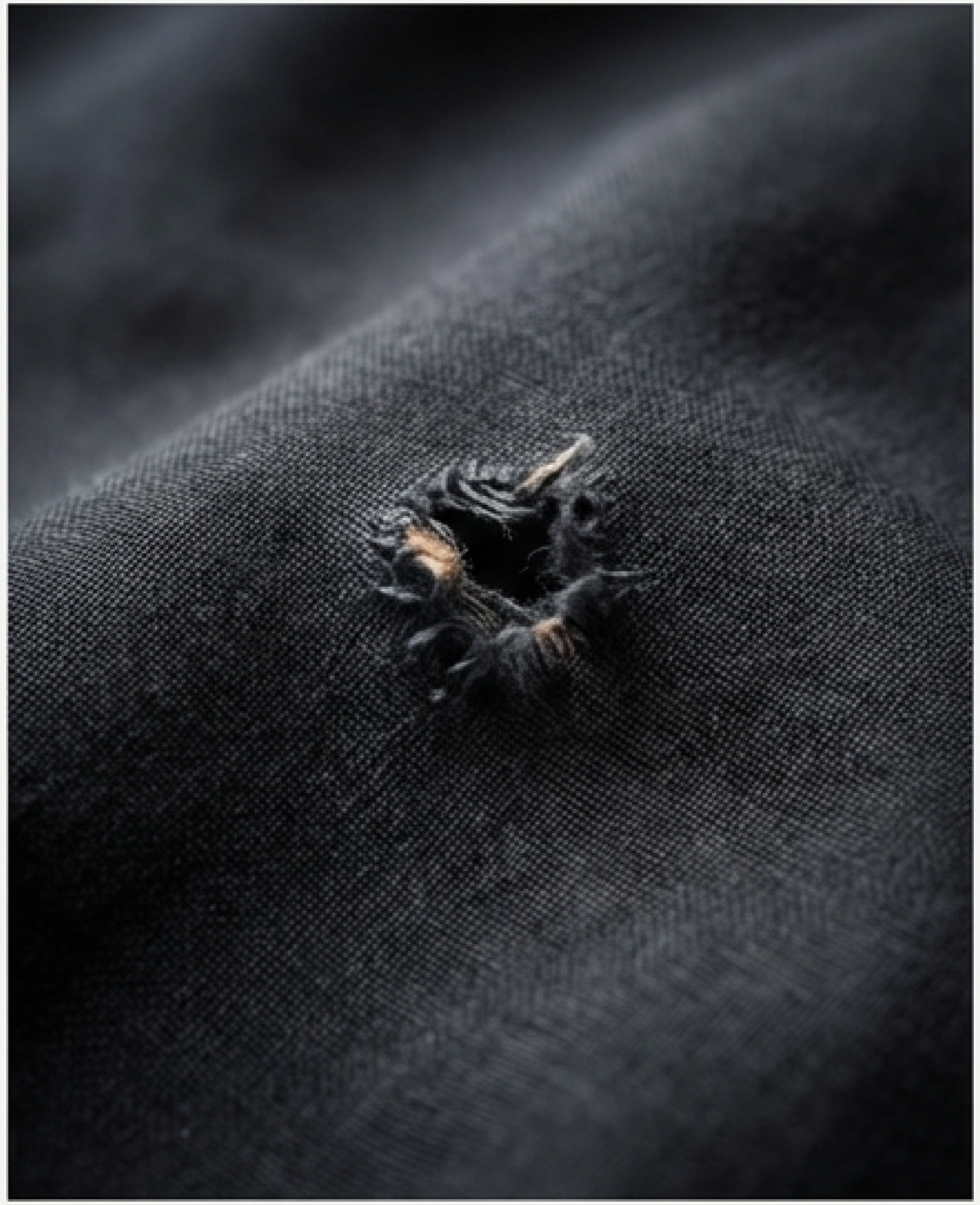
The Classic Edge: Real-Time Fabric Inspection Without Deep Learning

A case study in deploying a robust, interpretable multi-detector pipeline on a low-power edge device.

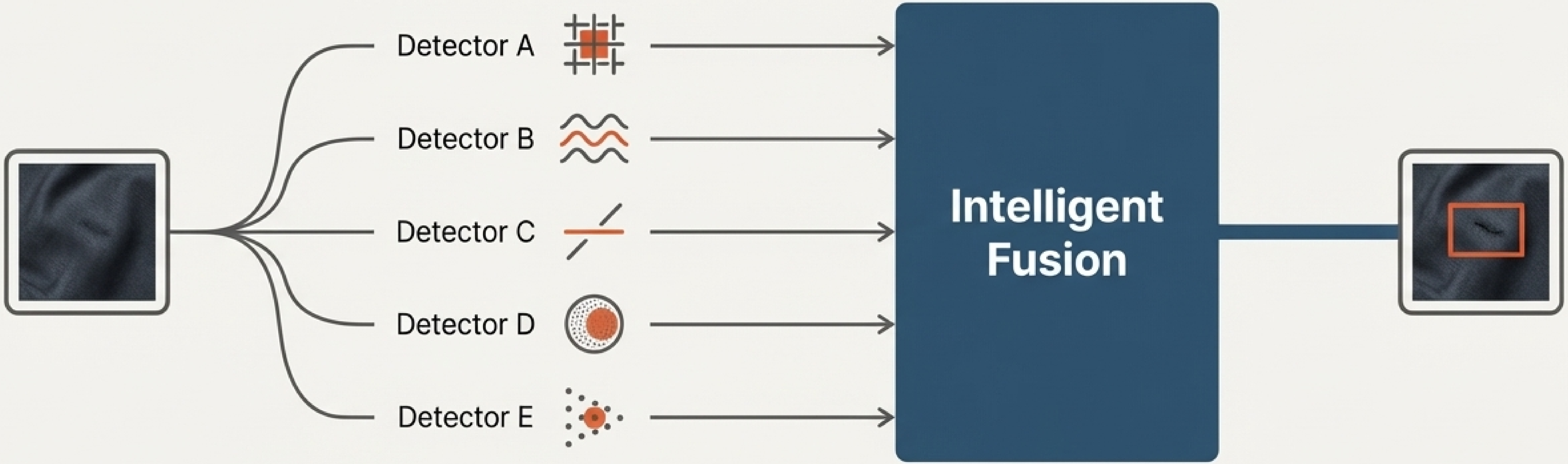
The Challenge: Industrial Inspection on a Tight Leash



The goal is to detect and localize fabric defects in real-time. However, the solution must meet strict operational constraints:

- **Real-Time Performance:** Must process live video feeds without lag on low-cost hardware.
- **Edge Deployment:** Must run efficiently on a low-power device (NVIDIA Jetson Nano).
- **Full Interpretability:** Every detection must be explainable, with no ‘black box’ components.
- **Data Independence:** Must operate without requiring large, labeled training datasets.

Our Approach: A Multi-Detector Classical Vision Pipeline



Instead of a single, monolithic deep learning model, we designed a system where multiple independent classicalial methods analyze each image in parallel. Their findings are then intelligently fused to produce a final, high-confidence result.

No Training Required

Deploys immediately without data collection or model training cycles.

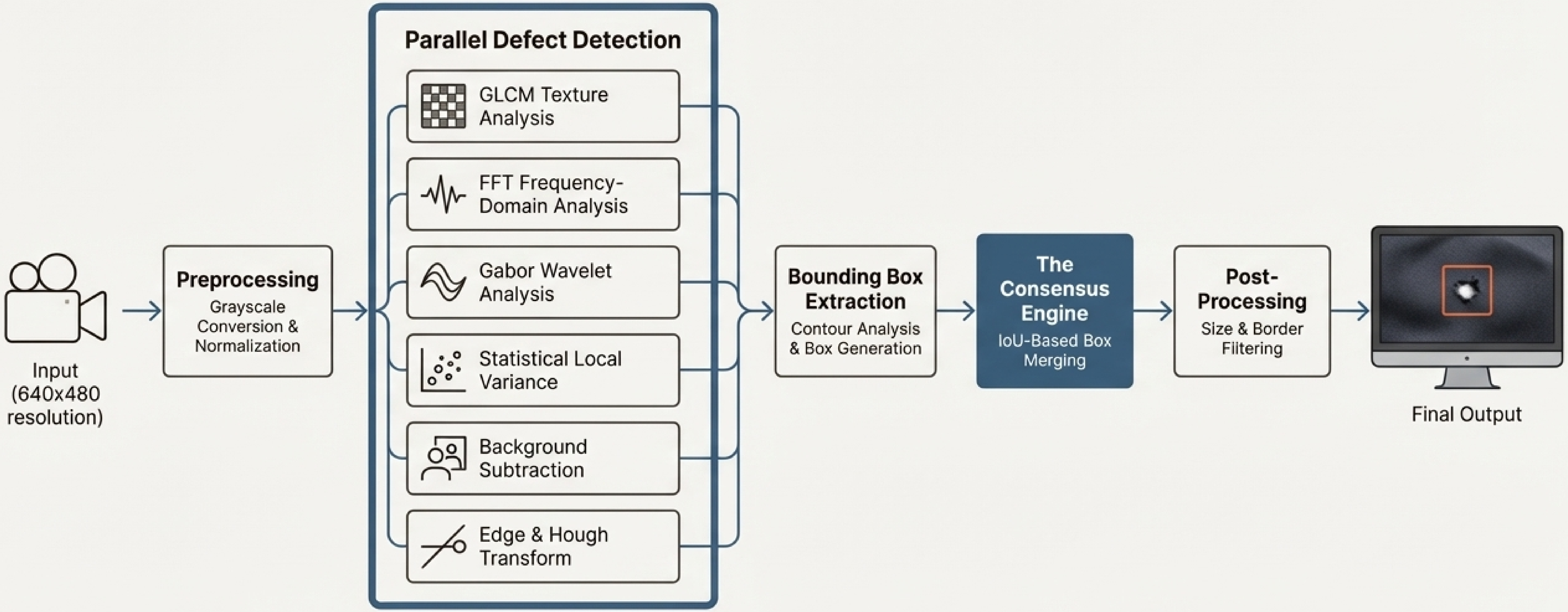
Explainable by Design

The logic of each detector (e.g., texture analysis, edge detection) is fully transparent.

Computational Efficiency

Tailor-made for the performance profile of low-power edge devices.

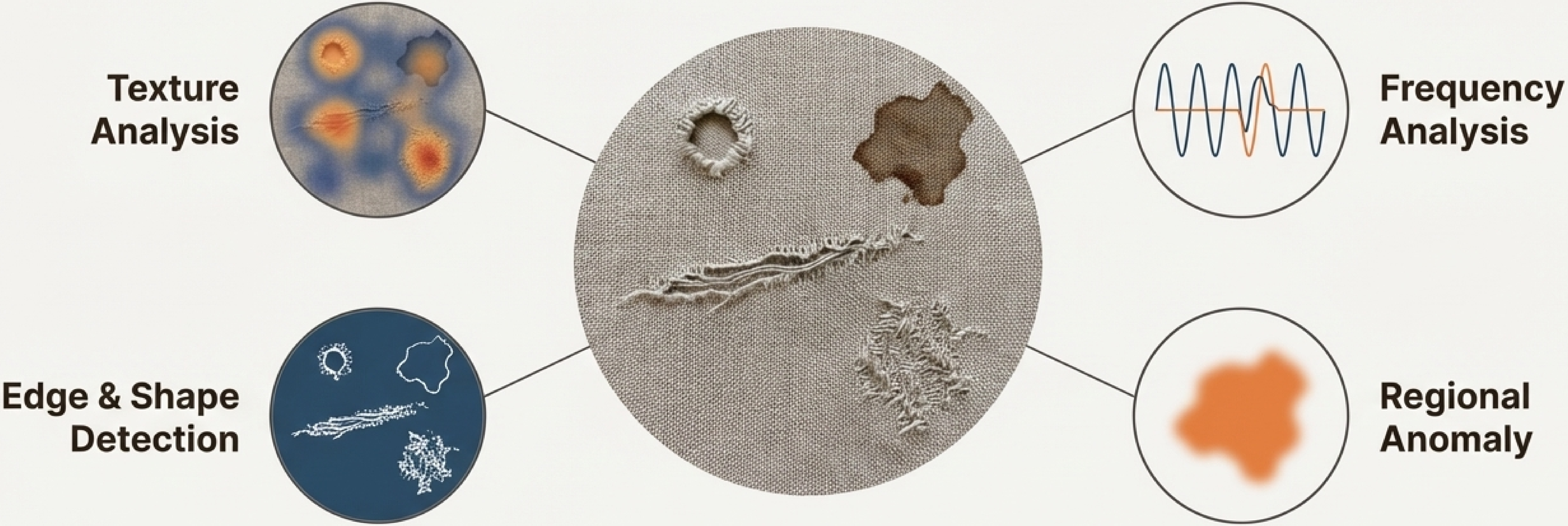
The System Architecture



NotebookLM

A Team of Specialists: The Parallel Detection Engine

No single classical algorithm can reliably detect all types of fabric defects. Our system deploys a team of specialized detectors, each with a unique expertise, to analyze the image simultaneously.



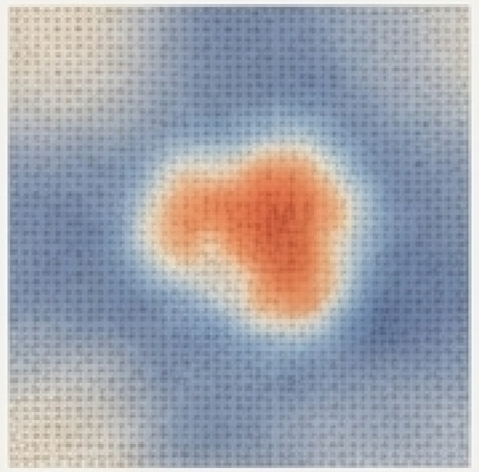
NotebookLM

Specialists for Texture and Repetitive Patterns

GLCM Texture Analysis

Method: Computes Gray-Level Co-occurrence Matrix contrast features in a sliding window. Z-score normalization highlights anomalies.

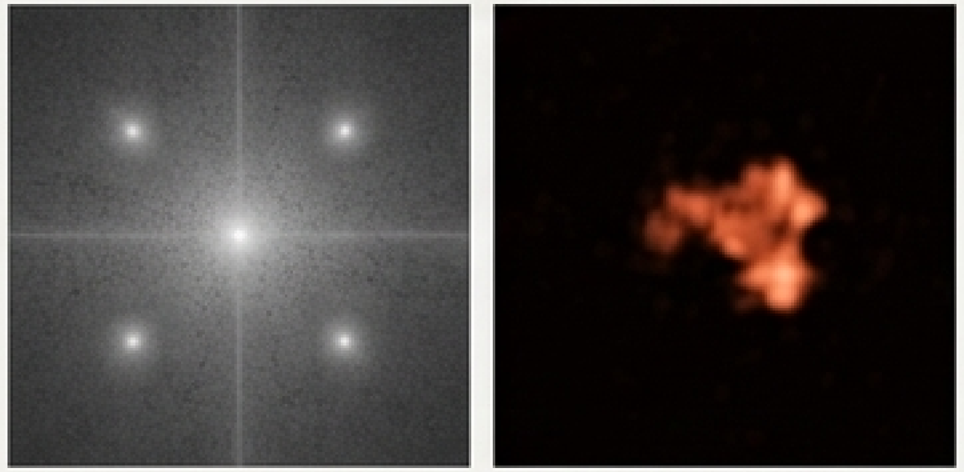
Specialty: Detecting small, subtle texture irregularities.



FFT Frequency-Domain Analysis

Method: Models the fabric's periodic texture with a 2D Fast Fourier Transform. The residual between the original and the reconstructed texture reveals defects.

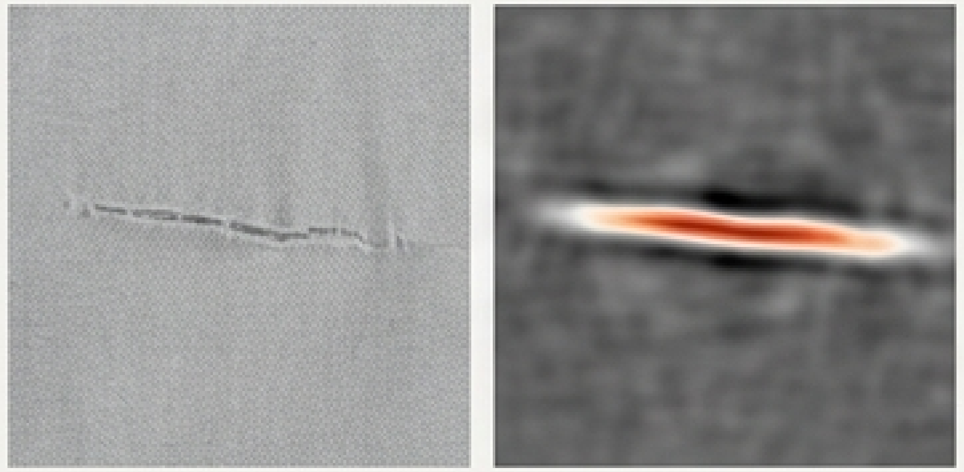
Specialty: Identifying flaws that break the fabric's regular, repeating pattern.



Gabor Wavelet Analysis

Method: A bank of Gabor filters at multiple orientations and frequencies computes local texture energy.

Specialty: Finding directional defects like misweaves and thread irregularities.

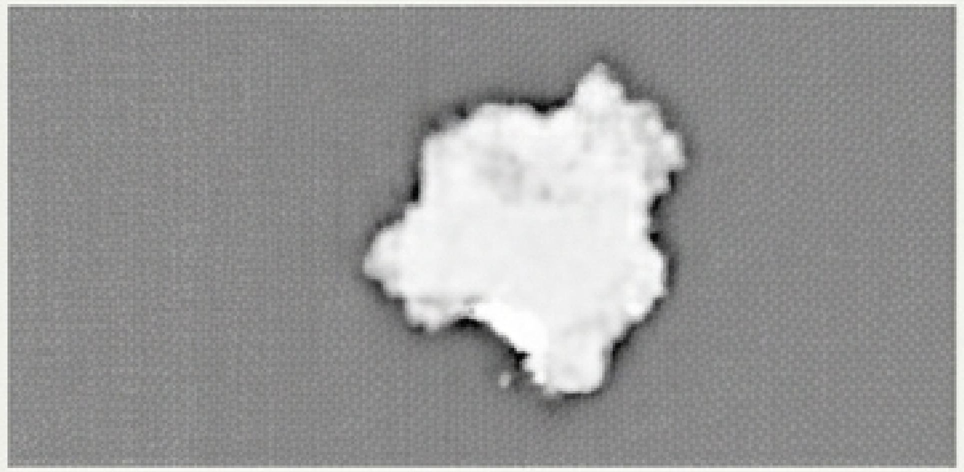


Specialists for Structural and Regional Flaws

Statistical Local Variance

Method: Calculates local standard deviation in a sliding window. Z-score thresholding identifies regions of abnormal texture.

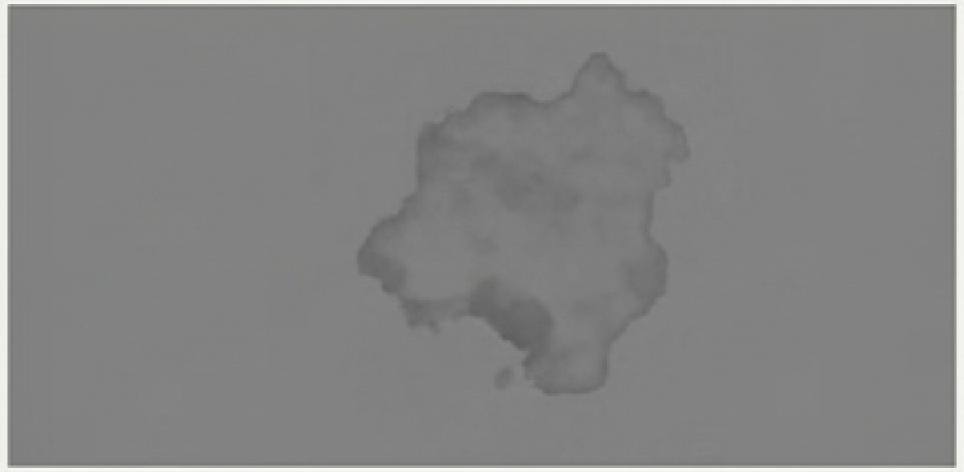
Specialty: Highly effective for holes, tears, and abrupt fabric damage.



Background Subtraction

Method: Estimates the regular fabric background using a median filter. Subtracting this from the original image highlights anomalies.

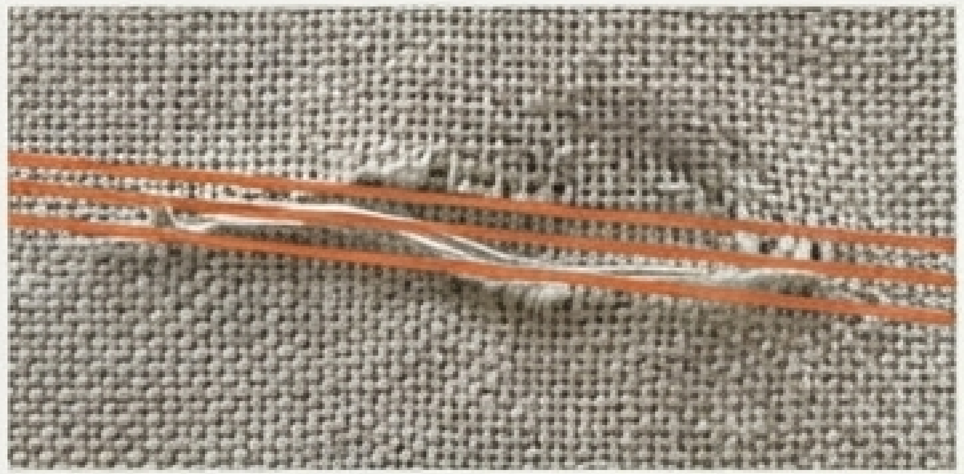
Specialty: Excels at finding non-structural defects like stains and faded regions.



Edge & Hough Transform

Method: Applies Canny edge detection followed by a probabilistic Hough transform to find line segments.

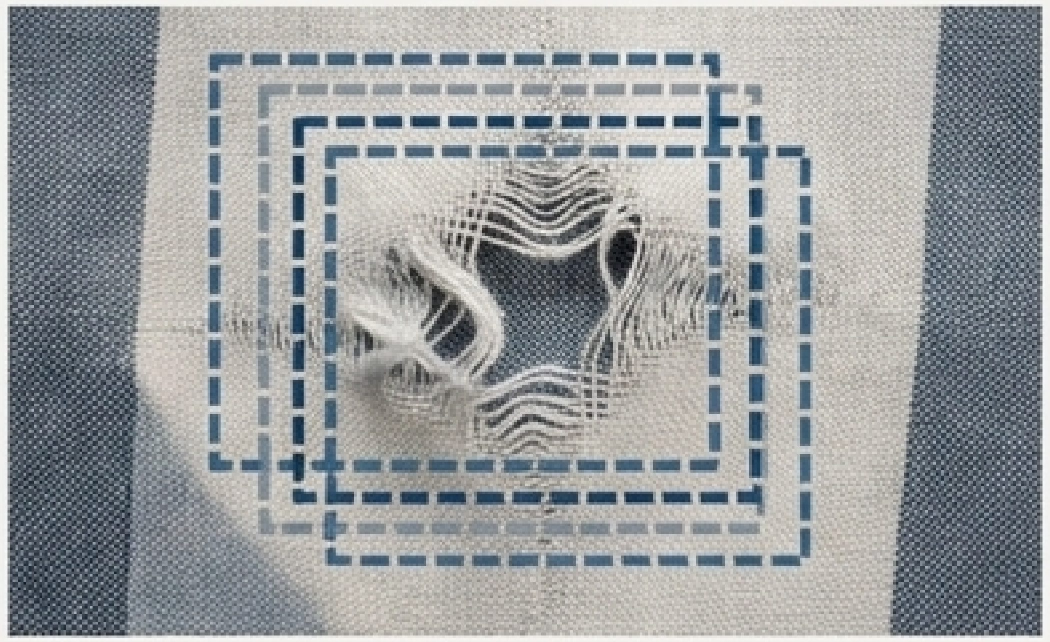
Specialty: Pinpointing linear defects such as thread breaks or snags.



The Consensus Engine: Forging a Unified Result

- **Problem:** With multiple detectors, a single defect might be flagged by several specialists, resulting in redundant, overlapping bounding boxes.
- **Solution:** An IoU (Intersection over Union)-based fusion strategy acts as a consensus engine.
  1. Calculate Overlap: The IoU metric is computed for all pairs of bounding boxes.
  2. Merge High-Confidence Groups: Boxes with an IoU above a set threshold are grouped and merged into a single, encompassing bounding box.
- **Outcome:** This process reduces duplicates, stabilizes localization, and increases the confidence of the final detection.

Before



After

Final Refinement: Eliminating Noise and Artifacts

To eliminate common false positives generated by imaging conditions or algorithmic artifacts, two key filters are applied to the merged bounding boxes.

Filter 1: Size Filtering

Removes detections that are unrealistically large (e.g., covering most of the image), which are often caused by illumination changes or frequency artifacts.

Filter 2: Border Filtering

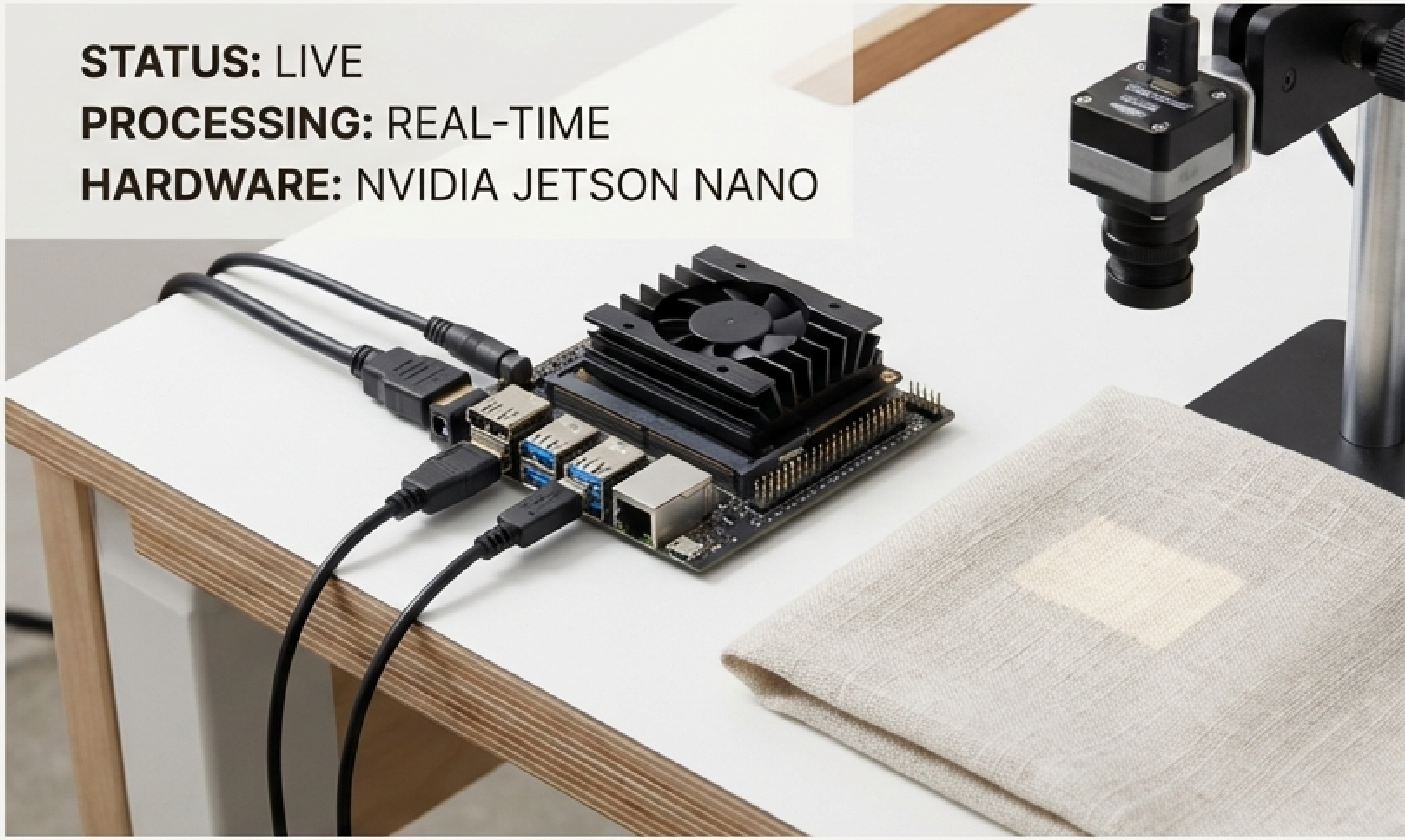
Discards boxes that touch the absolute edges of the image, as these are typically caused by padding or camera frame artifacts.

These simple, rule-based filters significantly improve the precision and reliability of the final output.

The Moment of Truth: Live on Jetson N Nano

- **Deployment:** The entire classical vision pipeline was deployed on an NVIDIA Jetson Nano.

- **Process:**
  - The pipeline was converted into modular, optimized Python scripts.
  - All disk I/O and plotting were removed from the runtime loop to maximize speed.
  - The system was configured to process a live USB camera feed.



Performance on the Edge: Efficient, Stable, and Real-Time

Real-Time Speed

Successfully processes the 640x480 camera feed at a real-time frame rate.

Low Resource Usage

Operates within the low memory and power envelope of the Jetson Nano.

Robust Detection

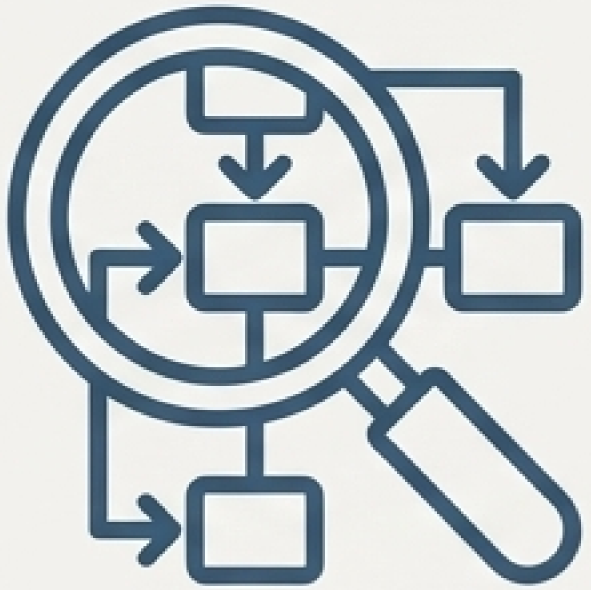
Demonstrates visual correctness and stability across various fabric types and defect classes.

Zero Training Time

Achieved 100% of this performance without any labeled training data or GPU-intensive training cycles.

Validating the Classical Approach

By deliberately choosing a classical computer vision pipeline, we built a system that is not only effective but also uniquely suited for the target application.



Interpretability

Every step is transparent and debuggable.

Efficiency

Natively designed for low-power, real-time edge computing.

Agility

No dependency on data collection, allowing for rapid deployment and adaptation.

Future Work and Potential Enhancements

Detector Confidence Voting

Weight the influence of different detectors in the fusion process based on their confidence.

Runtime Optimization

Further code profiling and optimization to increase FPS.

Conveyor Belt Integration

Adapt the system for use in a live industrial manufacturing line.

Optional Deep Learning Verifier

Explore a hybrid model where a lightweight classifier verifies potential defects found by the classical pipeline.

“This project demonstrates that **robust real-time fabric defect detection** can be achieved using **classical computer vision alone**, making it suitable for low-power edge devices such as Jetson Nano without relying on deep learning.”