VOL. 12 NO. 9,
MAY, 2026
E-ISSN 3026-958X
P-ISSN 3027-1169

African Scholar Publications & Research International
www.africanscholarpub.com

Journal of Science Innovation & Technology Research (JSITR)

LLM-Assisted UI Element Grounding and Design-Rationale Cards for Mobile Interfaces

Nathan Han
Computer Science, University of Southern California, Los Angeles, CA, USA
Corresponding Author: nathan_han_usc@gmail.com
DOI: https://doi.org/10.70382/ajsitr.v12i9.089

Abstract

User-interface screenshots contain dense visual and semantic structure: text labels, icons, buttons, lists, cards, toolbars, form inputs, and navigation regions frequently coexist on one screen. This study presents a compact, evidence-fused pipeline for LLM-assisted UI element grounding and design-rationale cards. The pipeline proposes UI elements from screenshot-aligned hierarchy candidates, classifies their control types, grounds natural referring expressions to element boxes, infers layout-level design tags, and produces concise rationale cards whose language is constrained by structured evidence. Experiments used 1,460 mobile UI screenshots and corresponding JSON view hierarchies, yielding 31,968 parsed nodes and 30,101 labeled elements across 25 component categories. All tasks used screen-disjoint training, validation, and test partitions. The evidence-fused detector reached 0.950 AP50 and 0.998 Recall@100 on the test screens. The evidence-fused control classifier reached 0.932 accuracy and 0.942 weighted F1. Across 2,161 generated referring trials, multi-evidence grounding reached 0.876 Top-1 accuracy, 0.979 Top-3 accuracy, and 0.928 mean reciprocal rank. Predicted component labels substantially improved layout tagging over geometry-only rules for forms, navigation, cards, image-heavy screens, and advertisements. Across 643 cards per output condition, the evidence-card generator preserved role, position, and action evidence with a contradiction rate of 0.000 under deterministic consistency checks. This last result establishes internal agreement with the evidence record, not human-rated usefulness or trust. Overall, the findings show that screenshot geometry and hierarchy metadata can be combined into an inspectable grounding-and-explanation layer, while

also indicating that hierarchy dependence, rare-class imbalance, generated grounding phrases, and the absence of user evaluation limit broader deployment claims.

Keywords: Mobile UI understanding, UI element grounding, design-rationale cards, layout analysis, large language models.

Introduction

Mobile interfaces are visually organized documents whose regions have functional as well as perceptual meaning. A single screenshot may contain a toolbar, tab strip, list, product card, form field, advertisement, modal, and icon row. An automated assistant that interacts with such a screen must determine not only where an element is located but also what role it plays and what visible or structural evidence supports that interpretation. Large UI corpora have made this problem measurable by connecting screenshots to view hierarchies, element semantics, and design topics (Deka et al., 2017; Leiva et al., 2020; Wu et al., 2023). Recent work on converting sketches into interactive prototypes further illustrates why structural correspondence and visual consistency must be evaluated together when interface representations move across modalities (Chen & Li, 2025).

The present study addresses element-level grounding with an evidence-linked explanation layer. Element grounding connects a referring expression such as "top search field," "bottom navigation tab," or "profile image on the card" to one element box. A design-rationale card then makes the selection inspectable by reporting the element's role, supporting evidence, position, likely action, and contribution to the screen design. This combined problem matters for accessibility support, app testing, design search, screen summarization, and agent planning. Mobile-screen research also suggests that interface analysis should remain sensitive to app-specific context and design adaptation rather than assuming that all screens share one generic visual grammar (J. Zhang, 2025).

Language models can make structured UI outputs easier to read, but unconstrained generation can also produce fluent claims that are not tied to the selected element. Transformer and large-language-model research established the capacity of attention-based models to generate and adapt language from context (Brown et al., 2020; Vaswani et al., 2017), yet fluency alone does not guarantee faithful grounding. This concern is particularly salient for interfaces, where an unsupported action statement may turn a decorative image into an apparent button or may confuse a container with its clickable child. Interface-microcopy studies have likewise shown that semantic interpretation and explanation must be linked to the actual textual evidence presented to the model (Xu et al., 2025). For this reason, the language layer in this study is evidence-bound. It may verbalize only fields produced by the detector, classifier, grounding scorer, and layout analyzer. Closely related design-rationale research has treated layout metadata

as a source for explainable decision cards rather than as a prompt for unrestricted commentary (Wu et al., 2025), while evidence-card interfaces have emphasized provenance, ranking transparency, and visible evidence hierarchy (Jin, 2025b). The present pipeline applies that principle at the level of individual mobile UI elements: geometry establishes where an object is, hierarchy metadata helps determine what it is, grounding links a phrase to the object, and a controlled language layer explains the selection.

The empirical focus is intentionally compact and fully inspectable. View hierarchies commonly contain invisible nodes, duplicate nested bounds, text-only items, decorative images, and interactive containers whose children do not align neatly with the visible target. Rather than assuming clean annotations, the method recursively extracts hierarchy candidates, filters them by visibility and geometry, maps all boxes to screenshot coordinates, and evaluates each stage using a screen-disjoint split. The study addresses three research questions:

1. How far can a lightweight evidence-fused pipeline go on hierarchy-backed mobile screenshots? 2. For which tasks and interface categories does geometry alone fail? 3. Can design-rationale cards remain concise enough for inspection while staying tied to element-level evidence?

The paper makes four contributions. First, it presents one pipeline for element proposal, control classification, referring-expression grounding, layout-tag inference, and evidence-bound rationale-card generation. Second, it evaluates that pipeline on 1,460 screens and 30,101 labeled elements through 11 tables and 8 figures. Third, it compares geometry-only, metadata-only, and evidence-fused configurations to isolate which evidence sources contribute at each stage. Fourth, it defines a rationale-card schema whose statements can be checked against structured screen evidence. The contribution is a reproducible structured baseline, not a claim that a named LLM was benchmarked, that screenshot-only detection was solved, or that human trust and task performance were established.

Literature Review

UI Corpora, Representations, and Layout Semantics

Rico established a large mobile-app resource that connects screenshots, view hierarchies, and interaction traces for data-driven design applications (Deka et al., 2017). Enrico subsequently provided a curated mobile-screen collection organized around interpretable design topics (Leiva et al., 2020). For web interfaces, WebUI paired visual pages with web semantics and showed that structural information can improve element detection, screen classification, and screen similarity (Wu et al., 2023). These corpora differ in platform and scale, but they share a central lesson: a UI element is not adequately represented by pixels alone when developer-side semantics and hierarchy structure are available.

Search and representation-learning studies have reinforced this point. VINS treated mobile UI retrieval as a domain-specific visual-search problem rather than an ordinary image-search task (Bunian et al., 2021). Work on denoising raw mobile layouts showed that hierarchy-derived datasets require explicit data-quality control because automatically collected structures

contain noise and redundant nodes (Li et al., 2022). UIBert learned multimodal UI representations from visual, textual, and structural signals (Bai et al., 2021), whereas Screen2Vec represented both whole GUI screens and individual components in a semantic embedding space (Li et al., 2021). Lexi further modeled the "language" of UI structure through self-supervision (Banerjee et al., 2022), and Pix2Struct demonstrated the value of screenshot parsing as a visual-language pretraining objective (Lee et al., 2023).

This literature motivates the present separation between location, type, and screen context. Geometry is informative for stable bands, repeated rows, and relative position, but it is often insufficient when two controls occupy similar rectangles. Layout-aware progressive rendering provides an analogous example: region geometry helps prioritize visual slices, while layout semantics determine which slices are functionally important (Wang et al., 2025). Similarly, prototype-generation research evaluates both structural and visual consistency because a geometrically plausible screen may still have the wrong functional organization (Chen & Li, 2025). The current study therefore treats geometry as one evidence source rather than as a complete account of interface semantics.

Element text also contributes meaning that cannot be recovered from shape. Dark-pattern detection from interface microcopy demonstrates that words, control context, and explanation should be analyzed together (Xu et al., 2025). Mobile-review and screen-analysis work likewise connects observed interface features to adaptive design decisions instead of treating the screenshot as an isolated image (J. Zhang, 2025). These findings support a control classifier that combines location and size with class names, resource identifiers, text, icon labels, and clickability.

Element Detection, Ranking, and Grounding

General object-detection research supplies useful evaluation concepts for UI element proposals. Faster R-CNN formalized region proposals within an end-to-end detection framework (Ren et al., 2015); focal loss addressed dense detection under class imbalance (Lin et al., 2017); YOLOv3 demonstrated a contrasting one-stage design (Redmon & Farhadi, 2018); and the PASCAL VOC challenge established widely used intersection-over-union and average-precision conventions (Everingham et al., 2010). The present study borrows AP50 and recall-at-K reporting, but its candidates are hierarchy-backed and must not be interpreted as a pixel-only detector benchmark.

Referring-expression grounding is also a ranking problem. A useful grounding system should retrieve the correct target early while retaining alternatives when several elements share a label or region. Research on hybrid retrieval and listwise reranking shows that candidate generation and final ranking can benefit from different evidence sources (Meng et al., 2026). Evidence-calibrated retrieval further distinguishes broad candidate coverage from selective downstream commitment when the available record is incomplete (Zhong et al., 2025). These studies operate in textual domains, but the ranking distinction transfers methodologically to UI

grounding: high proposal recall is valuable only if later semantic and spatial evidence can prioritize the intended element.

Reliable ranking further requires provenance. Evidence-chain RAG links source attribution to deterministic explanations rather than allowing the narrative layer to obscure how an answer was selected (Jin, 2025a). Retrieval-grounded failure analysis similarly separates retrieved operational evidence from the wording of a generated narrative (Sun et al., 2026). Prototype-based explanation offers another relevant pattern: retrieved normal examples can constrain how an anomalous case is described without implying that retrieval itself proves the diagnosis (Xin, 2025). In the present setting, element label, token overlap, clickability, region, and geometry form an inspectable evidence chain for the selected target.

Context-sensitive retrieval is also important when the same surface words can support different actions. A therapist-facing copilot, for example, ranks evidence from multi-turn context before presenting support options (Zhang & Zhang, 2025a). UI grounding has an analogous ambiguity: "bottom icon" may refer to several visually similar candidates, and the correct target may depend on whether it is clickable, belongs to navigation, or matches nearby text. This motivates the comparison among random, position-only, token-only, and multi-evidence scorers.

Design Rationale and Evidence-Card Interfaces

Design-rationale systems were developed to record not only a design outcome but also the reasoning and alternatives behind it (Lee, 1997). Human-AI guidance later emphasized visibility of system status, support for efficient correction, and alignment between explanations and user goals (Amershi et al., 2019). More general interpretability research has warned that explanation quality must be evaluated against explicit purposes rather than assumed from plausibility (Doshi-Velez & Kim, 2017). Reviews of human-XAI interfaces similarly distinguish faithful evidence presentation from visually persuasive but weakly supported explanation (Chromik & Butz, 2021).

Recent card-based interface studies operationalize these ideas through structured fields. Text-grounded design-rationale cards transform layout metadata into inspectable UI/UX decisions (Wu et al., 2025). Scientific-search evidence cards organize source evidence, ranking information, and visual hierarchy for retrieval transparency (Jin, 2025b). Explainable recommendation cards use an image, relation reason, operational or confidence cues, and an action control to separate evidence from recommendation (B. Zhang et al., 2025). Review-grounded recommendation research adds an explicit faithfulness check between the evidence record and the displayed reason (Chang et al., 2026). These systems are more directly relevant to the current work than free-form screenshot captioning because their output can be checked field by field.

The same structured-card pattern appears across domains, although cross-domain examples should be used as design analogies rather than as direct validation of mobile UI grounding. Privacy and data-integrity risk cards emphasize calibrated wording, visible evidence, and

human oversight rather than autonomous action (Su, Rao, et al., 2026). Infrastructure visual briefs and designer-editable creative briefs convert upstream evidence into decision-oriented interface fields (Liu et al., 2025; Mu et al., 2026), while a DevOps copilot study pairs retrieved runbook evidence with explicit command-safety checks (B. Zhang et al., 2026). Evidence-linked counseling cards similarly show how ranked support can be presented without collapsing evidence and recommendation into one sentence (Zhang & Zhang, 2025b).

Audience and task also shape the appropriate explanation. Teacher-facing LLM-ready explanations are intended for instructional intervention rather than for end-user autonomy (J. Zhang, 2026). Workload-explanation pipelines distinguish upstream computational forecasts from the language used to communicate them (He et al., 2026). These distinctions motivate the current five-field card—role, evidence, position, likely action, and design rationale—and the explicit separation between computational consistency and human-rated effectiveness.

Calibration, Abstention, and Evidence Sufficiency

An evidence-bound interface should communicate when the available record is insufficient. Trust-calibrated RAG studies evaluate answerability and abstention rather than assuming that every retrieved context supports a response (Chen & Xu, 2026). Evidence-calibrated retrieval likewise connects retrieval coverage, abstention, and hallucination proxies (Zhong et al., 2025). Claim-verification work ranks evidence before producing a narrative conclusion (Su, Chen, et al., 2026), and conformal alert control illustrates how calibrated escalation can be paired with a structured incident ticket (Xin, 2026).

Compact quality signals can make these decisions usable at the interface layer. Edge-oriented quality predictors have been proposed as sources of concise, machine-readable quality tokens rather than as replacements for downstream judgment (Zhou et al., 2025). In the current study, the card consistency checks play a narrower role: they test whether generated fields agree with upstream element evidence. They do not estimate human trust, semantic completeness, or the probability that the selected UI element is correct.

Synthesis and Research Gap

Prior work establishes strong foundations for UI corpora, multimodal representations, ranking, design rationale, and evidence-oriented cards. However, these strands are often evaluated separately: an element detector may not expose a readable rationale, while a card interface may assume that the underlying item has already been correctly selected. The gap addressed here is a single, auditable chain from hierarchy-backed proposal through control classification and phrase grounding to layout interpretation and evidence-bound language. The study tests where geometry is sufficient, where semantics add value, and what deterministic consistency can—and cannot—establish about explanation cards.

Materials and Methods

Study design and corpus

The experiments used 1,460 mobile UI screenshots and their corresponding JSON view hierarchies. Each hierarchy was parsed recursively, and every node with valid bounds, a visible region, and a component label was converted into an element record. Bounds were clipped to the screenshot frame, normalized to relative coordinates, and retained in both pixel and normalized forms. The parser recorded parent-child depth, number of children, clickability, text length, resource-ID length, class-name length, icon class, component label, screen orientation, and coarse screen region. This procedure produced 31,968 parsed hierarchy nodes and 30,101 labeled elements across 25 component labels.

The final manuscript should supply the corpus release identifier, download location, license, and collection or version information in a data-availability statement. The present report describes the observed corpus and does not infer that it is identical to Rico, Enrico, WebUI, or another named release.

Screen-Disjoint Partition and Preprocessing

Screen identifiers—not element identifiers—determined training, validation, and test membership. The training set contained 1,022 screens and 21,447 labeled elements; validation contained 219 screens and 3,974 elements; and testing contained 219 screens and 4,680 elements. This screen-disjoint design prevented near-duplicate widgets from the same screen from appearing in more than one partition. Candidate geometry was clipped to the valid screenshot area, and widths, heights, areas, and centers were normalized by the screenshot dimensions. Coarse location features represented top, middle, bottom, left, center, and right regions.

Candidate Proposals and Element Detection

The proposal stage retained original hierarchy-node bounds and supplemented them with structural anchors. Anchor rules generated candidate boxes for toolbar-height strips, full-width list rows, card-like rectangles, input-like horizontal strips, and bottom-navigation bands. A geometry score used normalized area, aspect ratio, border proximity, coarse region, and candidate source type. A metadata-prior score used component label, clickability, text length, and resource-ID tokens. The evidence-fused proposal score combined geometry and metadata using a regularized logistic classifier trained on the training partition and calibrated on validation candidates.

Detection was evaluated on the test partition using AP50, Precision@100, Recall@50, Recall@100, Recall@140, and mean best intersection over union. AP50 used an IoU threshold of 0.50. Recall@K measured the fraction of ground-truth elements recovered among the top K candidates on each screen. These metrics follow established object-detection and proposal-evaluation practice (Everingham et al., 2010; Lin et al., 2017; Redmon & Farhadi, 2018; Ren et al., 2015), while the interpretation remained specific to hierarchy-backed proposals.

Control Classification

The control classifier predicted a component label for each visible element. Four feature configurations were compared:

1. Geometry only: normalized center, width, height, area, aspect ratio, coarse screen region, hierarchy depth, child count, and clickability.
2. String metadata: tokenized class name, resource ID, visible text, icon class, and shape or category flags.
3. Geometry plus metadata: concatenation of the first two feature families.
4. Evidence fused: geometry and metadata augmented with detector confidence and layout context.

The reported classifier was a linear support-vector classifier with sparse text features, implemented with scikit-learn. This choice preserves a relatively auditable decision surface for short metadata strings and follows the established use of support-vector methods for high-dimensional classification (Cortes & Vapnik, 1995; Pedregosa et al., 2011). Less frequent labels remained in corpus summaries; for classification, extremely small classes were grouped into Other only when their support was too low for stable per-class testing.

Referring-Expression Grounding

Grounding trials were generated from held-out test elements with enough component, text or icon, and region evidence to form a distinct referring phrase. Phrases combined component type, visible text or icon class when available, and a coarse location such as top, middle, bottom, left, center, or right. Because these queries were generated from the structured record, the evaluation measures controlled grounding under known evidence rather than performance on naturally authored user instructions.

Four scorers were compared. Random ranked candidates uniformly. Position only used region agreement and center distance. Token match used overlap with the component label, visible text, icon label, resource ID, and class name. Multi-evidence grounding combined token evidence, predicted label, clickability, geometry, and region agreement. The design reflects the broader distinction between candidate retrieval and downstream reranking (Meng et al., 2026; Zhong et al., 2025), but those external systems were not implemented in this experiment. Performance was measured with Top-1 accuracy, Top-3 accuracy, mean reciprocal rank (MRR), median rank, and median center error.

Layout-tag Inference

The layout analyzer assigned seven screen-level tags: toolbar, list_screen, form_screen, card_screen, navigation_screen, image_heavy, and ad_supported. Reference tags were derived from hierarchy labels together with screen-level geometry. The geometry-only analyzer used widths, heights, repeated rows, top and bottom bands, and image-area ratios. The predicted-label analyzer used the evidence-fused component predictions. Comparing these analyzers isolated whether semantic component recognition was necessary for categories

that could not be identified reliably from rectangles alone.

The tags were selected to represent visual organization and likely user-facing function. This choice is consistent with HCI principles that treat navigation, consistency, visibility, and feedback as central aspects of interface interpretation (Dix et al., 2004; Nielsen, 1994; Shneiderman et al., 2016). Because both the reference tags and predictions depend on structured hierarchy evidence, the results are interpreted as an internal layout benchmark rather than an independently annotated human design judgment.

Evidence-Bound Design-Rationale Cards

For each selected grounded element, the system produced a compact record with five fields:

1. **Role:** predicted component label plus supporting text or icon evidence.
2. **Evidence:** screen region, clickability, box geometry, nearby text, and relevant layout tag.
3. **Position:** a verbalized location derived from normalized geometry.
4. **Likely action:** an action derived from clickability and control type. 5. Design rationale: a constrained sentence explaining the element's local design function.

Three output types were evaluated. Caption-only used the label text. Generic rationale used role, position, and action fields. Evidence card exposed the full structured record. The schema follows the principle that design language should remain traceable to layout metadata (Wu et al., 2025) and that provenance fields should be separate from the surface narrative (Jin, 2025a). It does not imply that the models or evaluation protocols from those studies were reproduced here.

Card quality was measured by supported-field coverage, role consistency, position consistency, action consistency, contradiction rate, and mean word count. Consistency checks compared card fields with the detected element evidence and grounding facts. A contradiction therefore meant disagreement with the structured record; the checks did not assess whether a human considered a card useful, appropriately concise, or persuasive.

Statistical Reporting and Reproducibility Boundaries

Metrics were computed from saved run files for the training, validation, and test partitions. Corpus tables report counts and normalized geometry after valid-bounds filtering. Classification reports accuracy, macro precision, macro recall, macro F1, weighted F1, and selected per-class metrics over test elements. Grounding metrics are aggregated across generated test queries. Layout metrics are computed at screen level, and rationale-card metrics are computed over all generated cards.

The pipeline is intended as a lightweight, inspectable baseline rather than a replacement for large visual detectors or multimodal transformers. The current report does not provide confidence intervals, significance tests, a split seed, full hyperparameters, the exact calibration procedure, software-version details, or a code repository. Those items should be supplied for a fully reproducible release and should not be inferred from the reported point estimates.



Figure 1. Experimental pipeline for element grounding and rationale-card generation

Figure 1: End-to-end pipeline for element proposals, classification, grounding, layout analysis, and evidence-bound design-rationale-card generation.

Results

The results are reported in the same order as the five pipeline tasks. Tables 1–4 and Figure 2 describe the corpus and element geometry. Tables 5–6 and Figures 3–4 report element proposals and detection. Tables 7–8 and Figure 5 report control classification. Table 9 and Figure 6 evaluate referring-expression grounding. Table 10 and Figure 7 report layout tagging. Table 11 and Figure 8 summarize the rationale-card conditions. All model comparisons use the screen-disjoint partitions described above.

Corpus profile and experimental partitions

Table 1. Corpus summary.

Screens	All nodes	Labeled elements	Component labels	Clickable elements	Mean elements/screen	Median elements/screen	Portrait screens	Landscape screens
1,460	31,968	30,101	25.00	8,967	20.65	15.00	1,458	2.00

Note. Counts were computed after recursive hierarchy parsing and valid-bounds filtering.

The corpus contained 1,460 screens, 31,968 parsed hierarchy nodes, and 30,101 labeled elements across 25 component labels. Of the labeled elements, 8,967 were clickable. The mean number of labeled elements per screen was 20.65, whereas the median was 15.00, indicating a long tail of dense list and dashboard screens. The orientation distribution was highly asymmetric: 1,458 screens were portrait and 2 were landscape.

Table 2. Component-label distribution for the 20 most frequent classes.

Component label	Count	Share
Text	10,674	0.355
Image	4,766	0.158
Icon	4,117	0.137
List Item	3,578	0.119
Text Button	3,240	0.108
Toolbar	918	0.030
Input	649	0.022

Component label	Count	Share
Card	620	0.021
Web View	448	0.015
Advertisement	265	0.009
Background Image	154	0.005
Multi-Tab	125	0.004
Radio Button	116	0.004
Pager Indicator	100	0.003
Drawer	81	0.003
Modal	60	0.002
Checkbox	49	0.002
On/Off Switch	48	0.002
Map View	21	0.001
Slider	20	0.001

Note. Share is the fraction of all 30,101 labeled elements; the five rarest classes are omitted from this display.

Text was the largest class with 10,674 elements and a share of 0.355. Image contained 4,766 elements (0.158), Icon contained 4,117 (0.137), List Item contained 3,578 (0.119), and Text Button contained 3,240 (0.108). The next most frequent labels were Toolbar, 918 (0.030); Input, 649 (0.022); Card, 620 (0.021); Web View, 448 (0.015); and Advertisement, 265 (0.009). Background Image, Multi-Tab, Radio Button, Pager Indicator, Drawer, Modal, Checkbox, On/Off Switch, Map View, and Slider each represented 0.005 or less of the corpus.

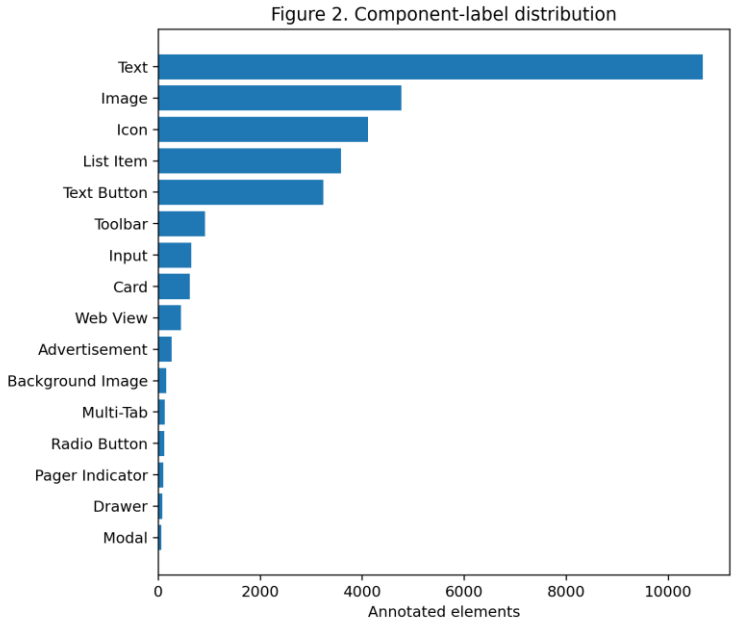


Figure 2. Label distribution across the parsed element corpus.

Table 3. Screen-disjoint experimental split.

Split	Screens	Elements	Clickable	Labels	Mean elements/screen
Train	1,022	21,447	6,244	25	21.03
val	219	3,974	1,302	22	18.15
test	219	4,680	1,421	23	21.37

Note. Screens, rather than elements, define the train, validation, and test partitions.

The training partition contained 1,022 screens, 21,447 elements, 6,244 clickable elements, and all 25 labels, with a mean of 21.03 elements per screen. The validation partition contained 219 screens, 3,974 elements, 1,302 clickable elements, and 22 labels, with a mean of 18.15. The test partition contained 219 screens, 4,680 elements, 1,421 clickable elements, and 23 labels, with a mean of 21.37.

Table 4. Geometry and clickability by component label.

Component label	Count	Mean area	Median width	Median height	Clickable rate
Text	10,674	0.022	0.316	0.032	0.007
Image	4,766	0.054	0.200	0.073	0.216
Icon	4,117	0.007	0.107	0.062	0.542
List Item	3,578	0.101	1.000	0.089	0.287
Text Button	3,240	0.035	0.464	0.066	1.000
Toolbar	918	0.078	1.000	0.077	0.014
Input	649	0.040	0.751	0.057	0.965
Card	620	0.156	0.922	0.201	0.489
Web View	448	0.096	0.778	0.068	0.542
Advertisement	265	0.074	1.000	0.068	0.045
Background Image	154	0.898	1.000	0.902	0.039
Multi-Tab	125	0.066	1.000	0.066	0.000
Radio Button	116	0.019	0.235	0.062	1.000
Pager Indicator	100	0.024	1.000	0.027	0.010
Drawer	81	0.700	0.739	0.902	0.210
Modal	60	0.914	0.922	0.951	0.000
Checkbox	49	0.004	0.078	0.044	0.061
On/Off Switch	48	0.206	0.595	0.067	0.104

Note. Areas, widths, and heights are normalized to the screenshot frame.

The geometry summary shows why location and shape offer useful but incomplete supervision. Toolbar and List Item had median widths of 1.000, with median heights of 0.077 and 0.089, respectively. Icon had a mean area of 0.007 and a clickable rate of 0.542. Text Button had a clickable rate of 1.000, and Input had a clickable rate of 0.965. By contrast, Image, Web View, Advertisement, Card, and Background Image occupied overlapping geometric regimes despite differing functions. Background Image had a mean normalized area

of 0.898, Drawer 0.700, and Modal 0.914, but clickability alone did not consistently distinguish these large containers.

Element proposals and detection

Table 5. Element-detection results on the test split.

Method	AP50	Precision@100	Recall@50	Recall@100	Recall@140	Mean best IoU
Geometry/anchor score	0.548	0.574	0.909	0.994	1.000	0.986
Metadata prior score	0.946	0.575	0.968	0.998	1.000	0.994
Evidence-fused proposal	0.950	0.574	0.970	0.998	1.000	0.994

Note. AP50 uses an intersection-over-union threshold of .50. Recall@K is the fraction of ground-truth elements recovered among the top K candidates per screen.

Geometry/anchor scoring reached 0.548 AP50, 0.574 Precision@100, 0.909 Recall@50, 0.994 Recall@100, 1.000 Recall@140, and 0.986 mean best IoU. Metadata-prior scoring increased AP50 to 0.946 and Recall@50 to 0.968; its Precision@100 was 0.575, Recall@100 was 0.998, Recall@140 was 1.000, and mean best IoU was 0.994. Evidence fusion reached the highest AP50, 0.950, and the highest Recall@50, 0.970. Its Precision@100 was 0.574, Recall@100 was 0.998, Recall@140 was 1.000, and mean best IoU was 0.994. Thus, evidence fusion improved AP50 by 0.402 over geometry/anchor scoring, but it tied metadata on Recall@100, Recall@140, and mean best IoU and did not exceed metadata on Precision@100.

Table 6. Detection recall@100 by component label.

Label	Support	Recall@100
Text	1,678	0.998
Icon	664	1.000
Image	661	0.988
Text Button	528	1.000
List Item	457	1.000
Toolbar	136	1.000
Card	113	1.000
Input	83	1.000
Advertisement	50	1.000
Web View	50	1.000
Radio Button	29	1.000
Pager Indicator	20	1.000
Multi-Tab	19	1.000
On/Off Switch	13	1.000
Drawer	11	1.000
Checkbox	10	1.000

Note. The rows shown have the largest test support.

At Recall@100, Text reached 0.998 over 1,678 instances, and Image reached 0.988 over 661 instances. Icon (664), Text Button (528), List Item (457), Toolbar (136), Card (113), Input (83), Advertisement (50), Web View (50), Radio Button (29), Pager Indicator (20), Multi-Tab (19), On/Off Switch (13), Drawer (11), and Checkbox (10) each reached 1.000. The remaining misses were therefore concentrated in Text and Image rather than in the displayed control classes.

Figure 3. Ground-truth elements and top UI proposals

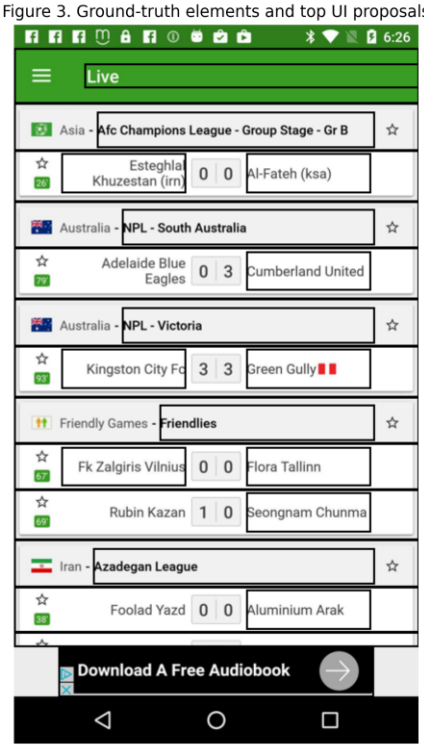


Figure 3. High-ranked element proposals on a held-out mobile-interface screen.

Figure 4. Detection precision-recall at IoU 0.50

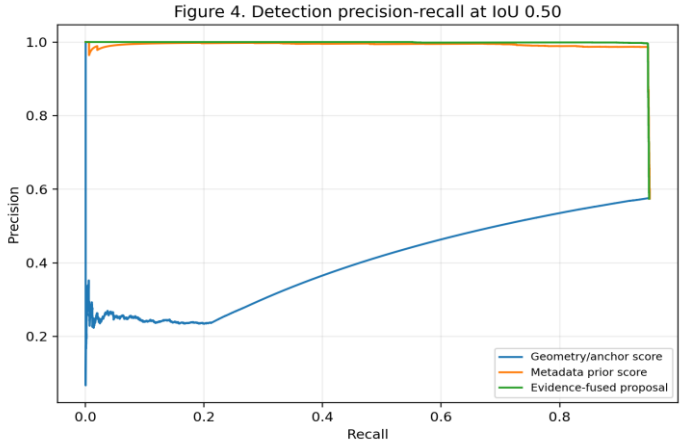


Figure 4: Detection precision–recall curves for geometry, metadata, and evidence-fused proposal scoring.

The precision-recall comparison is consistent with the tabular results. Geometry recovered most elements at high K but ranked many plausible nested or anchor boxes ahead of true elements. Metadata improved early ranking by using labels, text, clickability, and resource identifiers. Evidence fusion retained the high-recall behavior while producing the largest AP50. Precision@100 remained close to 0.574–0.575 because an average test screen contained approximately 21 labeled elements while each top-100 set intentionally retained many additional anchors.

Control classification

Table 7. Control-classification comparison.

Model	Accuracy	Macro precision	Macro recall	Macro F1	Weighted F1
Geometry only	0.363	0.210	0.343	0.208	0.398
String metadata	0.779	0.576	0.667	0.570	0.813
Geometry + metadata	0.812	0.665	0.787	0.662	0.840
Evidence-fused	0.932	0.749	0.856	0.762	0.942

Note. Metrics were computed on test elements under screen-disjoint training.

Geometry alone reached 0.363 accuracy, 0.210 macro precision, 0.343 macro recall, 0.208 macro F1, and 0.398 weighted F1. String metadata increased these values to 0.779 accuracy, 0.576 macro precision, 0.667 macro recall, 0.570 macro F1, and 0.813 weighted F1. Geometry plus metadata reached 0.812 accuracy and 0.840 weighted F1. The evidence-fused configuration reached 0.932 accuracy, 0.749 macro precision, 0.856 macro recall, 0.762 macro F1, and 0.942 weighted F1. Its accuracy exceeded geometry alone by 0.568.

Table 8. Per-class performance for the evidence-fused classifier.

Label	Precision	Recall	F1	Support
Text	0.999	0.980	0.990	1,689
Image	0.968	0.834	0.896	686.0
Icon	1.000	1.000	1.000	676.0
Text Button	0.946	0.836	0.888	529.0
List Item	0.953	0.959	0.956	461.0
Toolbar	1.000	0.942	0.970	138.0
Card	0.974	0.982	0.978	113.0
Web View	0.989	0.967	0.978	92.00
Input	0.349	0.869	0.498	84.00
Advertisement	0.836	0.927	0.879	55.00
Radio Button	0.552	0.533	0.542	30.00
Background Image	0.929	0.963	0.945	27.00
Pager Indicator	0.667	0.800	0.727	20.00
Multi-Tab	0.900	0.947	0.923	19.00
Drawer	0.800	0.800	0.800	15.00

Label	Precision	Recall	F1	Support
On/Off Switch	0.182	0.308	0.229	13.00
Modal	0.917	1.000	0.957	11.00
Checkbox	1.000	0.700	0.824	10.00
Other	0.167	0.833	0.278	6.00
Map View	0.571	0.800	0.667	5.00

Note. The rows shown have the highest test support; rare classes remain in the evaluation.

Performance was strongest for high-support and structurally distinctive labels. Text reached 0.999 precision, 0.980 recall, and 0.990 F1 over 1,689 instances. Icon reached 1.000 on precision, recall, and F1 over 676 instances. List Item reached 0.956 F1, Toolbar 0.970, Card 0.978, Web View 0.978, Background Image 0.945, Multi-Tab 0.923, and Modal 0.957. Image and Text Button reached F1 values of 0.896 and 0.888, respectively.

Lower-support or overlapping controls remained less stable. Input had 0.349 precision, 0.869 recall, and 0.498 F1 over 84 instances. Radio Button reached 0.542 F1 over 30 instances. On/Off Switch reached 0.182 precision, 0.308 recall, and 0.229 F1 over 13 instances. Other reached 0.278 F1 over 6 instances, and Map View reached 0.667 F1 over 5. These values show that the aggregate weighted F1 was driven mainly by common classes and should not be generalized to every rare control.

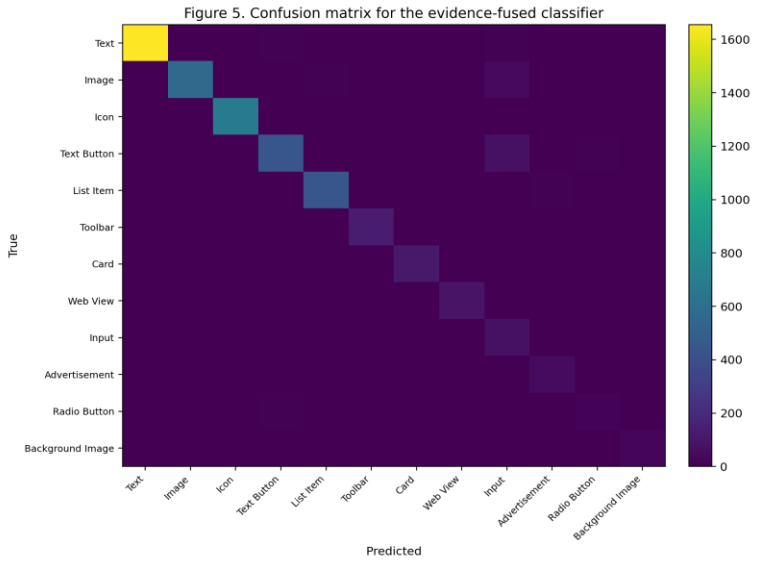


Figure 5: Confusion matrix for selected high-support classes in the evidence-fused component classifier.

Referring-expression grounding

Table 9. Referring-expression grounding results.

Method	Trials	Top-1	Top-3	MRR	Median rank	Median center error
Multi-evidence grounding	2,161	0.876	0.979	0.928	1.000	0.000
Position only	2,161	0.380	0.753	0.588	2.00	0.074
Random	2,161	0.021	0.196	0.180	9.00	0.439
Token match	2,161	0.729	0.906	0.826	1.000	0.000

Note. Trials were generated from held-out elements with label, text or icon, and region evidence.

Random ranking reached 0.021 Top-1 accuracy, 0.196 Top-3 accuracy, 0.180 MRR, a median rank of 9.00, and a median center error of 0.439. Position-only grounding improved to 0.380 Top-1, 0.753 Top-3, and 0.588 MRR, with a median rank of 2.00 and center error of 0.074. Token matching reached 0.729 Top-1, 0.906 Top-3, and 0.826 MRR, with a median rank of 1.000 and center error of 0.000. Multi-evidence grounding reached 0.876 Top-1, 0.979 Top-3, and 0.928 MRR, also with a median rank of 1.000 and center error of 0.000. The multi-evidence Top-1 result exceeded position only by 0.496 and token match by 0.147.

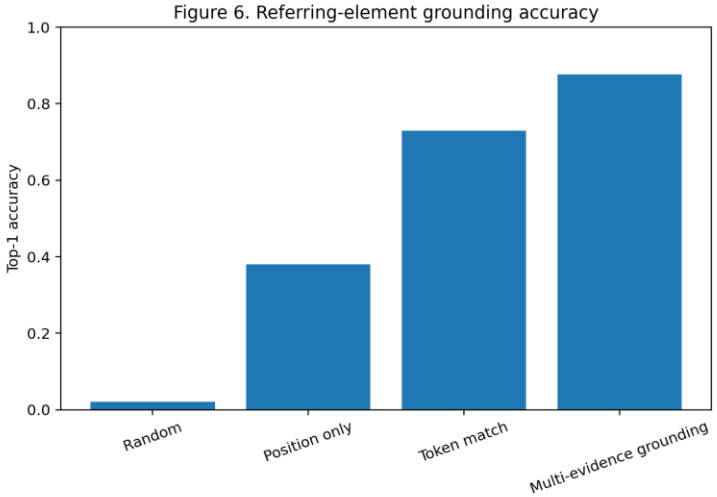


Figure 6: Top-1 accuracy for random, position-only, token-match, and multi-evidence referring-expression grounding.

The exact-target result was strongest when lexical, label, clickability, and spatial constraints agreed. Position alone remained ambiguous in regions with several controls, whereas token match could be ambiguous when two elements shared text or a component label. The high Top-3 value of 0.979 indicates that the correct element appeared among the first three candidates in most generated trials, but the queries were constructed from known evidence and do not represent naturally authored instructions.

Layout-tag performance

Table 10. Layout-tag analysis for geometry-only and predicted-label analyzers.

Analyzer	Layout tag	Precision	Recall	F1	Support
Geometry-only analyzer	toolbar	0.862	1.000	0.926	131
Geometry-only analyzer	list_screen	0.464	0.865	0.604	52
Geometry-only analyzer	form_screen	0.000	0.000	0.000	42
Geometry-only analyzer	card_screen	0.316	0.375	0.343	16
Geometry-only analyzer	navigation_screen	0.156	0.500	0.238	24
Geometry-only analyzer	image_heavy	0.585	0.925	0.717	93
Geometry-only analyzer	ad_supported	0.000	0.000	0.000	47
Predicted-label analyzer	toolbar	0.862	1.000	0.926	131
Predicted-label analyzer	list_screen	0.460	1.000	0.630	52
Predicted-label analyzer	form_screen	0.549	0.929	0.690	42
Predicted-label analyzer	card_screen	0.500	1.000	0.667	16
Predicted-label analyzer	navigation_screen	0.900	0.750	0.818	24
Predicted-label analyzer	image_heavy	0.975	0.839	0.902	93
Predicted-label analyzer	ad_supported	0.936	0.936	0.936	47

Note. Metrics are screen-level. The predicted-label analyzer uses outputs from the evidence-fused component classifier.

Geometry alone performed strongly for Toolbar, with 0.862 precision, 1.000 recall, and 0.926 F1 over 131 positive screens. It reached 0.604 F1 for List Screen and 0.717 for Image Heavy. However, geometry produced 0.000 F1 for Form Screen and Ad Supported, 0.238 for Navigation Screen, and 0.343 for Card Screen.

Using predicted component labels left Toolbar unchanged at 0.926 F1 and increased List Screen to 0.630. Form Screen rose to 0.549 precision, 0.929 recall, and 0.690 F1. Card Screen rose to 0.500 precision, 1.000 recall, and 0.667 F1. Navigation Screen reached 0.900 precision, 0.750 recall, and 0.818 F1. Image Heavy reached 0.975 precision, 0.839 recall, and 0.902 F1, while Ad Supported reached 0.936 for precision, recall, and F1.

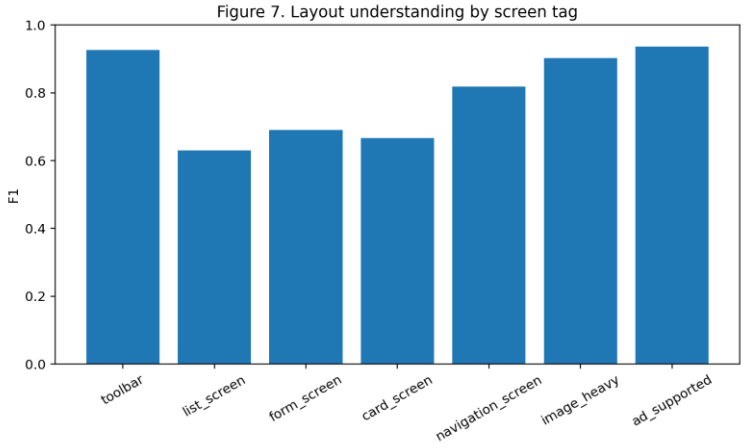


Figure 7. Predicted-label analyzer F1 by layout tag.

Rationale-card consistency

Table 11. Design-rationale-card consistency.

Card type	Cards	Supported fields	Role consistency	Position consistency	Action consistency	Contradiction rate	Mean words
Caption-only	643	0.250	1.000	0.000	0.000	0.000	2.53
Evidence card	643	1.000	1.000	1.000	1.000	0.000	64.94
Generic rationale	643	1.000	1.000	1.000	1.000	0.000	27.53

Note. Consistency was checked against detected-element evidence and generated grounding facts. A zero contradiction rate indicates same-source internal consistency, not human-rated usefulness.

Each condition contained 643 cards. Caption-only cards had supported-field coverage of 0.250, role consistency of 1.000, position consistency of 0.000, action consistency of 0.000, a contradiction rate of 0.000, and a mean length of 2.53 words. Evidence cards reached 1.000 for supported fields, role consistency, position consistency, and action consistency, with a contradiction rate of 0.000 and a mean length of 64.94 words. Generic rationales also reached 1.000 on the four coverage or consistency measures and 0.000 contradiction, with a mean length of 27.53 words.

Figure 8. Design-rationale card schema with data-backed evidence

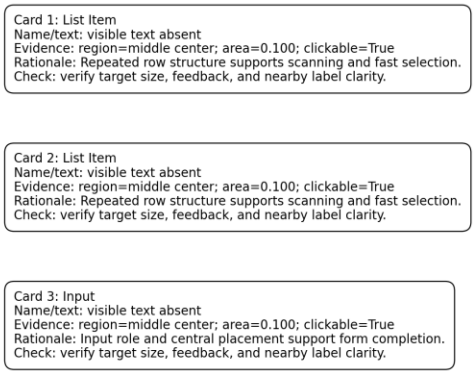


Figure 8: Design-rationale-card schema and examples grounded in structured UI evidence.

The results establish that both full evidence cards and generic rationales agreed with the structured fields checked by the evaluator. The evidence cards were 37.41 words longer on average than generic rationales and 62.41 words longer than captions. Because generic rationales and evidence cards received identical consistency scores, the measured difference is one of exposed detail and length, not a demonstrated advantage in human usefulness.

Discussion

What Evidence Fusion Contributed

The most consistent empirical pattern was that spatial and semantic evidence served different functions. Geometry/anchor scoring already achieved 0.994 Recall@100 because hierarchy bounds are closely aligned with visible UI structure. Its AP50 was only 0.548, however, indicating that geometry alone did not rank true targets early when many nested or anchor boxes were plausible. Metadata raised AP50 to 0.946, and evidence fusion increased it slightly further to 0.950 while retaining 0.998 Recall@100. This result supports a staged interpretation: geometry supplies broad candidate coverage, while labels, text, identifiers, clickability, and context improve prioritization.

The classification results show a larger semantic contribution. Geometry alone reached 0.363 accuracy and 0.398 weighted F1, whereas string metadata reached 0.779 and 0.813. Combining geometry with metadata raised accuracy to 0.812, and adding detector confidence and layout context produced 0.932 accuracy and 0.942 weighted F1. This pattern is compatible with multimodal UI research in which screen position, text, and hierarchy semantics contribute complementary information (Bai et al., 2021; Lee et al., 2023). It also explains why a single modality is unlikely to be sufficient: two components can share a box shape, while one token can occur in several positions or control types.

The magnitude of these gains must nevertheless be interpreted within the corpus design. The proposal stage starts from JSON hierarchy candidates and retains hierarchy labels among its metadata signals. It therefore evaluates ranking and filtering of structurally informed proposals, not end-to-end detection from raw pixels. The AP50 of 0.950 should not be compared directly with screenshot-only detection systems unless candidate generation and annotation access are matched. The result is best understood as a strong hierarchy-backed baseline whose evidence chain can later accommodate a learned visual proposal module.

Rare Controls and Error Structure

Weighted averages concealed substantial class variation. Text, Icon, Toolbar, Card, Web View, and several other common classes had high F1, whereas Input reached 0.498 F1 and On/Off Switch reached 0.229. The latter had only 13 test instances. These errors are plausible given the corpus: an input may resemble a long text strip or search bar, and a switch may be represented by a broader clickable container rather than by the visible toggle alone. Nested hierarchies further create several legitimate-looking targets around a single perceptual object. This issue is not solved merely by adding more card language. If the upstream object is wrong, a fluent rationale can faithfully describe the wrong record. UI dataset-denoising work has similarly emphasized that hierarchy quality affects every downstream task (Li et al., 2022). A practical extension should therefore add rare-control screens, report class-balanced sampling or reweighting, and distinguish perceptual objects from interaction containers. Evaluation could also separate exact-box selection from action-equivalent selection when a parent and child lead to the same interaction.

Grounding as Constrained Ranking

The grounding results demonstrate why neither region nor text should be the sole selection signal. Position only reached 0.380 Top-1 accuracy because several controls can occupy the same coarse region. Token match reached 0.729, showing that text, icon, class name, and resource identifier were more discriminative, but duplicate labels remained possible. Multi-evidence grounding reached 0.876 Top-1 and 0.979 Top-3 by combining lexical, component, action, and spatial constraints.

This separation between coverage and selection is consistent with retrieval work that first builds a candidate set and then applies a more selective ranking policy (Meng et al., 2026; Zhong et al., 2025). Provenance-oriented systems add that the selected item should remain linked to the evidence that raised it in the ranking (Jin, 2025a). In the present pipeline, that link is expressed through label, token, clickability, region, and geometry fields. The result makes a grounded target inspectable, but it does not establish performance on spontaneous human requests because all 2,161 phrases were generated from known element evidence.

Future grounding evaluation should therefore include naturally authored references from multiple users, including incomplete and ambiguous instructions. A selective-refusal or clarification condition could be added when two candidates remain close. Retrieved-prototype explanations and deterministic failure narratives offer relevant patterns for preserving evidence during this escalation step (Sun et al., 2026; Xin, 2025), although their operational domains differ from mobile UI interaction.

From Component Labels to Layout Interpretation

Layout tagging exposed a clearer boundary between visual regularity and function. Geometry recognized Toolbar at 0.926 F1 and Image Heavy at 0.717 because top bands and image area have stable shapes. It failed entirely on Form Screen and Ad Supported and reached only 0.238 F1 on Navigation Screen. Predicted labels increased these values to 0.690, 0.936, and 0.818, respectively. Card Screen also increased from 0.343 to 0.667.

The result indicates that some layout tags are semantic hypotheses rather than arrangements of rectangles. A long horizontal region can be an input, advertisement, tab bar, or card footer. Layout-aware visual processing likewise distinguishes where a region appears from why it should be prioritized (Wang et al., 2025). In a design tool, this distinction could support different views: a geometric overlay for box inspection and a semantic layer for form, navigation, advertisement, or card-level analysis.

However, the reference layout tags were themselves derived from hierarchy labels and geometry. The predicted-label analysis therefore measures consistency with a rule-derived target rather than agreement with independent designers. External validation should ask multiple annotators to assign screen-level tags, document inter-rater agreement, and test whether the inferred tags help a real review or navigation task.

Rationale-Card Scope and Interface Implications

The card experiment was designed to test field-level consistency, not open-ended writing quality. Caption-only cards exposed one quarter of the supported fields and averaged 2.53 words. Generic rationales averaged 27.53 words, and evidence cards averaged 64.94. Both longer conditions reached 1.000 on the evaluated coverage and consistency checks and 0.000 contradiction. Consequently, the study does not show that a 64.94-word card is more useful than a 27.53-word rationale; it shows that the expanded card can expose more structured detail without contradicting the fields checked by the evaluator.

A progressive-disclosure design is therefore more defensible than displaying every field at once. The default view could show the role and target, while an expanded view reveals geometry, clickability, nearby text, layout context, and consistency checks. Evidence-card and visual-brief research has similarly separated concise decision cues from inspectable support (Jin, 2025b; Liu et al., 2025; Mu et al., 2026). Evidence-linked recommendation interfaces also suggest that a reason, confidence or operational cue, and action should remain visually distinct (Zhang & Zhang, 2025b). These studies motivate the presentation strategy but do not replace user testing in the mobile grounding context.

The likely-action field requires particular care. Clickability and control type can support a bounded statement such as "likely tappable," but they do not prove what will happen after activation. Human-oversight cards emphasize the importance of separating observed evidence from recommended or predicted action (Su, Rao, et al., 2026), while command-safety evaluation retains an explicit review step rather than presenting generated operational guidance as an autonomous command (B. Zhang et al., 2026). The same principle should govern UI agents: the card may support confirmation and correction, but the current experiment does not validate autonomous execution.

Calibration, Faithfulness, and Human Evaluation

The 0.000 contradiction rate is narrower than a general faithfulness claim. The checks compare generated fields with the structured record used to produce them, so an upstream misclassification can remain internally consistent. Evidence-calibrated retrieval research distinguishes source coverage, answerability, and abstention because a coherent output is not necessarily supported by sufficient evidence (Zhong et al., 2025). Trust-calibrated interfaces similarly use refusal or uncertainty cues when the evidence cannot sustain an answer (Chen & Xu, 2026). The current pipeline should add an evidence-sufficiency score and a clarification or abstention path before it is described as trust calibrated.

Claim verification and conformal alerting offer complementary design ideas. Evidence can be ranked before a narrative is produced (Su, Chen, et al., 2026), and calibrated thresholds can determine when a structured ticket should be escalated rather than automatically accepted (Xin, 2026). For UI grounding, analogous controls could include a confidence badge, alternative targets, or a request for confirmation when Top-1 and Top-2 scores are close. Compact quality tokens may also provide a machine-readable cue for downstream interfaces,

but they should summarize model evidence rather than substitute for user judgment (Zhou et al., 2025).

Human evaluation remains essential. A card may pass role, position, and action checks while being verbose, unclear, or poorly matched to a user's task. Existing human-AI guidelines emphasize feedback, correction, and context-appropriate explanation (Amershi et al., 2019), and audience-specific work shows that the same computational evidence may require different presentation for designers, teachers, operators, or end users (He et al., 2026; J. Zhang, 2026). A future study should therefore compare captions, generic rationales, and expandable evidence cards on target-selection accuracy, review time, comprehension, perceived workload, calibrated trust, and correction success.

Limitations and Future Work

First, the evaluation is limited to hierarchy-backed mobile screenshots. It depends on component labels, clickability, text, resource IDs, icon classes, and node bounds. A screenshot-only detector would require stronger visual representations and additional annotation. Transfer to web pages, tablets, or cross-platform interfaces should be evaluated separately because responsive grids, hover states, accessibility trees, and viewport conventions differ from Android-style view hierarchies (Wu et al., 2023).

Second, the corpus provenance is incomplete in the current manuscript. The final version must identify the dataset release, version, license, access route, and any preprocessing exclusions. It should also report the random seed, hyperparameters, calibration procedure, software versions, and code or run-file availability. Without these items, "reproducible" should be used only for the documented pipeline logic and not for an independently executable artifact.

Third, the class distribution is strongly imbalanced. Text, Image, Icon, List Item, and Text Button dominate, while Slider, Map View, Checkbox, On/Off Switch, Modal, and Drawer have limited support. Additional rare-control data, class-balanced analysis, and uncertainty estimates are needed before claims about safety-critical automation. Confidence intervals or bootstrap intervals would also clarify whether small differences such as AP50 0.946 versus 0.950 are stable.

Fourth, grounding phrases and layout reference tags are derived from the same structured evidence used by the models. The resulting evaluation is valuable for controlled ablation, but it may overestimate performance on natural language and independently interpreted screen structure. Human-authored expressions, external layout annotations, cross-app splits, and cross-corpus tests would provide stronger evidence.

Fifth, the rationale-card evaluation is computational. No designers, testers, accessibility specialists, or end users assessed comprehension, trust, preference, or task performance. Card length, visual hierarchy, and progressive disclosure should be tested experimentally. Structured recommendation-card and design-brief studies offer interface patterns that can inform such a study (B. Zhang et al., 2025; Chang et al., 2026; Mu et al., 2026), but cross-domain examples do not validate the present mobile UI cards.

Finally, the language layer is intentionally constrained and no named LLM, prompt, decoding setting, or free-form generation baseline is empirically evaluated. The term "LLM-assisted" therefore refers to an LLM-compatible, evidence-bound language interface rather than to demonstrated superiority of one large language model. Future work could compare template realization with several language models, but every output should retain field-level provenance, abstention when evidence is insufficient, and a user-visible correction path.

Conclusion

This study presented a unified, evidence-fused pipeline for hierarchy-backed mobile UI element proposals, control classification, referring-expression grounding, screen-level layout tagging, and design-rationale cards. On 1,460 screenshots, the pipeline parsed 31,968 hierarchy nodes and 30,101 labeled elements across 25 component categories. The evidence-fused detector reached 0.950 AP50 and 0.998 Recall@100. The evidence-fused classifier reached 0.932 accuracy and 0.942 weighted F1. Across 2,161 generated grounding trials, the multi-evidence scorer reached 0.876 Top-1 accuracy, 0.979 Top-3 accuracy, and 0.928 MRR. The experiments show that geometry and hierarchy semantics contribute at different stages. Geometry provides broad candidate coverage and identifies visually stable bands, while text, resource identifiers, clickability, predicted labels, and layout context improve component and target disambiguation. Predicted labels were particularly important for Form Screen, Navigation Screen, Card Screen, Image Heavy, and Ad Supported tags. Evidence cards preserved the checked role, position, and action fields across 643 cards with a contradiction rate of 0.000 under deterministic internal checks.

These findings support an inspectable evidence chain, not a claim of screenshot-only detection, autonomous UI action, or validated human trust. The next steps are to document and release the data pipeline, evaluate natural referring expressions and independently annotated layouts, add uncertainty and abstention, and conduct task-based user studies of compact and expanded cards. Under those boundaries, evidence-bound rationale cards provide a useful interface between structured mobile UI analysis and human-readable review.

References

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Teevan, J., Kikin-Gil, R., & Horvitz, E. (2019). Guidelines for human-AI interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Article 3, pp. 1–13). Association for Computing Machinery. https://doi.org/10.1145/3290605.3300233

Bai, C., Zang, X., Xu, Y., Sunkara, S., Rastogi, A., Chen, J., & Agüera y Arcas, B. (2021). UIBert: Learning generic multimodal representations for UI understanding. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence* (pp. 1705–1712). International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2021/235

Banerjee, P., Mahajan, S., Arora, K., Baral, C., & Riva, O. (2022). Lexi: Self-supervised learning of the UI language. In Y. Goldberg, Z. Kozareva, & Y. Zhang (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022* (pp. 6992–7007). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.findings-emnlp.519

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., …

Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems, 33*, 1877–1901.

Bunian, S., Li, K., Jemmali, C., Harteveld, C., Fu, Y., & El-Nasr, M. S. (2021). VINS: Visual search for mobile user interface design. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Article 423, pp. 1–14). Association for Computing Machinery. https://doi.org/10.1145/3411764.3445762

Chang, X., Lu, Y., & Zhong, Z. S. (2026). Review-grounded explainable recommendation with faithfulness evaluation on Amazon reviews. *JEECS (Journal of Electrical Engineering and Computer Sciences), 11*(1), 9–22. https://doi.org/10.54732/jeecs.v11i1.2

Chen, Y., & Li, M. (2025). From hand-drawn sketches to interactive web prototypes: A reproducible vision-language approach with structural and visual consistency evaluation. *Journal of Technology Informatics and Engineering, 4*(2), 364–384. https://doi.org/10.51903/jtie.v4i2.490

Chen, Y., & Xu, H. (2026). Trust-calibrated multilingual RAG for humanitarian information platforms: Empirical evaluation on OMoS-QA for migration information access. *International Journal of Graphic Design, 4*(1), 141–164. https://doi.org/10.51903/ijgd.v4i1.3552

Chromik, M., & Butz, A. (2021). Human-XAI interaction: A review and design principles for explanation user interfaces. In C. Ardito, R. Lanzilotti, A. Malizia, H. Petrie, A. Piccinno, G. Desolda, & K. Inkpen (Eds.), *Human-computer interaction—INTERACT 2021* (Lecture Notes in Computer Science, Vol. 12933, pp. 619–640). Springer. https://doi.org/10.1007/978-3-030-85616-8_36

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273–297. https://doi.org/10.1007/BF00994018

Deka, B., Huang, Z., Franzen, C., Hibschman, J., Afergan, D., Li, Y., Nichols, J., & Kumar, R. (2017). Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (pp. 845–854). Association for Computing Machinery. https://doi.org/10.1145/3126594.3126651

Dix, A., Finlay, J., Abowd, G. D., & Beale, R. (2004). *Human-computer interaction* (3rd ed.). Pearson Education.

Doshi-Velez, F., & Kim, B. (2017). *Towards a rigorous science of interpretable machine learning* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.1702.08608

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision, 88*(2), 303–338. https://doi.org/10.1007/s11263-009-0275-4

He, S., Nie, J., & Li, C. (2026). Power-aware inventory planning for AI infrastructure using job-level forecasting and LLM workload explanations. *Journal of Technology Informatics and Engineering, 5*(1), 341–359. https://doi.org/10.51903/jtie.v5i1.548

Jin, J. (2025a). Evidence-chain reliable RAG: Hallucination detection, source attribution, and deterministic provenance explanations. *Journal of Technology Informatics and Engineering, 4*(2), 520–533. https://doi.org/10.51903/jtie.v4i2.535

Jin, J. (2025b). LLM-style evidence cards for scientific search interfaces: A UI/UX design framework for retrieval transparency, ranking trust, and visual evidence hierarchy. *International Journal of Graphic Design, 3*(2), 397–414. https://journal.stekom.ac.id/index.php/ijgd/article/view/3698

Lee, J. (1997). Design rationale systems: Understanding the issues. *IEEE Expert, 12*(3), 78–85. https://doi.org/10.1109/64.592267

Lee, K., Joshi, M., Turc, I. R., Hu, H., Liu, F., Eisenschlos, J. M., Khandelwal, U., Shaw, P., Chang, M.-W., & Toutanova, K. (2023). Pix2Struct: Screenshot parsing as pretraining for visual language understanding. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett (Eds.), *Proceedings of the 40th International Conference on Machine Learning* (Proceedings of Machine Learning Research, Vol. 202, pp. 18893–18912). PMLR. https://proceedings.mlr.press/v202/lee23g.html

Leiva, L. A., Hota, A., & Oulasvirta, A. (2020). Enrico: A dataset for topic modeling of mobile UI designs. In *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services* (Article 9, pp. 1–4). Association for Computing Machinery. https://doi.org/10.1145/3406324.3410710

Li, G., Baechler, G., Tragut, M., & Li, Y. (2022). Learning to denoise raw mobile UI layouts for improving datasets at scale. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (Article 67, pp. 1–13). Association for Computing Machinery. https://doi.org/10.1145/3491102.3502042

Li, T. J.-J., Popowski, L., Mitchell, T. M., & Myers, B. A. (2021). Screen2Vec: Semantic embedding of GUI screens and GUI components. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Article 578, pp. 1–15). Association for Computing Machinery. https://doi.org/10.1145/3411764.3445049

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2980–2988). IEEE. https://doi.org/10.1109/ICCV.2017.324

Liu, G., He, S., & Wong, H. (2025). LLM-compatible visual brief cards for AI infrastructure capacity dashboards: A UI/UX framework for turning forecast risk into graphic design decisions. *International Journal of Graphic Design, 3*(1), 196–213. https://doi.org/10.51903/ijgd.v3i1.3723

Meng, S., Chen, J., & Zheng, I. (2026). LLM-inspired offline reranking for financial search: Query rewriting, hybrid retrieval, and listwise relevance ranking on FiQA. *Journal of Technology Informatics and Engineering, 5*(1), 361–378. https://doi.org/10.51903/jtie.v5i1.537

Mu, J., Lu, Y., & Hwang, E. (2026). Structured visual brief interfaces for advertising design: A UI/UX framework for turning creative intentions into designer-editable graphic design cards. *International Journal of Graphic Design, 4*(1), 192–208. https://doi.org/10.51903/ijgd.v4i1.3702

Nielsen, J. (1994). Enhancing the explanatory power of usability heuristics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 152–158). Association for Computing Machinery. https://doi.org/10.1145/191666.191729

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825–2830.

Redmon, J., & Farhadi, A. (2018). *YOLOv3: An incremental improvement* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.1804.02767

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems, 28*, 91–99.

Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2016). *Designing the user interface: Strategies for effective human-computer interaction* (6th ed.). Pearson.

Su, W., Chen, S., & Qian, E. (2026). Narrative-aware scientific claim verification agent with evidence ranking for ClimateCheck. *Journal of Technology Informatics and Engineering, 5*(1), 327–340. https://doi.org/10.51903/jtie.v5i1.549

Su, W., Rao, H., & Ma, E. (2026). Privacy and data-integrity risk cards for LLM agents: A UI/UX design framework for secure human oversight under prompt-injection attacks. *International Journal of Graphic Design, 4*(1), 186–191. https://doi.org/10.51903/ijgd.v4i1.3699

Sun, X., Zhong, Z. S., & Wu, Q. (2026). Retrieval-grounded HDFS log anomaly detection and deterministic failure narrative generation. *Journal of Computational Systems and Applications, 3*(1), 15–30. https://doi.org/10.64229/j6d7fr94

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30*, 5998–6008.

Wang, H., Ren, Y., & Chang, X. (2025). Layout-aware progressive PDF rendering: AI prioritization of PDF slices to reduce time-to-functional-first-frame on FUNSD. *Journal of Technology Informatics and Engineering, 4*(2), 425–446. https://doi.org/10.51903/jtie.v4i2.523

Wu, J., Wang, S., Shen, S., Peng, Y.-H., Nichols, J., & Bigham, J. P. (2023). WebUI: A dataset for enhancing visual UI understanding with web semantics. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Article 286, pp. 1–14). Association for Computing Machinery. https://doi.org/10.1145/3544548.3581158

Wu, Q., Meng, S., & Zhao, J. (2025). Text-grounded LLM-assisted design rationale interfaces: Turning advertising layout metadata into explainable UI/UX decision cards. *International Journal of Graphic Design, 3*(1), 216–240. https://doi.org/10.51903/ijgd.v3i1.3713

Xin, Q. (2025). Explaining OpenStack failure-injection log anomalies with retrieved normal prototypes. *Emerging Information Science and Technology, 6*(2), 125–146. https://doi.org/10.18196/eist.v6i2.31232

Xin, Q. (2026). Log anomaly detection with conformal alert control and evidence-grounded incident ticket generation. *Aviation Electronics, Information Technology, Telecommunications, Electricals, and Controls, 8*(2), 247–264. https://doi.org/10.28989/avitec.v8i2.3974

Xu, H., Chen, Y., & Med, A. (2025). Automatic detection and explanation of dark patterns from interface microcopy: Empirical comparison of BERT-style encoders, RoBERTa-style encoders, and LLM-style decoders on the ec-darkpattern dataset. *Journal of Technology Informatics and Engineering, 4*(3), 590–612. https://doi.org/10.51903/jtie.v4i3.491

Zhang, B., Ren, Y., & Zou, J. (2025). LLM-style explainable e-commerce recommendation cards: A UI/UX design framework for trust-calibrated product recommendation. *International Journal of Graphic Design, 3*(2), 381–396. https://journal.stekom.ac.id/index.php/ijgd/article/view/3697

Zhang, B., Sun, X., Liu, G., & Zhou, B. (2026). LLM-style DevOps copilot for cloud-native troubleshooting: Retrieval-augmented runbook generation and command-safety evaluation. *Journal of Technology Informatics and Engineering, 5*(2), 104–118. https://doi.org/10.51903/jtie.v5i2.534

Zhang, J. (2025). Adaptive user interface design for volleyball learning apps: Empirical evidence from Google Play reviews and mobile screen analysis. *International Journal of Graphic Design, 3*(1), 175–195. https://journal.stekom.ac.id/index.php/ijgd/article/view/3618

Zhang, J. (2026). Early warning, grade prediction, and teacher-facing LLM-ready explanations toward an open volleyball course: Reproducible evidence from four public education datasets. *Journal of Technology Informatics and Engineering, 5*(2), 20–44. https://doi.org/10.51903/jtie.v5i2.525

Zhang, Y., & Zhang, H. (2025a). A therapist-facing session copilot for live counseling support: Reasoning-guided retrieval and ranking from multi-turn counseling dialogues. *Journal of Technology Informatics and Engineering, 4*(2), 464–486. https://doi.org/10.51903/jtie.v4i2.547

Zhang, Y., & Zhang, H. (2025b). Visualizing the right counseling support: Evidence-linked recommendation cards for explainable mental health intake interfaces. *International Journal of Graphic Design, 3*(1), 214–229. https://doi.org/10.51903/ijgd.v3i1.3722

Zhong, Z. S., Chen, J., Zhong, E., & Sun, X. (2025). Evidence-calibrated RAG for unanswerable question answering: Retrieval coverage, abstention calibration, and hallucination-proxy analysis on SQuAD 2.0. *Journal of Technology Informatics and Engineering, 4*(2), 502–520. https://doi.org/10.51903/jtie.v4i2.536

Zhou, B., Wang, H., & Chang, X. (2025). Distilling VMAF into an edge-deployable quality predictor: A pilot shot-level proxy with LLM-ready quality tokens. *Journal of Technology Informatics and Engineering, 4*(2), 447–463. https://doi.org/10.51903/jtie.v4i2.522