Computational Engineering and Technology Innovations

www.ceti.reapress.com

Comput. Eng. Technol. Innov. Vol. 3, No. 3 (2026) 176–192.

Paper Type: Original Article

View Coverage and Evidence Gating for Vision-Based Robotic Quality Inspection

Victoria Nozick[1,*], Fatemeh Rasoulpour[2], Milos Milovancevic[1]

[1] Departmant of Operations and Information Management Group, Aston Business School, Aston University, B4 7ET Birmingham, United Kingdom; victorianozick79@gmail.com.

[2] Department of Computer Engineering, Institute of Higher Education, RVF6+JJ9 Tenekabon, Iran; rasoulpour.72@gmail.com.

[3] Department of Mechanical Engineering, University of Nis, Serbia. milovancevic@masfak.ni.ac.rs.

Citation:

Received: 06 July 2025 *Revised: 18 September 2025* *Accepted: 08 November 2025*	Nozick, V., Rasoulpour, F., & Milovancevic, M. (2026). View coverage and evidence gating for vision-based robotic quality inspection. *Computational Engineering and Technology Innovations*, *3*(3), 176-192.

Abstract

Robotic quality inspection must translate observations of individual surfaces into a justified disposition of a physical item. An image without a visible defect does not establish that the item is defect free, particularly when different views expose different surfaces. This study develops an evidence-gated inspection architecture that binds image scores to item identity, acquisition epoch, required view coverage, model and calibration versions, evidence age, and independent execution permits. The architecture is evaluated through an object-disjoint public-data experiment, an executable supervisory replay, and a separate numerical geometry fixture. KolektorSDD contains 399 images from 50 defective items, including 52 images with visible defects. Fixed logistic-regression and extremely randomized-tree baselines achieved pooled held-out image AUROCs of 0.968 and 0.960, respectively. Nevertheless, a first-view binary policy released 44 of the 50 defective items with either model. Aggregating all available views reduced these escapes to six and fourteen; adding a fixed abstention interval and an eight-view release requirement reduced them to one and zero, while holding five and fourteen items. Nine adverse evidence or permit conditions prevented release in all 450 constructed fault traces. A 100,000-sample, explicitly assumed pose-error fixture agreed with the linearized covariance to within 0.87% relative Frobenius error. These results establish an auditable connection between perception evidence and supervisory disposition. They do not establish normal-item specificity, production throughput, physical robot performance, or safety compliance. The supplied code, split manifests, per-image predictions, decision traces, and equation sources support independent reproduction.

Keywords: Robotic inspection, Industrial vision, Multi-view evidence, Selective decision, Item traceability, Pose uncertainty.

1 | Introduction

An autonomous inspection station must answer a physical question: Which item may proceed, which item must be diverted, and which item requires additional evidence? A vision model usually answers a narrower question about an image or a localized image region. The gap between these two questions becomes consequential when an object has several inspected surfaces, a camera result arrives after the object has

Corresponding Author: victorianozick79@gmail.com jfea@aihe.ac.ir

https://doi.org/10.48314/ceti.v3i3.91

Licensee System Analytics. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0).

moved, or a robot receives a valid-looking decision for the wrong acquisition cycle. Improving image classification alone cannot resolve missing surface coverage or incorrect associations between observations and physical items.

The foundations of image formation, geometric calibration, feature extraction, and learning are well established [1], [2]. Their integration into an inspection cell nevertheless requires explicit contracts between perception and control. A defect score must refer to a particular item and view, remain interpretable under a known model version, and reach the control layer before its evidence becomes obsolete. A robot command must additionally satisfy the geometric and cell-level conditions required for the requested operation. These conditions concern different kinds of uncertainty and should be checked at the interface where each becomes relevant.

KolektorSDD exposes the distinction between image evidence and item quality particularly clearly [3], [4]. Its 399 images contain 347 negative surface views, yet every one of its 50 physical items is defective. A high-specificity image classifier can therefore produce many correct negative image decisions while a policy that releases the whole item after one such decision produces many defective releases. This is a structural acquisition problem before it is a model-selection problem. Conversely, a controller that holds every uncertain item can avoid observed escapes while providing poor service to normal production. Evaluation must account for disposition outcomes and unresolved items, rather than equating a low escape count with a complete inspection solution.

This paper develops a supervisory architecture in which the unit of disposition is the item and the unit of evidence is an identified view. The technical contribution is a precise evidence ledger and release invariant, coupled to an executable replay that separates classifier errors from coverage and association failures. A second contribution is a controlled public-data demonstration using fixed, transparent classifiers and the original physical-item-disjoint folds. A third contribution is an uncertainty-aware camera-to-robot interface, whose numerical example is explicitly a designed fixture rather than a measurement of the dataset acquisition system. Together these elements make the perception-to-action claim inspectable without presuming an unreported industrial deployment.

The study asks whether complete-view aggregation changes the apparent reliability of a fixed perception model, what additional effect a lower release threshold and HOLD action have, and whether common evidence faults can reach the RELEASE branch of the implemented supervisor. It also asks how a simple pose-uncertainty model changes a geometric clearance requirement. The empirical evidence is restricted to recorded industrial images and software execution. The physical robot, lighting hardware, PLC connection, and manufacturing execution system are specified interfaces for subsequent integration.

2 | Literature Synthesis and Research Position

Industrial visual inspection developed along several complementary directions. Tabernik et al. [3] introduced a segmentation-based method for learning from relatively few defective examples and released the Kolektor surface data. Bergmann et al. [5] established MVTec AD as a broad industrial anomaly-detection benchmark with normal training data and image-level and pixel-level evaluation. These contributions made defect perception reproducible, but their image protocols did not by themselves determine when a robot should release a multi-surface physical item. The distinction is an evaluation boundary, rather than a shortcoming of those datasets' stated objectives [3]. Roth et al. [6] proposed PatchCore, which represented nominal appearance using a memory bank of patch features. Batzner et al. [7] developed EfficientAD around a lightweight student–teacher design and incorporated global information for logical anomalies. Both works strengthened practical anomaly detection, including computational considerations, yet a reported model inference time remained different from an entire inspection-cell cycle. Acquisition, view changes, motion settling, communication, and decision escalation add work that cannot be inferred from image-model throughput. Neither method was reimplemented in the present experiment, and no numerical ranking against them is implied.

Zou et al. [8] introduced VisA and a self-supervised pretraining scheme that increased the diversity of objects and anomaly scenarios available for evaluation. Wang et al. [9] moved the benchmark focus toward real-world multi-view inspection through Real-IAD. Zhu et al. [10] subsequently combined RGB, pseudo-3D, and 3D observations in Real-IAD D3. These datasets made the organization of complementary observations increasingly central. Their existence also limits an expansive novelty claim: Multi-view inspection and multimodal evidence are established research directions. The present study concerns the conditions under which an accumulated set of observations is allowed to authorize a supervisory action.

Heckler-Kram et al. [11] extended industrial anomaly evaluation to difficult scenarios in MVTec AD 2, including changed lighting, transparent or overlapping objects, and very small defects. Their results showed why performance on a mature benchmark should not be treated as a general statement about production robustness. KolektorSDD is used here because its item identities and surface labels support an unusually clear coverage analysis, not because it represents the full range of industrial variability. Cross-facility transfer, illumination shift, and inspection of geometrically different products remain separate tests.

On the integration side, Silva and Paladini [12] studied smart machine vision for assembly-line decision making. Puttero et al. [13] reviewed collaborative robots for quality control and identified adoption barriers involving system integration, expertise, investment, data security, and regulatory requirements. These application and survey perspectives motivated an architecture that records the evidence and permits behind a decision. They did not supply measurements for the station specified here. *Table 1* compares the functions of the reviewed streams without mixing their reported benchmark scores.

Table 1. Comparison of relevant research streams and the interface addressed in this study.

Stream / Examples	Primary Contribution	Evidence Unit	Relation to This Study
KolektorSDD; MVTec AD	Supervised surface defects; normal-only anomaly benchmark	Image / pixel	Used item-preserving Kolektor folds; did not reproduce segmentation
PatchCore; EfficientAD	Patch memory; efficient student–teacher inference	Image / pixel	Possible score modules; not numerical comparators
VisA; Real-IAD	Diverse objects; complementary camera views	Image / multi-view item	Motivated distinct view and item evaluation
Real-IAD D3; MVTec AD 2	Complementary modalities; difficult acquisition conditions	Multi-modal sample / image	Define important transfer tests not performed here
Selective classification	Control error by abstaining on predictions	Prediction / selected subset	HOLD is separate from physical REJECT
Calibration; visual servoing	Relate sensing geometry to robot motion	Pose / feedback state	Geometric permit specified and simulated
Present study	Item-bound evidence ledger and gated disposition	Image → item → request	Measured image scores; executed software replay

Descriptions concern stated research scope, rather than an exhaustive audit of every implementation.

The HOLD action has a longer statistical history than modern industrial vision. Chow [14] formulated the error–rejection tradeoff. El-Yaniv and Wiener [15] analyzed selective classification through its risk–coverage relationship, and Geifman and El-Yaniv [16] extended selective prediction to trained deep networks. In those settings, rejection meant abstaining from a prediction. In this paper, REJECT denotes physical disposition of a suspected defective item, whereas HOLD denotes abstention or unresolved evidence. Keeping these meanings distinct prevents a statistical reject option from being confused with a robot diversion command.

Guo et al. [17] showed that predictive confidence and empirical correctness required separate calibration analysis. This distinction matters even for non-neural baselines: Class weighting and finite training data can yield useful ranking scores without production-calibrated defect probabilities. The present thresholds are fixed operating rules on scores. They are not interpreted as posterior defect probabilities or as certified

confidence limits. Saito and Rehmsmeier [18] emphasized precision–recall analysis for imbalanced classification accordingly, both AUROC and average precision are reported, together with the underlying image-level confusion counts.

Robotic geometry supplied another independent body of prior work. Zhang [19] described planar-pattern camera calibration. Tsai and Lenz [20] addressed hand/eye calibration, and Park and Martin [21] treated the rigid-transform equation on the Euclidean group. Chaumette and Hutchinson [22] distinguished image-based and position-based visual servo control and discussed their performance and stability. These works provided established transformations and control concepts. The contribution here is the placement of a geometric permit beside the evidence permit; no new calibration estimator, visual-servo law, or trajectory planner is claimed.

Macenski et al. [23] described ROS 2 as a modular robotics software architecture. Lamport's [24] treatment of event ordering provided the foundational distinction between causal ordering and assumptions about clocks. Sculley et al. [25] drew attention to dependencies surrounding the learned component of machine-learning systems. These perspectives supported explicit item epochs, version records, and audit logs. An epoch prevents observations from an earlier acquisition cycle from silently satisfying a later cycle, while a monotonic receipt clock supports a local age check. Neither feature alone proves end-to-end timing correctness.

The official ISO catalogue distinguished robot-level requirements in ISO 10218-1:2025 from application and cell integration requirements in ISO 10218-2:2025 [26], [27]. The supervisor is an ordinary quality-decision component that consumes a separate cell permit. Its image model and software gate are not safety-rated functions, and the replay does not constitute a standards-conformity assessment. *Table 2* states the remaining research gaps as concrete evaluation needs; *Table 3* summarizes the evolution of the relevant ideas.

Table 2. Research gaps, the present response, and the evidence that remains necessary.

Question	Present Response	Remaining Evidence
Can a negative view justify item release?	Formal coverage definition and perfect-local-classifier counterexample	Task-specific proof that planned views cover relevant defects
How are unresolved items counted?	REJECT / HOLD / RELEASE counts on all defective items	Normal items, adjudication workload and prospective cost
Can old or mismatched evidence authorize release?	Epoch-bound ledger; 450 adverse traces	Measured acquisition, network and PLC event histories
Is the target geometrically admissible?	Ellipsoid support bound and numerical pose fixture	Measured calibration covariance, tool volume and path checks
Does software quality imply cell safety?	Independent cell permit required	Application-specific risk assessment and safety validation

Table 3. Selected milestones in the evolution from image recognition to auditable robotic inspection.

Period	Milestone	Implication for Inspection
1970–1978	Chow: Abstention, Lamport: Event ordering	Prediction refusal and evidence ordering are distinct design problems
1989–2006	Hand/eye calibration, camera calibration, visual servoing	Geometric estimation and feedback enable vision-guided motion
2010–2017	Selective classification and confidence calibration	Score quality and willingness to decide require separate evaluation

Table 3. Continued.

Period	Milestone	Implication for Inspection
2019–2022	KolektorSDD, MVTec AD, PatchCore, VisA	Public image protocols made perception results reproducible
2024–2026	EfficientAD, Real-IAD, Real-IAD D3, MVTec AD 2	Efficiency, multiple views/modalities and acquisition shift gained emphasis
Present study	Evidence-gated supervisory replay	Physical item disposition is evaluated separately from image accuracy

This synthesis was selective rather than systematic. Primary publisher, author, institutional, standards, and dataset records were checked through 21 October 2026. Search themes included industrial anomaly detection, multi-view inspection, selective classification, hand/eye calibration, robotics middleware, and evidence traceability. The companion source ledger records access restrictions and metadata discrepancies. No source-specific performance number was imported into the experimental results.

3 | Evidence and Disposition Model

3.1 | From a Surface Observation to an Item Decision

Let an item be indexed by i, an acquisition epoch by e, and a planned view by v. The view plan V_i is finite and fixed before inspection starts. A binary annotation y_{iv} indicates a visible defect in a recorded view. The item label concerns the union of defects over its relevant inspected surfaces:

$$y_{iv} = 1\{\max_x M_{iv}(x) > 0\}, \qquad Y_i = \max_{v \in V_i} y_{iv}.$$

Here M_{iv} is the supplied annotation mask and x is an image pixel. The first expression is how image labels were extracted in the experiment. The second is an item-label definition when the plan covers the relevant surfaces and the labels are complete; it is not a claim that an arbitrary finite camera plan observes every possible defect. Every item in the analyzed archive had at least one positive recorded view. A negative view establishes only the absence of an annotated visible defect in that view.

An evidence record contains the item ID, epoch, view ID, image hash, local receipt time, model version, calibration version, and score. The image hash belongs in the audit manifest; the reference supervisor checks the identity and version fields and a bounded finite score. The production interface should authenticate the association between an image and its record. A consistent metadata string cannot by itself detect an upstream sensor that labels the wrong physical item.

Let B_{ie} denote the ledger indexed by unique view ID, t the decision time, t_v a local receipt time, and Δ the permitted receipt age. Define $I_{ie} = 1$ when the ledger has no identity, epoch, version, duplicate, or score-validity fault. Define $F_{ie}(t) = 1$ when every stored receipt age lies in the permitted interval. Completeness and eligibility then take the form:

$$C_{ie} = 1\{\mathrm{keys}(B_{ie}) = V_i\}, \quad F_{ie}(t) = 1\{0 \le t - t_v \le \Delta \ \forall v \in B_{ie}\}.$$

The reference implementation latches a validity fault for the current epoch when it receives a duplicate or inconsistent record. Recovering from that fault requires a new, explicitly started inspection epoch. This conservative rule sacrifices availability to keep evidence repair unambiguous. It does not require every physical system to forbid retransmission: An implementation that supports idempotent retransmissions must specify when two records are identical and how conflicts are handled. Each view supplies a score $s_{iv} \in [0,1]$. The accumulated score is:

$$q_{ie} = \max_{v \in \mathrm{keys}(B_{ie})} s_{iv}, \qquad q_{ie} = 0 \text{ when the ledger is empty.}$$

The maximum is a transparent pooling rule: One strongly suspicious surface can prevent release. It is neither an estimate of the probability that the item is defective nor a correction for differences in score calibration between models. Let J_i be an independent cell permit and K_i a geometric permit. Let $\ell < u$ be fixed lower and upper score thresholds. At the requested finalization time, the supervisor returns

$$a_{ie} = \begin{cases} \text{HOLD}, & I_{ie}F_{ie}J_iK_i = 0, \\ \text{REJECT}, & I_{ie}F_{ie}J_iK_i = 1,\ q_{ie} \geq u, \\ \text{RELEASE}, & I_{ie}F_{ie}J_iK_iC_{ie} = 1,\ q_{ie} < \ell, \\ \text{HOLD}, & \text{otherwise.} \end{cases}$$

During planned acquisition, an incomplete clean ledger remains in ACQUIRE; the equation describes finalization or a deadline, when unresolved acquisition becomes HOLD. REJECT is allowed before full coverage because positive evidence may already justify diversion. The decision is still a request to the execution layer. Actual movement requires an action-specific permit and a command handshake; HOLD leaves the item under the cell's defined retention procedure.

Proposition 1 (Release invariant). For an epoch whose ledger starts empty and is modified only by the specified ingestion rule, every RELEASE returned by the finalization rule satisfies:

$$a_{ie} = \text{RELEASE} \Rightarrow I_{ie} = F_{ie} = J_i = K_i = C_{ie} = 1 \quad \text{and} \quad q_{ie} < \ell.$$

The proof follows directly from the RELEASE branch and the ingestion invariant. Initially there are no keys. A valid new record can add only its own planned view key; an invalid or duplicate record latches a fault and cannot produce a clean ledger. Induction over arrivals therefore preserves the association between stored keys and accepted evidence for the current item and epoch. Finalization tests the full key set and the permits before entering RELEASE. The result is a property of the supervisor under its interface assumptions. It gives no bound on defect-classification error and does not ensure that a declared physical identity is truthful.

An action request should include the item ID, epoch, requested route, evidence digest, decision version, and a unique command token. A matching acknowledgment closes that request; a timeout, mismatched item, or repeated token returns control to a defined recovery procedure. These handshake semantics are specified here but are not exercised by the image replay. A manufacturing execution system may store the closed decision record, while a PLC controls station sequencing and the independent safety system governs motion permission. *Fig. 1* locates these interfaces.

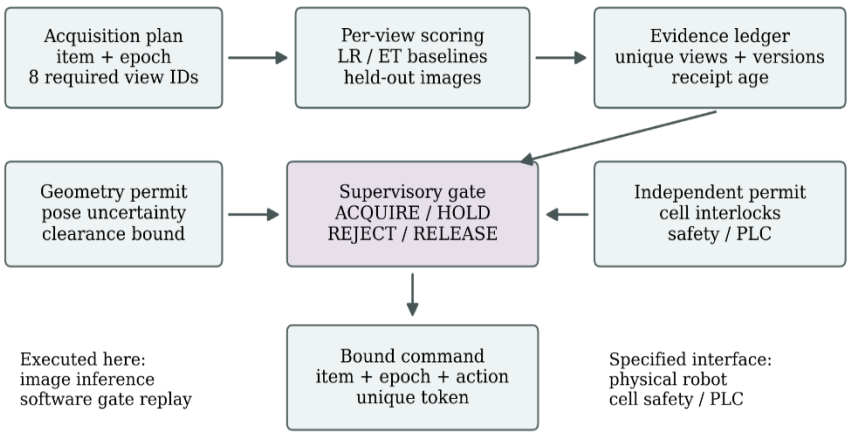


Fig. 1. Perception-to-disposition architecture.

Recorded-image inference and the supervisory gate were executed in software. The physical robot, PLC connection, command acknowledgment, and cell permits are specified integration interfaces. A quality RELEASE is not a safety permission.

3.2 | Coverage, Uncertainty, and Decision Cost

Suppose a policy observes one predetermined view v_0 and releases whenever that image is classified negative. Even a perfect local image classifier has defective-item escape fraction:

$$E_{\text{first}} = \frac{1}{n_+} \sum_{i:Y_i=1} 1\{y_{iv_0} = 0\},$$

where n_+ is the number of defective items. The error arises from unobserved surfaces rather than an incorrect local prediction. If a view is chosen uniformly from the m_i recorded views of item i, and r_i of those views show a defect, its expected escape fraction is:

$$\mathbb{E}[E_{\text{one}}] = \frac{1}{n_+} \sum_{i:Y_i=1} \left(1 - \frac{r_i}{m_i}\right).$$

No independence assumption between views is needed. Under maximum pooling and a fixed binary threshold, adding a valid view cannot change a REJECT into a RELEASE. A completeness requirement can additionally replace an unresolved release by HOLD. These are monotonic properties of the decision rule; they are not claims that extra views always improve a noisy classifier's specificity. More views can also accumulate false alarms on normal items. Probability aggregation requires stronger assumptions than score pooling. If $p_{iv} = P(A_{iv} \mid \mathcal{E}_{ie})$ are conditional probabilities of view-defect events given the same accumulated evidence, the union obeys:

$$\max_v p_{iv} \;\le\; P\left(\bigcup_v A_{iv} \mid \mathcal{E}_{ie}\right) \;\le\; \min\left(1, \sum_v p_{iv}\right)$$

The upper bound is a conditional union bound. A product formula would additionally require conditional independence. Per-image classifier outputs conditioned on separate images are not automatically the common-evidence marginals in this expression. Accordingly, the experiment does not substitute the LR or ET outputs into this probability bound. The distinction prevents an operational score from acquiring an unsupported probabilistic interpretation. For comparison, consider a genuinely calibrated item-defect probability p, escape cost c_E, false-rejection cost c_R, and hold cost c_H. A simple one-step loss model is:

$$L_{\text{release}}(p) = c_E p, \qquad L_{\text{reject}}(p) = c_R(1-p), \qquad L_{\text{hold}}(p) = c_H.$$

Minimizing these losses gives a release region below $\min\{c_R/(c_E + c_R), c_H/c_E\}$ and a rejection region above $\max\{c_R/(c_E + c_R), 1 - c_H/c_R\}$. With costs 100, 1, and 0.2, respectively, those boundaries are 0.002 and 0.8. This constructed example explains why consequences and calibration matter. It does not justify the empirical score thresholds 0.1 and 0.5, which were chosen as transparent fixed settings rather than estimated economic optima. Reinspection value, queue capacity, inspection time, and delayed escape cost would require a sequential decision model.

3.3 | Geometry and the robot Execution Permit

A disposition route may be fixed by fixtures, whereas targeted reinspection or manipulation requires a reliable relation between image coordinates and robot coordinates. Let T_{AB} map coordinates from the source frame B into the target frame A. For an eye-in-hand arrangement, the camera-to-base transform satisfies:

$$T_{BC} = T_{BE} T_{EC}, \qquad \tilde{p}_B = T_{BC} \tilde{p}_C.$$

Here B, E, and C denote robot base, end effector, and camera; the tilde denotes homogeneous coordinates. The product is standard rigid-body geometry. A real implementation must estimate the camera intrinsics, hand/eye transform, robot pose error, and any fixture uncertainty using measurements appropriate to the cell. The present archive does not contain those measurements. For a fronto-parallel point at known depth z, focal length f, and principal point (c_x, c_y), a simplified projection model is:

$$p_C = \left[z(u - c_x)/f,\ z(v - c_y)/f,\ z\right]^T, \qquad J_{uv} = \begin{bmatrix} z/f & 0 \\ 0 & z/f \\ 0 & 0 \end{bmatrix},$$

with small independent pixel error, translation error, and rotation-vector error, first-order propagation at a nominal identity orientation gives:

$$\Sigma_B \approx J_{uv}\Sigma_{uv}J_{uv}^T + \Sigma_t + [p_C]_\times \Sigma_\theta [p_C]_\times^T.$$

The skew matrix represents the rotational lever arm. The full point vector, including depth, belongs in that term. Using only its lateral offset would severely understate the effect of rotation in the designed example below. A general calibrated pose requires rotated Jacobians, depth-error terms, and any cross-covariances; the simplified formula is not a substitute for them. Assume a valid three-dimensional error ellipsoid with matrix Σ_B, radius parameter ρ, and nominal target $\hat{p}_B$. The target remains inside a permitted half-space $h^T p \le b$ for every error in that ellipsoid when,

$$h^T \hat{p}_B + \sqrt{\rho\, h^T \Sigma_B h} \le b.$$

To verify the expression, write an admissible error as $\Sigma_B^{1/2} w$ with $\| w \|_2 \le \sqrt{\rho}$. Cauchy–Schwarz bounds its projection onto h by the square-root term, and equality is attained in the aligned direction. The statement is a deterministic support-function result for the stated ellipsoid. If the pose error were exactly Gaussian with the stated covariance, choosing $\rho = \chi^2_{3,0.95}$ would assign 95% probability to that ellipsoid. Estimated, biased, correlated, or non-Gaussian errors require their own validation. A point-clearance condition also omits tool extent, swept volume, obstacles, and stopping behavior; those remain responsibilities of the motion and safety layers.

4 | Public-Data Protocol and Software Implementation

The fine-annotation KolektorSDD archive and its official three-fold split archive were downloaded from the dataset provider. Archive URLs and SHA-256 hashes are supplied in the companion. Direct enumeration found 399 JPEG images, 399 annotation masks, and 50 item folders. Forty-nine items had eight recorded views, and kos21 had Part0 through Part6, with Part7 absent. Original files were 500 pixels wide and 1235–1285 pixels high. This measured height range differs slightly from the website summary and is retained here to describe the actual analyzed archive. Masks yielded 52 positive and 347 negative image labels. Forty-eight items had one positive view and two had two.

The experiment used the provider's three disjoint test folds. Training contained 33, 33, and 34 items; testing contained 17, 17, and 16 items, respectively. All views of an item stayed on one side of every training/test boundary, and each image received exactly one held-out prediction. The script asserted these conditions before fitting. This prevents the direct leakage that would occur if correlated views of the same physical item were randomly distributed between training and testing. It does not produce an independent cross-factory test.

Each grayscale image was resized to 256×640 pixels using bilinear interpolation and scaled to the interval [0, 1]. This intentionally economical baseline resolution differs from the dataset authors' recommended benchmark resolution. Thus, even with the official item folds, the study is not an exact reproduction of their segmentation protocol. No masks, bounding boxes, item IDs, view IDs, or fold IDs were provided as model features. The masks were used solely to derive image labels and inspect dataset composition.

A fixed extractor produced 106 scalar features. Nine quantiles at 0, 1, 5, 25, 50, 75, 95, 99, and 100 percent, plus mean and standard deviation, summarized raw intensity; three Gaussian-background-minus-image residuals with standard deviations 2, 8, and 24 pixels; three grayscale-closing-minus-image residuals with rectangular sizes 3 × 21, 21 × 3, and 9 × 9; and Sobel gradient magnitude. These eight groups supplied 88 features. For each closing residual, thresholds 0.05, 0.10, and 0.20 additionally supplied foreground-area fraction and largest connected-component fraction, giving 18 further features. Connected components used the SciPy default two-dimensional connectivity. The design emphasizes dark local discontinuities and texture rather than claiming general defect invariance.

The first baseline was standardized, L2-regularized Logistic Regression (LR), with $C = 1$, balanced class weights, and a maximum of 3000 iterations. Standardization was fitted only on the training fold. The second was an extremely randomized-tree ensemble (ET), following the model family of Geurts et al. [28], with 400 trees, square-root feature sampling, minimum leaf size two, balanced class weights, and one worker. Scikit-learn supplied both implementations [29]. Random seed 71 was fixed. No hyperparameter search, threshold fitting, feature selection on test labels, or pretrained feature model was used. A class-weighted model's positive-class output is called a score throughout.

Table 4. Fixed experimental protocol and reporting units.

Component	Fixed Setting	Evaluation / Boundary
Archive	399 images; 52 visible defects; 50 defective items	All items have ≥1 positive view; no normal item cohort
Outer folds	Official 3 folds; test items 17 / 17 / 16	Test images 135 / 136 / 128; positives 18 / 18 / 16
Image input	Grayscale; bilinear 256 × 640; intensity / 255	No annotation pixels or IDs supplied as features
Features	106 fixed contrast, morphology and gradient summaries	No test-label feature selection
LR	Standard scaler; C = 1; balanced; max_iter = 3000	Training-fold scaler only; seed 71
ET	400 trees; sqrt features; min leaf 2; balanced	One worker; seed 71; no parameter search
Uncertainty summary	2000 item-bootstrap resamples; seed 7101	Fixed out-of-fold models; no refitting
Disposition	Upper score 0.5; lower score 0.1; plan views 0–7	Fixed thresholds; no probabilistic interpretation

Image AUROC and average precision were computed from the 399 pooled held-out scores. At score threshold 0.5, sensitivity, specificity, balanced accuracy, and confusion counts were also calculated. Pooled metrics can differ from a simple average of fold metrics, particularly when fold models produce differently scaled scores. Both pooled and fold results are retained in the companion. For descriptive uncertainty, 2000 bootstrap replicates sampled the 50 item IDs with replacement and retained all recorded views belonging to each sampled item.

Percentile intervals followed the bootstrap framework described by Efron and Tibshirani [30]. The models were not refitted in these resamples. These intervals describe variation in the fixed out-of-fold predictions under item resampling; they do not include training-algorithm instability or uncertainty from other factories.

Three disposition policies were then applied to each item's held-out scores. First-view binary used Part 0 and threshold 0.5. All-view binary used the maximum score over all available views with the same threshold. All-view gated used the same maximum, upper threshold 0.5, lower threshold 0.1, and an explicitly assumed plan requiring view IDs 0–7 before release. The eight-view requirement is a replay contract, not an inference that a missing archive image represented a real acquisition failure. Part labels determine a reproducible chosen order; no acquisition timestamps were supplied. Different views depict different surface regions, so this replay does not simulate repeated independent examinations of the same defect. For the 50 known defective items, the principal quantities are:

$$E = \frac{N_{release,+}}{50}, \qquad H = \frac{N_{hold,+}}{50}, \qquad R = \frac{N_{reject,+}}{50}, \qquad E + H + R = 1.$$

Here an escape is a quality RELEASE of an item known to be defective. HOLD is unresolved, not a correct rejection. Because no normal physical items are present, item-level false rejection, specificity, balanced accuracy, production prevalence, and economic utility cannot be estimated. Image-level specificity refers only to non-defective surface images, including views of defective items.

The reference gate was exercised separately from the learned-score comparison. Every recorded view was assigned score zero to expose the RELEASE path rather than letting a high defect score trivially block it. A clean trace and nine adverse conditions were instantiated for each of the 50 items: Missing final view; replacement of the final view with a duplicate first view; wrong item; old epoch; expired receipt; wrong model version; wrong calibration version; denied cell permit; and denied geometry permit. Finalization used local time 1, receipt time 0, and age limit 10 in arbitrary replay-time units; the expired record had receipt time −20. These values define software test conditions, not measured station timing. Six score-boundary cases and three invalid-score cases additionally checked branch behavior.

A separate geometry fixture assumed $z = 0.6$ m, $f = 800$ pixels, nominal $p_C = (0.04,0,0.6)$ m, independent isotropic pixel standard deviation 0.7 pixel, translation standard deviation 2 mm, and rotation-vector standard deviation 0.2°. It generated 100,000 samples with seed 7102, applied the exact rotation-vector transform to each perturbed camera point, and compared the empirical covariance with the first-order expression. Known depth, independence, zero bias, Gaussian inputs, and identity nominal orientation were deliberately simplifying assumptions. Neither the fixture parameters nor its conclusions were estimated from KolektorSDD robot poses.

5 | Results

Fig. 2 shows where positive labels occur across the recorded surfaces. Six items had a visible defect in Part0. Consequently, a perfect classifier of that first image would release the other 44 defective items, an 88% escape fraction for that policy. Selecting one recorded view uniformly per item gave an expected oracle escape fraction of 86.96%. These values quantify the coverage limitation without relying on a trained classifier. They should not be interpreted as the failure rate of an actual Kolektor inspection process; the one-view release policy is a deliberately incomplete counterfactual.

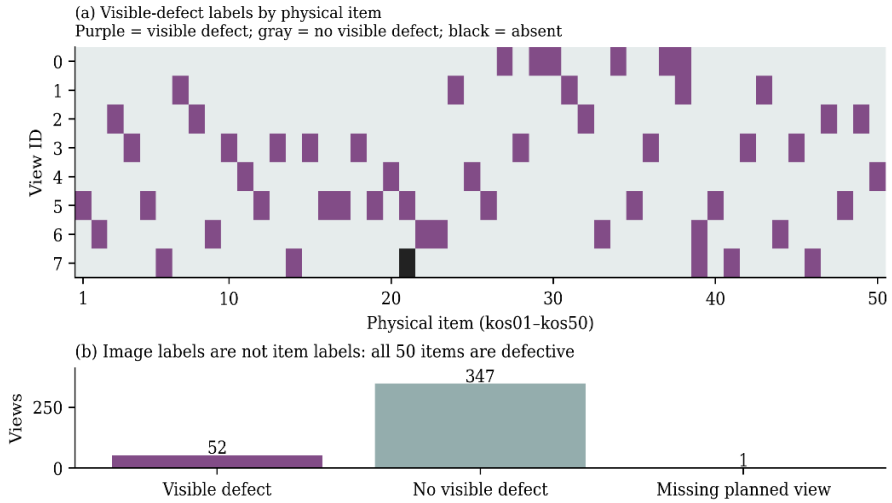


Fig. 2. Recorded visible-defect labels for the 50 physical items.

The black cell is the absent Part7 of kos21 under the eight-view replay plan. Negative image labels coexist with positive item labels because the images expose different surfaces. Plot generated from the downloaded masks; no source image or published figure was reproduced.

LR achieved AUROC 0.968 and average precision 0.888. ET achieved AUROC 0.960 and average precision 0.894. The item-bootstrap AUROC intervals were 0.943–0.989 for LR and 0.917–0.988 for ET. The paired ET-minus-LR AUROC interval, −0.036 to 0.011, did not support a clear ranking advantage. LR's higher sensitivity at threshold 0.5 came with more false-positive image decisions, while ET's higher specificity came with more missed positive images. *Table 5* and *Fig. 3* show why selecting a model from its AUROC alone would conceal the operating-point tradeoff.

Table 5. Image-level results from pooled held-out predictions.

Model	AUROC (95% CI)	AP (95% CI)	Sensitivity / Specificity	Balanced Accuracy	TP / FP / FN / TN
LR	0.968 (0.943–0.989)	0.888 (0.817–0.949)	0.846 / 0.951	0.899	44 / 17 / 8 / 330
ET	0.960 (0.917–0.988)	0.894 (0.825–0.951)	0.712 / 0.997	0.854	37 / 1 / 15 / 346

Confidence intervals resampled physical items and kept fitted fold models fixed. Sensitivity and specificity use threshold 0.5.

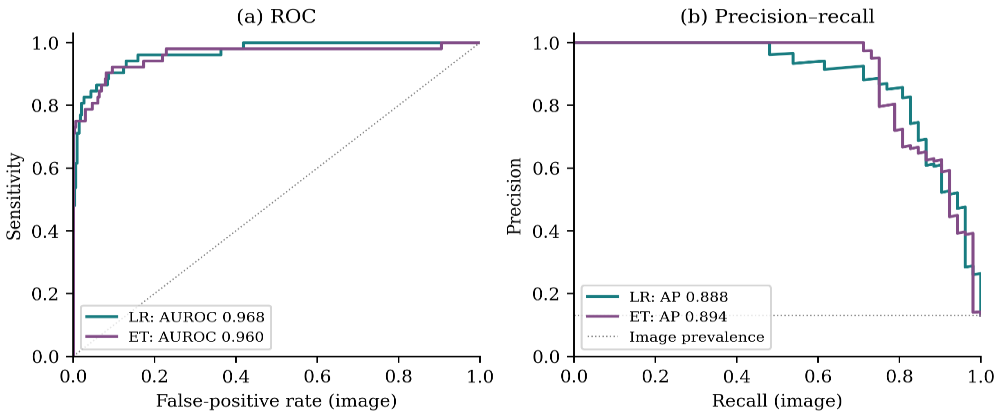


Fig. 3. ROC and precision–recall curves on 399 held-out image scores.

The precision reference is the image positive fraction, 52/399; it is not a production defect prevalence. Average precision uses scikit-learn's non-interpolated recall-weighted definition.

Both first-view policies released 44 defective items and rejected six. This numerical agreement with the first-view oracle does not mean that either classifier was perfect: One false-positive first view can compensate for one missed positive first view in the aggregate count. Complete confusion counts and per-image scores are therefore retained. Moving to all available views reduced LR escapes from 44 to six and ET escapes from 44 to fourteen. The aggregate maximum also accumulated false alarms on negative views, so the number of rejected items was not simply the number of true-positive images.

With the fixed abstention interval, LR released one item, held five, and rejected 44. ET released none, held fourteen, and rejected 36. For LR, the observed escape fraction fell from 12% under all-view binary to 2% under the gated policy; for ET it fell from 28% to 0%. The associated holds were 10% and 28%. A coverage-only ablation at the binary threshold produced the same counts as all-view binary in this particular cohort, because the incomplete item already exceeded the rejection threshold. Thus, the additional reductions in *Table 6* arose from abstention on low-to-intermediate item scores, while the explicit completeness check remained necessary to handle future low-score missing-view cases. The fault replay tests precisely that branch.

Table 6. Item disposition from held-out scores.

Model	Policy	REJECT	HOLD	RELEASE	Escape Fraction
Oracle	First view only	6	0	44	88%
LR	First view / binary	6	0	44	88%
LR	All available / binary	44	0	6	12%
LR	All / gated	44	5	1	2%
ET	First view / binary	6	0	44	88%
ET	All available / binary	36	0	14	28%
ET	All / gated	36	14	0	0%

All 50 items are defective. RELEASE therefore means an observed escape; HOLD remains unresolved. No normal-item operating characteristic can be inferred.

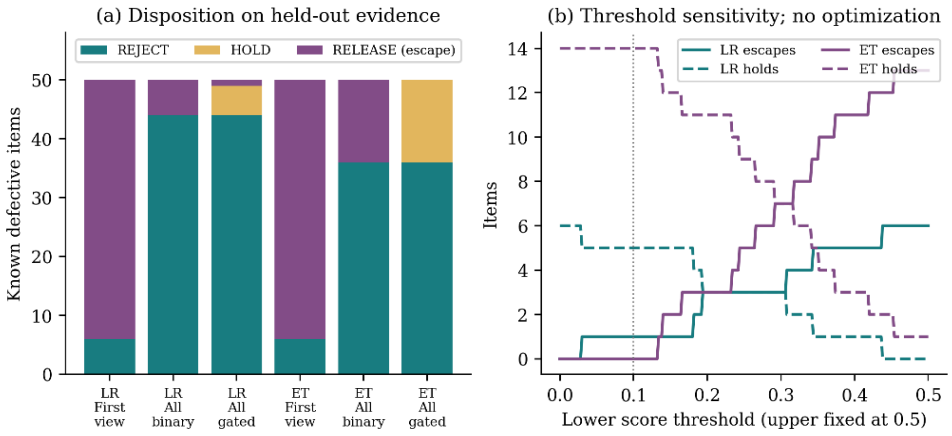


Fig. 4. Disposition counts and sensitivity to the lower release threshold.

The upper threshold remains 0.5. The vertical line marks the preselected lower threshold 0.1. The sweep is descriptive; no setting was selected from its test-cohort optimum. Higher lower thresholds increase release opportunities and can increase defective escapes.

The absence of an ET escape among these 50 items does not establish zero future risk. All items came from one dataset, the trained models overlapped in their training sets across folds, and no normal-item sample constrained the cost of abstention. Even under a simpler prospective independent-binomial design, zero events in 50 trials would leave a two-sided exact 95% upper bound of approximately 7.1%. That calculation illustrates the limited strength of a zero count; it is not presented as a calibrated confidence guarantee for this cross-validated replay.

The clean zero-score traces yielded 49 RELEASE decisions and one HOLD for the incomplete item. All 450 adverse traces ended in HOLD. A naive received-message counter would have counted eight messages in the duplicate replacement condition, but the unique-view ledger latched a duplicate fault. Wrong-item and old-epoch records likewise could not satisfy current-epoch evidence. At the score boundaries, 0 and 0.099999 produced RELEASE with complete valid evidence, 0.1 and 0.499999 produced HOLD, and 0.5 and 1 produced REJECT. NaN, −1, and 1.1 were invalid scores and produced HOLD. *Table 7* reports the constructed cases without interpreting their chosen frequencies as real fault probabilities.

Table 7. Executed supervisory fault replay with all perception scores set to zero.

Trace Family	Changed Condition	HOLD	RELEASE
Clean	Recorded roster; all scores zero	1	49
Missing	Final recorded view removed	50	0
Duplicate	Final record replaced by first view ID	50	0
Wrong item	One record names another item	50	0
Old epoch	One record uses epoch 70 instead of 71	50	0
Expired	One receipt age 21 exceeds limit 10	50	0
Wrong model	One model version differs	50	0
Wrong calibration	One calibration version differs	50	0
Cell permit denied	Independent permit false	50	0
Geometry permit denied	Geometric permit false	50	0

Each row comprises 50 traces, one per physical-item view roster. A blocked release is a software behavior, not a measured physical stop.

The geometry fixture produced position RMS error 4.621 mm. The empirical covariance differed from the first-order prediction by 0.864% in relative Frobenius norm. The nominal 95% three-dimensional ellipsoid contained 94.952% of simulated errors. These values support the local approximation for the selected small-error fixture. They do not identify the errors of a physical camera–robot chain. The ellipsoid's required clearances along the x, y, and z axes were 8.228, 8.237, and 5.605 mm, respectively. *Fig. 5* shows how increasing rotational uncertainty or camera distance enlarges the lateral clearance even when pixel uncertainty remains fixed.

Table 8. Designed geometry fixture and numerical checks.

Quantity	Value	Meaning
Camera point / focal length	(0.04, 0, 0.6) m / 800 px	Known depth; nominal identity orientation
Input standard deviations	0.7 px; 2 mm; 0.2°	Independent Gaussian pixel / translation / rotation errors
Samples / seed	100,000 / 7102	Exact rotation-vector Monte Carlo
Position RMS	4.621 mm	Simulated Euclidean position error
Relative covariance error	0.864%	Frobenius error against first-order model
95% ellipsoid coverage	94.952%	Simulated errors inside chi-square ellipsoid
Axis clearances x / y / z	8.228 / 8.237 / 5.605 mm	Support radius using chi-square(3) = 7.8147

All input uncertainties are assumptions; all output statistics were computed from the supplied simulation.

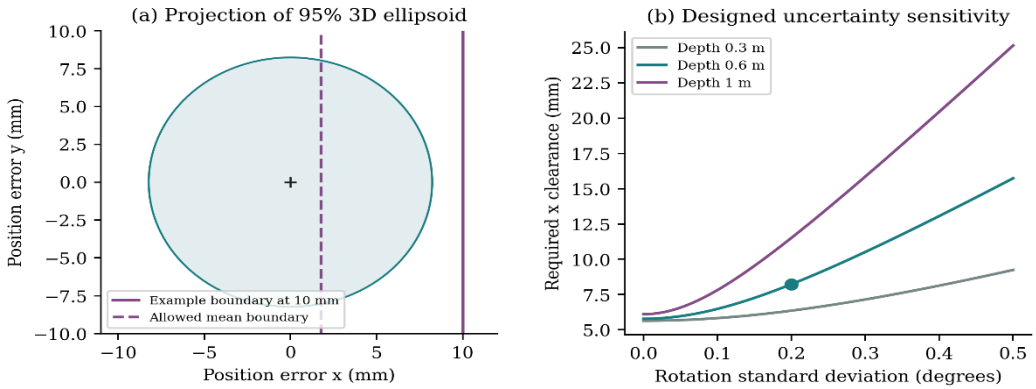


Fig. 5. Uncertainty-aware geometric permit.

Panel (a) projects the 95% three-dimensional ellipsoid onto the x–y plane; the 10 mm half-space boundary is an illustrative design choice. Panel (b) varies the assumed rotational standard deviation and depth while holding pixel and translation uncertainties fixed. This is a constructed model analysis, not measured robot accuracy or collision validation.

6 | Discussion and Integration Implications

The central finding is that an image metric and an item-release claim are different objects of evaluation. On this dataset, a locally perfect first-view decision could still release 88% of defective items. A classifier improvement cannot repair a surface that was never observed. The appropriate integration target is therefore an item-level acquisition and evidence contract, with perception evaluated inside that contract. This perspective is useful for fixed-camera multi-surface inspection, robot-carried cameras, and stations where reorientation exposes previously hidden surfaces.

The gate also makes the price of caution visible. ET's zero observed escapes required fourteen unresolved items, whereas LR released one defective item and left five unresolved. Without normal physical items, neither outcome establishes an acceptable production policy. A future prospective study should enroll normal and defective items under the same acquisition plan, preserve item and batch identities, and freeze thresholds before evaluation. It should measure false rejection, HOLD burden, review time, and final adjudication alongside escape counts. A method that simply routes a difficult cohort to HOLD can have value as an escalation component, but cannot be evaluated as a fully autonomous final inspector.

The study deliberately retained weakly parameterized, transparent classifiers. The 106-feature representation captured useful local contrast structure, but resizing may suppress narrow defects and the feature design is tailored to dark discontinuities on these surfaces. Stronger segmentation or anomaly models should replace the score module in further work without relaxing the evidence requirements. Pixel localization would be necessary to justify targeted reinspection or defect-directed manipulation. The present models provide image scores only; the geometry fixture is not fed by a learned defect localization output. Linking those two components remains an integration experiment rather than an achieved capability.

Evidence validity and perceptual uncertainty also differ. A wrong epoch is a deterministic protocol violation under the declared interface; a low score on a complete valid item can still be a perceptual mistake. The fault experiments establish that the implemented gate responds to the listed violations. They cannot establish that the acquisition system assigns true physical identities, that clocks have acceptable transport delay, or that hashes authenticate trustworthy sensor content. Receipt age bounds only the time since the supervisor received a record. A deployed cell needs bounded acquisition-to-receipt delay or synchronized acquisition timestamps with independently validated clock assumptions.

A useful production contract would start an epoch when an item is positively located at the fixture, define the required view roster, bind exposures and poses to that epoch, and close the ledger before issuing a disposition request. A pending acknowledgment should prevent an item ID from being recycled prematurely. Model or calibration updates should start new epochs, preserving the old decision record. Any permissible evidence repair should be explicit and logged. These choices make failures diagnosable, but their timing and availability consequences need a real station experiment.

The geometric analysis reinforces the separation of permits. A high-confidence defect classification does not ensure a safe or accurate manipulator pose. At a 0.6 m lever arm, a small angular uncertainty contributed substantially to lateral position uncertainty. Conversely, a geometrically accurate target does not establish a justified quality release. The camera-to-robot model should eventually use measured correlated calibration residuals, depth uncertainty, tool geometry, and task-specific clearance, with physical checks that cover the intended operating envelope. The point ellipsoid in this paper is one input to such a process, not a complete motion certificate.

Throughput requires a further measurement layer. A serial eight-view cycle includes acquisition and settling for each view, inference, aggregation, communication, and disposition motion. Pipelining can overlap some of these operations, but queueing and HOLD handling can dominate tail latency. The approximately 73 seconds taken by the present feature extraction, fitting, and bootstrap script is a development-run duration that includes statistical resampling; it is not per-item inference time or line throughput. No robot cycle rate, latency guarantee, energy reduction, or economic return is inferred from it.

The evidence base has several remaining limits. There are only 50 physical items, one industrial surface family, and no independent batch or site holdout. The bootstrap conditions on fixed fitted models. The replay uses an assumed view plan and chosen event order, rather than recorded station timestamps. The fault catalogue is finite and intentionally synthetic. The pose inputs are assumed independent Gaussian quantities. These limits identify the next validation steps: Prospective mixed-quality items, alternative lighting and batches, a measured acquisition ledger, robot/PLC execution traces, and a task-specific safety assessment. They do not diminish the demonstrated logical distinction between a negative image and a releasable item.

7 | Conclusion

Vision-based robotic inspection benefits from treating release as an evidence-dependent supervisory action. The proposed architecture requires complete planned view coverage for release and binds every accepted observation to an item, epoch, model, calibration, and local evidence-age check. Public-data experiments showed that strong image discrimination coexisted with severe defective-item escapes under a one-view policy. Maximum aggregation and an explicit HOLD region reduced observed escapes while exposing the corresponding unresolved workload. Executed fault traces verified the specified release gate, and an independent numerical fixture illustrated the contribution of pose uncertainty to geometric permission. The resulting contribution is a reproducible perception-to-disposition study with a defined path to physical integration, rather than a claim of an already validated autonomous production cell.

Data and code Availability

The analysis uses the fine-annotation KolektorSDD archive and the provider's three-fold split file [3], [4]. The provider distributes the dataset under CC BY-NC-SA 4.0 and specifies separate contact for commercial use. The companion includes download URLs, archive hashes, an image manifest, derived features, held-out scores, item decisions, fault traces, geometry results, scripts, figures, and bibliographic verification records. Original image and mask archives are not redistributed. The companion preserves the dataset attribution and license information for derived records. All numerical results in this paper can be regenerated with the documented Python environment; no GPU or physical robot is required for those software studies.

References

[1] Szeliski, R. (2011). *Computer vision: Algorithms and applications* (Vol. 1). Springer, London. http://mesh.brown.edu/engn1610/szeliski/03-imageprocessing.pdf

[2] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. https://aikosh.indiaai.gov.in/static/deep+learning+ian+goodfellow.pdf

[3] Tabernik, D., Šela, S., Skvarč, J., & Skočaj, D. (2020). Segmentation-based deep-learning approach for surface-defect detection. *Journal of Intelligent Manufacturing*, *31*(3), 759–776. https://doi.org/10.1007/s10845-019-01476-x

[4] Visual Cognitive Systems Laboratory. (2019). *Kolektor surface-defect dataset (KolektorSDD)*. https://www.indatlas.com/datasets/kolektor-surface-defect-dataset-kolektorsdd/

[5] Bergmann, P., Fauser, M., Sattlegger, D., & Steger, C. (2019). MVTec ad: A comprehensive real-world dataset for unsupervised anomaly detection. *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 9584–9592). IEEE. https://doi.org/10.1109/CVPR.2019.00982

[6] Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., & Gehler, P. (2022). Towards total recall in industrial anomaly detection. *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 14298–14308). IEEE. https://doi.org/10.1109/CVPR52688.2022.01392

[7] Batzner, K., Heckler, L., & König, R. (2024). EfficientAD: Accurate visual anomaly detection at millisecond-level latencies. *2024 IEEE/CVF winter conference on applications of computer vision (WACV)* (pp. 127–137). IEEE. https://doi.org/10.1109/WACV57701.2024.00020

[8] Zou, Y., Jeong, J., Pemula, L., Zhang, D., & Dabeer, O. (2022). SPot-the-difference self-supervised pre-training for anomaly detection and segmentation. *Computer vision - ECCV 2022* (pp. 392–408). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-20056-4_23

[9] Wang, C., Zhu, W., Gao, B. B., Gan, Z., Zhang, J., Gu, Z., ..., & Ma, L. (2024). Real-iad: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. *2024 IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 22883-22892). IEEE. https://doi.org/10.1109/CVPR52733.2024.02159

[10] Zhu, W., Wang, L., Zhou, Z., Wang, C., Pan, Y., Zhang, R. (2025). Real-iad 3D: A real-world 2D/pseudo-3D/3D dataset for industrial anomaly detection. *2025 IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 15214–15223). IEEE. https://doi.org/10.1109/CVPR52734.2025.01417

[11] Heckler-Kram, L., Neudeck, J. H., Scheler, U., König, R., & Steger, C. (2026). The MVTec AD 2 dataset: Advanced scenarios for unsupervised anomaly detection. *International Journal of Computer Vision*, *134*(4), 175. https://doi.org/10.1007/s11263-026-02743-0

[12] Silva, C. A. de S., & Paladini, E. P. (2025). Smart machine vision system to improve decision-making on the assembly line. *Machines*, *13*(2), 98. https://doi.org/10.3390/machines13020098

[13] Puttero, S., Verna, E., Genta, G., & Galetto, M. (2026). Collaborative robots for quality control: An overview of recent studies and emerging trends. *Journal of Intelligent Manufacturing*, *37*(4), 1345–1381. https://doi.org/10.1007/s10845-025-02600-w

[14] Chow, C. (1970). On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, *16*(1), 41–46. https://doi.org/10.1109/TIT.1970.1054406

[15] El-Yaniv, R. (2010). On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, *11*(5), 1605–1641. https://doi.org/10.5555/1756006.1859904

[16] Geifman, Y., & El-Yaniv, R. (2017). Selective classification for deep neural networks. *Advances in neural information processing systems* (Vol. 30, pp. 1–10). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2017/file/4a8423d5e91fda00bb7e46540e2b0cf1-Paper.pdf

[17] Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. *Proceedings of the 34th international conference on machine learning* (pp. 1321–1330). PMLR. https://proceedings.mlr.press/v70/guo17a.html

[18] Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PloS One*, *10*(3), e0118432. https://doi.org/10.1371/journal.pone.0118432

[19] Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*(11), 1330–1334. https://doi.org/10.1109/34.888718

[20] Tsai, R. Y., Lenz, R. K. (1989). A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Transactions on Robotics and Automation*, *5*(3), 345–358. https://doi.org/10.1109/70.34770

[21] Park, F. C., & Martin, B. J. (1994). Robot sensor calibration: Solving AX=XB on the Euclidean group. *IEEE Transactions on Robotics and Automation*, *10*(5), 717–721. https://doi.org/10.1109/70.326576

[22] Chaumette, F., & Hutchinson, S. (2006). Visual servo control. I. Basic approaches. *IEEE Robotics & Automation Magazine*, *13*(4), 82–90. https://doi.org/10.1109/MRA.2006.250573

[23] Macenski, S., Foote, T., Gerkey, B., Lalancette, C., & Woodall, W. (2022). Robot operating system 2: Design, architecture, and uses in the wild. *Science Robotics*, *7*(66), eabm6074. https://doi.org/10.1126/scirobotics.abm6074

[24] Lamport, L. (2019). Time, clocks, and the ordering of events in a distributed system. *Concurrency: The works of leslie lamport* (pp. 179–196). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3335772.3335934

[25] Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., …, & Dennison, D. (2015). Hidden technical debt in machine learning systems. *Advances in neural information processing systems*. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2015/file/86df7dcfd896fcaf2674f757a2463eba

[26] International Organization for Standardization. (2025). *ISO 10218-1:2025: Robotics — safety requirements — Part 1: Industrial robots. ISO*. https://www.iso.org/standard/73933.html

[27] International Organization for Standardization. (2025). *ISO 10218-2:2025: Robotics — safety requirements — Part 2: Industrial robot applications and robot cells*. https://www.iso.org/standard/73934.html

[28] Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, *63*(1), 3–42. https://doi.org/10.1007/s10994-006-6226-1

[29] Fabian, P. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825. https://doi.org/10.5555/1953048.2078195

[30] Efron, B., & Tibshirani, R. J. (1994). *An introduction to the bootstrap*. Chapman and Hall/CRC. https://doi.org/10.1201/9780429246593