

# An AI-Driven Context-Aware Framework for Indoor Item Discovery and Shelf-Level Navigation in Large-Scale Retail Stores

Suresh Kumar

Walmart, USA

Corresponding author: Suresh Kumar, Email: Sureshkumar89@gmail.com

Finding a specific item in a large retail store requires more than navigating to an aisle. A useful assistance system must interpret a shopper's request, identify the intended product variant, determine where that product is currently displayed, guide the shopper to a suitable viewing position, and verify the item on a crowded shelf. These requirements connect indoor positioning, visual recognition, catalog retrieval, and human interaction, yet their errors arise at different spatial and semantic scales. This paper proposes a context-aware framework that coordinates these functions through a versioned semantic store graph and an explicit evidence state. The framework separates global wayfinding from local shelf inspection and withholds exact item confirmation when location, identity, or freshness evidence is insufficient. It incorporates accessibility constraints, uncertain inventory, changing planograms, and optional visual, inertial, and radio measurements. The research contribution is a system design and analytical evaluation, rather than a claim of deployed retail performance. Reproducible calculations examine target width, localization uncertainty, map freshness, correlated sensing, and disruption-aware routing. An executed Monte Carlo check uses 100,000 synthetic samples at each of five assumed error levels to verify an idealized interval-assignment model. For a 0.60 m target interval, the analytical assignment probability decreases from 86.64% at a 0.20 m error standard deviation to 29.23% at 0.80 m. These values describe the stated model, not a trained system or a store trial. The analysis motivates separate confidence requirements for aisle guidance and shelf confirmation. A prospective evaluation protocol defines end-to-end success, uncertainty calibration, generalization, accessibility, and operational maintenance measures. The proposed framework provides a testable basis for retail navigation research while identifying the empirical evidence needed before operational claims can be made.

**Keywords:** indoor navigation; retail item discovery; context awareness; shelf localization; multimodal perception; uncertainty; semantic maps; assistive shopping.

## **1. Introduction**

Large stores distribute products across departments, aisles, bays, shelf tiers, promotional displays, and temporary locations. A shopper may know the function of an item without knowing its catalog name, or may know an exact brand but be unable to distinguish nearby package variants. Item discovery consequently includes a semantic problem and a spatial problem. A system that solves only one can still leave the shopper searching. An accurate aisle label does not establish which shelf contains the requested size, and a correct image classification does not establish a safe route to the display.

This paper treats successful discovery as a sequence that ends with verified correspondence between the requested item and a visible or otherwise confirmed stock-keeping unit, abbreviated SKU. The shopper first expresses an intention, such as locating unsweetened oat milk in a specified package size. The system resolves that intention against the store catalog, identifies candidate display locations, and guides the shopper toward an approach position. Local inspection then links the physical package to the chosen SKU. Completion requires appropriate evidence and a confirmation event; proximity to a map coordinate alone does not complete the task.

The distinction matters because uncertainty changes throughout the journey. During cross-store travel, a rough position may be adequate to select an aisle. Near a shelf, the same error may place a pointer over several different products. Conversely, a highly accurate camera-relative estimate can still point at the wrong item when the underlying planogram is outdated. A navigation controller must therefore reason about position, product identity, and map validity together, without treating them as interchangeable measures of confidence.

Existing research provides useful components. Dense retail detection addresses crowded visual scenes [1]. Grocery classification datasets support semantic product recognition [2]. Visual-inertial systems estimate motion from cameras and inertial measurements [3,4]. Image-language models permit flexible semantic matching [5]. Their combination, however, needs an explicit contract for what each component knows, how old its evidence is, and what action is justified. The framework proposed here makes that contract the center of the design.

The main contribution is a hierarchical assistance architecture with four linked elements: a semantic graph of store locations and product observations; a multimodal evidence state that preserves uncertainty and provenance; a controller that changes sensing and guidance behavior across navigation stages; and an evaluation design that separates verified item discovery from arrival near a target. These are proposed design contributions. The paper does not claim a new detector, a new SLAM algorithm, or universal novelty over every retail assistance system.

The analysis asks three questions. First, how should semantic intent, location estimates, and shelf evidence be represented so that one uncertain component cannot silently become a definitive instruction? Second, when does local verification become necessary even if global navigation appears accurate? Third, which measurements would establish whether the integrated framework improves shopping assistance under realistic changes in layout and product appearance? The numerical study answers a restricted part of the second question; the architecture and prospective protocol address the first and third.

The scope is customer assistance in a mapped indoor retail environment. Autonomous robot control, payment processing, storewide surveillance, and automatic inventory accounting are outside the primary task. Inventory and planogram services are inputs whose reliability must be evaluated. The proposed interface supports optional camera use and alternative guidance modes, but its accessibility benefits remain hypotheses until tested with the relevant users. All figures in this manuscript are original schematics or plots generated for this analysis.

## **2. Research Approach and Evidence Boundaries**

The research follows a constructive design approach. It defines a practical problem, translates that problem into requirements, specifies an artifact, analyzes selected consequences of the artifact, and proposes a falsifiable empirical evaluation. This sequence is appropriate when the contribution concerns coordination between established techniques rather than a new trained model. The present artifact is an architecture and its decision rules. No complete mobile application, instrumented store deployment, or trained shelf recognition pipeline was available for evaluation.

Three types of evidence are kept separate. Published work establishes the existence and intended capabilities of component methods. Original analytical reasoning establishes consequences under explicit assumptions. Synthetic sampling checks the implementation of one analytical expression. A fourth category, future field evidence, is described only as a proposed study. Tables and figures identify which category they belong to. This distinction prevents a convenient numerical example from being read as a measured improvement in a deployed system.

The literature selection is targeted rather than systematic. Sources were chosen to cover dense detection, product semantics, visual–inertial estimation, uncertainty calibration, mobile computation, and navigation evaluation. Primary research pages and proceedings records were consulted for bibliographic verification. This paper does not report a database search protocol, screening count, or exhaustive review. Its literature synthesis supports specific design choices and defines gaps in the evaluation required for this artifact.

Analytical evaluation is useful here because a spatial scale mismatch can be demonstrated without a proprietary store dataset. If a target occupies a narrow interval and relative position error is broad, the probability of assigning the correct interval is limited even under an ideal unbiased model. The calculation exposes a requirement for local evidence. It cannot determine which sensing technology will achieve a particular error distribution, nor does it measure a product detector’s ability to distinguish visually similar packages.

The evaluation also examines consequences of stale locations and correlated measurements. These are deliberately simple models that make assumptions visible. The freshness calculation assumes a constant location-change hazard and treats any change as invalidating the stored location. The fusion calculation assumes equal-variance estimates of the same scalar quantity. The routing example uses a single disruption cost. More realistic models would require observations of actual store operations and user behavior.

A strong empirical extension would preserve this separation. Component benchmarks would measure perception and localization against their own ground truth. An integrated task study would then measure whether shoppers find the correct requested item. Reporting only the best component accuracy would not establish end-to-end utility. Similarly, a user preference for an interface would not by itself establish correct SKU identification. The protocol in Section 12 defines these outcomes independently so that failures remain interpretable.

## **3. Related Work and Research Positioning**

### **3.1 Retail Perception and Product Identity**

Goldman and colleagues introduced SKU-110K to study detection in densely packed scenes [1]. Their work is relevant to the first step of shelf inspection: separating adjacent objects when packages are close together. Detection boxes, however, are not the same as exact SKU identities. A retail assistance system must connect a detected package to a catalog variant and must account for cases in which the visible front does not contain enough information for that distinction.

Klasson and colleagues supplied a grocery dataset with hierarchical labels and visual and semantic product information [2]. This supports the idea that product recognition can use relationships between fine and coarse categories. Wei and colleagues introduced the RPC checkout dataset [6], which addresses a different retail arrangement. These datasets can support component development, but neither dataset alone establishes the performance of navigation to a shelf in a changing store. The framework therefore separates dataset suitability for recognition from suitability for the integrated task.

Transformer-based detection, represented by DETR [7], and open-set detection, represented by Grounding DINO [8], provide alternative ways to locate image regions. CLIP connects image and text representations [5]. Their relevance is architectural: a shopper's phrase can help retrieve candidate products or regions. Similarity scores must still be checked against explicit attributes, such as package size and flavor. A generic semantic match should not authorize a claim that the exact requested variant has been found.

The proposed recognition pipeline consequently uses two stages. The first produces a short list of plausible products from category, language, appearance, and location context. The second verifies attributes through higher-resolution crops, readable label text, barcode evidence where available, and catalog constraints. This is an original design choice for the framework rather than a claim about the reported behavior of any cited model. It allows inexpensive broad retrieval while reserving stronger evidence for the final decision.

### **3.2 Indoor Positioning and Map Alignment**

ORB-SLAM3 supports visual, visual-inertial, and multiple-map estimation [3], while VINS-Mono provides a monocular visual-inertial estimation approach [4]. These methods motivate using motion and visual structure together. Their benchmark results cannot be transferred directly to shelves, lighting conditions, phone handling patterns, or crowding in a retail store. The present design treats them as candidate estimator families whose accuracy, recovery behavior, and resource use must be measured in the intended setting.

Global map alignment is a separate problem from short-term motion estimation. A phone can estimate movement consistently within its own local frame while the relationship between that frame and the store plan is wrong. Repeated aisle structures can create ambiguous visual associations. The framework therefore stores the estimated transform into the store frame with its own validity and uncertainty. A successful local tracking state does not automatically establish that the shopper is in the correct aisle.

Radio observations can add absolute or region-level constraints, provided that the selected device and infrastructure support them. This manuscript does not assign a universal accuracy to Bluetooth, Wi-Fi, or ultra-wideband systems. Shelf materials, anchor geometry, signal blockage, calibration, and device behavior can alter the measurement distribution. Radio inputs enter the framework as observations with tested likelihood models, rather than as technology labels that imply a guaranteed positioning precision.

### **3.3 Context and Uncertainty**

Dey's formulation of context-aware computing provides a conceptual basis for using situational information to change application behavior [9]. In this framework, relevant context includes the active task, the age of a product-location observation, current route accessibility, available sensors, and the shopper's chosen interaction mode. Context is not simply a large collection of user attributes. Each stored field should affect a specific retrieval, sensing, or guidance decision.

Guo and colleagues showed the importance of calibration when neural-network confidence is interpreted probabilistically [10]. Angelopoulos and Bates explain conformal prediction as a framework for uncertainty sets under specified assumptions [11]. These works motivate evaluating uncertainty separately from classification accuracy. They do not remove the need to test calibration under changes in store, device, packaging, and time. In the proposed system, the controller may request more evidence even when a model returns a high raw score.

### 3.4 Navigation Evaluation and Mobile Deployment

Anderson and colleagues discuss standardized evaluation for embodied navigation, including success-sensitive path measures [12]. A retail task requires an additional distinction between reaching a region and finding the intended SKU. The paper adapts that evaluation perspective by defining verified item success, accessible-route feasibility, and the cost of clarification. A short path to the wrong product is a failed task regardless of its geometric efficiency.

MobileNets demonstrates a model family designed for efficient mobile vision [13]. It motivates evaluating a lightweight local inference option, although the framework does not prescribe a particular network. Federated learning offers a possible distributed training mechanism [14], but it is not required by the architecture and does not automatically resolve privacy or data-quality issues. The initial deployment model instead emphasizes on-device processing where practical and tightly scoped operational data collection.

Table 1 summarizes the role of the reviewed components. The gap addressed here is an explicit coordination and evaluation problem: the cited capabilities must be connected without erasing distinctions between object detection, exact identity, position, and task completion. The table is a qualitative positioning matrix, not a performance ranking.

**Table 1.** Research positioning and component boundaries

| Research area            | Representative work                | Role in this framework           | What it does not establish                |
|--------------------------|------------------------------------|----------------------------------|-------------------------------------------|
| Dense detection          | Goldman et al. [1]                 | Locate packed product regions    | Exact SKU identity or navigation success  |
| Product semantics        | Klasson et al. [2]; Wei et al. [6] | Develop recognition components   | Performance on changing shelf routes      |
| Visual and inertial pose | Campos et al. [3]; Qin et al. [4]  | Estimate motion and pose         | Guaranteed retail shelf precision         |
| Language and vision      | Radford et al. [5]; Liu et al. [8] | Retrieve semantic candidates     | Exact variant correctness                 |
| Calibration              | Guo et al. [10]                    | Interpret confidence             | Calibration after arbitrary domain shifts |
| Navigation evaluation    | Anderson et al. [12]               | Define success-sensitive metrics | Verified retail SKU discovery             |

## 4. Problem Formulation and System Requirements

### 4.1 Task Definition

A discovery request consists of a desired item description, optional hard attributes, and optional preferences. Hard attributes define which products count as correct. Preferences rank acceptable alternatives but cannot silently replace hard constraints. For example, a request for a particular package size differs from a preference for a lower price. If no candidate satisfies the hard attributes, the system should explain the mismatch and ask whether alternatives are acceptable.

The store is represented as traversable space linked to product-bearing structures. A shelf target is not just a point in a floor plan. It includes a store identifier, department, aisle, side, bay, tier, optional facing, and observation time. A product can occupy several locations simultaneously. A seasonal display and a regular aisle should therefore be represented as two candidate locations for one SKU, rather than conflicting values in a single location field.

The navigation goal is an approach region from which the relevant shelf can be inspected, not the physical center of the product. This distinction avoids planning a route into a fixture. The final approach may depend on camera field of view, shelf height, available space, and the shopper's mobility needs. A successful route ends at a feasible observation position; successful discovery additionally requires verified correspondence to the requested item.

## 4.2 State and Observations

At time  $t$ , the system maintains a belief over pose, map alignment, and product-location hypotheses. It also maintains a task state, a sensor-availability state, and an evidence ledger. The ledger records where an assertion came from, when it was observed, and whether it has been independently verified. A catalog description and a camera observation are different evidence types even if both refer to the same product identifier.

Equation 1 expresses the proposed state compactly. The pose belief  $b$  includes uncertainty;  $G$  is the versioned store graph;  $Q$  is the resolved target set;  $C$  is operational and user-selected context;  $E$  is the evidence ledger. The state is a design abstraction rather than a claim that a single monolithic model should learn all variables jointly. Separate modules may own different fields, provided that their update semantics are consistent.

$$S_t = (b_t, G_t, Q_t, C_t, E_t) \quad (1)$$

Table 2 states the functional requirements used throughout the design. The acceptance evidence column deliberately names observable tests rather than unsupported accuracy targets. Numeric thresholds should be fixed during validation and then evaluated on held-out data. This avoids selecting thresholds after viewing favorable test outcomes.

**Table 2.** Proposed functional requirements and acceptance evidence

| Requirement         | Proposed behavior                                                | Evidence needed                              |
|---------------------|------------------------------------------------------------------|----------------------------------------------|
| Intent fidelity     | Preserve exact attributes and request approval for substitutions | Exact-variant retrieval tests                |
| Spatial hierarchy   | Represent aisle, bay, tier, and approach region separately       | Surveyed shelf and route ground truth        |
| Uncertainty control | Withhold precise instructions when evidence is weak              | Calibration and risk-coverage analysis       |
| Freshness           | Retain observation time and graph version                        | Tests after deliberate layout changes        |
| Accessible routing  | Remove edges violating hard constraints                          | Participant and route feasibility evaluation |
| Recovery            | Offer relocalization, clarification, or assistance               | Failure recovery time and outcome            |
| Auditability        | Record evidence behind final confirmation                        | Task-level error attribution                 |

## 4.3 Optimization Objective

The system should minimize the expected effort required to complete a correct discovery task, subject to route feasibility and evidence requirements. Effort includes walking time, scanning time,

clarification, and recovery from errors. Some constraints are hard, such as an inaccessible edge, while other costs are soft, such as an optional preference for less walking. Hard constraints must not be converted into small penalties that a shortest-path algorithm can trade away.

Equation 2 gives a high-level objective. The policy  $\pi$  maps evidence states to actions.  $T$  denotes time,  $N$  the number of interaction requests,  $F$  an incorrect final confirmation, and  $\lambda$  values express an application-defined trade-off. These coefficients are not estimated in this paper. The equation identifies what must be measured before an optimization claim can be tested. It also makes clear why the shortest geometric path may not minimize total task cost.

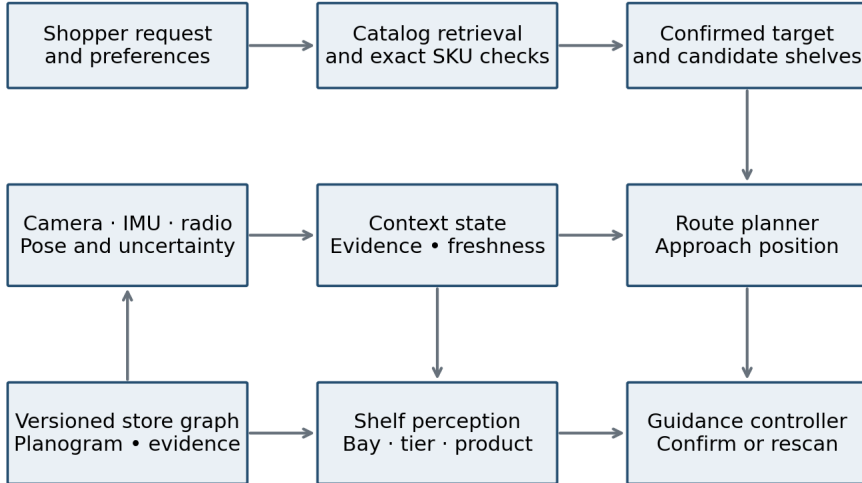
$$\pi^* = \arg \min_{\pi} \mathbb{E}_{\pi}[T + \lambda_N N + \lambda_F \mathbf{1}_F] \quad (2)$$

A practical implementation can approximate this objective with a hierarchical policy. It resolves the target first, chooses a feasible approach route, requests local inspection, and confirms only when evidence crosses calibrated thresholds. Such a controller is easier to inspect than a single end-to-end policy during early development. Learned policies can later replace individual decisions if they preserve the same task definitions and reporting requirements.

## 5. Proposed Framework Architecture

### 5.1 Modules and Information Flow

Figure 1 presents the proposed architecture. The request service converts language, typed text, or a selected catalog entry into a structured item query. The retrieval service returns candidate SKUs and the attributes that justify them. The store graph supplies candidate physical locations. A pose service estimates where the device is relative to the store, while a context service records which actions are currently feasible. The controller combines these outputs to select navigation, inspection, clarification, or confirmation.



Proposed architecture • arrows denote principal information flows

**Figure 1.** Proposed framework architecture. Original schematic; no deployed system is implied.

The architecture is asynchronous because its inputs change at different rates. Inertial measurements may arrive much more frequently than catalog updates, while shelf observations may be available only when the shopper chooses to scan. Each module therefore emits timestamped observations and validity information. The controller should not wait for every service before responding, but it must identify which evidence is missing and limit the specificity of its instruction accordingly.

The controller is the only component that authorizes user-facing task completion. A detector cannot independently announce that an item has been found, and a route planner cannot independently conclude that the requested product is available. This separation makes it possible to trace a false confirmation to its contributing evidence. It also supports incremental deployment: a store can begin with catalog lookup and aisle navigation, then add local verification without changing the definition of a completed task.

## **5.2 Retrieval and Intent Resolution**

The request service represents explicit constraints as structured fields. A query such as “the unsweetened one-liter oat drink” produces category, sweetness, and volume constraints. If the brand is absent, it remains unspecified. The system may retrieve several acceptable candidates, but it should not infer a brand from an advertising preference or from an unrelated previous purchase. Any personalization should be optional and visible in the candidate selection.

Candidate retrieval can combine lexical search with semantic embeddings. Lexical matching preserves exact model numbers, sizes, and brand spellings; semantic retrieval helps with ordinary descriptions and synonyms. A reranking step checks whether each candidate satisfies the hard attributes. The output includes a reason code such as exact match, acceptable category match, or requires clarification. This explanation is operationally useful because later shelf verification can focus on the attributes that remain unresolved.

Substitution requires a separate user decision. An unavailable requested SKU should not be replaced automatically merely because a nearby package has a similar embedding. The system should describe the relevant difference and request acceptance. For products whose attributes have material consequences, the interface should support direct label inspection rather than infer suitability from package appearance. The framework does not certify product composition; it preserves the distinction between catalog information and observed packaging.

## **5.3 Controller and Evidence Ledger**

The ledger stores an assertion together with source, observation time, confidence interpretation, and graph version. It also stores dependencies. For example, a shelf-location assertion derived from a camera frame depends on the pose estimate used to place that frame. If map alignment later changes, the assertion may need to be recomputed or invalidated. Without dependency tracking, stale geometric assumptions can survive as apparently independent product evidence.

Confidence values are meaningful only with a defined event. The probability that a package belongs to a category differs from the probability that it matches an exact SKU, and both differ from the probability that a shelf location is still valid. The ledger therefore uses typed confidence fields. It does not multiply arbitrary model scores into a single number. A calibrated fusion model may combine evidence, but its target event and calibration dataset must be documented.

The controller produces an action plus a reason. “Move toward the next aisle” may depend on coarse location confidence. “Scan the middle shelf” may depend on a candidate bay and a feasible viewing position. “This is the requested item” requires stronger identity evidence. The reason record helps developers inspect failures and helps the interface choose a concise explanation when a rescan is

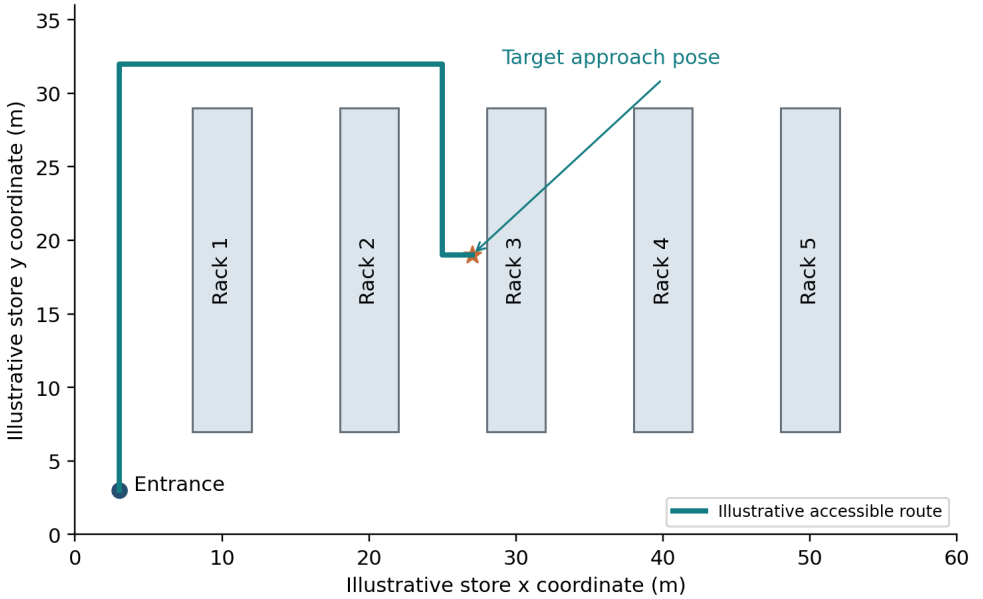
needed. It also prevents a product model’s numeric confidence from being displayed as a misleading universal certainty score.

## 6. Semantic Store Graph and Data Maintenance

### 6.1 Hierarchical Spatial Representation

The proposed store graph links geometric navigation nodes to semantic entities. Traversable nodes represent aisle junctions, corridor segments, and approach regions. Product-bearing nodes represent fixtures, bays, tiers, and optional facing intervals. Product nodes refer to stable catalog identifiers. Observation edges connect products to locations with timestamps and evidence provenance. This separation allows a product to move without redefining the physical shelf structure.

Figure 2 illustrates the relationship between a floor plan and an approach route. It is a schematic layout, not a surveyed store. The route remains in open space and terminates beside the target fixture. A local shelf frame then expresses positions along the fixture and above the floor. This hierarchical representation limits the precision demanded from each part of the map: global navigation needs connected free space, while inspection needs a local shelf coordinate system.



**Figure 2.** Illustrative store geometry and route to a shelf approach position. Coordinates are synthetic.

Table 3 defines the principal entities. Geometry and time are both essential. A location record without a coordinate frame is ambiguous, while a correct coordinate without a verification time may be operationally misleading. The data model also distinguishes unavailable information from a negative assertion. “No recent observation” should not be stored as “product absent.”

**Table 3.** Proposed semantic store graph entities

| Entity              | Essential fields                                  | Purpose                                       |
|---------------------|---------------------------------------------------|-----------------------------------------------|
| Store frame         | Store ID; origin; units; survey version           | Common geometric reference                    |
| Navigation edge     | Endpoints; distance; clearance; status; timestamp | Feasible traversal and cost                   |
| Fixture             | Fixture ID; pose; dimensions; frame ID            | Link map geometry to shelves                  |
| Shelf location      | Aisle side; bay; tier; facing interval            | Represent local target position               |
| SKU                 | Catalog ID; variant; size; label version          | Preserve exact product identity               |
| Observation         | Source; time; uncertainty; dependencies           | Represent evidence without overwriting intent |
| Location hypothesis | SKU ID; shelf ID; confidence event; validity      | Support multiple displays and uncertainty     |
| Task                | Request constraints; active target; outcome       | Connect guidance to verified success          |

## 6.2 Planograms and Observed Layouts

A planogram describes intended product arrangement. The observed shelf may differ because of replenishment, misplaced items, temporary promotions, or incomplete updates. The framework stores intended and observed locations separately. Agreement increases confidence; disagreement creates a reconciliation task. It should not be resolved automatically by overwriting whichever source arrived first.

An observation should identify its spatial granularity. A staff member may confirm that a product is in an aisle without confirming its tier. A shelf image may locate a package within a bay without reading the exact variant. The graph must preserve those partial assertions. Promoting an aisle-level observation to a facing-level location would introduce unsupported precision even if the user interface looks confident.

Store-map versions define internally consistent snapshots. Route computation should reference a known graph version, and the controller should check whether a new version changes a relevant edge or target. If an update affects a distant department, the active route need not be recomputed. If an aisle closes or the target fixture moves, the route and inspection plan must be revised. This selective invalidation reduces unnecessary interruption while maintaining consistency.

## 6.3 Freshness and Conflicting Evidence

Freshness is a property of a claim in context. A permanent wall map can remain valid longer than a promotional product location. The system therefore assigns validity policies by entity and observation type. It should learn change patterns only after enough operational data are available. Initially, conservative category-specific policies can be configured and tested, with the age of evidence retained for evaluation.

Conflicting observations should remain inspectable. Two shoppers may report different positions because the SKU has multiple displays, one scan was misidentified, or the product moved between scans. A conflict handler can retain multiple hypotheses and seek a stronger observation. Repeated evidence from the same device trajectory is not automatically independent corroboration. Provenance is needed to avoid counting copies of one mistake as several confirmations.

Operational maintenance is part of the research problem. A system that performs well immediately after a complete survey may become expensive if every shelf change requires manual remapping. Field evaluation should therefore report the labor required to create and maintain the graph, the delay

between a physical change and a corrected instruction, and the proportion of tasks using outdated locations. These costs belong beside user-facing performance measures.

## 7. Multimodal Localization and Uncertainty Management

### 7.1 Sensor Roles

The proposed pose service accepts visual motion, inertial motion, radio measurements, and recognized landmarks. Their roles are complementary. Inertial measurements support short-term motion prediction; visual observations constrain movement and local structure; radio or surveyed landmarks can support alignment to the store. The architecture permits subsets of these sensors because ordinary customer devices will not provide a uniform hardware configuration.

Table 4 states how each observation would be used and what can invalidate it. The entries are design considerations, not measured comparisons. In particular, the table does not assume that a radio observation resolves shelf tier or that a camera always resolves global aisle identity. A useful estimator must indicate when an input is unavailable, inconsistent, or too weak to improve the current belief.

**Table 4.** Candidate sensing inputs and limitations

| Observation            | Proposed role                     | Failure condition to test           |
|------------------------|-----------------------------------|-------------------------------------|
| Camera motion          | Local relative pose               | Blur, occlusion, repeated textures  |
| Inertial measurements  | Short-term movement prediction    | Bias, phone handling, timing errors |
| Radio observations     | Absolute or regional constraints  | Blockage, geometry, device mismatch |
| Aisle sign or landmark | Resolve global location ambiguity | Unreadable or relocated signs       |
| Shelf structure        | Local bay and tier geometry       | Oblique views and missing dividers  |
| Label text or barcode  | Distinguish exact SKU variants    | Small print, glare, hidden labels   |

### 7.2 Belief Updates and Map Constraints

Equation 3 expresses a Bayesian filtering view of pose estimation. The prediction step uses the previous belief and motion input; the measurement likelihood updates that belief. The map contributes constraints on feasible movement. In a practical implementation, an extended Kalman filter, particle filter, or factor-graph estimator could approximate the update. The architecture does not require a specific estimator, but it requires that the reported uncertainty match the approximation being used.

$$b_t(x) \propto p(z_t \mid x, G_t) \int p(x \mid x', u_t) b_{t-1}(x') dx' \quad (3)$$

Here  $x$  is pose,  $z$  is the current observation,  $u$  is the motion input, and  $x'$  is the previous pose. The proportionality constant normalizes the belief.

Map constraints should reject physically impossible trajectories without forcing every uncertain observation onto the nearest aisle. Aggressive snapping can make a wrong estimate appear stable. If two nearby aisles remain plausible, the estimator should retain ambiguity and the controller should request a discriminating landmark or stop issuing precise turn instructions. A broad but honest belief is more useful than an artificially narrow belief centered on the wrong corridor.

Time synchronization matters because visual and inertial measurements refer to motion at different instants. A delayed image can appear inconsistent with the current heading even when both sensors are functioning. The pose service should preserve observation time separately from processing time. Evaluation should include timestamp errors and variable processing delay because they can produce failures that would otherwise be attributed to model accuracy.

### 7.3 Correlation and Overconfidence

Sensor fusion can become overconfident when supposedly independent estimates share information. A shelf position inferred from a camera and a pose estimated using that same image are not independent. Two model heads trained on identical features may also share error modes. A fusion rule that ignores this dependence can shrink uncertainty without adding equivalent information. Section 11 includes an equal-variance example that quantifies the effect for two scalar estimates.

A practical evidence ledger should therefore record common ancestors. When two assertions derive from one frame, the system can avoid treating them as independent votes. A conservative fusion method may retain the larger uncertainty or use a model that explicitly represents cross-covariance. The correct choice depends on the estimator and available validation data. The requirement is that confidence gain must correspond to new evidence rather than repeated processing of the same evidence.

### 7.4 Recovery from Lost Localization

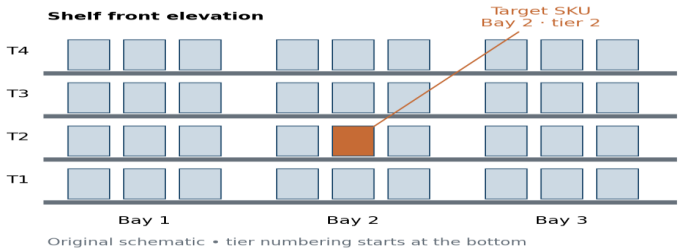
Lost localization is a normal operating state, not an exceptional software crash. The controller should recognize disagreement between predicted movement and incoming observations, excessive uncertainty, or failed map alignment. It then pauses precise navigation and offers a recovery action suited to the user's situation. Possible actions include reading an aisle sign, scanning a known fixture, returning to a recent landmark, or requesting staff assistance.

Recovery should not require constant camera use. Some shoppers may decline video capture, have difficulty holding a phone steadily, or be unable to obtain a useful view. The system should then operate at a coarser level and explain the resulting limitation. Evaluation must count these tasks, including unresolved cases, rather than removing them as inconvenient outliers. Otherwise the reported success rate would apply only to users able to satisfy the system's preferred sensing conditions.

## 8. Shelf Perception and Exact Item Verification

### 8.1 Shelf Coordinates and Visual Grouping

Once the shopper reaches an approach region, the system estimates a local shelf frame. The horizontal coordinate identifies position along the fixture, the vertical coordinate identifies tier, and the depth coordinate distinguishes the shelf plane from foreground clutter. Figure 3 shows the relationship between bay, tier, and target package. Tier numbers in this manuscript increase from the bottom; a deployed system should use one convention consistently and explain it to users.



**Figure 3.** Shelf front elevation showing bay, tier, and a target package. Original conceptual image.

Shelf geometry can be estimated from visible dividers and horizontal edges, combined with the mapped fixture dimensions where available. Product detections are then grouped relative to that geometry. This step is distinct from exact recognition. A package can be assigned to a tier even when its identity is uncertain, and a package can be recognized while its spatial relation to the shelf remains ambiguous because of occlusion or an oblique view.

Perspective must be handled explicitly. A screen-space vertical ordering is not always a physical tier ordering when the phone is tilted. Likewise, the left edge of an image need not correspond to the left edge of a bay. The system should transform observations into a shelf-relative frame or avoid precise spatial instructions until geometry is sufficiently constrained. This requirement is especially important for augmented overlays that can look authoritative despite a weak alignment.

## 8.2 Candidate Recognition and Attribute Verification

The first recognition pass should preserve several plausible candidates. A package image may reveal brand and color but not size, and several products may share most of their artwork. A second pass can inspect text, distinctive visual features, or a barcode. Catalog constraints narrow the candidate set, but a strong location prior should not overpower contradictory label evidence. Otherwise a misplaced product could be confirmed simply because the planogram expected the correct item there.

Equation 4 gives an illustrative candidate-ranking score. Visual similarity, text compatibility, catalog constraints, and spatial consistency contribute separate terms. The coefficients are not learned in this paper, and the resulting score is not a calibrated probability. Calibration must follow any learned ranking stage before its output is used to control confirmation. The explicit constraint term can reject a candidate when an observed attribute conflicts with the requested SKU.

$$s(k) = \alpha s_{\text{visual}}(k) + \beta s_{\text{text}}(k) + \gamma s_{\text{catalog}}(k) + \delta s_{\text{space}}(k) \quad (4)$$

Open-set behavior is necessary because a store can introduce products that were absent from training. The system should be allowed to return “unrecognized package” or “several possible variants.” It can then show catalog candidates or ask for a different view. Forcing every crop into a known class produces confident errors that are difficult for users to notice. An evaluation set should include unseen products and near-duplicate variants, not only familiar classes.

## 8.3 Verification and Confirmation

Confirmation should combine identity evidence with a user-visible description of the selected product. A concise statement can identify brand, variant, and size, with an optional image or barcode check. The user can reject the suggestion, which returns the controller to inspection rather than completing the task. This interaction creates an explicit outcome that can be audited during evaluation.

A visible package is not proof that the full requested quantity is available. The framework distinguishes finding one item from satisfying a shopping-list quantity. It also distinguishes a catalog availability signal from visual confirmation. If the requested package is not visible, the interface should offer a bounded next step rather than repeatedly pointing at an empty location. A time-limited rescan, alternate display, or staff request can prevent indefinite search loops.

The final confidence threshold should depend on the meaning of the confirmation, not on a desire to maximize completion count. A threshold that permits many wrong variants can make the system appear fast while reducing utility. The evaluation therefore reports both coverage, the fraction of tasks receiving a confirmation, and selective risk, the error rate among those confirmations. This pair reveals whether high apparent accuracy comes from declining most difficult cases.

## 9. Context-Aware Route Planning and Guidance

### 9.1 Feasible Approach Routes

A route planner operates on the traversable graph after removing edges that violate hard constraints. It then chooses a path to an approach region associated with an acceptable target location. The target region can change if a display is inaccessible or if another verified display offers a better viewing position. The planner should use a snapshot of edge costs and record that snapshot so that route choices can be reproduced during evaluation.

Equation 5 defines a candidate edge cost in time units. Walking time is combined with expected delay, uncertainty-related recovery cost, and interaction cost. The terms should not double count the same event. For example, a congestion estimate already included in walking time should not also be added as a full separate delay. Nonnegative costs permit standard shortest-path methods; a geometric travel-time lower bound can serve as a heuristic only when it remains admissible under the chosen cost model.

$$c_e(t) = \frac{d_e}{v_e(t)} + \mathbb{E}[D_e(t)] + \mathbb{E}[R_e(t)] + I_e(t) \quad (5)$$

Here  $d$  is edge distance,  $v$  is expected walking speed,  $D$  is additional delay,  $R$  is recovery time, and  $I$  is interaction time attributed to the edge.

A route that is slightly longer can still reduce expected completion time if it avoids a high risk of interruption. Section 11 demonstrates this with a two-route calculation. That calculation does not establish that the proposed planner outperforms a baseline across stores. It identifies a condition under which incorporating disruption probability changes the preferred route. Actual probabilities and recovery costs must be estimated and validated before the planner can claim an operational advantage.

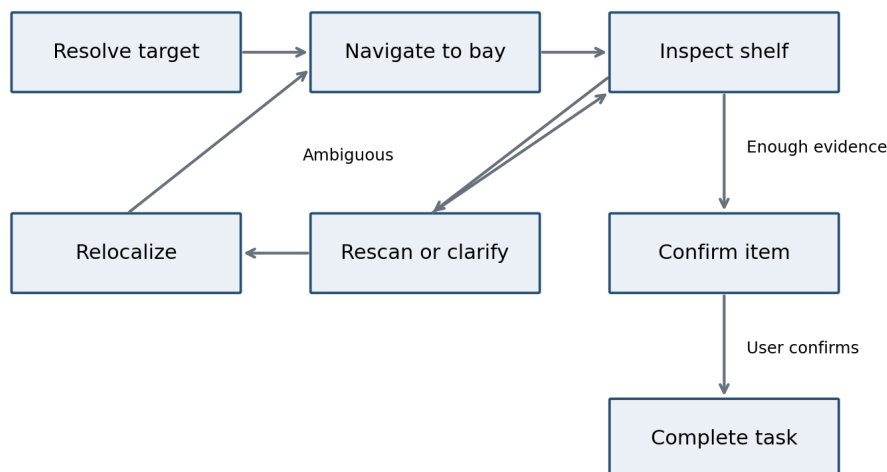
### 9.2 Multiple Items and Changing Targets

For a shopping list, the system must choose an order as well as a path. An exact global optimum may be unnecessary for an interactive service; a practical heuristic can choose a sequence and revise it when item availability changes. The objective should include confirmation effort and acceptable substitutions, not just distances between shelf coordinates. Two nearby items can still be expensive to find if they require difficult visual discrimination.

Replanning should be bounded to avoid distracting the shopper. Minor changes in estimated congestion should not repeatedly reverse the next instruction. A hysteresis rule can require a minimum expected improvement before changing an active route, except when a hard constraint becomes invalid. The interface should explain a route change in plain terms, such as a blocked aisle or a relocated display. This makes the change understandable and reduces the chance that the user interprets inconsistent instructions as their own mistake.

### 9.3 Stage-Specific Interaction

Figure 4 shows the controller's principal states. The distinction between navigating and inspecting permits different interfaces. Navigation can prioritize a small number of landmark-based instructions. Inspection can prioritize target appearance, tier, and local direction. When evidence is ambiguous, the controller requests a useful action rather than increasing the number of unsupported details in the instruction.



**Figure 4.** Principal controller states and recovery transitions. Original design schematic.

A useful clarification reduces uncertainty that affects the next decision. Asking the shopper to scan the same uninformative view repeatedly wastes effort. The controller can instead request a closer view of a distinguishing label, a wider view of shelf boundaries, or an aisle sign needed for relocalization. A formal information-gain policy is a possible extension, but the initial system can use explicit rules tied to known failure conditions.

## 9.4 Accessibility and User Control

Accessibility affects route feasibility, instruction format, and the physical demands of sensing. A route should account for user-selected mobility constraints and known store barriers. The interface should support readable text, spoken instructions, and optional haptic cues, with controls for repetition and pace. These design features are proposed requirements, not evidence that the application is accessible to every user group.

Shelf reachability and item visibility are separate constraints. An approach position may provide a clear camera view of a high shelf without allowing the shopper to retrieve the product. The system should distinguish “item located” from “item retrieved” and offer assistance when retrieval is not feasible. A user should also be able to end guidance or switch to staff support without the system treating that choice as sensor failure.

Future evaluation should involve users with varied needs from the design stage. Successful use by developers or sighted participants does not establish suitability for people with visual impairments. Similarly, a path that appears wide enough in a static map may be obstructed by temporary displays. The empirical protocol must include both planned accessibility constraints and changing obstacles observed during actual tasks.

## **10. Implementation Design and Computational Considerations**

### **10.1 On-device and Server Responsibilities**

The proposed implementation separates latency-sensitive functions from functions that benefit from centralized store data. The device maintains the active task, recent pose belief, cached route, and immediate interface state. A store service maintains catalog records, graph versions, and verified product-location observations. Image processing can run locally when resources permit. An optional server path may handle more demanding recognition, but its use should depend on connectivity, user choice, and the information required for the task.

This division supports partial operation during network interruption. A cached map and confirmed target can support coarse guidance while updates are unavailable. The interface must indicate when product-location evidence is old and avoid presenting cached availability as current. After reconnection, the controller reconciles graph versions before applying new instructions. A reconnect event should not automatically restart the task or discard the shopper's confirmed item selection.

The service boundary should expose structured evidence rather than only a final answer string. A recognition response should contain candidate identifiers, visible attributes, observation time, model version, and calibrated confidence semantics. A route response should contain graph version, feasibility assumptions, and target approach region. These fields make evaluation and debugging possible without requiring the application to infer why another service reached its conclusion.

### **10.2 Latency and Resource Accounting**

End-to-end latency includes sensing, preprocessing, inference, network transfer when used, evidence fusion, and interface rendering. Measuring inference alone would understate the time a shopper waits. The evaluation should report median and upper-tail response time under realistic device load, including thermal throttling and background activity. It should also report whether the result arrived in time to affect the instruction being displayed.

Computation should adapt to the current stage. Continuous high-resolution recognition is unnecessary while the shopper crosses a long corridor. Conversely, a brief higher-resolution inspection may be justified when distinguishing package variants. A staged policy can reduce camera and computation use without changing the definition of correctness. The proposed system should measure energy per completed task, because energy per frame can be misleading when one method requires many more frames.

Large catalogs introduce retrieval and update costs. Candidate retrieval can first restrict the catalog by store and explicit attributes, then use semantic ranking within the remaining set. Cached embeddings require versioning when product descriptions or models change. A stale embedding index can return a discontinued variant even when the authoritative catalog has been updated. Evaluation should include catalog update latency and the handling of missing or duplicated identifiers.

### **10.3 Model Lifecycle and Deployment Controls**

A model release should be associated with training-data versions, validation results, calibration parameters, and supported devices. A detector update may change confidence distributions even if its average accuracy improves. Confirmation thresholds should therefore be revalidated rather than copied automatically. The active application should record model versions in task logs so that a change in failure rates can be traced to a release.

Deployment can begin in shadow mode, where the system records recommendations without issuing precise shelf confirmations to users. Staff can compare those recommendations with ground truth and identify systematic mismatches. A limited pilot can then expose guidance to consenting participants with a clear fallback. This staged approach creates evidence for readiness; it does not substitute for an independent test set or permit selective removal of unsuccessful tasks.

The system should support rollback of model and graph changes independently. A poor recognition update should not require reverting a valid store-layout correction. Conversely, a faulty graph update should not force a model retraining cycle. Independent versioning reduces the operational scope of a failure and improves the interpretability of post-deployment evaluation.

## 11. Analytical Evaluation and Synthetic Verification

### 11.1 Evaluation Scope and Reproducibility

This section reports computations executed for this manuscript. It does not report training, detector benchmarking, live radio measurements, or shopper trials. The purpose is to test several implications of the proposed architecture under transparent assumptions. Table 5 lists the chosen values. They are illustrative inputs selected to span meaningful uncertainty regimes; they are not fitted to a retail dataset or attributed to a particular device.

**Table 5.** Inputs to the analytical and synthetic study

| Quantity               | Chosen value                   | Interpretation                         |
|------------------------|--------------------------------|----------------------------------------|
| Target widths          | 0.30, 0.60, 1.20 m             | Illustrative one-dimensional intervals |
| Sampled error $\sigma$ | 0.10, 0.20, 0.40, 0.80, 1.60 m | Assumed Gaussian standard deviations   |
| Samples per $\sigma$   | 100,000                        | Independent synthetic errors           |
| Random seed            | 20260915                       | One sequential NumPy generator         |
| Location-change rates  | 0.01, 0.03, 0.08 per hour      | Assumed constant hazards               |
| Route times            | 80 s and 110 s                 | Undisrupted travel times               |
| Disruption penalty     | 180 s                          | Added once to the short route          |
| Fusion correlation     | 0 to 1                         | Equal-variance scalar model            |
| Additional scan        | 6 s; recovery cost 60 s        | Illustrative break-even calculation    |

The first calculation treats correct shelf assignment as falling within a one-dimensional interval centered on the true target. The second models the probability that an observed location has not changed. The third compares two routes with different disruption costs. The fourth examines correlated estimates. An executed Monte Carlo check verifies the first calculation at five error levels. Together, the calculations support architectural requirements but do not estimate an integrated system’s expected real-world accuracy.

The distinction between analytical sensitivity and empirical evaluation is central. The figures show how specified mathematical models behave when their inputs change. They do not show measured gains from the proposed framework over deployed alternatives. In particular, the numerical effect of reducing localization error should not be interpreted as evidence that the architecture can achieve that reduction. Establishing achievable error requires the component and field studies described later.

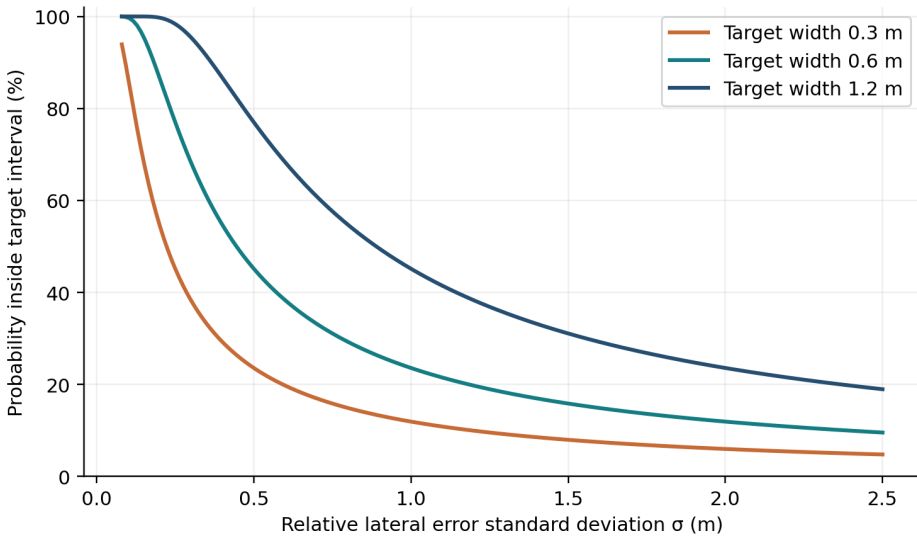
### 11.2 Relative Error and Target Width

Let the relative lateral error between an estimated target position and the correct shelf interval follow a zero-mean Gaussian distribution with standard deviation  $\sigma$ . Let  $w$  denote interval width. Correct assignment occurs when the error magnitude is no greater than half the interval width. Equation 6

gives the probability, where  $\Phi$  is the standard normal cumulative distribution function. The expression assumes no systematic bias, correct target identity, and a current map.

$$P_{\text{interval}} = P(|e| \leq w/2) = 2\Phi\left(\frac{w}{2\sigma}\right) - 1 \quad (6)$$

Figure 5 plots the probability for widths of 0.30 m, 0.60 m, and 1.20 m. The widths represent illustrative target intervals; they are not reported shelf dimensions from a store. At fixed uncertainty, a wider interval is easier to assign correctly. At fixed interval width, rising uncertainty reduces assignment probability. This demonstrates why an aisle-level location estimate cannot be treated as sufficient evidence for a narrow shelf target.



**Figure 5.** Analytical interval-assignment probability under unbiased Gaussian relative error. Curves are model calculations, not measured sensor performance.

For the 0.60 m interval, the model gives 99.73% probability at  $\sigma = 0.10$  m, 86.64% at 0.20 m, 54.67% at 0.40 m, 29.23% at 0.80 m, and 14.87% at 1.60 m. These are conditional spatial assignment probabilities. They exclude product-recognition errors, vertical-tier errors, occlusion, and stale layouts. Including such failures would generally reduce end-to-end success unless compensating verification actions were added.

The model also makes a distinction between global and relative accuracy. Local shelf geometry can sometimes support a useful relative estimate even when the global position is uncertain. Conversely, accurate global position does not guarantee correct local target assignment when the shelf map is wrong. The architecture consequently stores uncertainty at both scales and requires an explicit transform between them.

A biased error distribution would change the calculation. If the mean error is  $\mu$ , the probability becomes the difference between the normal cumulative probabilities at the two interval boundaries after subtracting  $\mu$ . A systematic offset can therefore cause repeated wrong-bay assignments that an

unbiased model would underestimate. Future experiments should report signed error and bias, not only a pooled root-mean-square value.

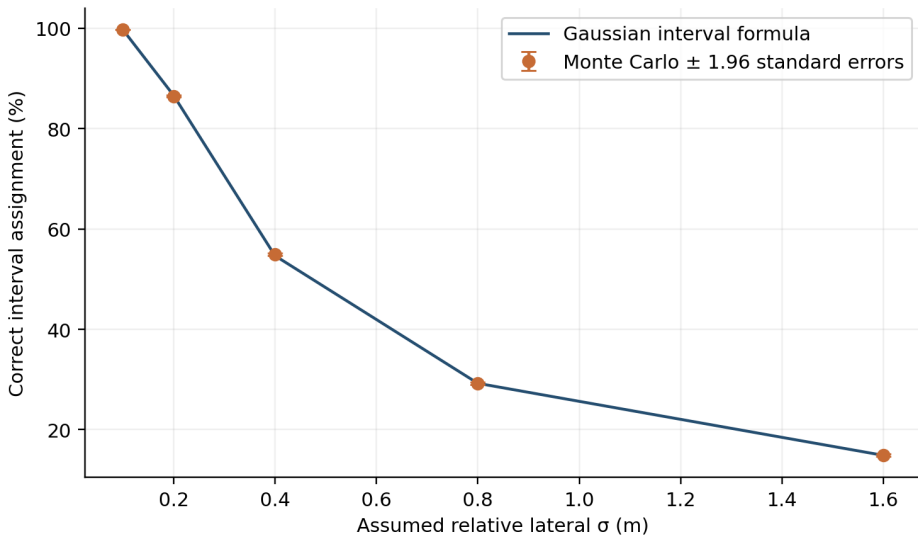
### 11.3 Executed Monte Carlo Check

The synthetic check draws 100,000 independent Gaussian errors for each of the five  $\sigma$  values listed above. A fixed NumPy generator seed of 20260915 is used, and each sample is counted as correct when its absolute value does not exceed 0.30 m. The sample proportion is compared with Equation 6. Table 6 gives the calculated probability, simulated proportion, and an approximate 95% interval half-width based on the binomial standard error.

**Table 6.** Synthetic interval-assignment results for a 0.60 m target

| $\sigma$ in m | Analytical % | Monte Carlo % | 95% half-width in pp |
|---------------|--------------|---------------|----------------------|
| 0.10          | 99.730       | 99.713        | 0.033                |
| 0.20          | 86.639       | 86.484        | 0.212                |
| 0.40          | 54.675       | 54.938        | 0.308                |
| 0.80          | 29.234       | 29.222        | 0.282                |
| 1.60          | 14.873       | 14.942        | 0.221                |

Figure 6 shows that the Monte Carlo estimates are close to the analytical curve at the tested settings. The largest absolute difference in this run is approximately 0.263 percentage points at  $\sigma = 0.40$  m. All five analytical values lie within the corresponding displayed approximate intervals in this particular run. This agreement checks the implementation of the stated sampling model. It does not validate the Gaussian assumption against retail observations.



**Figure 6.** Executed synthetic check of the Gaussian interval formula. Error bars show approximate 95% sampling intervals; they exclude model uncertainty.

The interval half-width quantifies sampling uncertainty conditional on the synthetic model. It does not include uncertainty about model form, target width, device behavior, or map correctness. Increasing the number of samples would narrow the Monte Carlo interval without resolving any of those external uncertainties. This is why very precise simulation output can still have limited practical predictive value.

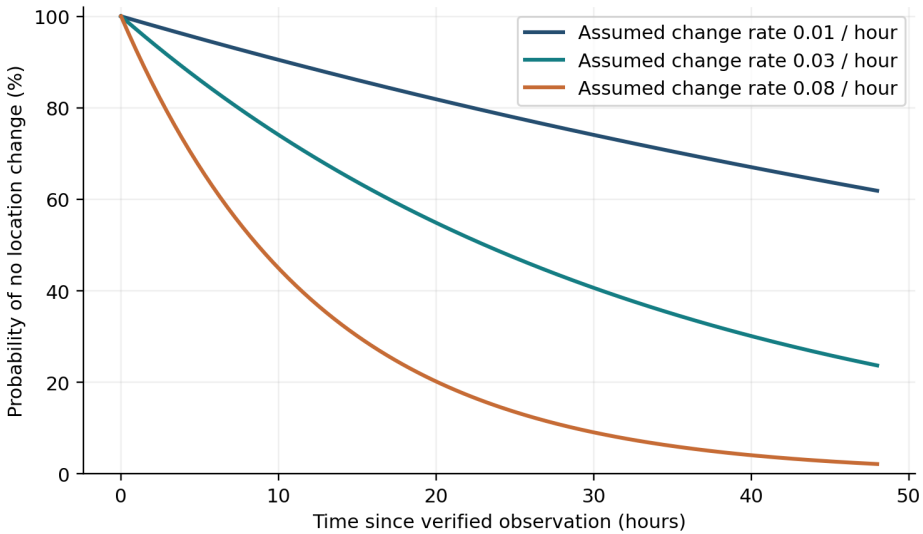
The simulation also treats errors as independent. Real walking trajectories contain temporal correlation, and several consecutive frames may share the same alignment error. A future trajectory-level simulation should model that dependence if it is used to compare recovery policies. Counting correlated frames as independent samples would exaggerate the effective sample size and produce misleading uncertainty intervals.

### 11.4 Freshness of Product-Location Evidence

Assume location changes occur according to a Poisson process with constant rate  $\lambda$  per hour, and treat the stored location as valid only if no change has occurred since verification. Equation 7 gives the no-change probability after  $\Delta t$  hours. It is a deliberately conservative validity proxy: a product might move away and return, but this simple model still treats any intervening change as invalidating the old assertion.

$$P_{\text{unchanged}}(\Delta t) = \exp(-\lambda \Delta t) \quad (7)$$

Figure 7 compares rates of 0.01, 0.03, and 0.08 changes per hour. At the middle rate, the no-change probability is approximately 83.53% after six hours, 69.77% after twelve hours, and 48.68% after twenty-four hours. These values are calculated assumptions, not observed retail relocation frequencies. The purpose is to show that evidence age can materially alter the reliability of a product-location assertion.



**Figure 7.** Analytical no-change probability at three assumed location-change rates. Rates are illustrative rather than measured.

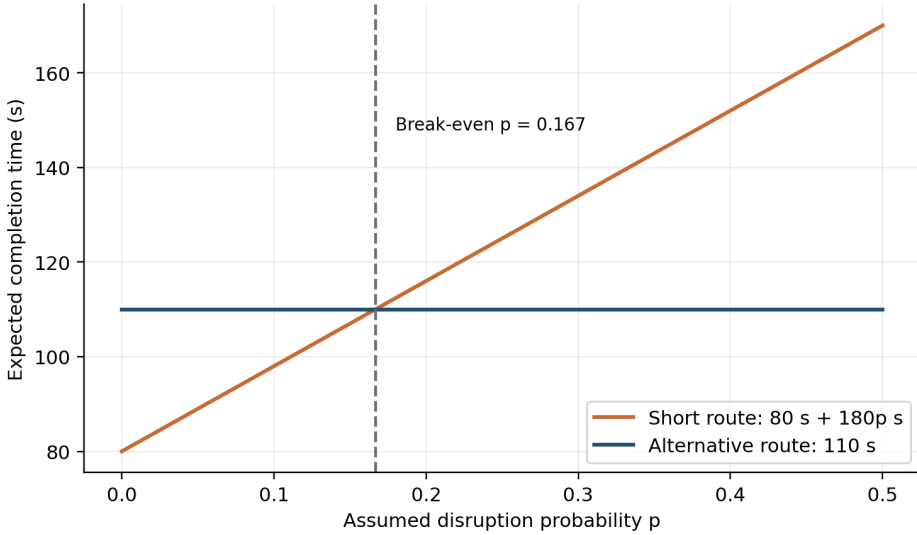
The freshness model should not be confused with stock availability. A product can remain assigned to the same shelf while all visible units are sold. Conversely, stock can be available in a back room while the shelf is empty. Location validity and availability therefore require different observations. The framework records them separately and avoids using one as proof of the other.

A deployment would need to estimate change behavior by display type and operational period. Promotional fixtures may differ from regular shelves, and replenishment periods may violate the constant-rate assumption. A survival model with covariates or a change-detection process could replace the simple exponential model. Such extensions would be justified by measured data, not by the smooth appearance of an illustrative curve.

### 11.5 Disruption-Aware Route Choice

Consider a short route requiring 80 seconds in the absence of disruption. A disruption adds 180 seconds, and its probability is  $p$ . A second route requires 110 seconds and, for this restricted example, has no additional disruption cost. Equation 8 compares their expected times. The alternative route becomes preferable when  $p$  exceeds one-sixth, approximately 0.167. Figure 8 shows the crossing point.

$$T_A = 80 + 180p, \quad T_B = 110, \quad T_B < T_A \Leftrightarrow p > \frac{1}{6} \quad (8)$$



**Figure 8.** Expected time of two illustrative routes as disruption probability changes. The dashed line marks the calculated break-even probability.

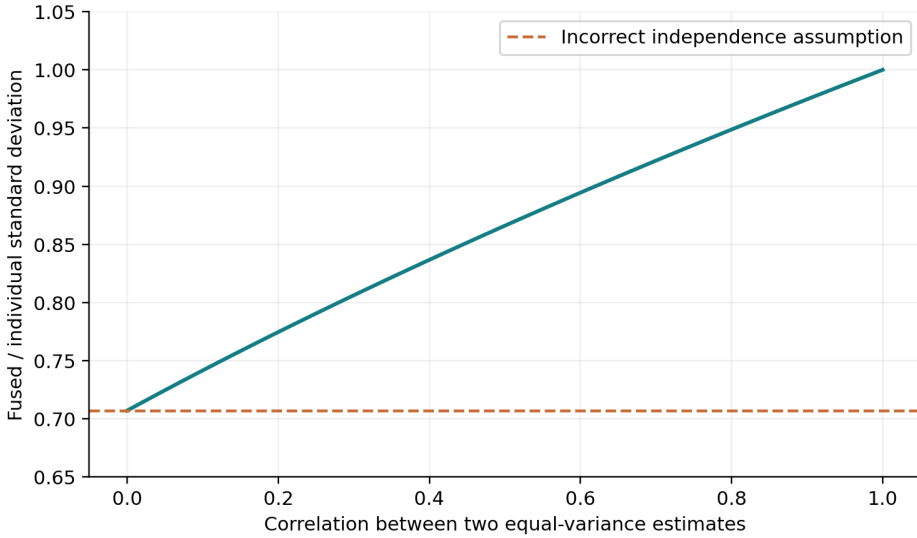
At  $p = 0.25$ , the short route has an expected time of 125 seconds, compared with 110 seconds for the alternative. The example demonstrates the value of considering consequences as well as path length. It does not estimate actual store congestion or prove a 15-second saving for users. If the alternative also has a disruption probability, its expected penalty must be included and the crossing point changes.

The example assumes that disruption is encountered once and has a fixed extra cost. In a real graph, interruptions can affect several edges and can be correlated. A display placed across an aisle may invalidate multiple paths simultaneously. The route planner should avoid summing independent penalties when the same event affects several edges. A scenario-based model or shared event identifier can represent that dependence.

## 11.6 Correlated Sensing

For two unbiased estimates of the same scalar quantity with equal variance  $\sigma^2$  and correlation  $\rho$ , their equally weighted average has the variance given in Equation 9. The resulting standard-deviation ratio is plotted in Figure 9. When  $\rho = 0$ , the ratio is approximately 0.707. When  $\rho = 0.8$ , it is approximately 0.949, so the uncertainty reduction is much smaller.

$$\text{Var}\left(\frac{X_1 + X_2}{2}\right) = \frac{\sigma^2(1 + \rho)}{2} \quad (9)$$



**Figure 9.** Uncertainty reduction from averaging two equal-variance estimates. Correlated inputs yield less reduction than an independence assumption predicts.

If the system assumes independence when the true correlation is 0.8, it reports a ratio near 0.707 even though the correct value for this model is near 0.949. The resulting overconfidence can cause a premature transition to exact shelf guidance. The numerical example supports retaining dependencies between observations. It does not imply that every sensor pair has equal variance or a known fixed correlation.

In practice, shared map errors may dominate individual sensor noise. Averaging several observations referenced to the same incorrectly aligned map does not remove the common offset. A field study should therefore include changes to map alignment and compare nominal uncertainty with actual error. Calibration should be assessed at the task-relevant spatial scale, including bay and tier assignment, rather than only at the level of raw sensor residuals.

### **11.7 Value of an Additional Scan**

An additional scan is worthwhile when its expected reduction in recovery cost exceeds its interaction cost. Under a simple model, let  $c$  be scan time,  $R$  the cost of recovering from an incorrect decision, and  $p_0$  and  $p_1$  the error probabilities before and after scanning. Equation 10 gives the break-even condition. It assumes the scan's only benefit is reducing this error probability and that the recovery cost is constant.

$$c < (p_0 - p_1)R \quad (10)$$

For an illustrative six-second scan and a sixty-second recovery cost, the error probability must fall by more than 0.10 for the scan to reduce expected time. This calculation discourages indiscriminate rescanning. A new view that provides little additional information can slow the task even when it marginally improves a model score. The controller should request observations that address a specific uncertainty, such as package size or shelf alignment.

The relevant cost may include effort and accessibility burden as well as elapsed time. A camera action that is easy for one user may be difficult for another. The policy should therefore allow alternate evidence paths and user-selected interaction modes. Future studies can estimate these costs from observed task behavior and user reports, but should avoid treating one group's average as a universal preference.

### **11.8 Interpretation of the Numerical Study**

The calculations support four design implications. Narrow targets require stronger relative localization than broad regions. Old product-location evidence can become unreliable even when its original observation was accurate. Correlated inputs offer less information than independent inputs. Finally, the fastest route or action in isolation may not minimize total task effort once error recovery is considered. These implications follow from the stated models and motivate the architecture's separation of stages and evidence types.

The study does not establish end-to-end superiority, achievable localization precision, recognition accuracy, or user satisfaction. No detector was trained, no real images were classified, and no shoppers were recruited for the reported numbers. The original contribution at this stage is a transparent design and a set of testable conditions. The next section defines the experiments needed to determine whether those conditions translate into practical benefits.

## **12. Prospective Empirical Evaluation Protocol**

### **12.1 Study Stages and Comparison Conditions**

A future evaluation should proceed through component tests, integrated technical trials, and a shopper study. Component tests establish whether perception and localization work under the intended conditions. Integrated trials establish that the data services and controller preserve those results through a complete task. The shopper study examines whether the resulting guidance helps people find the correct item with acceptable effort. Each stage should have a frozen protocol before its final test data are inspected.

The baseline conditions should isolate the proposed contribution. A catalog-and-aisle baseline measures the value of basic lookup. A static shortest-path baseline adds navigation without dynamic evidence or shelf verification. A shelf-recognition condition tests local visual assistance. The full framework adds evidence freshness, uncertainty-dependent transitions, and recovery. All conditions should receive the same catalog snapshot and comparable device resources so that infrastructure differences do not dominate the comparison.

Ablations should remove one mechanism at a time where feasible. Useful ablations include no freshness handling, no local verification, no uncertainty gating, and no dependency-aware fusion. Their purpose is to test whether each mechanism affects the outcomes it was designed to influence. They should not be interpreted as a ranking of arbitrary incomplete products. A negative ablation result may indicate an unnecessary component or a dataset that does not exercise the relevant failure mode.

## **12.2 Data Collection and Annotation**

A store dataset should include synchronized device observations, surveyed or independently verified pose references, product identifiers, shelf annotations, and map versions. Collection should span ordinary operating variation: different times, replenishment, temporary displays, lighting, and crowd levels. A single clean aisle captured once would not represent the changing conditions that motivate the architecture.

Ground truth should separate product identity from location. A package annotation needs an exact catalog identifier where visible evidence permits it, and an uncertainty or unreadable label when it does not. Annotators should not infer a hidden size from the planogram. Shelf annotations should include aisle side, bay, tier, and visibility. Disagreement should be resolved through a documented process, with a sample double-annotated to assess reliability.

Temporal data need careful handling. An image captured before a display change should not be evaluated against a later planogram as if the image were wrong. Every annotation should reference the relevant observation time and map version. The dataset should also retain negative cases, such as an empty target location or an unseen product. Removing such cases would bias evaluation toward tasks the system can already solve.

## **12.3 Splits and Generalization**

Training, validation, and test partitions should be separated by the unit that creates dependence. Randomly splitting adjacent video frames leaks nearly identical views into all partitions. A stronger design separates sessions and dates, and includes a held-out store where feasible. Product-level holdouts can assess recognition of unfamiliar SKUs, while packaging-version holdouts can test robustness to redesigned labels.

The validation set determines model selection, confidence thresholds, and route-policy parameters. The final test set should remain unused until those choices are frozen. If the system is adapted to a new store, the amount of adaptation data and staff effort should be reported. “Generalization” should not describe a condition in which the complete test-store layout and product imagery were incorporated into training without disclosure.

Device generalization requires its own split. Camera quality, image stabilization, inertial behavior, and compute limits can change the pipeline. A model that succeeds on a developer handset may fail on a lower-resource phone. The report should identify supported device classes and show performance by class rather than only a pooled average. Unsupported configurations should be reported as such.

## **12.4 Outcome Measures**

The primary outcome is verified discovery of the requested SKU within a prespecified task window. The numerator counts tasks that meet the exact item constraints and obtain an appropriate confirmation; the denominator includes all assigned tasks. Aborted, unresolved, and incorrect tasks remain visible. Retrieval by the user can be a separate outcome when reaching or handling the item matters.

Table 7 defines the main metrics. No numeric target in this table is presented as an achieved result. The metrics are selected to avoid hiding failure behind a single aggregate score. A system can have low average walking distance but high false confirmations, or high confirmation accuracy but very low coverage. Both cases require different design changes.

**Table 7.** Proposed empirical outcome measures

| Metric             | Definition                                           | Reporting requirement                        |
|--------------------|------------------------------------------------------|----------------------------------------------|
| Verified discovery | Correct requested SKU confirmed / all assigned tasks | Include unresolved and incorrect tasks       |
| False confirmation | Wrong SKU confirmations / all confirmations          | Report with confirmation coverage            |
| Task time          | Elapsed time to outcome or task limit                | Include failures and censoring rules         |
| Path efficiency    | Success-weighted feasible path ratio                 | Use identical hard constraints               |
| Localization error | Pose error against independent reference             | Median, tails, bias, and aisle errors        |
| Shelf assignment   | Correct bay and tier among annotated cases           | Report both conditional and end-to-end rates |
| Calibration        | Agreement of confidence and correctness              | Define the event and subgroup                |
| Recovery cost      | Time and interactions after detected failure         | Separate recovery types                      |
| Resource use       | Energy and latency per complete task                 | Report device and workload                   |
| Maintenance effort | Staff time and update delay                          | Include map and catalog changes              |

Equation 11 defines a success-weighted path measure adapted to this task. Success  $S$  is one only for verified correct discovery,  $L$  is the shortest feasible reference path under the same hard constraints, and  $P$  is the distance actually traveled. A task that ends at the wrong product receives zero. The metric should be reported beside completion time and interaction cost, because it does not account for time spent inspecting a shelf.

$$\text{SPL}_{\text{item}} = \frac{1}{n} \sum_{i=1}^n S_i \frac{L_i}{\max(P_i, L_i)} \quad (11)$$

The ratio is defined for positive reference distance. A successful task requiring no movement is assigned a path ratio of one.

## 12.5 Statistical Analysis

The statistical plan should recognize clustering by participant, store, and product. Repeated attempts by one participant are not independent shoppers, and neighboring SKUs may share visual difficulty. A mixed-effects analysis or a cluster-aware resampling method can account for these dependencies. The report should give effect sizes and uncertainty intervals rather than relying only on whether a test crosses a significance threshold.

For paired conditions, task order should be counterbalanced to reduce learning effects. A participant who has already found a product may remember its location in a later condition. Different but matched item sets, or a design that explicitly models order, can reduce this problem. The study should record prior store familiarity because it can strongly influence navigation time independently of the system.

Sample size should be chosen after a pilot estimates task variability and clustering, using a prespecified smallest effect of practical interest. This manuscript does not invent a participant count or a power result without those inputs. A future report should distinguish the pilot from the confirmatory test and disclose any changes made after the pilot. Exploratory analyses should be labeled separately.

## 12.6 Calibration and Selective Prediction

Calibration analysis should compare predicted confidence with observed correctness for the exact event being claimed. Shelf identity calibration should not be computed using a broader product-category label. Reliability plots, proper scoring rules, and subgroup analysis can reveal whether a probability

estimate is useful. Calibration should be assessed after store and packaging changes, not only on an in-distribution validation set.

Selective prediction analysis varies the confirmation threshold and records coverage and error among accepted predictions. This makes the cost of abstention visible. A system that confirms only the easiest products may have impressive accepted accuracy while leaving most shoppers unsupported. The evaluation should also measure whether a declined confirmation leads to a useful fallback or simply ends the task.

Conformal prediction could provide candidate sets under suitable assumptions [11], but changes between calibration and deployment can weaken the intended coverage interpretation. The framework therefore treats uncertainty guarantees as conditional on their assumptions. A store-specific recalibration process should be evaluated as an operational procedure, including its data and maintenance costs.

### **12.7 User experience and Accessibility Outcomes**

The shopper study should measure workload, clarity, perceived control, and the number of times instructions require repetition. Observation can distinguish a user who ignores a correct instruction from a user who cannot interpret it. Interviews should ask about concrete episodes, such as an unclear tier instruction or an unnecessary rescans, rather than only requesting an overall satisfaction score.

Accessibility evaluation should include participants who use the relevant interaction modes and mobility supports. Their outcomes should not be collapsed into a single average that conceals systematic difficulties. Researchers should document accommodations, task modifications, and situations in which staff assistance was necessary. A request for assistance can be a successful fallback even when fully independent completion is not achieved, provided that the outcome definition is explicit.

The study should also examine distraction. A detailed screen overlay may improve shelf matching while reducing attention to surrounding shoppers or obstacles. The system should avoid encouraging continuous screen viewing during movement. A suitable design can separate movement instructions from stationary inspection, but the benefit of that design needs observation in realistic tasks.

## **13. Privacy Security and Operational Integrity**

The assistance task does not require identifying bystanders. Camera processing should focus on shelf regions and avoid retaining unrelated imagery where possible. A deployment should distinguish temporary processing needed for the current task from collection for model development. The latter requires a separate, understandable choice and a defined retention practice. The framework can operate with local image processing and structured observations, although the performance of that configuration remains to be measured.

Navigation logs can reveal shopping behavior even without raw video. A sequence of visited shelves and requested items may be sensitive. The proposed data design therefore keeps operational identifiers separate from customer identity and limits persistent task records to what the evaluation or service actually needs. Aggregation should be considered carefully because rare routes or product combinations may still be distinctive. The paper makes no claim that removing a name alone anonymizes a trajectory.

Integrity matters as much as confidentiality. A false catalog update, altered shelf location, or maliciously submitted observation could direct users to the wrong place. Store-authorized updates should be authenticated and versioned, and user observations should not automatically become

authoritative map corrections. Reconciliation can require corroboration or staff review according to the impact of the change. The system should retain enough provenance to reverse a faulty update.

The interface should also protect the integrity of the shopper's objective. Sponsored placement must not silently change the target or route objective. If alternatives or promotions are displayed, they should remain distinguishable from the requested-item guidance. Research evaluation should disable or explicitly account for such commercial influences because they can confound route length, item choice, and perceived usefulness.

Table 8 links failure conditions to observable symptoms and proposed responses. It is a design risk matrix rather than a record of incidents. Its purpose is to make recovery behavior testable, particularly when failures span several modules.

**Table 8.** Proposed failure responses and evaluation checks

| Failure                  | Observable symptom                | Proposed response                             | Evaluation check               |
|--------------------------|-----------------------------------|-----------------------------------------------|--------------------------------|
| Wrong aisle alignment    | Pose conflicts with landmark      | Pause precise guidance and relocalize         | Time to detect and recover     |
| Outdated location        | Expected product absent           | Check alternative display or refresh evidence | Stale-location task success    |
| Near-duplicate package   | Attributes conflict or unreadable | Request distinguishing label view             | Exact-variant error rate       |
| Correlated evidence      | Several scores share one source   | Preserve dependence in fusion                 | Confidence versus actual error |
| Blocked approach         | Route edge becomes infeasible     | Replan or request assistance                  | Constraint violation rate      |
| Network interruption     | Update unavailable                | Use bounded cached guidance                   | Offline task outcome           |
| Repeated unhelpful scans | Uncertainty does not decrease     | Change evidence request or stop               | Interaction burden             |

## 14. Discussion

### 14.1 Contribution of Hierarchical Evidence Control

The framework's main contribution is the link between semantic resolution, spatial guidance, and evidence-dependent confirmation. These functions can be implemented using different model families, but their interface must preserve what has actually been established. A product detector establishes a region hypothesis. A catalog match establishes a semantic candidate. A pose estimate establishes a spatial belief. The controller must decide whether those pieces support an instruction at the requested level of detail.

This interpretation changes how progress is measured. Arriving at an aisle is intermediate progress rather than task completion. A confident category match is useful but does not complete an exact-variant request. A rescan can be productive when it resolves a known ambiguity, even though it increases immediate interaction time. These distinctions allow the evaluation to attribute improvement to the intended mechanism instead of treating every extra step as inefficiency.

The analytical results show why this separation is necessary under the chosen models. Relative error and target width determine a spatial assignment limit. Freshness affects whether the target location should still be trusted. Correlation limits the information gained by combining estimates. None of these effects can be repaired merely by displaying more precise coordinates. The interface needs a policy for when to be specific and when to seek evidence.

## **14.2 Integration Cost and Practical Value**

The proposed architecture creates additional requirements for data management and operations. Maintaining a versioned graph, evidence provenance, and calibrated confidence is more involved than returning a static aisle number. Those costs are justified only if they produce measurable gains in verified discovery, reduced recovery effort, or better support for difficult tasks. The empirical study should therefore include a simple baseline rather than assuming that a more elaborate system is automatically better.

The framework may be most useful where exact item distinctions and layout changes are common enough to cause meaningful search difficulty. In a small familiar store, a catalog lookup may already solve most tasks. The design should permit a lightweight configuration and escalate sensing only when needed. This also provides a practical deployment path in which richer shelf data are collected for selected departments rather than required everywhere at launch.

A retailer's incentive may differ from a shopper's. Longer routes can increase exposure to products, whereas the shopper may want a fast direct task. The optimization objective should be explicit and evaluation should measure the shopper's stated goal. Otherwise an apparently intelligent route could serve an undisclosed commercial objective. Clear objectives are also necessary for comparing systems fairly.

## **15. Limitations and Future Work**

The largest limitation is the absence of an implemented and empirically evaluated integrated system. The manuscript specifies components and decision rules, but it does not show that a particular model combination meets latency, accuracy, or resource requirements. Its numerical study isolates simplified mechanisms. It cannot estimate overall task success or establish that the proposed framework is superior to existing retail applications.

The Gaussian interval model is one-dimensional and unbiased. Real shelf errors can be multimodal, biased, correlated over time, and affected by wrong-aisle associations. Vertical uncertainty and product-identity uncertainty are omitted. A future model should include a joint distribution over aisle, bay, tier, and SKU, with recovery actions that change the distribution. Empirical measurements should guide that model rather than merely expanding the number of simulated variables.

The freshness model assumes a constant hazard and treats any location change as invalidation. Store changes may follow schedules and may create multiple simultaneous displays. The two-route example assumes a single fixed disruption penalty. The fusion example considers equal-variance scalar estimates. These restrictions make the calculations interpretable but limit their predictive scope. Field data are required to test whether richer models improve decisions enough to justify their complexity.

The literature review is selective and includes foundational component research rather than an exhaustive comparison of current retail products. The proposed combination should therefore be treated as a research design, not a proven claim of being the first framework of its kind. A submission to a particular venue would benefit from a focused update of closely related systems and a benchmark aligned with that venue's expectations.

Future work should prioritize a small reproducible end-to-end prototype, a carefully annotated store dataset, and independent verification of exact SKU outcomes. The most informative initial study would include deliberate planogram changes and near-duplicate products, because those conditions exercise the framework's central claims. Accessibility co-design and evaluation should run alongside technical development, not be deferred until the interface is fixed.

## 16. Conclusion

This paper proposes an AI-driven, context-aware framework for indoor item discovery and shelf-level navigation. Its central design decision is to separate global movement, local inspection, and exact-item confirmation while connecting them through a versioned store graph and an evidence ledger. The framework accommodates ambiguous product requests, uncertain pose, outdated locations, sensor dependence, and user-selected interaction constraints.

The analytical evaluation demonstrates how target width, error magnitude, evidence age, and disruption cost change the decisions a navigation assistant should make under explicit assumptions. The executed synthetic check confirms the implementation of the interval-assignment calculation. These results support the architecture's rationale but do not establish real-store performance. The proposed empirical protocol defines how to test that performance through exact item success, calibration, task effort, generalization, and maintenance cost.

Reliable assistance requires more than a plausible route or a confident image label. It requires a defensible connection between the shopper's request, the current physical environment, and the evidence used to confirm completion. The framework makes that connection explicit and testable.

## 17. Data and Reproducibility Statement

No proprietary store dataset or participant data were used in the reported analysis. All quantitative results in Section 11 arise from the stated formulas and synthetic sampling assumptions. Appendix A contains a compact executable script that reproduces the numerical table and the principal analytical values. The diagrams are original conceptual illustrations. An empirical deployment, trained recognition model, and real-store evaluation dataset are future work.

## References

- [1] E. Goldman, R. Herzig, A. Eisenschtat, O. Ratzon, I. Levi, J. Goldberger, and T. Hassner, "Precise Detection in Densely Packed Scenes," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019. <https://arxiv.org/abs/1904.00853>
- [2] M. Klasson, C. Zhang, and H. Kjellström, "A Hierarchical Grocery Store Image Dataset with Visual and Semantic Labels," IEEE Winter Conference on Applications of Computer Vision, 2019. <https://arxiv.org/abs/1901.00711>
- [3] C. Campos, R. Elvira, J. J. Gómez Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM," IEEE Transactions on Robotics, vol. 37, no. 6, pp. 1874–1890, 2021. <https://arxiv.org/abs/2007.11898>
- [4] T. Qin, P. Li, and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," IEEE Transactions on Robotics, 2018. <https://doi.org/10.1109/TRO.2018.2853729>
- [5] A. Radford et al., "Learning Transferable Visual Models From Natural Language Supervision," Proceedings of the 38th International Conference on Machine Learning, PMLR, vol. 139, pp. 8748–8763, 2021. <https://proceedings.mlr.press/v139/radford21a.html>
- [6] X.-S. Wei, Q. Cui, L. Yang, P. Wang, and L. Liu, "RPC: A Large-Scale Retail Product Checkout Dataset," arXiv preprint arXiv:1901.07249, 2019. <https://arxiv.org/abs/1901.07249>
- [7] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End Object Detection with Transformers," arXiv preprint arXiv:2005.12872, 2020. <https://arxiv.org/abs/2005.12872>
- [8] S. Liu et al., "Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection," arXiv:2303.05499, 2023, revised 2024. <https://arxiv.org/abs/2303.05499>
- [9] A. K. Dey, "Understanding and Using Context," Personal and Ubiquitous Computing, vol. 5, pp. 4–7, 2001. <https://doi.org/10.1007/s007790170019>

- [10] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," Proceedings of the 34th International Conference on Machine Learning, PMLR, vol. 70, pp. 1321–1330, 2017. <https://proceedings.mlr.press/v70/guo17a.html>
- [11] A. N. Angelopoulos and S. Bates, "A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification," arXiv preprint arXiv:2107.07511, 2021. <https://arxiv.org/abs/2107.07511>
- [12] P. Anderson et al., "On Evaluation of Embodied Navigation Agents," arXiv preprint arXiv:1807.06757, 2018. <https://arxiv.org/abs/1807.06757>
- [13] A. G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv preprint arXiv:1704.04861, 2017. <https://arxiv.org/abs/1704.04861>
- [14] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, PMLR, vol. 54, pp. 1273–1282, 2017. <https://proceedings.mlr.press/v54/mcmahan17a.html>