstream subtype classification.

6.2 Scripts

`3_01_01_02_Mapped_Original_Value_Hovernet_NPIFs_to_BRCA_Subtypes.py / .ipynb` → Uses NPIFs from all tiles.

`3_01_01_06_Mapped_Original_Value_Hovernet_NPIFs_to_BRCA_Subtypes_Filtered_Tiles_Top25Q.py / .ipynb` → Uses NPIFs from top 25% cancer-enriched tiles.

6.3 Inputs

NPIF CSV files (from Step 4, calculated using Hover-Net outputs).

BRCA biomarker status file (provided in TCGA_BRCA_Metadata).

6.4 Outputs

Mapped NPIF + biomarker status CSV file for use in classification scripts.

7. BRCA Clinical Subtype Prediction Using HoverNet-Predicted NPIFs

7.1 What this does

Uses NPIF feature tables mapped to BRCA biomarker status (HER2, ER, PR) to train binary classifiers for each subtype (from Step 6).

Implements nested cross-validation with Logistic Regression (L1 penalty) for robust feature selection and prediction.

All Tiles version provides baseline performance. Top 25% Tiles version (cancer-enriched) is the final model used in the manuscript.

7.2 Scripts

`4_01_04_103_04_101_BRCA_Clinical_Subtype_Prediction_Using_All_HoverNet_Predicted_NPIFs_All_Tiles_Using_Lasso_Binary_Subtype_Classification.py / .ipynb` → Uses NPIFs from all tiles.

`4_01_04_103_04_103_BRCA_Clinical_Subtype_Prediction_Using_All_HoverNet_Predicted_NPIFs_Filtered_Tiles_Top25Q_Binary_Subtype_Classification.py / .ipynb` → Uses NPIFs from top 25% cancer-enriched tiles (final model).

7.3 Inputs

Mapped NPIF + biomarker status CSV files (from Step 4).

7.4 Outputs

Trained Lasso-regularized Logistic Regression models (saved for external validation).

ROC curves, AUC scores, confusion matrices, and accuracy metrics per subtype.

Saved scalers and feature selection details for reproducibility.

8. CPTAC-BRCA: Tile-level Nucleus Segmentation with Hover-Net

8.1 What this does

Reads pre-generated H&E tiles for CPTAC-BRCA.

Runs Hover-Net inference to segment & classify nuclei per tile (cancer, immune, fibroblast, epithelial, dead).

Saves per-tile predictions for downstream NPIF calculation.

For environment/dependencies, use the official Hover-Net repo: https://github.com/vqdang/hover_net (pin the commit/weights used in the manuscript).

8.2 Scripts

`2_01_22_02_Test_CPTAC_Dataset_ExtractMorphologicalFeaturesFromHnE.py / .ipynb`

`2_01_100_02_01_JobSubmissionCode.py / .ipynb` (batch/SLURM launcher)

8.3 Inputs

Directory of CPTAC H&E tile images (PNG/JPG), organized by slide/sample.

8.4 Outputs

Instance masks / overlay images with nuclei boundaries (per tile).

Prediction JSON/CSV (per tile) with nucleus polygons and predicted class.

Logs for QC.

9. CPTAC-BRCA: Morphology Computation from Hover-Net Predictions

9.1 What this does

Reads per-tile Hover-Net outputs (instance polygons + nucleus type). Computes morphology for each individual tumor nucleus (Area, Perimeter, Major/Minor Axis, Eccentricity, Circularity, etc.). Writes per-nucleus records at tile level

9.2 Scripts

`2_02_03_02_CPTAC_MorphologyCalculation_All_Slides.py / .ipynb`

`2_02_13_02_CPTAC_Job_Submission_MorphologyCalculation_All_Slides.py / .ipynb`

9.3 Inputs

Hover-Net prediction files (JSON per tile) for CPTAC-BRCA.

Tile → slide mapping via folder structure or a manifest.

9.4 Outputs

Per-nucleus morphology CSVs per tile (one row = one cancer nucleus in that tile).

10. CPTAC-BRCA: NPIF Computation (Top 25% Cancer-Enriched Tiles)

10.1 What this does

Loads per-nucleus morphology CSVs produced in Step 8 (tile/slide level). Ranks tiles per slide by cancer-nuclei abundance (count). Keeps the top 25% cancer-enriched tiles. From cancer nuclei only, computes slide-level NPIFs (12 features). Produces one NPIF row per slide.

10.2 Script

`2_03_02_05_CPTAC_BRCA_NPIFs_Calculation_HoverNetPrediction_Filtered_Tiles_Top25Q.py / .ipynb`

10.3 Inputs

Per-nucleus morphology files from Step 9.

10.4 Outputs

CPTAC_BRCA_NPIFs_Top25Q.csv (one row per slide with NPIFs)

11. CPTAC-BRCA: Map NPIFs to BRCA Biomarker Status (Top 25% Tiles)

11.1 What this does

Loads CPTAC NPIFs computed from the top 25% cancer-enriched tiles (Step 9). Merges with CPTAC BRCA biomarker status (HER2, ER, PR). Outputs a mapped table ready for subtype classification.

11.2 Script

```
3_01_01_07_CPTAC_Mapped_Original_Value_Hovernet_NPIFs_to_BRCA_Subtypes_Filtered_Tiles_Top25Q.py / .ipynb
```

11.3 Inputs

NPIF CSV (Top25%) from Step 8

CPTAC biomarker metadata with: sample_id, HER2_Status, ER_Status, PR_Status (values: Positive / Negative).

11.4 Outputs

CPTAC_BRCA_NPIFs_Top25Q_with_Status.csv containing:

sample_id, NPIF columns (12 features: mean/sd of Area, Perimeter, Major/Minor Axis, Eccentricity, Circularity), Biomarker columns: HER2_Status, ER_Status, PR_Status.

12. External Prediction on CPTAC-BRCA Using TCGA-Trained HoverNet-Predicted NPIF Models (Top 25% Tiles)

12.1 What this does

Applies the five outer-fold binary classifiers per subtype (HER2+, HR+, TNBC) trained on TCGA-BRCA NPIFs (from Step 6.4) to the CPTAC-BRCA dataset. Each fold’s saved feature subset and scaler is used to ensure exact preprocessing consistency. Produces per-fold and ensemble (mean probability) predictions for each subtype. Evaluates performance with ROC curves, AUC scores, confusion matrices, and accuracy metrics.

12.2 Scripts

`6_01_04_103_04_103_CPTAC_Prediction_Using_BRCA_Clinical_Subtype_Prediction_Using_HoverNet_Predicted_Model_All_NPIFs_Filtered_Tiles_Top25Q_Binary_Subtype_Classification.py / .ipynb`

→ Loads trained TCGA models and applies them to CPTAC NPIFs from the top 25% cancer-enriched tiles.

12.3 Inputs

CPTAC NPIFs (Top-25% tiles) mapped to BRCA biomarker status (HER2, ER, PR) from Step 9. Saved TCGA training outputs for each subtype (from Step 5.4):

12.4 Outputs

For each subtype (HER2+, HR+, TNBC), the outputs include ensemble prediction CSVs with the sample ID, true label, probability (averaged across the 5 fold-specific models), performance plots such as ROC curves with AUC for both per-fold and ensemble results, and confusion matrices with corresponding accuracy values; as well as saved metrics including accuracy, precision, recall, and F1-score for each subtype.

We executed an equivalent end-to-end pipeline for the **POST-NAT-BRCA** dataset, mirroring the methodology used for **CPTAC-BRCA**, and incorporating the following scripts:

Tile Extraction & HoVer-Net Inference:

```
2_01_100_02_POST_NAT_JobSubmissionCode.py / .ipynb
2_01_22_02_Test_POST_NAT_Dataset_ExtractMorphologicalFeaturesFromHnE.py / .ipynb
```

Morphology Feature Calculation:

```
2_02_03_02_POST_NAT_MorphologyCalculation_All_Slides.py / .ipynb
2_02_13_02_POST_NAT_Job_Submission_MorphologyCalculation_All_Slides.py / .ipynb
```

NPIF Calculation from Filtered Tiles (Top 25% cancer nuclei):

```
2_03_02_05_POST_NAT_BRCA_NPIFs_Calculation_HoverNetPrediction_Filtered_Tiles_Top25Q.py / .ipynb
```

Mapping NPIFs to BRCA Clinical Subtypes:

```
3_01_01_07_POST_NAT_Mapped_Original_Value_Hovernet_NPIFs_to_BRCA_Subtypes_Filtered_Tiles_Top25Q.py / .ipynb
```

Subtype Prediction using Lasso Models:

```
6_01_04_103_04_103_Lasso_POST_NAT_Prediction_Using_BRCA_Clinical_Subtype_Prediction_Using_HoverNet_Predicted_Model_All_NPIFs_Filtered_Tiles_Top25Q_Binary_Subtype_Classification
```

This pipeline included: Extracting and segmenting tiles from H&E slides using HoVer-Net. Computing NPIFs for tumor nuclei and selecting top 25% tiles based on cancer nuclei count. Mapping features to HER2, ER, and PR status to derive molecular subtype labels. Applying

L1-penalized logistic regression models (trained on TCGA_BRCA_FFPE NPIFs) for binary and multiclass subtype classification.

Survival Analysis with EXPAND Features

1.1 What this does

- Evaluates the ability of EXPAND NPIFs to predict patient survival in TCGA-BRCA, compared against PIFs, PathAI-derived HIFs and nuHIFs.
- For each subtype (HER2+, HR+, TNBC), builds a multivariate Cox regression model using:
    - Selected feature set (NPIFs, PIFs, HIFs, or nuHIFs) after collinearity filtering.
    - Age as a confounder.
- Derives subtype-specific risk scores and stratifies patients into high- vs low-risk groups using a fixed 0.5 threshold on quantile-normalized scores ([10%, 90%] interval).
- Performs Kaplan-Meier survival analysis for Overall Survival (OS) and Progression-Free Survival (PFS).

1.2 Scripts

Feature Mapping to Survival:

- `5_01_01_mapped_hovernet_npifs_to_tcga_survival.py / .ipynb`

  `5_01_02_mapped_pathai_hifs_to_tcga_survival.py / .ipynb`

  `5_01_03_mapped_pathai_nuhifs_to_tcga_survival.py / .ipynb`

  `5_01_04_mapped_pathai_pifs_to_tcga_survival.py / .ipynb`

CoxPH + Cross-Validation Models (OS):

- `6_01_01_all_npifs_OS_analysis_with_age_cv.py / .ipynb`
- `6_01_02_01_all_original_pathai_hifs_OS_analysis_with_age_cv_fixed_threshold.py / .ipynb`

  `6_01_03_01_all_original_pathai_nuhifs_OS_analysis_with_age_cv_fixed_threshold.py / .ipynb`

  `6_01_04_01_all_original_pathai_pifs_OS_analysis_with_age_cv_fixed_threshold.py / .ipynb`

1.3 Inputs

- Survival metadata: OS and PFS times + event status from TCGA-BRCA.
- Feature tables: NPIFs (HoverNet), HIFs, nuHIFs, PIFs (from previous steps).
- Clinical covariates: Patient age.

1.4 Outputs

- Subtype-specific CoxPH models (per feature set).
- Kaplan-Meier plots stratifying high- vs low-risk patients.
- Performance comparison across NPIFs, PIFs, HIFs, and nuHIFs.