Machine Vision & Computer Vision

Visual inspection, defect detection, and camera-guided automation.

`machine-vision-computer-vision` · Revision Draft · 2026-07-07 · Status: **Draft**

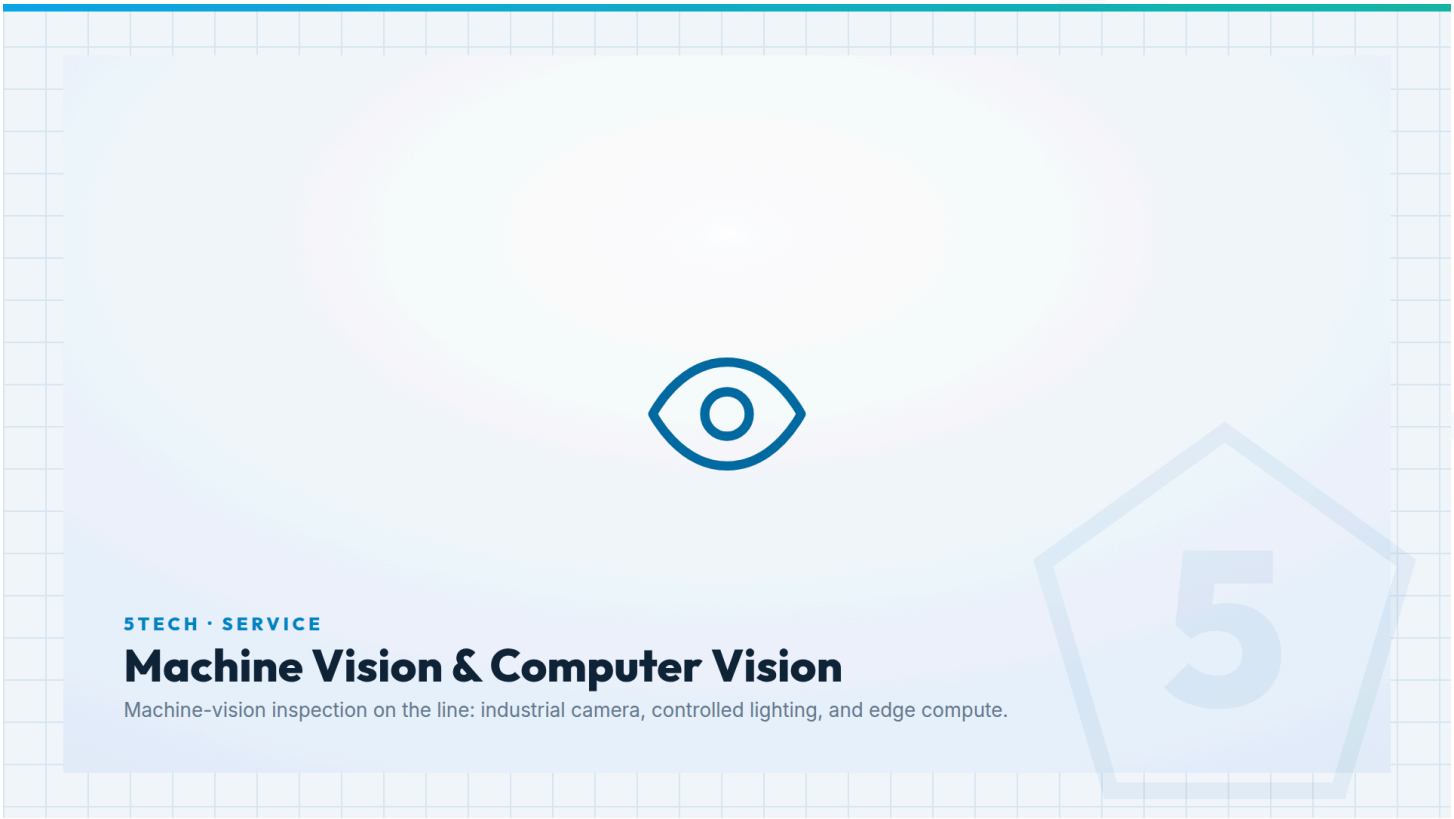

Figure FIG-001 — Machine-vision inspection on the line: industrial camera, controlled lighting, and edge compute.

We build camera-based systems that inspect, measure, count, and guide — from classical machine vision to modern deep-learning models running at the edge. The goal is a system that holds up on the line, not just on a benchmark. Most vision success or failure is decided by optics, not algorithms, so we begin with a feasibility study on your real product images and lock down camera, lens, and lighting before touching a model. Only then do we choose between classical vision and deep learning, based on which is more reliable and maintainable for your task.

Who this is for

- Manufacturers with manual inspection or QA bottlenecks

- Lines that need counting, measuring, or presence/absence checks
- Teams needing vision to guide robots or sorting equipment

Problems we solve

- Manual visual inspection that is slow, costly, or inconsistent
- Defects escaping to customers, or scrap discovered too late
- Existing vision setups that fail with lighting or product variation
- Uncertainty about whether a vision approach is even feasible

What you get (deliverables)

- Feasibility assessment with sample images from your product
- Camera, lens, and lighting selection for stable results
- Inspection / detection model (classical or deep learning)
- Edge deployment integrated with your line or robot
- Validation against real defects and an accuracy report

Every project is validated against known-good and known-bad samples with an accuracy report you can act on — not a vague claim of being "AI-powered". Because we also do robotics, embedded, and software, we integrate the camera into the rest of your line — from trigger to reject mechanism to data — instead of leaving you to stitch vendors together.

How we work

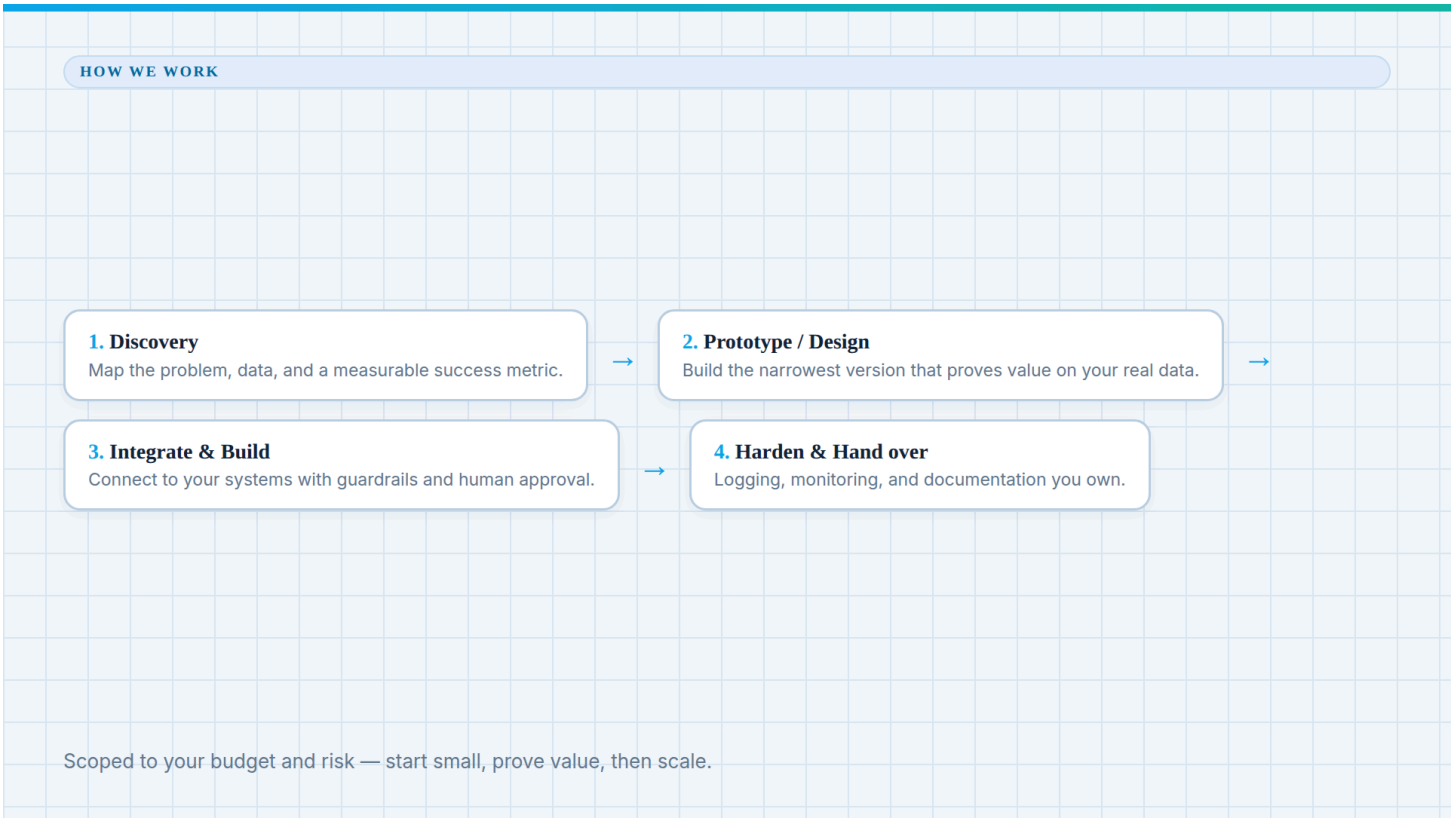

Figure FIG-002 — How a machine-vision engagement runs: image study, optics, model, then deploy and validate.

Most engagements begin with a low-cost feasibility study on your samples, so you can decide with evidence instead of a sales promise, and only commit to a full build once the approach is proven on your product.

1. **Image study** — Collect representative images and confirm the problem is solvable with vision.
2. **Rig the optics** — Lock down camera, lens, and lighting; most vision success is decided here.
3. **Model & tune** — Build and tune the detection/measurement model on your real data.
4. **Deploy & validate** — Run at line speed at the edge and validate against known-good/known-bad samples.

We are deliberately honest about feasibility: some inspection problems are solved better and more cheaply with classical vision, some genuinely need deep learning, and a few are not reliably solvable with cameras at all — and we will say so after studying your images.

Example use cases

- Automated visual inspection and quality control

- Defect detection on high-speed production lines
- Camera-based counting, measuring, and sorting
- Computer vision for manufacturing and food processing
- Vision guidance for robotic pick and placement

Related services & industries

Related services

- Robotics & Automation — `robotics-automation`
- AI Automation & Agents — `ai-automation-agents`
- Embedded Systems & IoT — `embedded-sensors-iot`

Industries served: Manufacturing · Food processing · Warehousing · QA & inspection.

Related industries: `manufacturing-automation` · `warehouse-robotics` · `retail-ai-automation`.

Common questions

Will it work with our product and lighting? We start with a feasibility study on your real images and control the optics and lighting, which is where most machine-vision projects succeed or fail.

Classical vision or deep learning? Whichever is more reliable and maintainable for your task. Many inspection problems are solved better and cheaper with classical vision; others need deep learning. We recommend based on your data.

Can it run at line speed? Yes — we deploy optimized models at the edge so inspection keeps up with production throughput.

> Voice: capability-based and honest — no invented metrics or client names (see the live `services.ts` for the established tone).

Figures

ID	TITLE	STATUS	FUTURE FILENAME	PURPOSE
FIG-001	Industrial Hero Render	Missing	`images/hero_service.png`	Cover / hero image
FIG-002	Engagement Process	Missing	`diagrams/process.png`	The service engagement process