A Proposed Framework for Robust Visual Product Identification in Dynamic Environments

Klaudia Éva Zeleny[1], Tamás Ruppert[2] and Andrea Kő[1]

[1] Department of Information Systems, Corvinus University of Budapest, Fővám tér 13-15, Budapest, 1093, Hungary; klaudia.zeleny@stud.uni-corvinus.hu, andrea.ko@uni-corvinus.hu

[2] Department of System Engineering, University of Pannonia, Egyetem str. 10., Veszprém, 8200, Hungary; ruppert.tamas@mk.uni-pannon.hu

Abstract: *Reliable visual product identification remains a challenging task in dynamic indoor environments, where variability in viewpoint, illumination, and object placement limits the applicability of conventional scanning-based approaches. This is particularly relevant in high-mix low-volume intralogistics settings, where incomplete observations and inconsistent acquisition conditions are common. This paper proposes a modular framework for robust visual product identification based on machine-readable codes. The approach formulates identification as a system-level reasoning problem, integrating detection, geometric normalization, decoding, validation, recovery, and temporal aggregation within a unified architecture. Frame-level observations are treated as uncertain evidence, which is accumulated and constrained at the sequence level in order to support reliable identification decisions. The individual components draw on established computer vision and decoding techniques; the contribution lies in their system-level integration with explicit uncertainty handling and temporal reasoning, rather than in new detection, decoding, or fusion algorithms. The proposed framework emphasizes robustness through structured integration of visual evidence rather than isolated recognition performance, providing a foundation for perception-driven identification in visually dynamic environments. A preliminary evaluation on a controlled test set, shows that detection performance is close to ceiling, while frame-level decoding alone succeeds on only about one third of ground-truth instances (35.6%); aggregating evidence across frames within the proposed sequence-level reasoning layer raises identification success to 73.5%, providing initial evidence that the proposed architecture improves reliability over frame-level decoding alone.*

Keywords: computer vision; intralogistics; DataMatrix; product localization; high-mix low-volume

1 Introduction

Visual product identification plays a central role in modern intralogistics systems, where physical objects must be continuously linked to digital representations in order to support tracking, inventory management, and operational decision-making. In warehouse environments, this identification is typically based on machine-readable codes such as barcodes, QR codes, or DataMatrix symbols, which provide a simple and cost-effective mechanism for associating physical items with digital records [1]. At the same time, increasing system complexity and the need for real-time operational visibility have strengthened the role of data-driven and perception-based approaches in intralogistics systems [2]. While identifier-based tracking remains widely adopted, its effectiveness depends on reliable acquisition of visual information. In structured environments, this is typically achieved through manual scanning or controlled sensing setups. However, in dynamic indoor environments characterized by variability in layout, illumination, and object placement, such approaches become increasingly difficult to apply consistently.

These challenges are particularly pronounced in high-mix low-volume (HMLV) intralogistics environments, where product diversity, low repetition rates, and frequent reconfiguration of storage conditions limit the applicability of standardized workflows. In such environments, objects may appear under varying viewpoints, distances, and lighting conditions, and visual identifiers may be partially occluded or embedded in cluttered scenes. As a result, reliable visual identification cannot be reduced to a single-step detection or decoding problem, but must explicitly account for environmental variability and uncertainty in visual observations [3].

In practical warehouse operations, these limitations often manifest as discrepancies between physical and digital inventory states. Location updates typically rely on manual scanning or operator input, which may be delayed or omitted under time pressure. Consequently, operators frequently need to search for items despite the presence of machine-readable identifiers, leading to inefficiencies and reduced trust in system data. This highlights the need for identification approaches that operate continuously and robustly under realistic acquisition conditions. Existing research on visual code recognition has achieved significant progress in detection and decoding methods, particularly through deep learning–based object detectors. However, these approaches are often evaluated under controlled conditions and typically treat detection and decoding as largely independent stages. As a result, the interaction between perception uncertainty, validation logic, and temporal observation remains insufficiently addressed, particularly in settings where visual information is incomplete or inconsistent across frames [4].

This work addresses this gap by formulating visual product identification as a system-level reasoning problem. Instead of treating decoding as an isolated task, the proposed approach considers identification as the result of structured integration of detection, geometric normalization, decoding, recovery, validation and temporal

aggregation. Within this perspective, frame-level outputs are interpreted as uncertain observations that must be accumulated and constrained in order to support reliable identification decisions. The main contribution of the paper is the formulation of a modular framework for robust visual product identification in dynamic environments. This contribution is architectural and methodological rather than algorithmic: detection, decoding, and geometric normalization rely on established computer vision techniques, and the novelty lies in how these components are integrated, together with explicit validation, recovery, and temporal reasoning mechanisms, into a single system-level architecture. The framework explicitly separates frame-level perception from sequence-level reasoning and introduces mechanisms for uncertainty handling, including validation against prior knowledge and recovery strategies for degraded observations. In addition, the framework's evaluation is designed around controlled acquisition variability, enabling analysis of how component-level uncertainty propagates to system-level identification outcomes.

The remainder of the paper presents the proposed framework and its architectural structure, reports the corresponding frame-level and sequence-level results, and closes with a discussion of implications and limitations, with conclusions drawn at the end.

2 Related Work

A structured literature review was conducted to map existing research at the intersection of computer vision, visual code recognition, and intralogistics applications. The search was performed in the Scopus database, focusing on peer-reviewed journal articles and conference papers published from 2019 onwards. The search logic combined four conceptual dimensions:

(i) Computer vision and deep learning
(ii) Two-dimensional code recognition (e.g., QR and DataMatrix)
(iii) Logistics and warehouse-related applications
(iv) High-mix low-volume environments

As no publication was found that simultaneously covered all four dimensions, the analysis was based on partial overlaps between these domains. The final literature corpus includes both review papers and research articles, enabling a combined synthesis of methodological trends and applied system designs.

2.1 Visual code recognition and industrial perception

Visual code recognition in industrial contexts is predominantly formulated as a two-stage problem, consisting of localization and decoding. In recent literature, deep

learning-based object detectors have become the dominant approach for localization, while decoding is typically performed using classical decoding algorithms.

Across both review papers and applied studies, convolutional neural networks, particularly modern one-stage detectors such as the YOLO family are widely adopted due to their favorable trade-off between localization accuracy and computational efficiency [5][6]. These models are frequently used for detecting small visual markers, barcodes, or DataMatrix symbols in industrial inspection, warehouse automation, and mobile scanning scenarios [7-10]. Two-stage detectors such as Faster R-CNN remain relevant in applications where higher localization precision is required [4][11].

A consistent architectural pattern is the use of hybrid pipelines, in which deep learning-based detection is followed by classical decoding libraries such as libdmtx or ZBar [4][12]. In such systems, neural networks are primarily responsible for identifying candidate regions of interest, while symbolic interpretation is performed by deterministic decoders. This separation reflects both the maturity of classical decoding methods and the reliability requirements of industrial applications.

Despite strong performance reported in controlled settings, several studies highlight that detection and decoding accuracy may degrade significantly under challenging conditions such as occlusion, motion blur, or non-frontal viewpoints [13]. As a result, visual code recognition is often treated as a frame-level task, where each image is processed independently, with limited consideration of temporal consistency across observations.

2.2 Vision systems in warehouse and intralogistics environments

Computer vision has been increasingly applied in warehouse automation, inventory tracking, and robotic systems. Existing approaches address tasks such as object detection, localization, navigation, and scene understanding, often integrated into broader perception pipelines [14][15].

In these contexts, perception systems are frequently evaluated in structured or semi-controlled environments, including predefined layouts, fixed camera placements, or constrained motion trajectories. For example, warehouse vision systems and UAV-based scanning solutions are often validated using static images or offline video sequences, with limited representation of continuous operation under dynamic conditions [7][16].

Similarly, localization and mapping approaches, particularly those based on SLAM and visual odometry, rely on geometric and probabilistic frameworks, sometimes augmented with learning-based components [17][18]. While these methods

demonstrate robustness in structured environments, their applicability to cluttered and dynamically changing warehouse settings remains constrained.

A recurring limitation across studies is the reliance on simplified assumptions regarding scene structure, sensor placement, and environmental variability. Static cameras, predefined viewpoints, and controlled illumination conditions are commonly assumed, whereas real intralogistics environments involve frequent layout changes, occlusions, and heterogeneous object configurations.

2.3 Dataset and evaluation limitations

The reviewed literature reveals several recurring limitations related to dataset design and evaluation protocols. Many industrial vision datasets are relatively small, task-specific, and often proprietary, reflecting practical constraints in data acquisition [11][12]. In the context of barcode and DataMatrix recognition, datasets are frequently collected under favorable imaging conditions and may not adequately represent real-world variability [13].

To address limited data availability, several studies rely on synthetic data generation or simulation-based augmentation [19][20]. While these approaches increase dataset size and variability, they introduce challenges related to domain transfer and realism.

Evaluation protocols are typically based on standard computer vision metrics such as mean Average Precision (mAP), precision, and recall [5]. Although these metrics provide consistent benchmarking at the component level, they may not fully capture system-level performance in operational environments. In particular, frame-level evaluation does not reflect the temporal dynamics of mobile acquisition scenarios, where objects may be observed across multiple frames under varying conditions.

Furthermore, datasets explicitly representing HMLV intralogistics environments are scarce. Existing datasets rarely capture the combination of high product diversity, dynamic scene configurations, partial occlusion, and variable lighting conditions characteristic of such settings. This limits the ability to systematically evaluate perception systems under realistic operational variability.

2.4 Research gap

The reviewed literature demonstrates significant progress in deep learning-based object detection, visual code recognition, and warehouse perception systems. However, the intersection of these domains, particularly in the context of dynamic HMLV intralogistics remains only partially explored. Table 1 summarizes the main limitations identified across the reviewed literature, the requirements they imply, and the specific components of the proposed framework (Section 4) that address them.

Table 1

Mapping from limitations identified in the reviewed literature (Section 2) to the requirements they imply and the framework components that address them.

Limitation in existing literature	Implied requirement	Addressed by
Detection, decoding, and perception treated as separate problems, mostly evaluated under controlled or semi-structured conditions	A unified architecture linking and jointly evaluating perception stages under variable acquisition conditions	End-to-end pipeline integrating detection, geometric normalization, decoding, and validation
No explicit mechanism for handling uncertainty from occlusion, distortion, or incomplete observations	A dedicated pathway to recover identification when primary decoding degrades or fails	Recovery stage: RANSAC-based geometric reconstruction and similarity-based matching
Evaluation predominantly performed at the frame level; identification reliability across multiple observations not considered	Reasoning over temporally extended observations rather than single frames	Sequence-level reasoning: track association, evidence accumulation, and the dominance/sufficiency decision rule
Lack of an integrated framework combining validation, recovery, and temporal reasoning; temporal evidence accumulation not systematically addressed	A system-level architecture in which validation, recovery, and temporal aggregation operate jointly rather than in isolation	Full proposed pipeline, explicitly separating frame-level perception from sequence-level decision-making
Datasets rarely reflect HMLV variability and heterogeneity; standardized benchmarks for such conditions are largely absent	A dataset design with systematically controlled, HMLV-representative variability to support stratified evaluation	Macro-scene-based dataset design with controlled illumination, viewpoint, occlusion, and blur variability

These observations suggest that the primary limitation of existing approaches is not the absence of effective detection or decoding methods, but the lack of an integrated framework that combines these components with validation, recovery, and temporal reasoning mechanisms; in particular, the interaction between environmental variability, decoding uncertainty, and system-level identification reliability remains insufficiently characterized in the literature.

This gap motivates the system-level perspective adopted in this work, where visual product identification is formulated as a multi-stage process integrating frame-level perception with sequence-level reasoning in order to support robust operation in HMLV-inspired environments.

3 Methodology

The proposed framework is developed from a system-level perspective in which data acquisition, model design, and decision logic are treated as interconnected elements of a unified visual identification process. Rather than focusing on isolated algorithmic components, the methodology emphasizes how these elements jointly contribute to reliable identification under visually variable conditions. The design is motivated by mobile visual acquisition scenarios, where a camera continuously observes the environment while moving through space. In such settings, the same visual identifier may appear across multiple frames under changing viewpoints, distances, and illumination conditions. As a result, identification cannot be reliably based on single-frame observations alone, but must account for temporal variability and partial visibility. This motivates a fundamental distinction between frame-level perception and sequence-level identification. Frame-level processing performs detection, geometric normalization, and decoding attempts on individual images, while sequence-level reasoning integrates these observations over time. The methodology therefore, treats identification as an evidence accumulation process, where frame-level outputs are interpreted as uncertain observations rather than final decisions.

The design of the framework is guided by several key principles. First, robustness is prioritized over idealized performance, emphasizing stable behavior under visually degraded conditions. Second, end-to-end identification reliability is considered more relevant than isolated component metrics, reflecting the integrated nature of the pipeline. Third, environmental variability is treated as a design factor, enabling systematic analysis of performance under controlled acquisition conditions. Fourth, modularity is incorporated through clearly defined processing stages, supporting extensibility and comparative evaluation. Finally, the system explicitly distinguishes between frame-level perception and sequence-level identification, enabling identification reliability to emerge from accumulated observations.

The architectural design is aligned with established data-driven development practices, particularly the CRISP–DM framework. While CRISP–DM describes an iterative lifecycle rather than a runtime system, the proposed architecture reflects its principles through the separation of data design, model development, and evaluation logic. This alignment supports methodological transparency, reproducibility, and systematic traceability between design choices and observed system behavior. These methodological principles are operationalized in the pipeline architecture presented in Section 4, where perception, validation, and temporal reasoning are integrated into a unified identification process.

Overall, the proposed system-level architecture provides a structured and extensible framework for investigating robust DataMatrix identification under visually variable acquisition conditions. The modular structure facilitates targeted

experimentation, comparative evaluation, and future extensions, such as the integration of additional sensing modalities or higher-level warehouse management interfaces.

4 Proposed Pipeline

The proposed framework follows a modular, pipeline-oriented architecture designed to support robust visual product identification under dynamic indoor imaging conditions. The system integrates frame-level perception and sequence-level temporal reasoning in order to reduce identification uncertainty arising from environmental variability. An overview of the framework is shown in Figure 1.

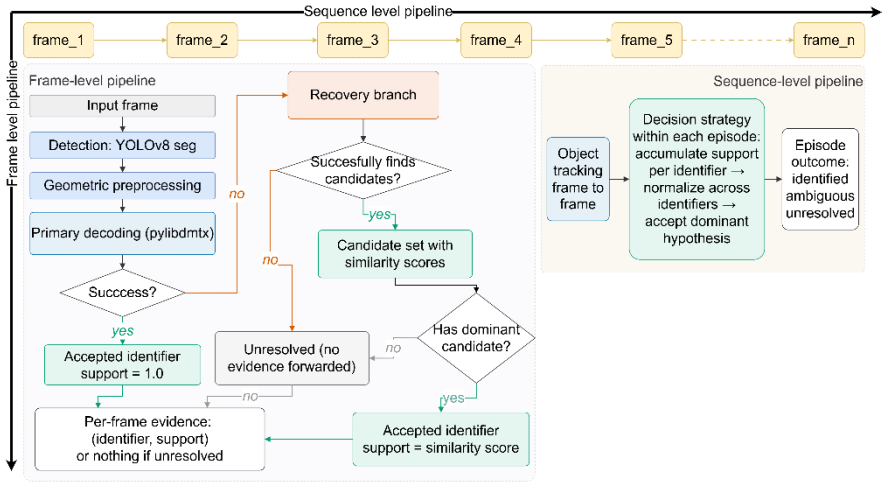

Figure 1. Overview of the proposed end-to-end DataMatrix identification pipeline.

At a conceptual level, the framework operates on two interacting processing layers. The frame-level layer transforms individual image frames into structured observations through detection, geometric normalization, decoding attempts, validation, and recovery mechanisms. The sequence-level layer aggregates these observations over time to support deferred identification decisions.

The input consists of image frames captured by a mobile camera in dynamic indoor environments characterized by variability in distance, viewpoint, illumination, motion, and partial occlusion. The framework does not assume controlled acquisition conditions.

4.1 Frame-level processing

The first stage performs visual **symbol detection** using a deep learning–based segmentation model. The model localizes candidate symbol regions and provides geometric information required for downstream normalization. Detection is implemented using a YOLO-based instance segmentation model [21]. Deep learning object detectors, particularly from the YOLO family, are widely employed in visual code recognition pipelines due to their favorable balance between localization accuracy and computational efficiency, as reflected in prior studies. Within the proposed framework, segmentation-based localization is selected instead of bounding-box detection in order to provide geometrically consistent symbol boundary estimates required for downstream normalization and decoding. For each input frame, the detection module produces segmentation masks and associated confidence estimates for candidate symbol regions. These outputs serve as geometric constraints for subsequent processing stages and directly influence the reliability of identification decisions at the sequence level.

Detected regions are forwarded to a geometric **preprocessing** stage, where each predicted segmentation mask is converted into a validated, ordered four-corner quadrilateral rather than a bounding box or a minimum-area rectangle, so that the perspective distortion of the symbol plane is preserved rather than discarded. The mask polygon's convex hull is simplified via iterative polygon approximation until a four-point quadrilateral is obtained; the resulting corners are ordered consistently and validated for convexity, minimum size, and consistency with the source mask area. A candidate region for which no reliable quadrilateral can be constructed is dropped entirely rather than forced into an approximate shape. Given a validated quadrilateral, a planar projective transformation (homography) [22] rectifies the symbol region into a fixed-size, canonical square crop. A quiet-zone margin is included around the rectified symbol, since decoding is empirically highly sensitive to its presence. The resulting rectified representation provides a standardized input for subsequent decoding and contributes to limiting geometric error propagation within the overall identification pipeline.

A primary **decoding** attempt is then performed using *pylibdmtx* [23], a standard ECC200-compliant [24] DataMatrix decoding library. The decoding output is treated as a candidate hypothesis rather than a final identification decision. Following geometric normalization, each rectified symbol region is processed by this decoding engine, which internally performs grid sampling, module binarization, timing pattern verification, and error-correction decoding. The decoding process produces either a candidate identifier or a decoding failure. Given the variability of real imaging conditions, decoding outputs are not treated as final identification decisions. Instead, decoding is interpreted as an observation step within a broader identification framework. To ensure consistency with prior system knowledge, candidate identifiers are subjected to an explicit validation procedure against a reference set of known identifiers: a decoded value is accepted only if it

matches a known identifier, and rejected otherwise. Within the proposed architecture, primary decoding represents the most direct pathway to identification when sufficient visual quality is available. However, the system does not assume reliable per-frame decoding. Instead, decoding outcomes are incorporated into a structured evidence accumulation process at the sequence level. Observations that are rejected by validation are forwarded to recovery mechanisms designed to handle degraded or incomplete symbol representations.

In realistic acquisition conditions, primary decoding may fail due to insufficient symbol resolution, motion-induced degradation, partial occlusion, or photometric distortions. To address such scenarios, the proposed framework incorporates a **recovery stage**, activated only when primary decoding is rejected by validation. The recovery stage first attempts a more robust geometric reconstruction of the symbol than the one-shot rectification used in the primary path: the DataMatrix's characteristic L-shaped finder pattern is detected, and its two solid boundary edges are localized using RANSAC line fitting [25] on the crop's binarized or edge-detected boundary pixels, rather than the coarser mask-derived corners used for primary decoding. This is expected to be more resilient to segmentation noise, partial occlusion, and blur, precisely the conditions under which primary decoding is most likely to have already failed. The intersection of the fitted boundary lines provides the symbol orientation and code boundaries, from which rotation and perspective correction are applied, the DataMatrix grid dimensions are estimated, and the binary cell matrix is reconstructed and rendered as a clean digital DataMatrix image. This reconstructed image is passed to a secondary decoding attempt, and any resulting identifier is validated against the reference set in the same way as a primary decoding output.

When the geometric reconstruction is too degraded to support a reliable secondary decode, for example when the symbol is heavily occluded or severely blurred, the same reconstructed representation is instead compared against a reference library of known digital DataMatrix renderings using image-based similarity measures. This comparison acts as a fallback rather than an alternative primary pathway: it does not require a fully reconstructed grid to succeed, and it ranks candidate identifiers by a similarity score, kept distinct from detection confidence, that reflects how closely the observed pattern matches each reference code. Rather than forwarding this ranking as-is, a frame-level acceptance rule is applied: the leading candidate is forwarded as the frame's contribution to sequence-level reasoning only if it is accepted as *dominant* by the following rule, reused unchanged at the sequence level (Section 4.2). Let $v_1 \geq v_2 \geq \ldots \geq v_k$ denote the ranked candidate values under consideration (here, similarity scores against the reference library). Candidate v_1 is accepted as dominant if:

$$\text{Dominant}(v_1, \ldots, v_k) = \begin{cases} \text{true}, & k = 1 \\ \frac{v_1 - v_2}{v_1} \geq \theta_M, & k = 2 \\ \frac{v_1 - v_2}{v_1 - v_k} \geq \theta_Q(k), & k = 3 \end{cases} \quad (1)$$

and rejected otherwise. For $k = 1$, a single surviving candidate is accepted directly: with no competitor to be judged against, even a weak match constitutes non-trivial evidence to be weighed later at the sequence level. For $k = 2$, dominance reduces to a relative margin between the two candidates, since no larger reference set exists to judge the gap against. For $k \geq 3$, dominance is evaluated using Dixon's Q test [26], a small-sample statistic for detecting a single outlier in a ranked set, comparing the gap to the runner-up against the full spread of the candidate set; $\theta_Q(k)$ is taken from Dixon's standard, sample-size-dependent critical-value table rather than fixed by hand. θ_M and the significance level underlying $\theta_Q(k)$ remain configurable parameters, not yet calibrated, consistent with the current implementation status of the temporal decision logic (Section 4.2). Algorithm 1 summarizes the procedure. When Equation (1) rejects dominance, for instance because multiple candidates are too close to distinguish, the frame contributes no evidence and is treated as unresolved at the frame level.

Algorithm 1 Dominance test over ranked candidate values $v_1 \geq \ldots \geq v_k$

if $k = 1$ **then**
    return accept v_1
else if $k = 2$ **then**
    $M \leftarrow (v_1 - v_2) / v_1$
    return accept v_1 **if** $M \geq \theta_M$, **else** reject
else
    $Q \leftarrow (v_1 - v_2) / (v_1 - v_k)$
    return accept v_1 **if** $Q \geq \theta_Q(k)$, **else** reject
end if

Outputs of the recovery stage, an accepted identifier from either secondary decoding or similarity-based matching, are propagated to the sequence-level reasoning layer together with primary decoding outcomes and detection confidence signals.

The output of the frame-level pipeline consists of structured **per-frame evidence**: at most one accepted identifier together with a support value, either 1.0 for primary or secondary decoding, or the accepted similarity score for a dominant recovery candidate, or an unresolved symbol observation when neither decoding nor a dominant recovery candidate is available.

4.2 Sequence-level reasoning

Frame-level observations corresponding to the same physical symbol are temporally associated to form **symbol tracks**. Evidence accumulation is performed at the track level, enabling identification decisions to be deferred until sufficient evidence becomes available. While the preceding components operate at the level

of individual frames, reliable visual product identification in dynamic environments requires reasoning over temporally extended observations. In mobile acquisition scenarios, the same DataMatrix symbol may be observed across multiple consecutive frames under varying imaging conditions, including changes in viewpoint, distance, illumination, and motion blur. As illustrated in Figure 1, each frame is processed independently by the frame-level pipeline, producing structured observations that are subsequently integrated at the sequence level.

The objective of temporal integration is to aggregate heterogeneous visual evidence associated with the same physical symbol over time. Instead of treating frame-level decoding outcomes as final decisions, the system maintains evolving symbol hypotheses whose reliability is progressively refined as new observations become available. This design reflects the practical characteristic of mobile vision systems, where decoding success is strongly influenced by transient imaging conditions.

Sequence-level processing begins with the temporal association of detected symbol candidates across consecutive frames. Each frame's detected quadrilaterals are associated with tracks carried from the previous frame using a weighted combination of three geometric cues rather than overlap alone, so that association remains stable under the scale and viewpoint changes typical of mobile acquisition. Let Q_t denote a track's last-known quadrilateral and Q_o an observation quadrilateral in the current frame, with areas a_t, a_o and centroids c_t, c_o (the mean of the four corners). Writing $\mathrm{IoU}(Q_t,Q_o)$ for mask overlap, $\hat{d} = \|c_t - c_o\|_2 / \sqrt{((a_t+a_o)/2)}$ for the scale-normalized centroid distance, and $\Delta a = |a_o - a_t| / \max(a_o,a_t)$ for the relative change in area, the match score is:

$$\mathrm{score}(Q_t, Q_0) = w_{\mathrm{iou}}\mathrm{IoU}(Q_t, Q_0) + w_c \max\left(0.1 - \frac{\hat{d}}{\rho_c}\right) + w_a \max\left(0.1 - \frac{\Delta a}{\rho_a}\right) \quad (2)$$

with $w_{iou} = 0.5$, $w_c = 0.3$, $w_a = 0.2$, $\rho_c = 2.0$, and $\rho_a = 1.0$. This positions the tracker as a lightweight variant of IoU-based tracking-by-detection approaches such as SORT [27], replacing their Kalman-filter motion model and Hungarian assignment with two additional geometry-only cues and greedy assignment — reasonable simplifications given the small number of concurrently visible, static symbols typical of this setting. The specific weight and threshold values reflect reasonable defaults set during development rather than fitted or ablated parameters, consistent with the systematic ablation of pipeline design choices identified as future work in Section 4. A (track, observation) pair is matched only if $\mathrm{score}(Q_t,Q_o) \geq \tau_{match} = 0.4$, pairs above threshold are assigned greedily by descending score with each track and observation used at most once, and a track is expired after 5 consecutive unmatched frames. A track's visible frames are consolidated into a chronological episode whenever consecutive visible frames are separated by at most 15 missing frames, so that brief interruptions, such as a momentary occlusion, do not split a single physical instance into multiple episodes.

Each episode accumulates structured evidence: the per-frame support values forwarded by frame-level processing (Section 4), together with detection

confidence and geometric measurements retained for diagnostic purposes. For each candidate identifier i observed within an episode, its support values are summed across all contributing frames into an accumulated support A_i, which is then normalized across all k candidate identifiers observed in the episode into $s_i = A_i / \Sigma_{j=1..k} A_j$, so that normalized support sums to one. The episode's identifier is accepted as the dominant hypothesis only if two conditions jointly hold: (i) *sufficiency*, the total accumulated evidence reaches a minimum floor:

$$\sum_{i=1}^{k} A_j \geq \theta_{\text{suff}} \tag{3}$$

and (ii) *dominance*, the same test introduced for the recovery branch (Equation (1), Algorithm 1), applied here to the ranked normalized support values $s_1 \geq \ldots \geq s_k$. Unlike the frame-level use of the dominance test in the recovery branch, the sequence level additionally requires sufficiency: an episode accumulates evidence over time, so premature acceptance from a single strong frame would forgo the benefit of further observations, whereas a recovery-branch decision is inherently a one-shot, single-frame judgment. This confidence-weighted voting strategy is one of the three temporal decision rules considered for the framework, alongside first-valid identification and unweighted majority voting; it is intended to allow the system to suppress spurious decoding results while reinforcing temporally stable interpretations. Figure 2 shows a worked example of this mechanism.

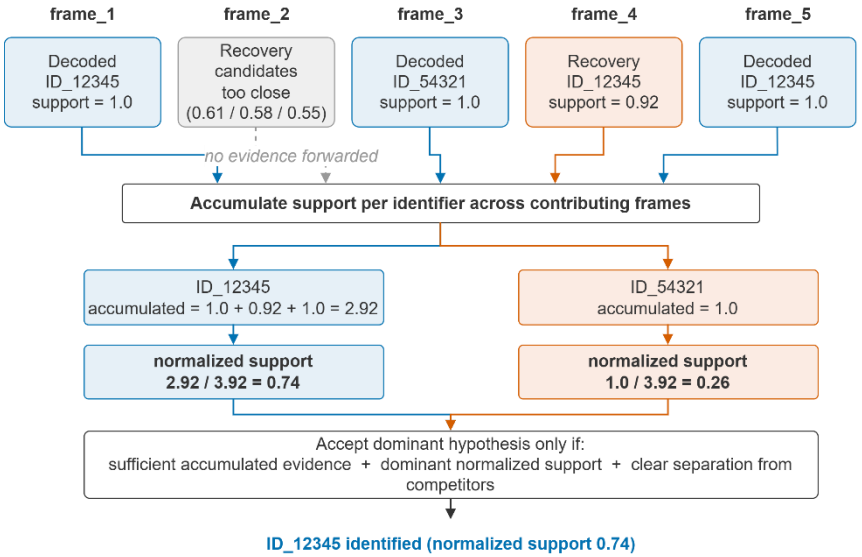

Figure 2. Worked example of the confidence-weighted voting mechanism. Three frames decode the same identifier directly (support 1.0), one contributes it via a dominant recovery candidate (support 0.92), one is unresolved and forwards no evidence, and one decodes a different identifier (support 1.0). Accumulating and normalizing these values yields a dominant, clearly separated hypothesis (0.74 normalized support), which is accepted as the episode's identifier.

Symbol tracks are initialized upon first observation and extended for as long as the tracker continues to associate new detections with them. An episode ends when the symbol is absent for longer than the configured gap tolerance, after which its accumulated evidence is passed to the temporal decision stage. The episode is then categorized as *identified* if conditions (i)–(ii) above are jointly satisfied, *ambiguous* if sufficiency (Equation (3)) holds but dominance (Equation (1)) does not, or *unresolved* if sufficiency itself does not hold.

From a system perspective, temporal integration constitutes a central robustness mechanism of the proposed framework. By explicitly separating frame-level perception from sequence-level decision making, the architecture enables identification reliability to emerge from cumulative visual evidence rather than isolated frame-level decoding outcomes.

4.3 Visual acquisition assumptions and evaluation context

The proposed identification pipeline is evaluated under visual acquisition conditions representative of mobile indoor sensing scenarios. Image data are captured using a handheld camera in an indoor environment, with DataMatrix codes physically attached to surfaces and objects at predefined locations, without fixed camera mounting or controlled lighting. Rather than assuming controlled imaging setups, such variability is treated as an inherent characteristic of the visual observation process and is explicitly incorporated into the evaluation logic of the system.

The dataset supporting the present study is conceived as an experimental instrument for controlled analysis of identification reliability rather than as a general-purpose benchmark. Visual scenes are organized into coherent acquisition contexts, referred to as *macro-scenes*, within which the physical configuration of symbols and surroundings remains fixed while acquisition parameters vary. The dataset comprises three macro-scenes, one per split, each defined by a distinct physical code layout and a disjoint code set, so that no code identifier or physical configuration is shared across splits; within each macro-scene, one video was recorded per illumination condition (F1–F3). Dataset partitioning into training, validation, and test splits is performed at the macro-scene level to prevent contextual leakage between splits and to enable a more realistic assessment of generalization behavior. The resulting split comprises 1216 training, 933 validation, and 922 test frames; the test split, used throughout Section 5, contains 1015 annotated DataMatrix instances. The three videos analyzed in the sequence-level evaluation (Section 5.3) correspond to the test macro-scene's three illumination conditions.

All visible DataMatrix instances are manually annotated with polygon-based segmentation masks rather than bounding boxes, normalized to image dimensions and stored in a format compatible with the YOLO segmentation training pipeline,

in order to support the geometry-aware preprocessing and decoding stages described in Section 4.

Environmental and object-level variability is recorded along four dimensions, summarized in Table 2 and used as grouping factors throughout Section 5: illumination, viewpoint, object visibility condition, and blur. Illumination is recorded at the frame level; viewpoint angle is derived automatically per object from the geometry of its detected quadrilateral; object condition combines a manually annotated occlusion flag with an automatically derived too-small classification based on a minimum pixel-size threshold; blur is annotated manually per object.

Table 2

Environmental and object-level variability dimensions used as grouping factors in Section 5.

Dimension	Category	Description
Illumination	F1	Good, homogeneous lighting, high contrast
	F2	Reduced, semi-dark lighting; ambient light only
	F3	Frontal light source; glare or reduced contrast
Viewpoint	S1 (Frontal)	Near-orthogonal viewing angle
	S2 (Oblique)	Rotated, oblique viewing angle
Object condition	Normal	Fully visible, unoccluded instance
	Occluded	Partially occluded instance
	Too small	Instance below a minimum pixel-size threshold
Blur	Blurred / Not blurred	Motion- or focus-induced image blur

This acquisition and dataset design perspective provides the methodological foundation for analyzing how component-level visual uncertainties propagate to episode-level identification outcomes. Consequently, the dataset serves as a controlled experimental context for investigating the robustness properties of the proposed system rather than as a standalone contribution.

4.4 System-level outcome

Final identification outcomes in the proposed framework are determined at the level of episodes rather than individual frames. The system categorizes each episode as *identified*, *ambiguous*, or *unresolved*. By retaining intermediate outputs across

pipeline stages, the architecture enables systematic analysis of how detection, geometric normalization, decoding reliability, and temporal aggregation jointly influence end-to-end identification performance.

Beyond symbolic recognition, episode-level identification events may also be interpreted as temporally grounded observations of product presence. In mobile acquisition scenarios, each resolved episode implicitly contains geometric and temporal information related to the viewing configuration of the sensing platform. While the present work does not implement a full spatial localization framework, this observation suggests that validated identification outcomes may serve as probabilistic indicators of product presence within a given physical region.

From a system design perspective, the proposed architecture therefore provides not only a perception pipeline but also a conceptual foundation for integrating visual identification with higher-level operational reasoning. In data-driven intralogistics systems, such observation-based identification signals may complement existing inventory tracking mechanisms by providing incremental updates under visually variable and dynamically evolving conditions.

Evaluation of the proposed framework follows the same frame-level and sequence-level structure. The results reported in Section 5 currently cover the detection and primary-decoding stages at the frame level, and, at the sequence level, an *ever_correct* criterion used as a lower-bound proxy for the *identified*/*ambiguous*/*unresolved* taxonomy pending implementation of the full temporal decision logic described in Section 4. Evaluation of the recovery branch, and a systematic ablation across the pipeline's design choices, remain directions for future work.

5 Results

The results reported in this section correspond to the primary-decoding pathway of the pipeline described in Section 4 (detection, crop extraction, primary decoding, and reference validation), evaluated both at the frame level and at the sequence level. The recovery branch, which is triggered only when primary decoding fails, has not yet been evaluated and is addressed separately in Section 5.4 as a direction for future work.

5.1 Detection performance

Detection performance was evaluated on a held-out test split comprising 922 frames and 1015 ground-truth DataMatrix instances, using the trained YOLO-based segmentation model described in Section 4. Table 3 reports precision, recall, F1, and mAP overall and by lighting condition, a frame-level grouping for which these

metrics are well defined, together with recall and mean matched IoU by object condition, viewpoint angle, and blur.

Table 3

Detection performance overall, by lighting condition, and by object condition, viewpoint, and blur.

Factor	Level	Precision	Recall	F1	mAP50	mAP50-95	Mean IoU	n
overall	–	0.98	0.99	0.98	0.98	0.93	0.97	1015
Lighting	F1	0.96	0.99	0.98	0.98	0.92	0.97	190
	F2	0.98	0.99	0.99	0.98	0.94	0.98	454
	F3	0.99	0.98	0.99	0.97	0.92	0.97	371
Object condition	Normal	1.00	0.99	1.00	0.99	0.94	0.97	863
	Occluded	1.00	0.95	0.97	0.94	0.91	0.98	150
	Too small	1.00	1.00	1.00	1.00	0.60	0.78	2
Viewpoint	Frontal	1.00	0.99	1.00	0.99	0.94	0.98	577
	Oblique	1.00	0.97	0.99	0.97	0.91	0.97	438
Blur	Blurred	1.00	0.98	0.99	0.98	0.93	0.97	738
	Not blurred	1.00	0.99	1.00	0.99	0.94	0.98	277

The gap between mAP50 and mAP50–95 reflects how detection quality changes as the overlap criterion between predicted and ground-truth masks is tightened. Figure 3 shows average precision as a function of the IoU matching threshold: AP remains close to 1.0 for thresholds up to approximately 0.8, then declines sharply toward IoU = 0.95. This indicates that the detector reliably localizes the correct region, while the predicted mask boundary is not pixel-perfect under the strictest overlap criteria.

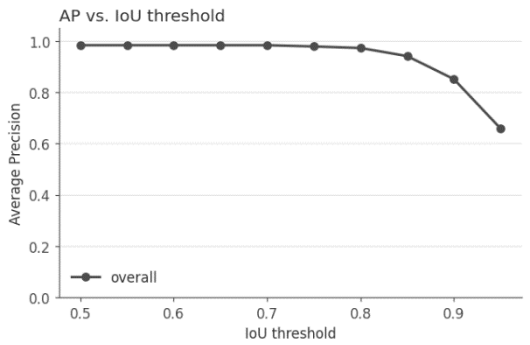

Figure 3. Detection average precision as a function of the IoU matching threshold.

Detection is therefore close to perfect and is largely unaffected by blur, occlusion, viewpoint, or lighting condition, with the small exception of reduced recall for occluded and oblique-viewpoint objects. Consequently, the decoding failures reported in the following subsection can be attributed to the decoding stage rather than to the detection stage.

5.2 Frame-level primary decoding results

Primary decoding was evaluated directly on ground-truth polygon crops from the same test split, which isolates the performance of the decoder itself from the quality of detection. A decoded value is counted as correct only if it matches the object's expected reference identifier (Section 4, reference validation). Table 4 reports the resulting decode rate, overall and broken down by blur, lighting condition, viewpoint angle, and occlusion.

Table 4

Primary decode rate on ground-truth crops, overall and by condition (test split, n=1015 ground-truth objects).

Factor	Level	Decode rate	n
Overall	–	35.6%	1015
Blur	Not blurred	68.6%	277
	Blurred	24.4%	738
Lighting	F1	29.0%	190
	F2	26.8%	454
	F3	51.6%	371
Viewpoint	Frontal (S1)	55.1%	577
	Oblique (S2)	11.9%	438
Object condition	Normal	42.4%	863
	Occluded	2.7%	150
Best case	F3, S1, not blurred, normal	100.0%	110

Blur is the single largest driver of primary decoding failure, with a larger effect than lighting condition, viewpoint angle, or occlusion individually. Under the most favorable combination of conditions, the decoder reaches its ceiling of essentially 100% decode rate, indicating that the losses observed above stem from acquisition conditions rather than from a fundamental limitation of the decoding algorithm itself.

5.3 Sequence-level results

Sequence-level evaluation was carried out on 34 episodes across three videos, automatically constructed by the sequence-tracking pipeline (Section 4.2) and annotated by a reviewer with each track's code identity and visibility conditions; each episode corresponds to one code's continuous ground-truth visibility span. Detection and primary decoding were run on every frame of each episode, and an episode is counted as successfully identified (*ever_correct*) if at least one frame within it produced a correct decode. This criterion serves as a proxy for the temporal decision logic described in Section 4 (first-valid, majority, or confidence-weighted voting), which has not yet been implemented; it establishes the ceiling that any such voting strategy could reach without acquiring evidence beyond what is already present in the sequence.

Table 5 reports the overall episode-level identification success rate, together with breakdowns by lighting condition and by whether the reviewer confirmed a fully visible frame within the episode.

Table 5

Episode-level sequence identification success rate (n = 34 episodes).

Breakdown	Category	Success rate	n
Overall	–	73.5%	34
Lighting	F1	69.2%	13
	F2	66.7%	12
	F3	88.9%	9
Has fully visible frame	Yes	95.5%	22
	No	33.3%	12

Nine of the 34 episodes never resolved. Inspection of the reviewer-recorded visibility notes shows that these cases are predominantly explained by occlusion, extreme viewing angle, or motion blur, rather than by episodes simply being too short to provide the decoder with an opportunity to succeed.

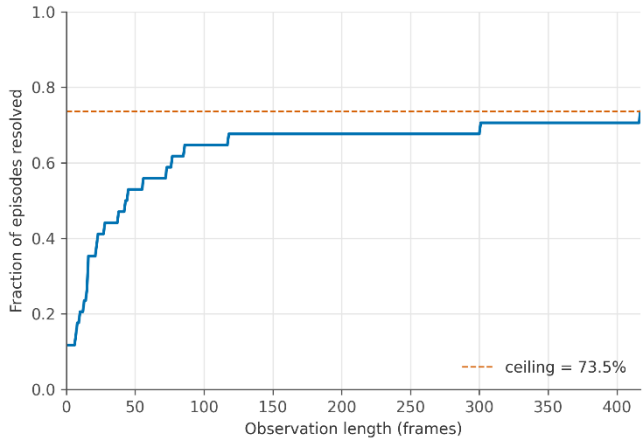

Figure 4. Cumulative fraction of episodes resolved as a function of observation length. The ceiling (dashed line) is the overall episode success rate reached once the full episode is observed.

Figure 4 shows how this accumulates as more frames are observed: only 11.8% of episodes are already resolved after a single frame, rising to 44.1% within 30 frames, and reaching the full 73.5% ceiling only once the entire episode is observed. This directly demonstrates the value of sequence-level reasoning over single-frame decisions: a substantial share of episodes that a single-frame system would report as unresolved are in fact identifiable given enough accumulated evidence over time.

Full per-frame processing of these episodes required 6901 frame-level decode attempts in total, which is computationally demanding for a system intended to run continuously. To quantify this cost, episode outcomes were recomputed while sampling only every 15th or every 30th frame, approximating a reduced processing budget. Table 6 shows that sampling every 30th frame, approximately one frame per second, reduces the number of processed frames by 96.7%, but lowers the overall episode success rate from 73.5% to 47.1%, with the cost distributed unevenly across lighting conditions.

Table 6

Trade-off between frame-sampling stride, computational cost, and episode-level identification success.

Stride (frames)	Frames processed	% of full cost	Episode success
1 (every frame)	6901	100.0%	73.5%
15	461	6.7%	58.8%
30	231	3.3%	47.1%

This trade-off motivates a cheaper alternative to uniform frame skipping: rather than processing or skipping frames independently of their content, a lightweight quality check could identify frames that are unlikely to decode before primary decoding is attempted, avoiding wasted decode attempts while retaining more of the identification accuracy than blind striding does. Section 5.4 presents a preliminary step in this direction.

5.4 Blur calibration as a future recovery-branch component

Motivated by the cost-accuracy trade-off identified above, we explored whether a low-cost, unsupervised sharpness classifier could identify frames that are unlikely to decode before primary decoding is attempted. The method operates directly on the rectified crop and does not require labeled training data: the grayscale intensity histogram of the crop is separated into two clusters using k-means, and the ratio between the depth of the valley separating the clusters and their peak heights, together with the proportion of dark pixels, is used to classify the crop as sharp or blurred. Figure 5 illustrates the method on a sharp and a blurred example, showing the crop, its intensity histogram with the detected peaks and valley, and the resulting pixel clustering.

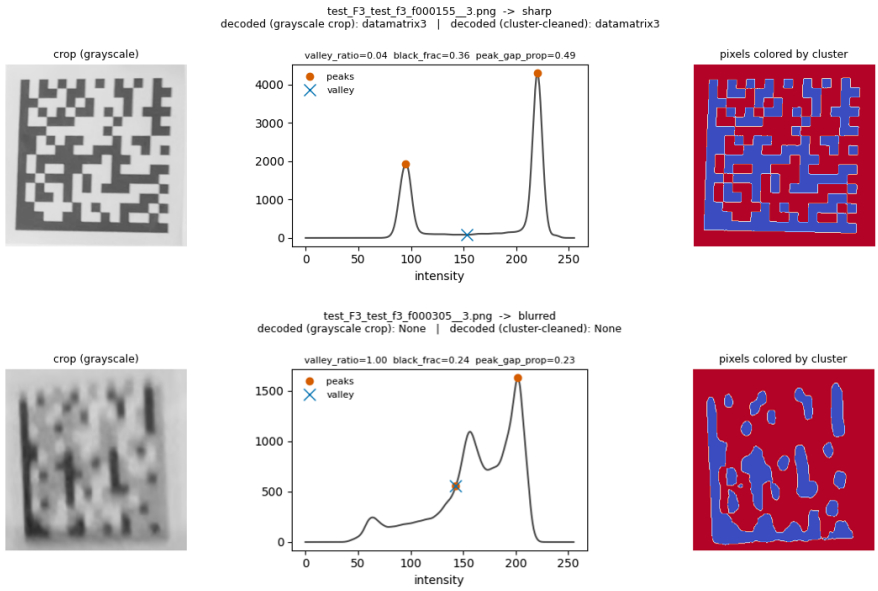

Figure 5. Sharp (top) and blurred (bottom) DataMatrix crop, with intensity histogram and pixel clustering used by the sharpness classifier.

On a small curated evaluation set of 24 manually labeled crops, this classifier reached 91.7% accuracy, with errors concentrated on borderline blurred cases

misclassified as sharp. On a further set of 10 visually ambiguous crops without ground-truth labels, the classifier labeled all cases as sharp, indicating that its current calibration is conservative and that evaluation on a larger and more diverse dataset is required before deployment.

These results are preliminary and are reported here as a direction for future work rather than as a validated pipeline component. Two integration points are envisaged. First, the classifier could act as a pre-decode gate within the frame-level pipeline, skipping primary decoding for frames that are unlikely to decode, which Section 5.3 suggests could reduce computational cost with a smaller accuracy penalty than uniform frame sampling. Second, the classifier could be used to trigger the recovery branch directly on detected blur, complementing its current activation solely on primary-decoding failure.

6 Discussion

The results reported in Section 5 show a clear separation between perception difficulty and decoding difficulty. Detection performance is close to ceiling under nearly all conditions (Table 3): the DataMatrix symbols are almost always found and localized correctly, regardless of blur, lighting, or viewpoint. Primary decoding performance follows a markedly different pattern, succeeding on only about a third of ground truth crops overall (Table 4), with this bottleneck driven predominantly by blur rather than by lighting or viewpoint individually. Sequence-level aggregation substantially narrows this gap: episodes are resolved correctly in roughly three out of four cases (Table 5), even though the underlying frame-level decode rate remains modest throughout. System reliability therefore does not derive from an unusually strong decoder, but from the accumulation of weak, individually unreliable frame-level evidence into a more robust sequence-level decision, consistent with the mechanism underlying the proposed framework.

Contextualizing these results against prior work is informative, subject to an important caveat. No publicly available benchmark covers detection, decoding, recovery, and sequence-level identification together under the kind of acquisition variability considered here, so the comparison that follows should be read as contextualization rather than a shared benchmark.

Table 7

Decode/recognition rates and recovery/temporal mechanisms reported in prior Data Matrix and barcode studies, contextualized against the present study. DM abbreviates Data Matrix. Reported rates not directly comparable across studies, due to differing datasets, acquisition protocols and code types.

Study	Domain	Reported decoding rate	Dataset conditions	Recovery / multi-frame use
Almeida et al. [4]	DM, self-localization	51.2% (458/895)	Real indoor scenes, cluttered, varying illumination and planes; no explicit blur factor	Recovery: none. Multi-frame: none (tracking proposed only to reduce latency)
Karrach & Pivarciová [28]	DM, classical geometric	95.6% (172/180)	Isolated code crops: synthetic plus web images plus industrial laser-marked codes; not full-scene detection	Recovery: none. Multi-frame: none
Karrach & Pivarčiová [29]	DM, classical geometric	99.1%, 100% at multi-scale	Isolated code crops, same three sources; method assumes a largely intact finder pattern	Recovery: cascaded geometric fallback. Multi-frame: none
Liao et al. [30]	DM, laser-marked metal	99.5% (vs. 8.5% libdmtx)	Isolated codes on aluminum, mobile phone, controlled lab lighting; single degradation type (metal glare and texture)	Recovery: unconditional CNN reconstruction. Multi-frame: none
Matuszczyk & Weichert [12]	DM, DPM polymer AM	86.0% (avg., corrected)	Close range (3–10 cm), limited angle (0 to ±30 degrees), low-contrast codes, controlled lighting	Recovery: unconditional encoding network. Multi-frame: none
Liu et al. [31]	DM, industrial defects	90.2% overall, 85–88% under defects	Curated single-defect-type categories (protrusion, interruption, rotation); not compounded degradation	Recovery: unconditional geometric reconstruction. Multi-frame: none
Rybakova et al. [32]	DM, semi-synthetic	90.3%	Semi-synthetic, code covers at least 25% of frame, ROI already given (no full-scene detection)	Recovery: cascaded geometric fallback. Multi-frame: none
Edula et al. [8]	QR code	33.5% rising to 50.3% with enhancement	Real photos with varied blur, glare, damage, perspective, and height	Recovery: unconditional super-resolution. Multi-frame: none
Muallim et al. [10]	QR / 1D barcode	15% to 68% (val), 0% to 90% (real-world)	Heavily augmented synthetic damage for validation; manually damaged real codes (writing, scratches, paint) for the real-world test	Recovery: failure-gated GAN restoration. Multi-frame: Kalman tracker (object continuity only)
Xu et al. (Lichao) [33]	1D barcode, video	78% rising to 89% (ever decoded)	Real warehouse video, but only 18 distinct barcodes total; 1D barcodes only	Recovery: none. Multi-frame: ever-decoded criterion only
Present study, frame level	DM, HMLV	**35.6%**	Blur, oblique viewpoint, occlusion, and illumination variation combined within the same test set, by design	Recovery: failure-gated (RANSAC + similarity). Multi-frame: N/A
Present study, sequence level	DM, HMLV	**73.5%**	Same	Recovery: failure-gated (RANSAC + similarity). Multi-frame: confidence-weighted accumulation

This comparison supports the central argument of the paper in one important respect. Every study that evaluates a plain decoder against the same decoder augmented with a failure-conditioned processing step reports a substantial improvement: Liao et al. [30] from 8.5% to 99.5%, Edula et al. [8] from 33.5% to 50.3%, Muallim et al. [10] from 15% to 68% and from 0% to 90%. This pattern recurs across independently developed methods and application domains,

supporting the premise that primary decoding alone is insufficient. More directly relevant to the present framework, none of the reviewed studies accumulates evidence across frames to resolve object identity. Where multiple observations of the same object are used at all, as in Almeida et al. [4], Muallim et al. [10], or in UAV-based factor-graph localization work [7] omitted from Table 7 because it reports a detection rather than a decoding metric, this serves latency reduction, geometric position refinement, or tracking continuity, rather than identification confidence. This gap is consistent across the reviewed literature, and the present study's own frame-to-sequence improvement from 35.6% to 73.5%, measured on the identical detector, decoder, and test conditions, constitutes the most direct available evidence that closing it is worthwhile, since this comparison is not confounded by cross-study differences in dataset or hardware.

The comparison also raises a legitimate question that warrants explicit consideration rather than omission. Several of the reviewed studies report decode rates of 85% to 99% using only classical geometric methods, on datasets that, while real, are narrower in scope than the present test set: a single defect type, a limited viewing angle, or a fixed close distance, rather than blur, oblique viewpoint, and occlusion combined within the same evaluation. This difference in acquisition difficulty is a plausible contributing factor to the lower frame-level rate observed in the present study. It does not, however, exclude the possibility that part of the observed gap reflects potential for improvement in the primary decoding and geometric preprocessing stages themselves, independent of the contribution of temporal reasoning. The recovery branch of the proposed framework has not yet been evaluated in this study; the improvements reported elsewhere are therefore cited as motivation for including a recovery mechanism within the architecture, not as validation of the specific implementation proposed here.

Consistent with this reasoning, several of the alternative geometric reconstruction techniques identified in this comparison address sub-problems closely related to the one the recovery branch is designed to solve. Hough-transform-based line fitting for the finder pattern [32], dashed timing-pattern edge localization guided by a two-level screening strategy [31], and CNN-based module-pair classification combined with graph-theoretic edge correction [30] each address the underlying task of reconstructing a decodable representation from a geometrically degraded DataMatrix symbol, using techniques distinct from the RANSAC-based boundary fitting employed here. None of these techniques has been incorporated into the present implementation, and none is claimed as part of the present contribution. They are noted as candidate directions for refining or extending the recovery branch in future work, alongside the systematic evaluation and design ablation already identified as necessary next steps.

Taken together, these results support the broader argument of the paper: that robustness under dynamic, HMLV-representative conditions is better pursued as a system-level evidence-integration problem than as a search for a marginally more accurate single-frame decoder, while also indicating that the primary-pipeline and

recovery-branch components of the system retain potential for independent improvement and validation.

Beyond the technical contributions discussed above, the proposed framework carries several anticipated operational implications for the intralogistics setting motivating this work. The manual scanning and location-tracking practices described in Section 1 are prone to delays and omissions under time pressure, producing discrepancies between physical and digital inventory states that require additional operator effort to resolve. A system capable of continuously and automatically identifying products as they move through the monitored environment could reduce reliance on deliberate manual scanning actions and provide more frequent, incremental updates to inventory records, thereby improving consistency between physical and digital state. The explicit validation of decoded identifiers against a reference set, combined with the sufficiency and dominance conditions underlying episode-level acceptance (Section 4.2), is designed to suppress spurious or low-confidence decodes before they propagate into identification decisions, which is expected to reduce the incidence of incorrect identifications relative to accepting any single successful decode at face value. The cumulative resolution behavior reported in Section 5.3 further indicates that identification does not require a single, fully favorable observation: episodes accumulate evidence and resolve progressively as additional frames become available, reaching the full success rate only once the entire episode is observed, suggesting that the framework can recover identifications from otherwise incomplete or degraded observation sequences rather than requiring a single high-quality scan. These implications remain anticipated rather than measured at the operational level. The present study reports frame-level and sequence-level identification performance under controlled acquisition variability (Section 5), not a comparison against manual scanning workflows or a deployed inventory system, and any claim regarding the magnitude of operational improvement would require dedicated experimentation beyond the scope of this work.

7 Conclusions

The proposed framework addresses visual product identification as a system-level problem rather than as an isolated decoding task. This perspective is particularly relevant in dynamic indoor environments, where visual observations are often incomplete, degraded or temporally inconsistent. Under such conditions, reliable identification depends not only on whether a symbol can be detected or decoded in a single frame, but also on how uncertainty is represented, constrained and aggregated across the processing chain.

A central implication of the proposed design is the explicit separation between frame-level perception and sequence-level decision making. Frame-level processing produces structured observations of varying reliability, while sequence-

level reasoning interprets these observations as evidence and integrates them over time. This shifts the emphasis from single-frame recognition toward cumulative identification reliability, which is more consistent with mobile acquisition scenarios.

The framework further highlights the role of prior knowledge and intermediate reasoning mechanisms in constraining identification outcomes. Candidate decoding results are evaluated against a predefined reference set, reducing the likelihood that corrupted or ambiguous observations propagate into higher-level decisions. In addition, the inclusion of recovery mechanisms enables partially informative observations to remain useful within the identification process. Together, these elements extend the framework beyond conventional decoding-centric approaches and support robustness under visually degraded conditions.

From a methodological perspective, the use of controlled acquisition variability and macro-scene-level separation enables analysis of how architectural choices interact with environmental degradation. This provides a more informative evaluation perspective than aggregate performance metrics alone by linking component-level uncertainty to system-level identification outcomes.

The main contribution of this paper lies in the architectural and methodological formulation of the problem. The proposed framework integrates detection, geometric normalization, decoding, recovery, validation, and temporal aggregation within a unified structure, while explicitly distinguishing between perception and identification processes. In addition, the evaluation design supports systematic analysis of reliability under controlled variability conditions.

At the same time, the framework should be interpreted within its current scope. The contribution lies primarily in conceptual system design and evaluation logic, supported by a preliminary empirical evaluation reported in Sections 5 and 6: detection performance is close to ceiling, frame-level primary decoding alone succeeds on only about a third of ground-truth instances, and sequence-level evidence accumulation raises identification success to roughly three in four episodes under an *ever_correct* proxy criterion. This evaluation covers only the primary-decoding pathway and the proxy sequence-level criterion; the recovery branch and the full sufficiency/dominance decision rule described in Section 4 remain unevaluated, so comprehensive, large-scale empirical validation still remains limited. Similarly, the interpretation of episode-level identification as an indicator of product presence is introduced at a conceptual level and not as a fully implemented operational solution.

Several limitations remain. The recovery branch and the full sufficiency/dominance temporal decision logic remain unevaluated, as noted above, and the framework has not yet been validated through large-scale experiments beyond the single pilot dataset used here, and the operational integration of identification outputs into warehouse systems has not been realized. Furthermore, the experimental setting reflects controlled dynamic conditions rather than full industrial deployment.

Future work will focus on empirical validation of the recovery branch and the full temporal decision logic at larger scale, as well as on extending the framework toward spatial inference and system-level integration. Overall, the proposed framework suggests that robust visual product identification in dynamic environments may be more appropriately treated as an evidence integration problem than as a single-step recognition task; the preliminary results reported in Sections 5 and 6 support this view, and provide a structured basis for further investigation in HMLV-inspired settings.

Acknowledgment

Project no. 2019-1.1.1-PIACI-KFI-2019-00312 has been implemented with the support provided by the Ministry of Innovation and Technology of Hungary from the National Research, Development and Innovation Fund, financed under the PIACI KFI funding scheme.

References

[1] J. J. Losada del Olmo, P. E. L. de Teruel, A. Ruiz, and F. J. García-Clemente: Computer vision on the edge: A scalable auto-id solution for industrial logistics. Procedia Computer Science, 265:276–284, 2025. 20th International Conference on Future Networks and Communications/ 22nd International Conference on Mobile Systems and Pervasive Computing/15th International Conference on Sustainable Energy Information Technology (FNC/MobiSPC/SEIT 2025).

[2] C. Tadjine, A. Ouafi, A. Benlamoudi, and A. Taleb-Ahmed: Computer vision in warehouse management automation: A survey on implemented methods with prototyping hardware. Engineering Applications of Artificial Intelligence, 160:111886, 2025.

[3] A. Simeth, A. A. Kumar, and P. Plapper: Flexible and robust detection for assembly automation with yolov5: a case study on hmlv manufacturing line. Journal of Intelligent Manufacturing, 36(5):3447–3463, 2025.

[4] T. Almeida, V. Santos, O. M. Mozos, and B. Lourenco: Comparative analysis of deep neural networks for the detection and decoding of data matrix landmarks in cluttered indoor environments. Journal of Intelligent & Robotic Systems, 103(1):13, 2021.

[5] T. Diwan, G. Anirudh, and J. V. Tembhurne: Object detection using yolo: challenges, architectural successors, datasets and applications. multimedia Tools and Applications, 82(6):9243–9275, 2023.

[6] A. M. Rekavandi, L. Xu, F. Boussaid, A.-K. Seghouane, S. Hoefs, and M. Bennamoun: A guide to image-and video-based small object detection using

deep learning: Case study of maritime surveillance. IEEE Transactions on Intelligent Transportation Systems, 2025.

[7] I. Kalinov, A. Petrovsky, V. Ilin, E. Pristanskiy, M. Kurenkov, V. Ramzhaev, I. Idrisov, and D. Tsetserukou: Warevision: Cnn barcode detection-based uav trajectory optimization for autonomous warehouse stocktaking. IEEE Robotics and Automation Letters, 5(4):6647–6653, 2020.

[8] V. Edula, K. Ammisetty, A. Kotha, D. Ravipati, A. Davuluri, et al.: A novel framework for qr code detection and decoding from obscure images using yolo object detection and real-esrgan image enhancement technique. In 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), pages 1–6. IEEE, 2023.

[9] D. Corpuz, L. Rosario, J. Cempron, P. L. Lozano, and J. Ilao: Qr code detection with perspective correction and decoding in real-world conditions using deep learning and enhanced image processing. Proc. Copyright, pages 685–690, 2025.

[10] T. Muallim, H. Kucuk, M. Bareket, and M. Kahraman: Lightweight deep learning model and novel dataset for restoring damaged barcodes and qr codes in logistics applications. Computer Modeling in Engineering & Sciences, 143(3):3557, 2025.

[11] T. Almeida, V. Santos, B. Lourenço, and P. Fonseca: Detection of data matrix encoded landmarks in unstructured environments using deep learning. In 2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), pages 74–80. IEEE, 2020.

[12] D. Matuszczyk and F. Weichert: Reading direct-part marking data matrix code in the context of polymer-based additive manufacturing. Sensors, 23(3):1619, 2023.

[13] R. Wudhikarn, P. Charoenkwan, and K. Malang: Deep learning in barcode recognition: A systematic literature review. IEEE Access, 10:8049–8072, 2022.

[14] Ö. Albayrak Ünal, B. Erkayman, and B. Usanmaz: Applications of artificial intelligence in inventory management: A systematic review of the literature. Archives of computational methods in engineering, 30(4):2605–2625, 2023.

[15] J. L. de Moura Junior, E. M. Frazzon, and G. de Lorena Diniz Chaves: Computer vision as a resource in smart warehouses: A systematic review. In International Joint conference on Industrial Engineering and Operations Management, pages 73–83. Springer, 2024.

[16] B. Chen and J. Huang: Drone-assisted qr code recognition method for warehouse management. In 2024 International Conference on Networking, Sensing and Control (ICNSC), pages 1–6. IEEE, 2024.

[17] A. Morar, A. Moldoveanu, I. Mocanu, F. Moldoveanu, I. E. Radoi, V. Asavei, A. Gradinaru, and A. Butean: A comprehensive survey of indoor localization methods based on computer vision. Sensors, 20(9):2641, 2020.

[18] L. R. Agostinho, N. M. Ricardo, M. I. Pereira, A. Hiolle, and A. M. Pinto: A practical survey on visual odometry for autonomous driving in challenging scenarios and conditions. IEEE Access, 10:72182–72205, 2022.

[19] O. M. Manyar, J. Cheng, R. Levine, V. Krishnan, J. Barbič, and S. K. Gupta: Physics informed synthetic image generation for deep learning-based detection of wrinkles and folds. Journal of Computing and Information Science in Engineering, 23(3):030903, 2023.

[20] F. Töper, J. M. Araya-Martinez, A. S. Reig, T. Tom, S. Sardari, and P. Ohlhausen: Leveraging synthetic training data for object detection to enhance autonomous depalletizing systems. In European Robotics Forum, pages 229–235. Springer, 2025.

[21] G. Jocher, A. Chaurasia, and J. Qiu: Ultralytics YOLOv8. https://github.com/ultralytics/ultralytics, 2023.

[22] R. Hartley, A. Zisserman, et al.: Multiple view geometry in computer vision, volume 2. Cambridge university press Cambridge, 2003.

[23] M. Laughton and libdmtx Contributors: Libdmtx: Open source data matrix software. https://github.com/dmtx/libdmtx, 2026. Accessed: 2026-08-10.

[24] GS1: GS1 DataMatrix guideline. https://ref.gs1.org/guidelines/datamatrix/, 2018.

[25] M. A. Fischler and R. C. Bolles: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM, 24(6):381–395, 1981.

[26] W. J. Dixon: Ratios involving extreme values. The Annals of Mathematical Statistics, 22(1):68–78, 1951.

[27] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft: Simple online and realtime tracking. In 2016 IEEE international conference on image processing (ICIP), pages 3464–3468. Ieee, 2016.

[28] L. Karrach and E. Pivarciová: Recognition of data matrix codes in images and their applications in production processes. Management Systems in Production Engineering, 28(3):154–161, 2020.

[29] L. Karrach and E. Pivarčiová: Comparative study of data matrix codes localization and recognition methods. Journal of Imaging, 7(9):163, 2021.

[30] L. Liao, J. Li, and C. Lu: Data extraction method for industrial data matrix codes based on local adjacent modules structure. Applied Sciences, 12(5), 2022.

[31] Y. Liu, Y. Song, G. Gu, J. Luo, T. Wang, and Q. Jiang: A data matrix code recognition method based on l-shaped dashed edge localization using central prior. Sensors, 24(13):4042, 2024.

[32] E. O. Rybakova, E. E. Limonova, and P. V. Bezmaternykh: Improving data matrix mobile recognition via fast hough transform and adaptive grid extractors. Computer Optics, 49(6):1182–1190, 2025.

[33] L. Xu, V. R. Kamat, and C. C. Menassa: Automatic barcode extraction for efficient large-scale inventory management. In Computing in Civil Engineering 2017, pages 340–348. ASCE, 2017.