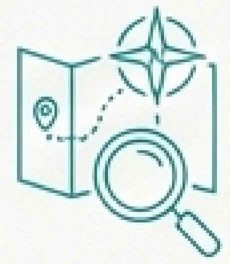
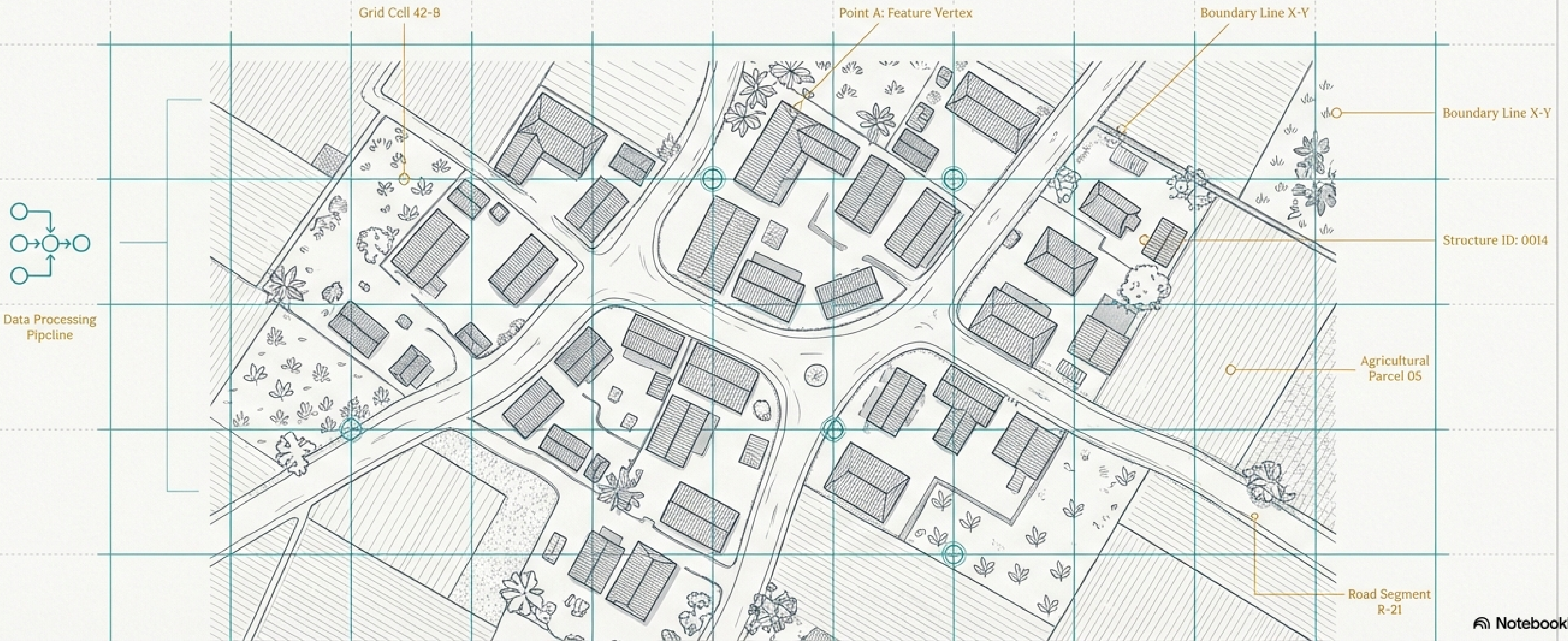
# The Digital Cartographer’s Workflow

A Technical and Conceptual Guide to Feature Extraction for the SVAMITVA Scheme



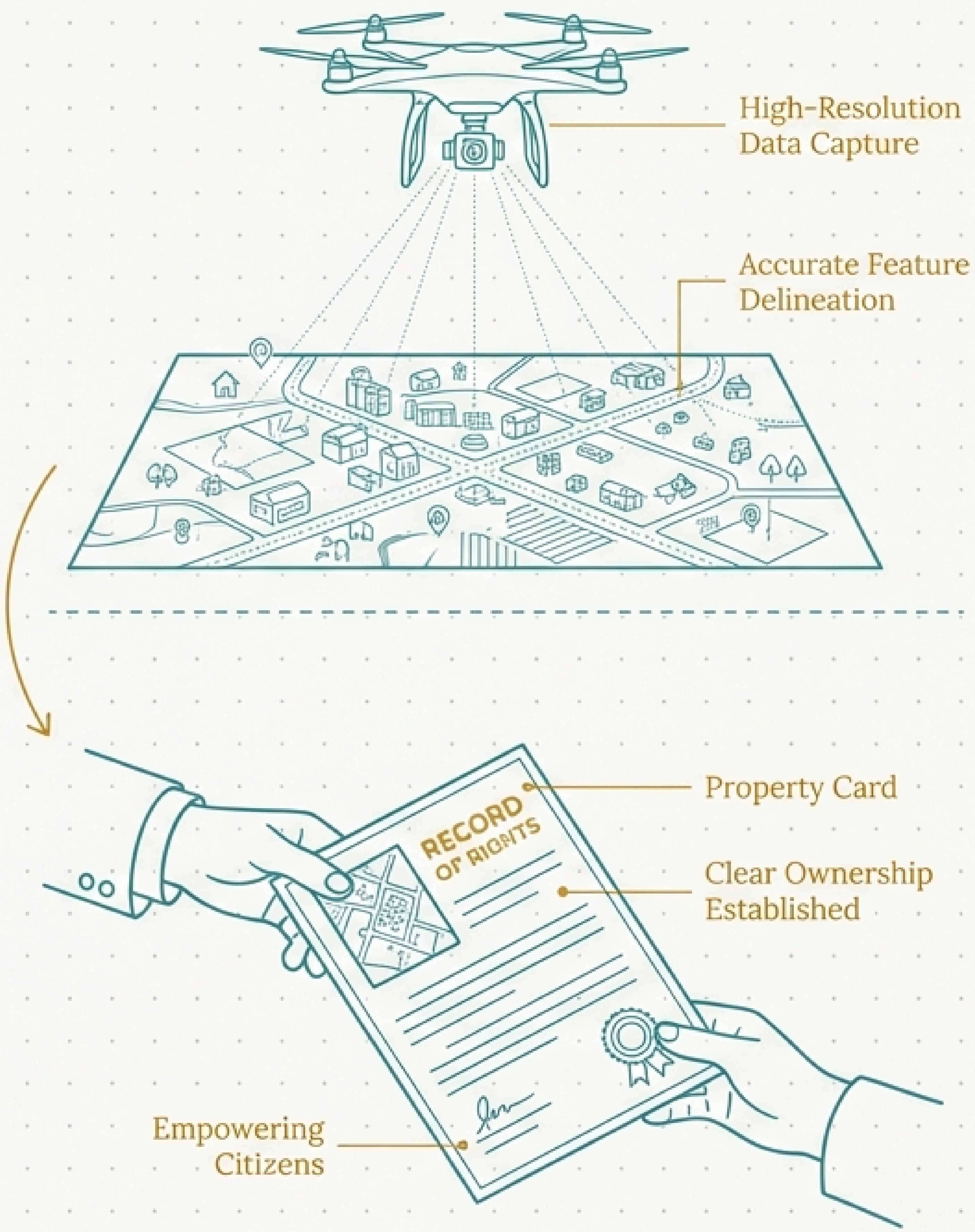



# The Mission: Defining Digital Boundaries for a Nation

**The SVAMITVA Scheme:** A reformative initiative to establish clear ownership of property in rural inhabited areas.

**The Core Mandate:** To create a high-resolution digital map and a record of rights for every village.

**The Foundation:** This process relies on accurately identifying and delineating features—buildings, roads, infrastructure—from high-resolution drone imagery.

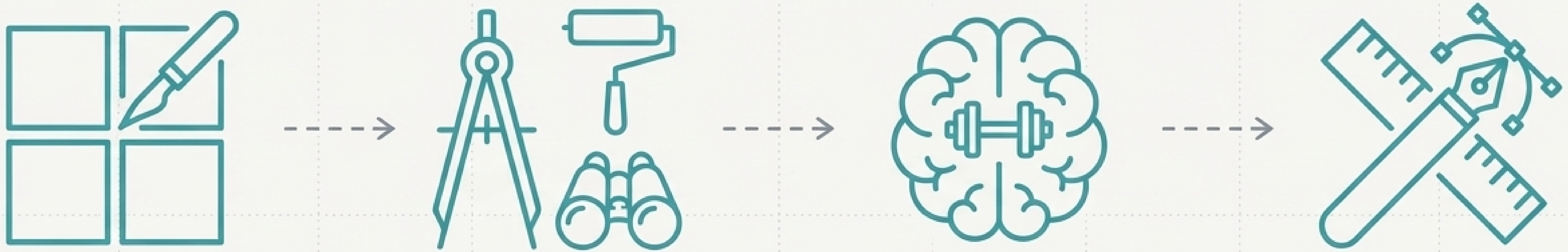


# The Challenge: From a Single Image to Millions of Data Points

A single high-resolution drone orthophoto is a vast, unstructured dataset. It contains millions of pixels but no inherent meaning. The objective is to build an intelligent pipeline that can automatically:

1. **Identify** every relevant feature with high precision.
2. **Classify** them correctly (e.g., building, road, well).
3. **Vectorise** them into clean, GIS-ready digital formats.

# Our Approach: A Four-Stage Automated Workflow



**1**

### Stage 1: Prepare the Canvas

Breaking down the vast map into manageable sections.

**2**

### Stage 2: Deploy the Specialists

Using multiple specialised AI models for different tasks.

**3**

### Stage 3: Conduct Rigorous Training

Teaching the AI to learn, not just memorise.

**4**

### Stage 4: Craft the Final Map

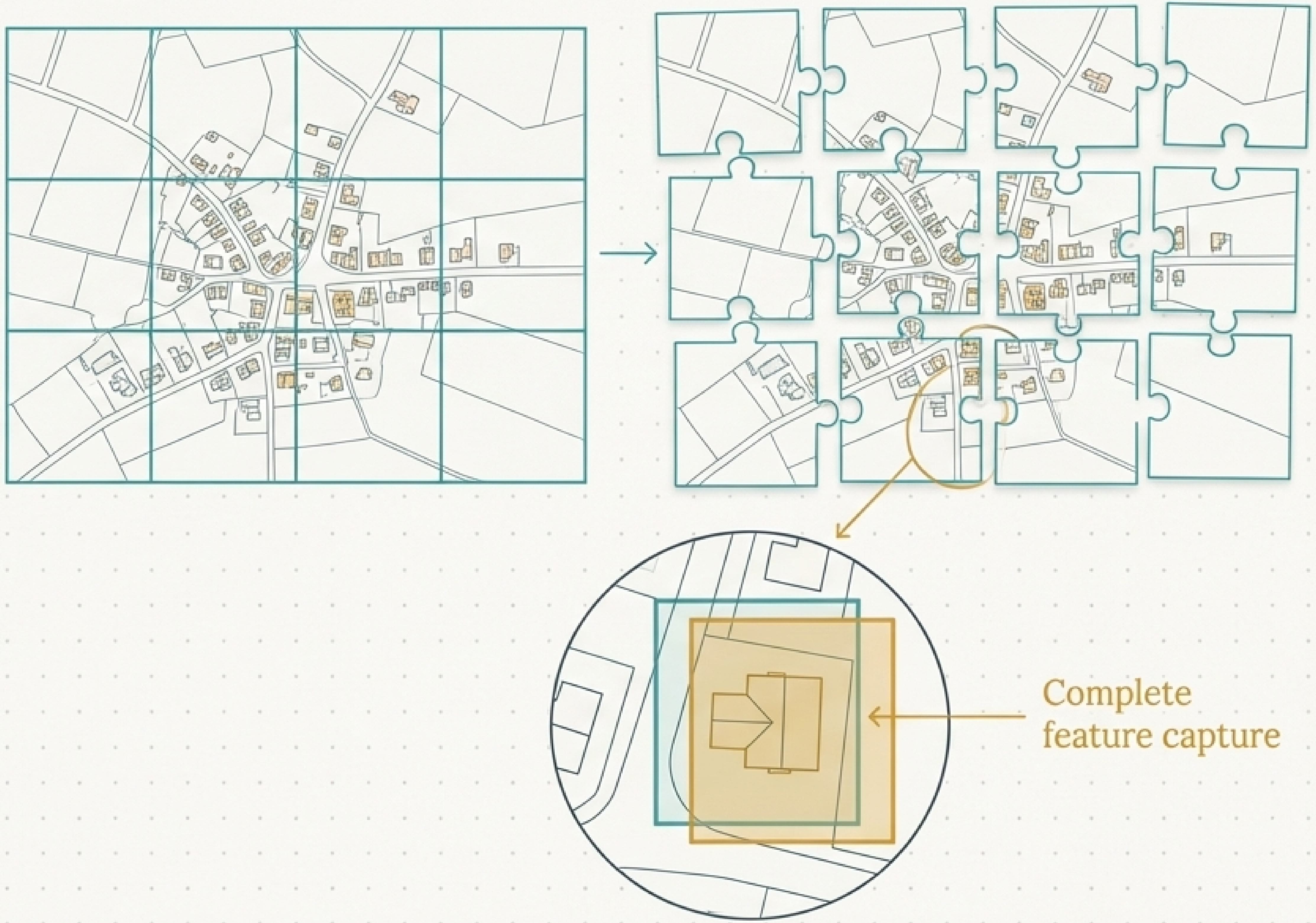
Assembling and refining the results into a perfect final product.

# Stage 1: Preparing the Canvas with with the Jigsaw Puzzle Approach

**The Problem Analogy:** A high-resolution drone map is like a giant wall poster. A computer cannot process the entire image at once; its memory (GPU) is insufficient.

**The Solution Analogy:** We cut the poster into small, manageable squares, like jigsaw puzzle pieces (tiles). The computer can easily analyse each piece individually.

**The Critical Detail (Overlap):** If a house is cut in half at the edge of a piece, the AI won't recognise it. By overlapping the cuts, we ensure every single feature appears whole in at least one puzzle piece.




NotebookLM

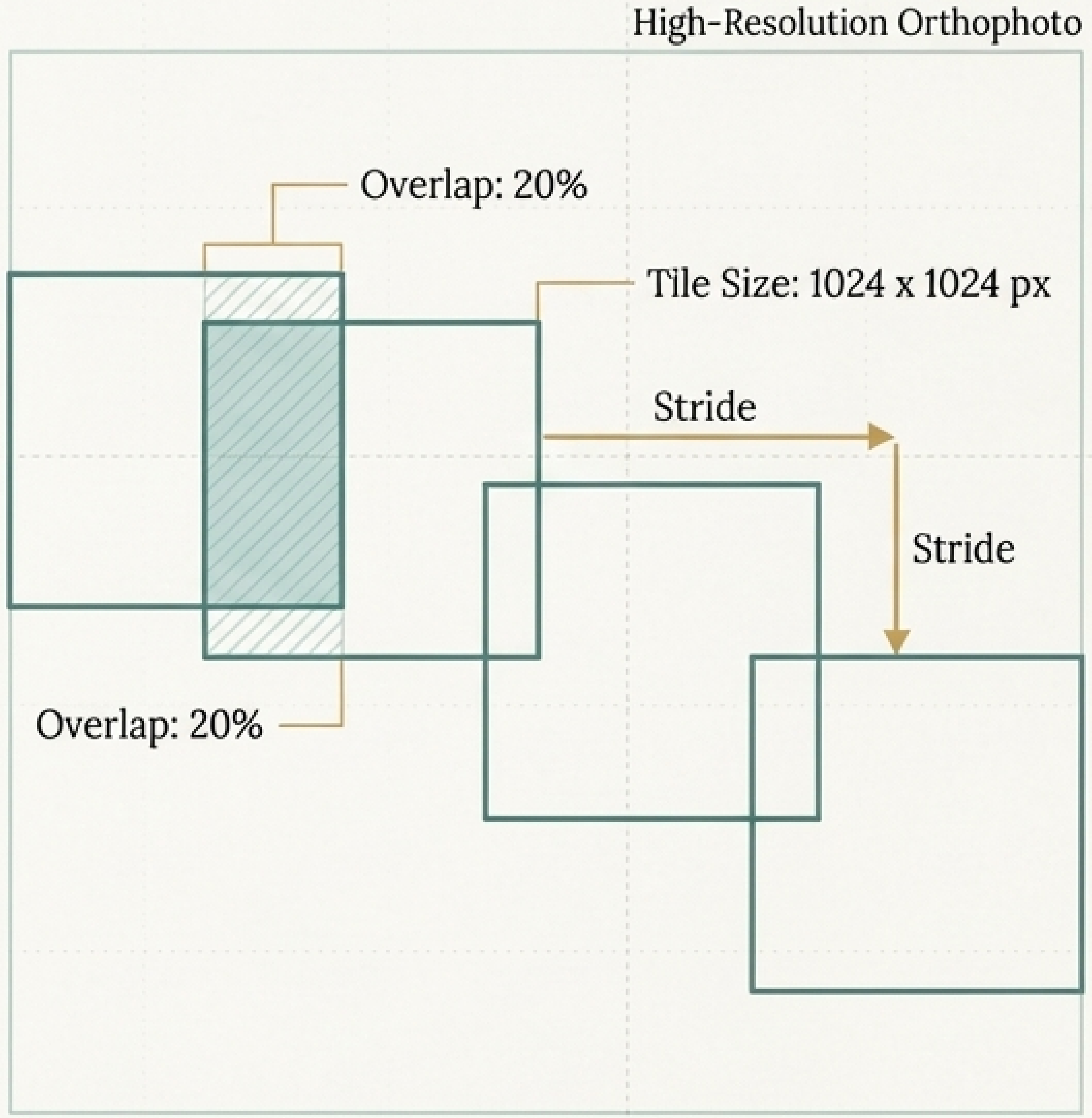
# Technical Details: Input Handling via Sliding Window Tiling

**Core Technique**: Sliding Window / Tiling Algorithm

**Objective**: Dimensionality reduction for GPU memory constraints and localised processing.

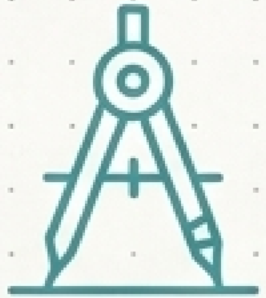
**Key Parameters**:

- **Input Dimensions**: High-Resolution Orthophotos (e.g., > 10,000 x 10,000 pixels).
- **Tile Size**: 512 x 512 or 1024 x 1024 pixels.
- **Stride/Overlap**: 15-20% overlap to **mitigate edge artefacts and ensure complete feature capture.**



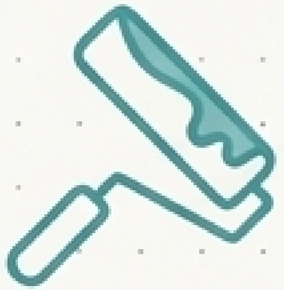
# Stage 2: Deploying a Team of Three AI Specialists

One single AI model cannot perform all feature extraction tasks perfectly.
We deploy a team of three highly specialised models, each with a distinct skill.



## The Architect

Carefully outlines each individual building and identifies its roof type. (Task: Instance Segmentation).

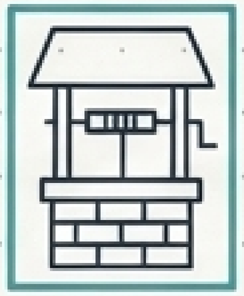


## The Painter

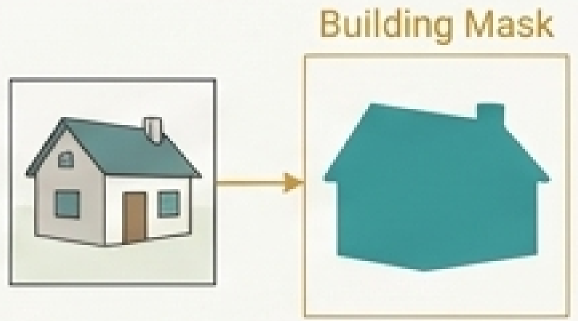
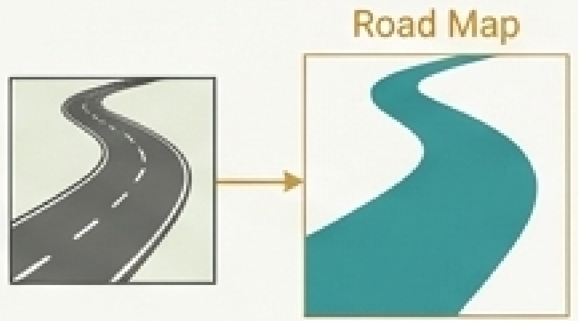
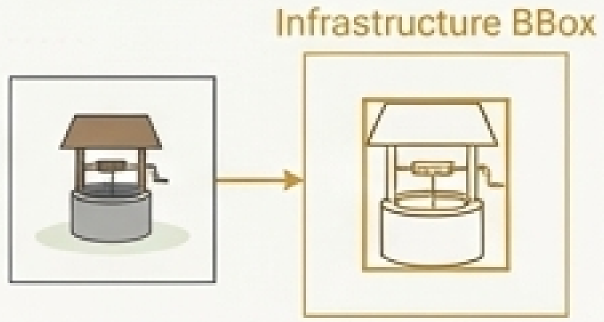
Colours in large, continuous areas like roads and water bodies. It cares about coverage, not counting individual items. (Task: Semantic Segmentation).

## The Scout

Quickly spots and flags small, distinct objects like transformers or wells. (Task: Object Detection).



# Technical Details: Multi-Model Pipeline Architectures

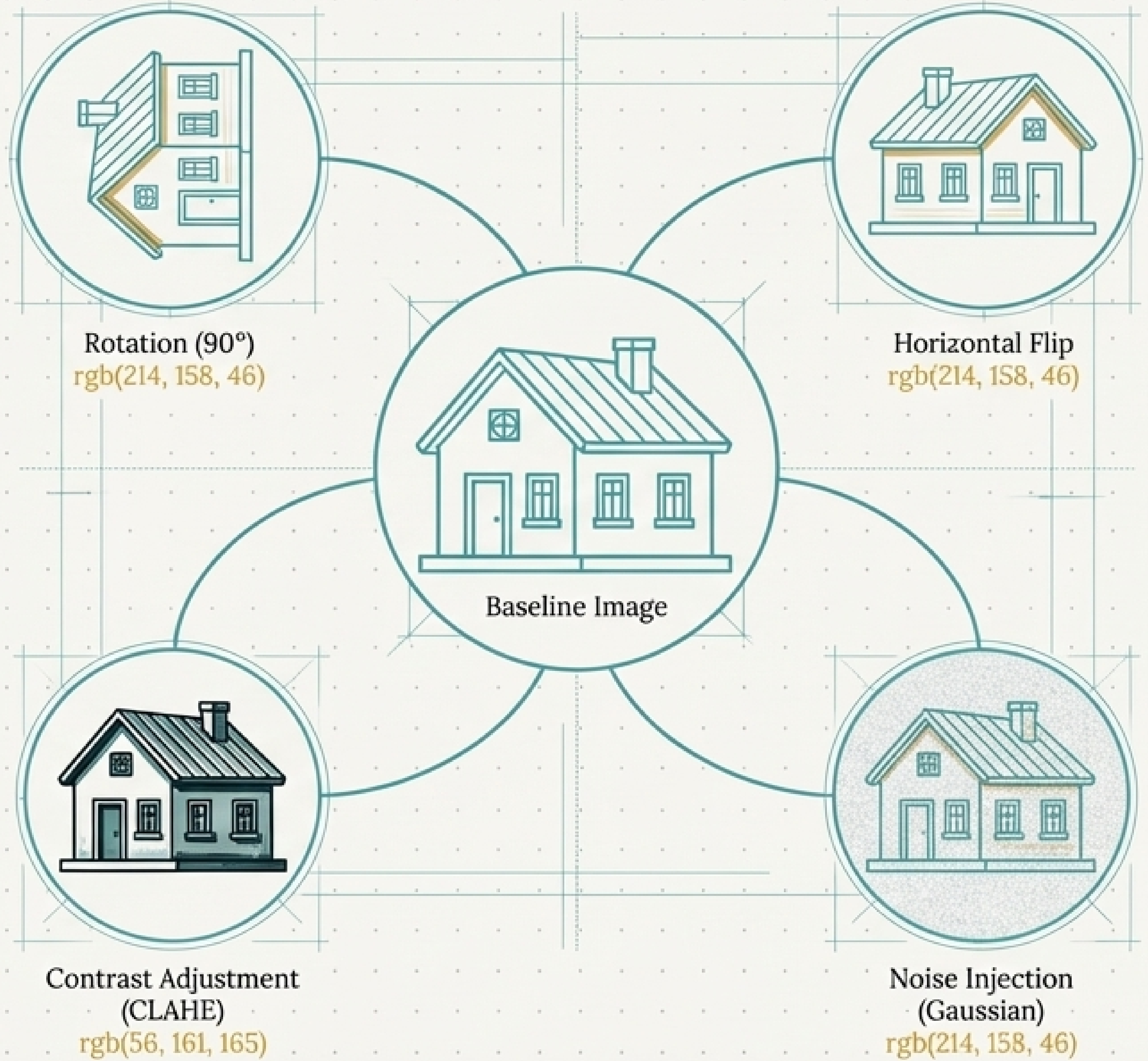
| Task | Architecture | Output |
|---|---|---|
| Task A: Instance Segmentation (Buildings) | **Mask R-CNN** or **Cascade R-CNN** | Binary Masks for each building instance + Classification Head for roof type (e.g., RCC, Tin, Tile).   |
| Task B: Semantic Segmentation (Roads/Water) | **DeepLabV3+** or **U-Net** with a ResNet backbone. | Pixel-wise classification maps.   |
| Task C: Object Detection (Infrastructure) | **YOLOv8** (You Only Look Once). | Bounding Boxes for small features like transformers and wells.   |

# Stage 3: Forging Expertise Through Rigorous Training

**The Challenge:** With a limited dataset (e.g., only 10 villages), the AI might simply memorise the specific houses it sees, failing to learn the general concept of a 'house'.

**The Solution - 'Trick Questions' (Augmentation):** We artificially alter the training images. We show the AI houses that are rotated, flipped, in different lighting, or with digital 'noise'. This forces it to learn the fundamental features of a house, making it smarter and more adaptable.

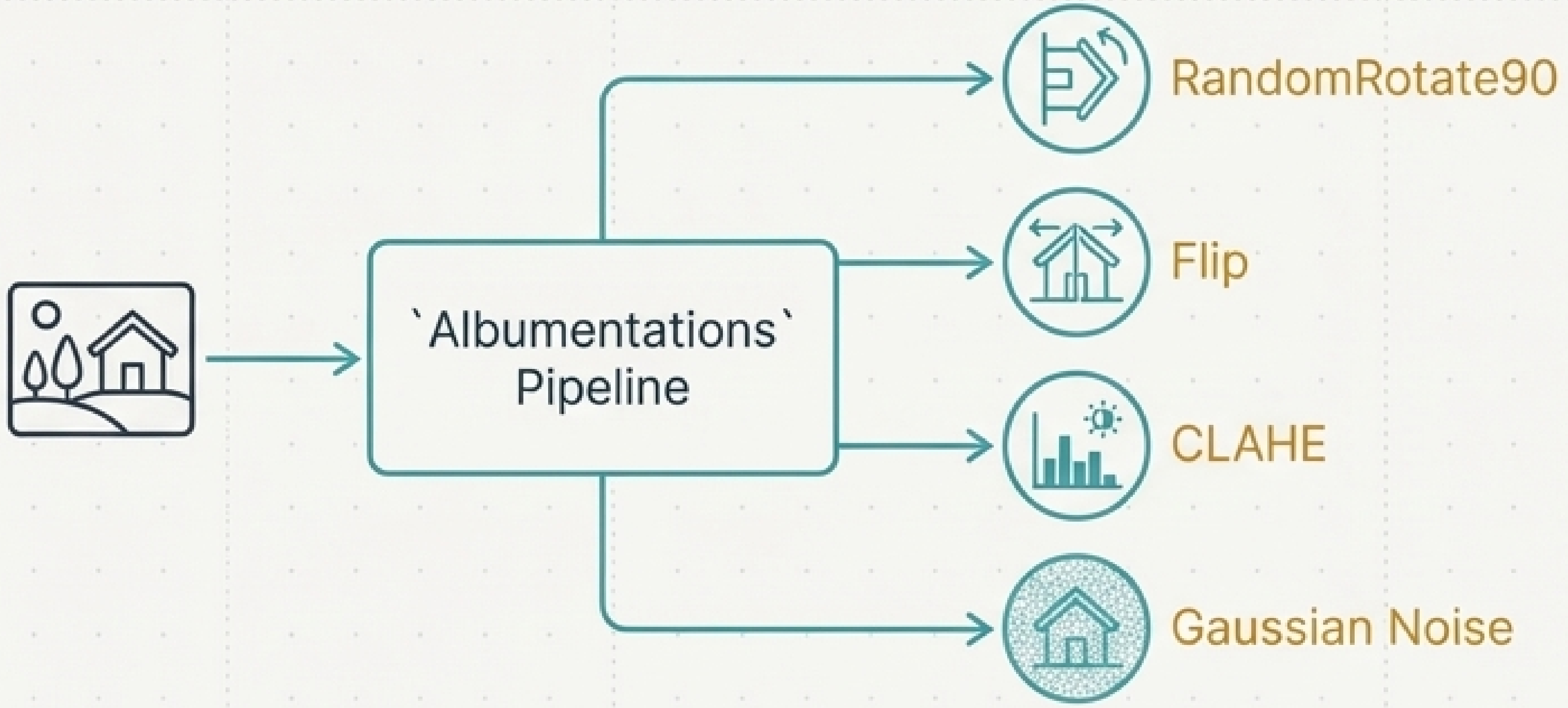
***The Priority System (Loss Optimisation):** We explicitly tell the AI: 'Finding a small, rare feature like a well is far more important than correctly identifying another patch of grass.' This focuses its learning on what matters most.



# Technical Details: Data Augmentation & Loss Functions

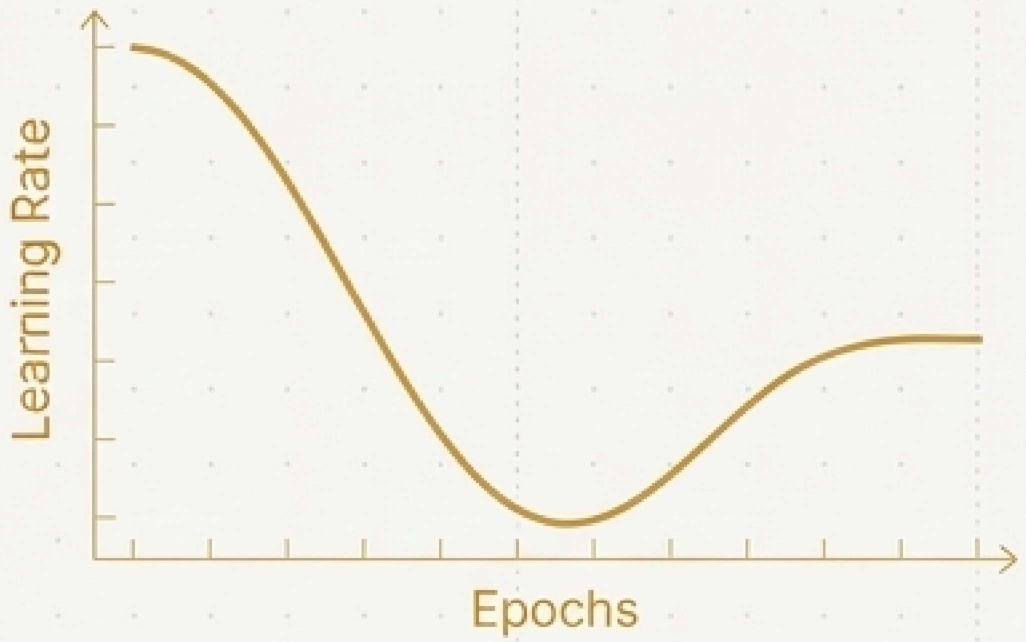
## Data Augmentation Pipeline

*Library*: Albumentations
*Techniques Employed*: RandomRotate90, Flip, CLAHE (Contrast Limited Adaptive Histogram Equalization), Gaussian Noise.




## Loss Optimisation

**Function:** Use of **Dice Loss** or **IoU Loss** to effectively handle the severe class imbalance between small foreground features (buildings) and the vast background.
**Optimizer: AdamW Optimizer.**
**Scheduler: Cosine Annealing** learning rate scheduler for stable convergence.



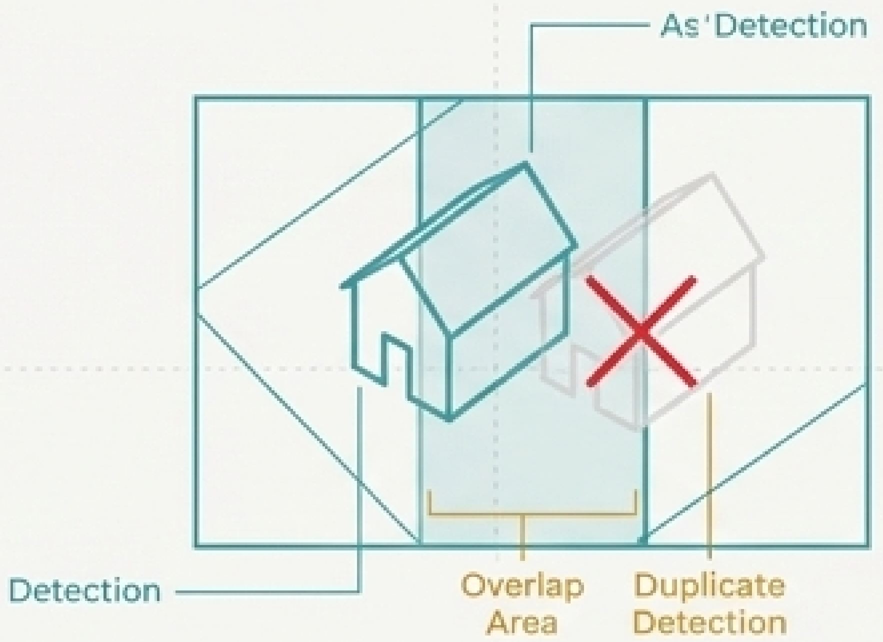
# Stage 4 : Crafting the Final Map Through Assembly and Refinement

Part 1

**Gluing the Pieces (Stitching)**

Lora Regular

We take all the individually analysed puzzle pieces (tiles) and stitch them back together to reconstruct the full, large-scale map with all the AI's predictions.

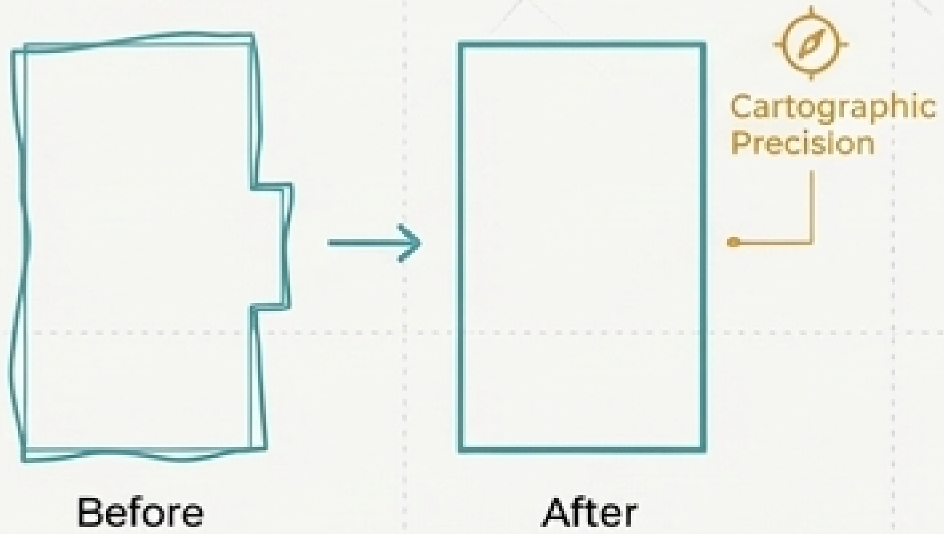



Part 2

**Removing the Doubles (NMS)**

Lora Regular

Because the tiles overlapped, the AI may have identified the same house twice. This step intelligently finds and deletes the duplicate detections, ensuring every feature is counted only once.




Part 3

**Straightening the Lines (Regularization)**

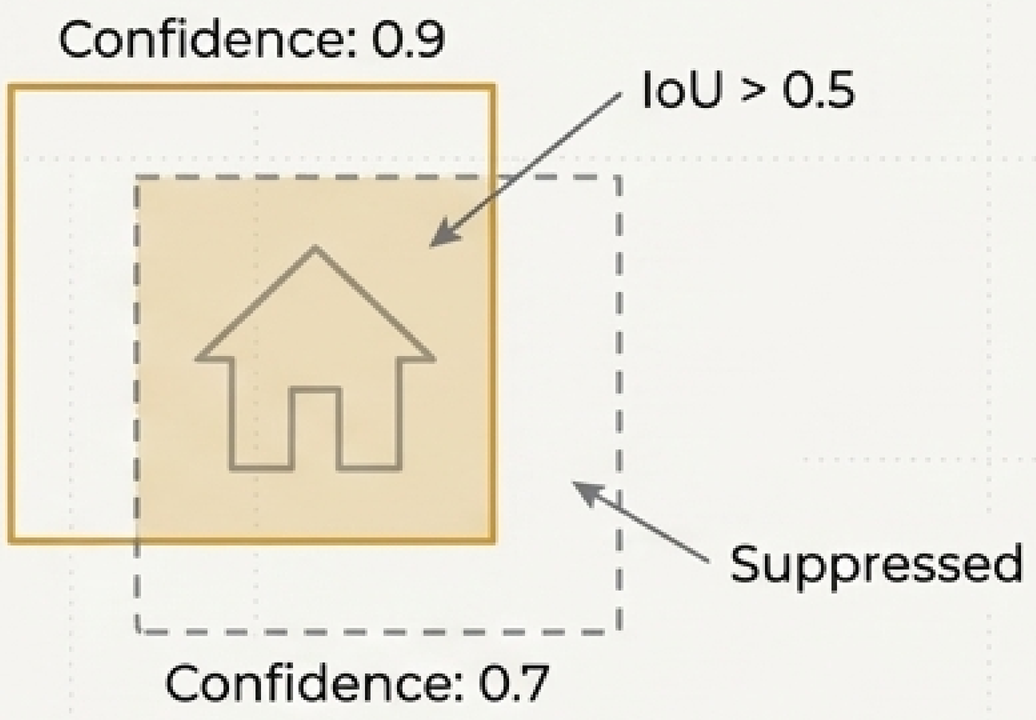
Lora Regular

The AI might draw a building's outline with slightly shaky or wavy lines. This final step applies a mathematical algorithm to 'snap' these lines into perfect, clean right angles, just as a human cartographer would.

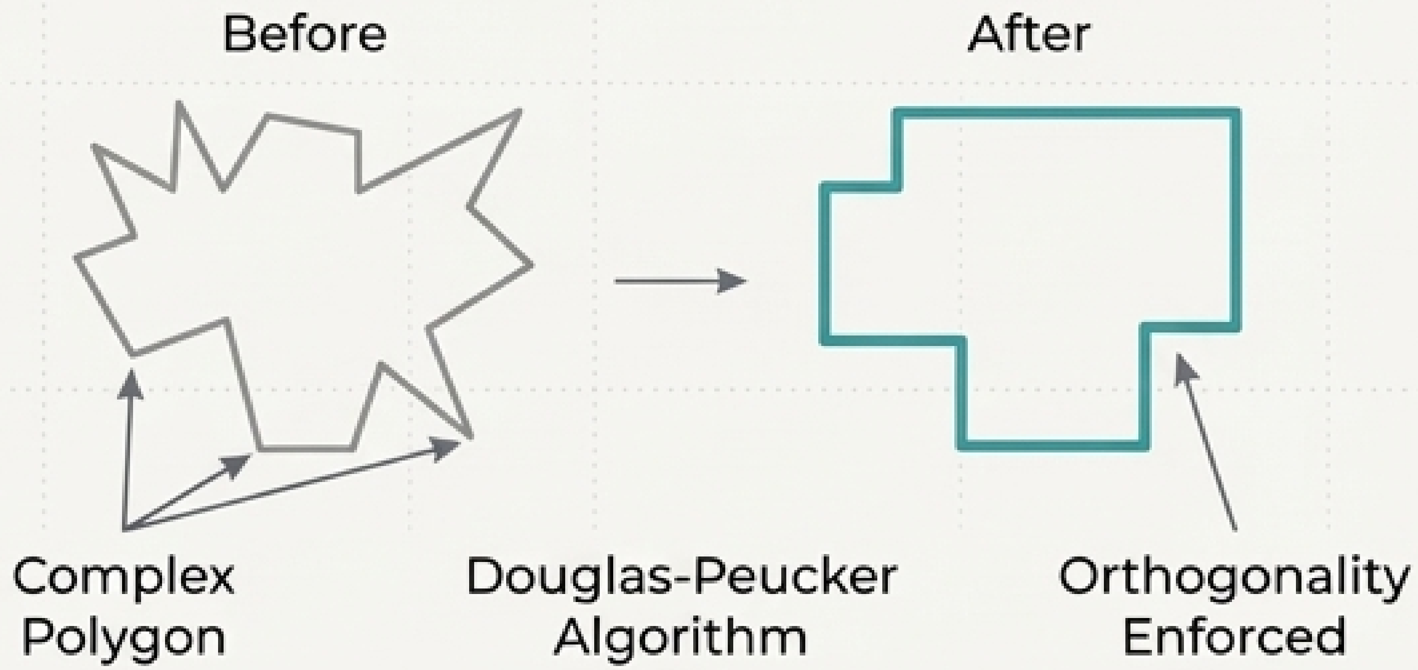
# Technical Details: Inference Stitching & Vectorization

- **Reconstruction**: Geospatial stitching of prediction tiles back to original map coordinates, generating a complete prediction GeoTIFF.
- **Conflict Resolution:** Non-Maximum Suppression (NMS) is applied to eliminate duplicate detections in overlapping regions. An Intersection over Union (IoU) threshold of 0.5 is used to determine duplicates.
- **Refinement & Vectorization: Polygon Regularization** algorithms (e.g., the Douglas-Peucker algorithm) are used to simplify contours. Orthogonality is enforced on building footprints to create clean, right-angled vectors.

## Non-Maximum Suppression (NMS)

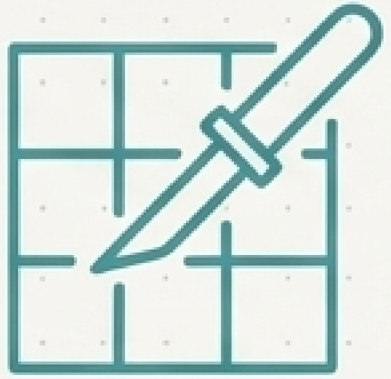



## Polygon Regularization



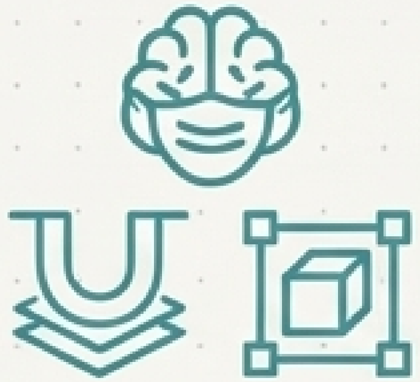
# The Digital Cartographer’s Workflow at a Glance

Goal: Achieve >95% Accuracy on SVAMITVA Feature Extraction.



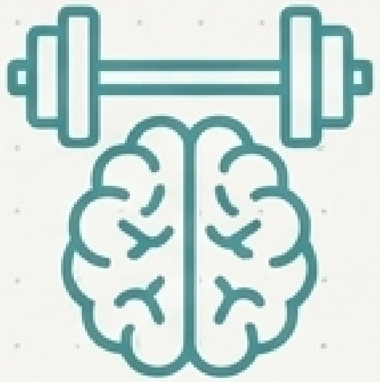
### CUT
### (Tiling)

Deconstruct high-resolution orthophotos into manageable tiles with strategic overlap.



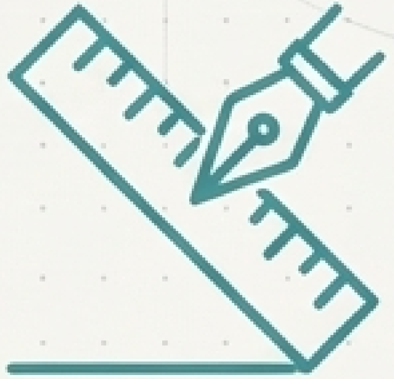
### ANALYSE
### (Multi-Model AI)

Deploy three specialist models (Mask R-CNN, U-Net, YOLOv8) to identify every feature type.



### TRAIN
### (Augmentation)

Force the models to learn robustly using a pipeline of visual ‘trick questions’.



### REFINE
### (Post-Processing)

Stitch predictions, eliminate duplicates (NMS), and regularise vector shapes for a clean final map.

# From Pixels to Policy: The Tangible Impact of Precision Mapping

**Property Rights:** Clean, accurate vectors form the basis for issuing property cards, resolving disputes, and enabling financial inclusion.

**Infrastructure Planning:** Accurate data on roads, buildings, and utilities allows for better planning of public services.

**Taxation & Development:** Creates a reliable record for fair property taxation and sustainable rural development.

By transforming raw pixels into structured data, this automated workflow provides the digital bedrock upon which the SVAMITVA scheme is built.