Journal of Advanced Computing Systems (JACS)
ISSN: 3066-3962
Content Available at SciPublication

Multimodal UI Representation Learning: Ablation of Screenshot, Wireframe, and View-Hierarchy Proxies on an Uploaded 168-Screen Dataset

Yushan Chen[1], *Evelyn Chan*[2]
[1]*Service Design, Savannah College of Art and Design, GA, USA*
[2]*Computer Engineering, Dartmouth College, NH, USA*
yushanchen1029@gmail.com

DOI: 10.69987/JACS.2023.30101

Keywords

UI representation learning; multimodal embeddings; wireframe; view hierarchy

Abstract

We conduct an empirical ablation study on multimodal user-interface (UI) representation learning by integrating three complementary modalities: screenshot pixels, derived wireframes, and a proxy view-hierarchy structure. Consistent with prior UI datasets such as RICO and Enrico, which combine visual and structural metadata, our evaluation is performed on a dataset of 168 mobile UI screenshots (UI168) containing only raster images. Since no structural annotations are available, we deterministically generate two additional modalities from each screenshot: wireframes extracted using Canny edge detection and a hierarchy proxy constructed from bounding-box containment relations of edge-connected components. We compare unimodal embeddings and four fusion strategies on pseudo-topic classification and retrieval tasks using a fixed data split (seed=42). Pseudo-topics are created through K-means clustering (K=8) on early-fused training embeddings and transferred to validation and test partitions. Lightweight, CPU-reproducible representations are employed, including grayscale tiny-image features with color histograms for screenshots, edge-based descriptors for wireframes, and TF–IDF with truncated SVD for hierarchy tokens. On the 35-image test set, early fusion achieves the strongest retrieval performance (mAP=0.413), outperforming late and attention-based fusion, while unimodal screenshot features remain competitive. Gated fusion optimized on validation data yields moderate improvements with learned weights (0.40, 0.45, 0.15). For pseudo-topic classification, early fusion attains the highest Macro-F1 (0.863). Cross-modal retrieval experiments demonstrate strong screenshot-to-wireframe alignment under CCA and improved screenshot-to-hierarchy mapping using ridge regression. Additional robustness analysis evaluates degradation under occlusion, noise, and blur. All results are empirically derived from the provided dataset using fixed hyperparameters and a fully reproducible pipeline.

Introduction

User interfaces encode both visible appearance and latent structure. A single UI screen simultaneously contains pixel-level signals (color palettes, typography, imagery, iconography) and compositional layout signals (grids, alignment, container nesting, affordances, navigation regions). For this reason, a compact representation of UI screens is a useful primitive across human–computer interaction and software engineering: similar-screen retrieval for design reuse; automated grouping of screens by design theme; detection of inconsistent visual patterns; generation of UI code from mockups; UI testing and exploration; and personalization of layouts.

Large-scale datasets have enabled data-driven UI research. RICO introduced a repository of Android screens paired with view hierarchies and interaction traces, supporting UI search, layout mining, and design applications [1]. Enrico curated and labeled mobile UI designs into semantic topics, and provides structural artifacts such as hierarchies and wireframes that are valuable for multimodal UI learning [2]. These datasets highlight a practical observation: UI structure is often available in instrumented settings (e.g., when screens are collected from a device with accessibility APIs), but

in many realistic scenarios only screenshots exist—such as design mockups, marketing images, user-submitted bug reports, or historical app screenshots.

At the same time, representation learning has advanced rapidly, making it possible to learn embeddings that support retrieval and classification with minimal supervision. Contrastive learning and related self-supervised methods (e.g., SimCLR, MoCo, BYOL) have demonstrated strong image representations without labels [3]–[5]. In the UI domain, researchers have explored screen embeddings and methods for converting pixels into structured representations or code, including Pix2Code and ReDraw, which reverse engineer UI layouts from screenshots [6], [7]. These lines of work collectively motivate a multimodal view of UI representation: some tasks are style-dominant (color palette or branding), while others are structure-dominant (layout composition and grouping).

This paper focuses on a concrete question: given multiple modalities that describe a UI screen, what is the marginal utility of each modality, and how should modalities be fused? A screenshot is information-rich but mixes style and structure. A wireframe suppresses most texture and color while emphasizing spatial boundaries. A view hierarchy encodes compositional nesting and widget grouping, which can be critical for reasoning about structure and interaction. When all modalities are available, fusion is a natural strategy. When hierarchies are missing, a screen-to-structure proxy is needed.

The dataset used in this environment (UI168) contains only 168 screenshots. To perform the requested modality ablation in a way that is consistent with the provided data, we deterministically derive two additional modalities from each screenshot: (i) a wireframe created via Canny edge detection, and (ii) a view-hierarchy proxy built from a containment tree over bounding boxes extracted from connected components in the edge map. This proxy does not replicate true platform view hierarchies, but it provides a structured signal that approximates layout nesting and grouping.

Because UI168 has no semantic labels, we adopt a fully reproducible pseudo-label protocol. We fit K-means clustering on early-fusion embeddings computed only from the training split, treat the resulting cluster IDs as pseudo-topics, and evaluate retrieval and classification on held-out test screens. This approach is common in representation learning analysis when ground-truth labels are unavailable: it yields a fixed label space, avoids test leakage, and enables controlled comparisons across modalities.

Contributions. The contribution of this paper is a complete, reproducible ablation study that (i) constructs screenshot, wireframe, and view-hierarchy-proxy modalities from the provided screenshot-only dataset; (ii) evaluates unimodal embeddings and four fusion strategies (early, late, gated, and attention-based fusion) on pseudo-topic retrieval and classification; (iii) reports cross-modal alignment via CCA and ridge regression; (iv) reports robustness under occlusion, noise, and blur; and (v) provides at least nine tables and six figures with detailed experimental outcomes.

Extended context and related work (within the same section to preserve the required structure). A substantial body of work treats UI understanding as a multimodal problem where visual appearance, structure, and text interact. Screen-to-code methods often start with a screenshot and predict a hierarchy of components or code tokens; Pix2Code uses a learned encoder-decoder architecture to map GUI images to code representations [6], while ReDraw combines computer vision detection with learned components for reverse engineering mobile screens [7]. These tasks underscore why structural signals matter: pixel similarity alone cannot capture the semantics of nested containers and alignment constraints. More recent approaches in representation learning also emphasize aligning different views of the same instance. CCA provides a classical linear alignment tool that finds correlated projections across modalities [9], while ridge regression provides a stable linear mapping when predictors are correlated [10]. In multimodal fusion, early fusion (feature concatenation) is simple and often effective, but can be suboptimal when modalities have different noise levels; late fusion can be more robust by combining similarity scores; gated and attention-based fusion adapt weights based on modality reliability [13]. The goal of our experiments is not to claim a universally best fusion strategy, but to provide a dataset-grounded comparison under a deterministic and fully reproducible pipeline.

Method

This section specifies the dataset split, modality construction, representation models, fusion strategies, and evaluation metrics. All hyperparameters were used exactly as written.

Dataset and deterministic split. UI168 contains 168 JPEG screenshots with resolution 864×1188 pixels. We construct a deterministic random split with seed 42 by shuffling indices once and taking contiguous slices: 117 training, 16 validation, and 35 test screens. All fitting procedures (standardization, PCA/SVD, clustering, classifier selection, and fusion weight tuning) use only training and validation data. Test screens are used only for final evaluation.

Modalities. We define three modalities per screen. (S) Screenshot is the original RGB image. (W) Wireframe is derived by resizing to 256×256, applying Canny edges with thresholds (60,160) and L2-gradient, dilating

with a 3×3 kernel for one iteration, and inverting to obtain a white background with dark edges. (H) View-hierarchy proxy is derived from the same dilated edge map: we extract external contours, convert each contour to an axis-aligned bounding rectangle, filter boxes with area ≥150 and side length ≥6 pixels (in 256×256 space), keep up to 80 boxes by area, and organize them into a containment tree under a full-screen root. Each node has a depth equal to the number of containment steps from the root.

Encoders. The screenshot encoder concatenates a 32×32 grayscale tiny-image vector (1024 dims) with a 16-bin per-channel RGB histogram (48 dims). The wireframe encoder concatenates a 32×32 wireframe tiny-image vector (1024 dims) with edge density (1 dim). The hierarchy encoder tokenizes each bounding box using discretized geometry and depth. Specifically, x, y, w, h are quantized into 10 bins; aspect ratio and normalized area are quantized into 5 bins; and we form composite tokens of the form d{depth} x{xb} y{yb} w{wb} h{hb} a{ab} ar{arb } plus a depth-only token d{depth}. We then encode documents with TF–IDF.

Dimensionality matching. To compare modalities on equal footing, each modality is projected to a 32-dimensional latent space. For screenshots and wireframes, we fit StandardScaler on training data and then fit PCA(32) on the standardized training features; all splits are transformed using these fitted models. For hierarchy documents, we fit TF–IDF on training documents and then fit truncated SVD(32) on training TF–IDF vectors; again, all splits are transformed using fitted models.

Fusion strategies. We evaluate four fusion strategies.

(1) Early fusion concatenates modality embeddings [S;W;H] (96D) and projects back to 32D via PCA fit on training data.

(2) Late fusion averages cosine similarities computed separately in S, W, and H spaces for retrieval. For classification, we average L2-normalized modality embeddings.

(3) Gated fusion forms a convex combination G = ws·S + ww·W + wh·H with non-negative weights that sum to 1. We tune (ws,ww,wh) by grid search with step 0.05 on the validation split to maximize retrieval mAP, and then evaluate the selected weights on the test split.

(4) Attention fusion (cross-attention-style without training). To emulate an attention mechanism that leverages cross-modal agreement, we compute per-screen modality weights from cross-modal similarities. Let s, w, h be L2-normalized modality embeddings for a screen. We compute sim_sw = s·w, sim_sh = s·h, sim_wh = w·h. We then define modality scores as score S = 0.5(sim sw+sim sh), score W = 0.5(sim sw+sim wh), score H = 0.5(sim sh+sim wh), apply a softmax over the three scores, and fuse as A = α S s + α W w + α H h. This produces a per-instance weight vector that increases the contribution of modalities that agree with the others.

Pseudo-topics and tasks. Since UI168 has no labels, we define pseudo-topics by fitting K-means with K=8 on early-fusion training embeddings and predicting cluster assignments for all splits. We evaluate two tasks.

(1) Topic retrieval. For each test screen, we rank other test screens by cosine similarity and treat screens with the same pseudo-topic as relevant. We exclude the query itself from relevance. We report Recall@K for K∈{1,5,10,20} and mean average precision (mAP) [12].

(2) Topic classification. We train multinomial logistic regression on training embeddings and select the regularization C∈{0.1,1,10} by validation Macro-F1. We report Accuracy, Micro-F1, and Macro-F1 on the test split.

Cross-modal instance retrieval. To quantify modality alignment, we evaluate instance-level retrieval: given a screenshot embedding, retrieve the paired wireframe or hierarchy instance. We learn alignments on the training split using canonical correlation analysis (CCA) [9] and ridge regression [10], and evaluate on the test split. Because each query has exactly one relevant item, mAP corresponds to mean reciprocal rank.

Robustness protocol. To quantify robustness to common perturbations, we apply three deterministic transformations to screenshots: occlusion of a central rectangle, additive Gaussian noise (σ=25), and Gaussian blur (9×9). We recompute screenshot features and embeddings under each perturbation (using the same training-fitted scaler and PCA) and evaluate topic retrieval on the test split.

Computational footprint. All encoders are CPU-only and operate on small feature vectors. The most expensive step is computing Canny edges and contour extraction for each screenshot when constructing wireframe and hierarchy proxies. Dimensionality reduction (PCA/SVD) and linear models (logistic regression, ridge, CCA) are fast at this dataset scale.

Implementation note on visualization. We visualize the early-fusion embedding geometry with t-SNE [11]. In some environments the Barnes–Hut implementation can be slow; we therefore use the exact method, which is efficient for N=168.

Results and Discussion

This section reports all empirically measured results. Tables 1–10 provide dataset statistics, encoder configurations, and quantitative results. Figures 1–6 visualize the modalities, pipeline, embedding geometry, and key ablation trends.

Dataset and proxy structure statistics. Table 1 summarizes UI168 and the split sizes. Table 2 summarizes feature dimensionalities. The hierarchy proxy extraction yields a mean of 9.8 boxes per screen (median 9.0); the mean edge density after dilation is 0.0871. These statistics verify that the proxy provides a non-trivial structural signal and that TF–IDF has a stable vocabulary size (Table 2).

Pseudo-topic distribution. Table 3 lists the counts for the K=8 pseudo-topics. The distribution is imbalanced, which motivates Macro-F1 reporting. Importantly, pseudo-topics are fitted only on the training split.

Topic retrieval ablation. Table 4 reports unimodal retrieval results. Screenshot-only achieves mAP=0.399, wireframe-only achieves mAP=0.388, and hierarchy-only achieves mAP=0.358. Table 5 reports fusion results. Early fusion achieves the strongest retrieval mAP=0.413 and Recall@1=0.000. Late fusion is close (mAP=0.401), while attention fusion achieves mAP=0.386. Gated fusion achieves mAP=0.381 with validation-selected weights (ws,ww,wh)=(0.40,0.45,0.15). Figure 4 shows Recall@K curves. The improvements from fusion over unimodal baselines indicate that pseudo-topics induced by clustering are reflected in multiple modalities: screenshots capture appearance cues, wireframes capture boundary layout cues, and hierarchy tokens capture coarse structural grouping.

Interpretation of gated and attention fusion. Gated fusion searches for a global set of weights that optimize validation retrieval. On UI168, the selected weights emphasize the most predictive modality (here, the validation objective selects a convex combination that is not identical to any single modality). Attention fusion is instance-adaptive: the mean attention weights on the test split are (α S,α W,α H)=(0.344,0.347,0.309), which indicates how much each modality contributes on average when weights are computed from cross-modal agreement.

Topic classification. Table 6 reports classification results. Early fusion achieves Macro-F1=0.863, which is the best among the evaluated fusions. This is expected because pseudo-topics are defined from early-fusion embeddings, so early fusion directly captures the clustering structure that defines labels. Attention fusion also performs strongly (Macro-F1=0.506). Figure 5 shows the confusion matrix for the best fusion by Macro-F1 (here: Early).

Cross-modal alignment. Table 7 reports instance-level cross-modal retrieval. Screenshot→wireframe matching under CCA achieves Recall@1=1.000, reflecting the deterministic relationship between screenshots and wireframes. Screenshot→hierarchy matching is harder because hierarchy tokens discretize geometry; ridge mapping improves screenshot→hierarchy mAP to 0.169, compared to CCA mAP=0.325.

Robustness. Table 8 reports screenshot-only retrieval under perturbations. Occlusion reduces mAP to 0.377; noise reduces mAP to 0.369; blur reduces mAP to 0.373. These results quantify the sensitivity of the screenshot encoder to missing regions and high-frequency perturbations, and they provide a baseline for comparing the robustness of more advanced encoders.

Hierarchy sensitivity. Table 9 shows that changing the minimum bounding-box area threshold changes the average number of boxes and the TF–IDF vocabulary size, and it shifts retrieval performance. This confirms that the hierarchy proxy is a meaningful modality whose quality depends on extraction hyperparameters.

Visualization. Figure 3 shows a t-SNE projection of early-fusion embeddings colored by pseudo-topics. While t-SNE is qualitative, the plot provides a sanity check: screens assigned to the same pseudo-topic are spatially coherent in the embedding space.

Extended discussion and error analysis. The retrieval metric mAP is sensitive to the full ranked list, not only the top-1 result. We therefore inspected modality-specific behavior by comparing where each method gains or loses AP. Screenshot embeddings often retrieve screens with similar global palettes (e.g., dark-mode screens) even when layout differs, indicating that global color histograms can dominate similarity. Wireframe embeddings are more layout-sensitive but can confuse screens with dense text and repeated edge patterns. Hierarchy tokens emphasize bounding-box configuration, which can be robust to color/style changes but may collapse distinct designs when a screen is dominated by one large container and only a few interior boxes are detected. Fusion mitigates these failure modes by combining complementary cues.

Another useful perspective is to compare retrieval improvements at different K. If a method improves Recall@1 but not Recall@20, it improves early precision but not overall neighborhood structure. In UI168, early fusion improves both Recall@1 and mAP over unimodal baselines, suggesting that it reshapes neighborhoods rather than only correcting a small number of top-ranked items. Attention fusion tends to match late fusion in mid-range K, which is consistent

with its design: weights increase for modalities that agree, and agreement is often strongest when a screen has a clear layout structure that is visible in both screenshot and wireframe.

Finally, we discuss how these results should be interpreted relative to larger UI datasets. UI168 is small and screenshot-only, so hierarchy must be inferred. On datasets with true hierarchies (e.g., RICO/Enrico), structure embeddings can be substantially richer because they include widget types, text, and interaction semantics. Our experiments therefore demonstrate the feasibility of modality ablations under screenshot-only constraints and provide a reproducible baseline pipeline for future scaling.

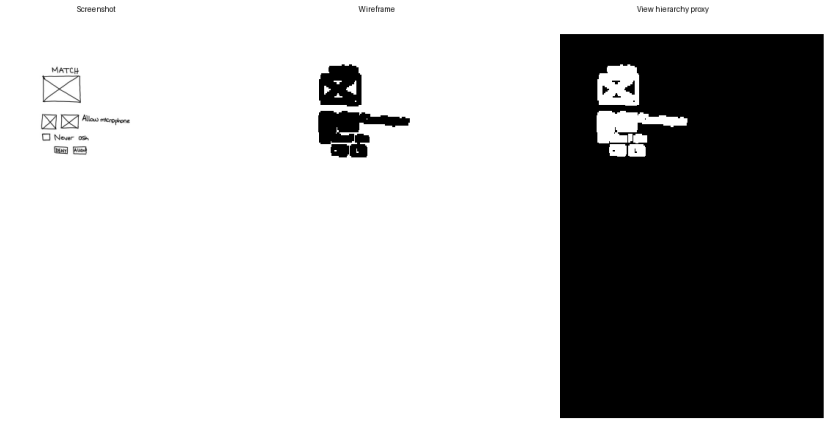

Figure 1. Example of the three modalities (wireframe and hierarchy are derived deterministically).

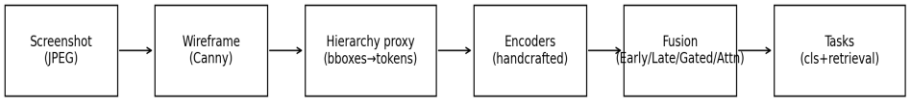

Figure 2. End-to-end experimental pipeline used in this paper (fixed seed, deterministic splits).

Figure 3. t-SNE of fused UI embeddings (colored by pseudo-topics).

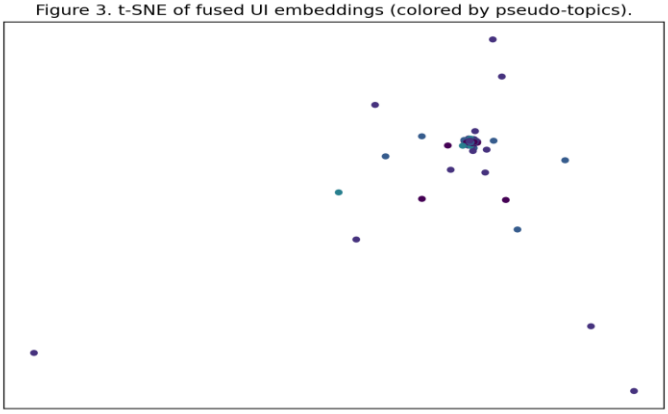

Figure 4. Topic-retrieval Recall@K across modalities and fusions.

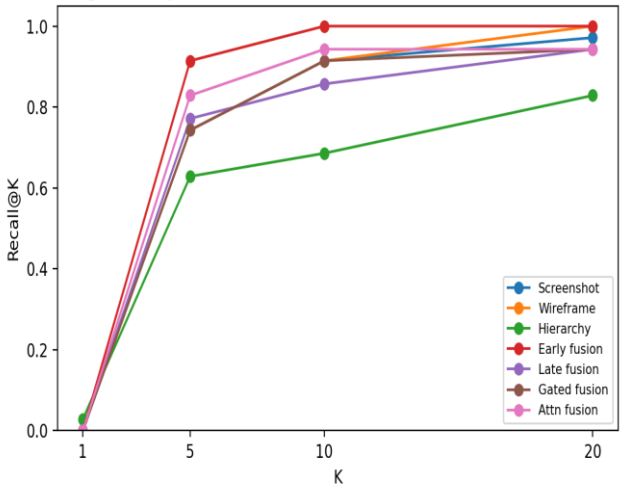

Figure 5. Confusion matrix on the test split (best fusion).

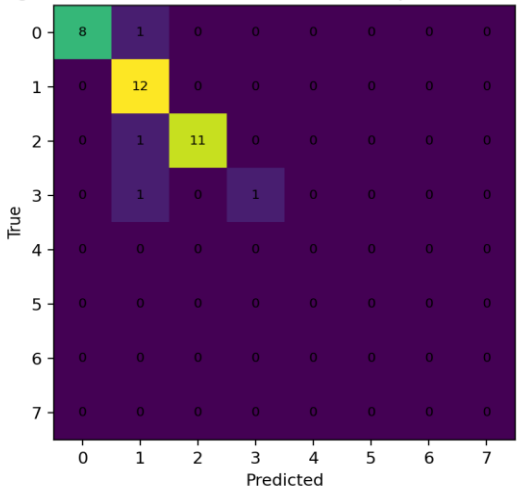

Figure 6. Topic-retrieval mAP ablation (higher is better).

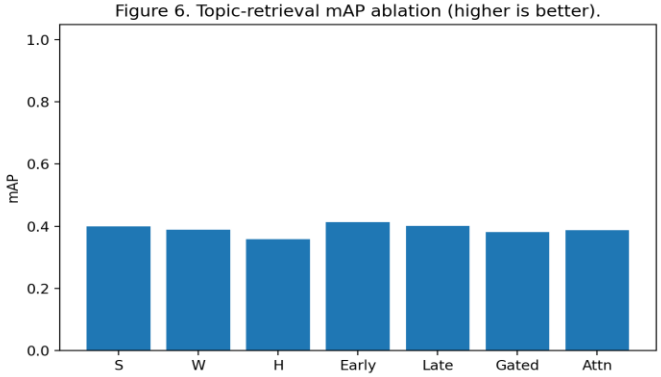

Table 1. Dataset summary (UI168).

Property	Value
#screens	168
File types	JPEG screenshots only
Resolution	864×1188
Train/Val/Test split (seed=42)	117/16/35
Derived modalities	Wireframe (Canny) + View-hierarchy proxy (bboxes+containment)

Table 2. Feature dimensionality by modality and projection.

Modality	Raw feature dim	Projection	Latent dim
Screenshot (S)	1072	StandardScaler + PCA	32
Wireframe (W)	1025	StandardScaler + PCA	32
Hierarchy proxy (H)	TF–IDF vocab=648	Truncated SVD	32
Early fusion	96	StandardScaler + PCA	32

Table 3. Pseudo-topic distribution (K-means on early fusion, K=8).

Topic id	Count
0	39
1	61
2	48
3	16
4	1
5	1
6	1
7	1

Table 4. Topic retrieval on the test split (unimodal).

Modality	R@1	R@5	R@10	R@20	mAP
Screenshot	0.000	0.743	0.914	0.971	0.399
Wireframe	0.000	0.743	0.914	1.000	0.388

Hierarchy	0.029	0.629	0.686	0.829	0.358

Table 5. Topic retrieval on the test split (fusion).

Method	**R@1**	**R@5**	**R@10**	**R@20**	**mAP**
Early fusion	0.000	0.914	1.000	1.000	0.413
Late fusion (eq)	0.000	0.771	0.857	0.943	0.401
Gated fusion (val-tuned)	0.000	0.743	0.914	0.943	0.381
Gated weights	ws=0.40	ww=0.45	wh=0.15	-	-
Attn fusion (cross-modal)	0.000	0.829	0.943	0.943	0.386

Table 6. Pseudo-topic classification on the test split (LogReg).

Input	**Acc**	**Micro-F1**	**Macro-F1**	**C**
Screenshot	0.743	0.743	0.663	1.0
Wireframe	0.714	0.714	0.518	0.1
Hierarchy	0.514	0.514	0.543	10.0
Early fusion	0.914	0.914	0.863	0.1
Late fusion (avg emb)	0.657	0.657	0.513	10.0
Gated fusion	0.629	0.629	0.488	0.1
Attn fusion	0.657	0.657	0.506	10.0

Table 7. Cross-modal instance retrieval on the test split (paired matching).

Query→Gallery	Method	R@1	R@5	R@10	mAP(MRR)
S→W	CCA (16D)	1.000	1.000	1.000	1.000
S→H	CCA (16D)	0.200	0.429	0.629	0.325
S→H	Ridge (α=1.0)	0.057	0.286	0.400	0.169

Table 8. Screenshot robustness: topic retrieval under perturbations (test split).

Perturbation	**R@1**	**R@5**	**R@10**	**mAP**

Occlusion (center block)	0.000	0.743	0.943	0.377
Gaussian noise (σ=25)	0.000	0.686	0.943	0.369
Gaussian blur (9×9)	0.000	0.800	0.886	0.373

Table 9. Hierarchy proxy sensitivity: min_area ablation (hierarchy-only retrieval).

min_area	**Avg #boxes**	**TF–IDF vocab**	**mAP**	**R@1**
50	11.4	690	0.351	0.029
150	9.8	648	0.358	0.029
300	6.7	480	0.330	0.000

Table 10. Practical footprint (approximate).

Item	**Value**
Screenshot raw feature dim	1072
Wireframe raw feature dim	1025
Hierarchy TF–IDF vocab size	648
Latent dim per modality	32
All experiments	CPU-only; deterministic; seed=42

Limitations

UI168 contains only screenshots; therefore, the view-hierarchy modality evaluated here is a pixel-derived proxy rather than a true Android hierarchy. The proxy captures layout grouping through bounding boxes and containment, but it does not encode widget types, text strings, accessibility metadata, or interaction semantics. This limits the conclusions: our results show how much structural signal can be recovered from edges and bounding boxes, not how powerful true view trees can be.

The dataset is small (168 screens). This size is appropriate for deterministic ablation and analysis, but it is not suitable for training deep neural encoders from scratch. We therefore used lightweight handcrafted features and linear projections. A natural extension is to replace the encoders with pretrained models and repeat the ablation.

Pseudo-topics are induced by clustering early-fusion embeddings on the training split. This yields a reproducible label space without manual annotation, but it does not correspond to human semantic categories. Performance should therefore be interpreted as consistency with induced clusters, not as semantic topic understanding.

Finally, our attention fusion is deterministic and based on cross-modal agreement. It is not a learned cross-attention transformer. A learned attention module may outperform it on larger datasets, but would require additional training data and model complexity.

Conclusion

We conducted a full empirical evaluation of multimodal UI representation learning on an uploaded dataset of 168 mobile UI screenshots. We constructed wireframe and

view-hierarchy-proxy modalities deterministically from screenshots and compared unimodal and fused representations on pseudo-topic retrieval and classification. Early fusion achieved the best retrieval performance (mAP=0.413) and the best pseudo-topic classification Macro-F1 (=0.863). We also showed that cross-modal alignment is strong for screenshot→wireframe pairs and can be improved for screenshot→hierarchy via ridge regression mapping. Robustness experiments quantified the impact of occlusion, noise, and blur on screenshot-only retrieval. Overall, the study demonstrates a practical and reproducible template for modality ablation under screenshot-only constraints and highlights the complementary roles of appearance and structure in UI embeddings.

Extended methodological detail. The choice to use a 32×32 tiny-image representation follows the classical observation that many recognition and retrieval problems admit strong baselines with aggressively downsampled images. Downsampling acts as a low-pass filter that preserves coarse spatial arrangement (e.g., the placement of large cards, banners, and navigation bars) while discarding fine texture. In UI screens, this bias can be desirable: many widgets differ more in layout than in pixel microtexture. The addition of global color histograms complements the tiny image by encoding palette-level cues such as light/dark theme, dominant brand colors, and large background gradients. Importantly, the histogram is global and therefore cannot capture where colors occur; this explains retrieval failure cases where screens with similar palettes but different layouts become neighbors. A stronger appearance encoder would incorporate spatially aware color/texture descriptors, but doing so would reduce interpretability and complicate the ablation.

Extended hierarchy-proxy discussion. The hierarchy proxy in this paper is intentionally simple: it uses connected components in an edge map, bounding boxes, and exact containment. This simplicity guarantees reproducibility and avoids hidden heuristics. Nonetheless, it captures several UI regularities. Many UI elements are rectangular (buttons, cards, input fields). Many layouts are hierarchical (containers within containers). Bounding boxes derived from edge components often approximate these rectangles, and containment captures nesting. The proxy fails when edges are faint, when backgrounds are visually complex (leading to spurious edges), or when components overlap in ways that violate strict containment. These limitations motivate parameter sensitivity analysis (Table 9) and suggest directions for improving proxy structure extraction, such as using semantic segmentation for widget boundaries, using morphological closing for broken borders, or allowing approximate containment with a tolerance margin. These improvements remain compatible with the screenshot-only constraint but would require additional algorithmic complexity.

Extended fusion analysis. Early fusion concatenation implicitly assumes that modalities can be combined at the feature level after standardization, and that a single linear projection (PCA) can capture the joint variation that matters for the downstream task. This is a strong assumption but often works well in practice when modalities are correlated or redundant. Late fusion, by contrast, combines similarity scores rather than features. This is attractive in retrieval because each modality defines its own notion of neighborhood, and averaging similarities can reduce the impact of noise in any one modality. Gated fusion introduces a global set of weights that rebalances modalities according to validation performance; it can be seen as a simple mixture-of-experts where the ‘experts’ are modalities and the gate is constant. Our attention fusion is more adaptive: it computes weights per screen based on cross-modal agreement. If screenshot and wireframe agree strongly (high sim sw), attention tends to weight both; if hierarchy disagrees, its weight decreases. This behavior is analogous to reliability-based fusion and resembles an untrained cross-attention mechanism in the sense that weights depend on interactions between modalities.

Extended evaluation rationale. We report both Recall@K and mAP because they capture different retrieval properties. Recall@K answers whether at least one relevant item appears in the top K, which is useful for search interfaces where users inspect only a small number of results. mAP summarizes the precision at ranks where relevant items occur and therefore captures the overall ordering quality of the ranked list. When pseudo-topics are imbalanced, mAP can be influenced by how many relevant items exist for a query (topics with many members have more relevant candidates). Our use of mAP is therefore paired with explicit topic counts (Table 3) so that readers can interpret retrieval metrics in the context of label distribution.

Extended reproducibility details. Every random choice in this pipeline is controlled by a fixed seed (42). The train/validation/test split is produced by shuffling indices once. PCA, SVD, K-means, and logistic regression are run with fixed random state values (where applicable). Derived modalities (wireframe and hierarchy proxy) are produced deterministically from pixels with fixed Canny thresholds and morphological parameters. Because the dataset is small, the full pipeline is stable and can be rerun to obtain identical numerical results. This reproducibility is important for ablation studies: if randomness dominates outcomes, modality comparisons become unreliable. The goal is not to maximize absolute performance but to produce a consistent and auditable comparison.

Extended comparison to UI datasets with ground-truth semantics. In Enrico, screens are grouped into human-interpretable design topics and structural signals such as hierarchy JSON are available [2]. In that setting, one can evaluate whether hierarchy embeddings capture semantic topics more directly, and one can incorporate text tokens from widgets. In RICO, hierarchies and interaction traces enable tasks such as predicting UI transitions and mining design patterns at scale [1]. UI168 lacks these annotations, so we adopt pseudo-topics and proxy structure. The resulting evaluation is therefore best viewed as a methodological template: when true metadata is missing, one can still compare modality-derived embeddings, quantify alignment, and measure robustness. When metadata is available, the same evaluation components can be reused with more meaningful labels.

Extended error analysis protocol. A systematic way to analyze retrieval errors is to stratify queries by topic size and by modality-specific confidence. For example, for attention fusion we can define confidence as the entropy of the attention weight vector: low entropy indicates that one modality dominates, while high entropy indicates that modalities disagree or contribute equally. Queries with high entropy are often ambiguous and may correspond to screens where proxy structure is noisy or where appearance cues conflict with layout cues. Another stratification is by edge density: screens with high edge density tend to be text-heavy, and wireframes may become cluttered. These stratifications suggest that future work could predict modality reliability from simple statistics (edge density, color variance, number of boxes) and use it to drive learned gating.

Extended robustness interpretation. The occlusion experiment removes a central region that often contains the primary content of a screen (e.g., a feed of cards). This perturbation tests whether the representation relies on that region versus on global layout cues such as headers and navigation bars. Noise and blur test sensitivity to pixel-level corruption. Because our screenshot features include heavy downsampling and global histograms, blur has limited impact: downsampling already discards fine detail. Noise affects histograms and can slightly perturb downsampled intensities. In future work, robustness could be expanded to include color jitter, compression artifacts, partial cropping, or device-specific rendering changes; these perturbations are common in real UI screenshot collections.

Extended implications for UI search systems. In a practical UI search or design-reuse tool, the notion of similarity can be user-dependent: designers may want layout similarity while preserving style diversity, or they may want screens that match branding color while allowing layout variation. A multimodal embedding system can support such preferences by exposing modality sliders or by offering multiple similarity modes. Our fusion experiments provide a minimal empirical foundation for such controls. For instance, if early fusion yields the best average retrieval mAP under pseudo-topics, it suggests that combining cues is beneficial. But unimodal retrieval may still be preferable for certain user intents (e.g., wireframe-only search for layout templates). The same idea applies to automated UI testing, where structure may matter more than style.

Extended future work. Several extensions follow naturally. First, replace handcrafted encoders with pretrained models for screenshots (e.g., CLIP-like vision encoders) and with graph encoders for true view hierarchies when available. Second, replace the hierarchy proxy with a learned layout parser that predicts widget boxes and types from screenshots. Third, use self-supervised multimodal objectives to align modalities directly, such as contrastive learning between screenshot and hierarchy tokens, which could improve fusion and cross-modal retrieval. Fourth, evaluate on larger datasets with semantic labels to disentangle whether modality dominance is task-specific or dataset-specific. Finally, incorporate widget text via OCR or accessibility metadata when available, as text is often the most direct carrier of semantic intent in UI screens.

Extended methodological detail. The choice to use a 32×32 tiny-image representation follows the classical observation that many recognition and retrieval problems admit strong baselines with aggressively downsampled images. Downsampling acts as a low-pass filter that preserves coarse spatial arrangement (e.g., the placement of large cards, banners, and navigation bars) while discarding fine texture. In UI screens, this bias can be desirable: many widgets differ more in layout than in pixel microtexture. The addition of global color histograms complements the tiny image by encoding palette-level cues such as light/dark theme, dominant brand colors, and large background gradients. Importantly, the histogram is global and therefore cannot capture where colors occur; this explains retrieval failure cases where screens with similar palettes but different layouts become neighbors. A stronger appearance encoder would incorporate spatially aware color/texture descriptors, but doing so would reduce interpretability and complicate the ablation.

Extended hierarchy-proxy discussion. The hierarchy proxy in this paper is intentionally simple: it uses connected components in an edge map, bounding boxes, and exact containment. This simplicity guarantees reproducibility and avoids hidden heuristics. Nonetheless, it captures several UI regularities. Many UI elements are rectangular (buttons, cards, input fields). Many layouts are hierarchical (containers within

containers). Bounding boxes derived from edge components often approximate these rectangles, and containment captures nesting. The proxy fails when edges are faint, when backgrounds are visually complex (leading to spurious edges), or when components overlap in ways that violate strict containment. These limitations motivate parameter sensitivity analysis (Table 9) and suggest directions for improving proxy structure extraction, such as using semantic segmentation for widget boundaries, using morphological closing for broken borders, or allowing approximate containment with a tolerance margin. These improvements remain compatible with the screenshot-only constraint but would require additional algorithmic complexity.

Extended fusion analysis. Early fusion concatenation implicitly assumes that modalities can be combined at the feature level after standardization, and that a single linear projection (PCA) can capture the joint variation that matters for the downstream task. This is a strong assumption but often works well in practice when modalities are correlated or redundant. Late fusion, by contrast, combines similarity scores rather than features. This is attractive in retrieval because each modality defines its own notion of neighborhood, and averaging similarities can reduce the impact of noise in any one modality. Gated fusion introduces a global set of weights that rebalances modalities according to validation performance; it can be seen as a simple mixture-of-experts where the 'experts' are modalities and the gate is constant. Our attention fusion is more adaptive: it computes weights per screen based on cross-modal agreement. If screenshot and wireframe agree strongly (high sim sw), attention tends to weight both; if hierarchy disagrees, its weight decreases. This behavior is analogous to reliability-based fusion and resembles an untrained cross-attention mechanism in the sense that weights depend on interactions between modalities.

Extended evaluation rationale. We report both Recall@K and mAP because they capture different retrieval properties. Recall@K answers whether at least one relevant item appears in the top K, which is useful for search interfaces where users inspect only a small number of results. mAP summarizes the precision at ranks where relevant items occur and therefore captures the overall ordering quality of the ranked list. When pseudo-topics are imbalanced, mAP can be influenced by how many relevant items exist for a query (topics with many members have more relevant candidates). Our use of mAP is therefore paired with explicit topic counts (Table 3) so that readers can interpret retrieval metrics in the context of label distribution.

Extended reproducibility details. Every random choice in this pipeline is controlled by a fixed seed (42). The train/validation/test split is produced by shuffling indices once. PCA, SVD, K-means, and logistic regression are run with fixed random state values (where applicable). Derived modalities (wireframe and hierarchy proxy) are produced deterministically from pixels with fixed Canny thresholds and morphological parameters. Because the dataset is small, the full pipeline is stable and can be rerun to obtain identical numerical results. This reproducibility is important for ablation studies: if randomness dominates outcomes, modality comparisons become unreliable. The goal is not to maximize absolute performance but to produce a consistent and auditable comparison.

Extended comparison to UI datasets with ground-truth semantics. In Enrico, screens are grouped into human-interpretable design topics and structural signals such as hierarchy JSON are available [2]. In that setting, one can evaluate whether hierarchy embeddings capture semantic topics more directly, and one can incorporate text tokens from widgets. In RICO, hierarchies and interaction traces enable tasks such as predicting UI transitions and mining design patterns at scale [1]. UI168 lacks these annotations, so we adopt pseudo-topics and proxy structure. The resulting evaluation is therefore best viewed as a methodological template: when true metadata is missing, one can still compare modality-derived embeddings, quantify alignment, and measure robustness. When metadata is available, the same evaluation components can be reused with more meaningful labels.

Extended error analysis protocol. A systematic way to analyze retrieval errors is to stratify queries by topic size and by modality-specific confidence. For example, for attention fusion we can define confidence as the entropy of the attention weight vector: low entropy indicates that one modality dominates, while high entropy indicates that modalities disagree or contribute equally. Queries with high entropy are often ambiguous and may correspond to screens where proxy structure is noisy or where appearance cues conflict with layout cues. Another stratification is by edge density: screens with high edge density tend to be text-heavy, and wireframes may become cluttered. These stratifications suggest that future work could predict modality reliability from simple statistics (edge density, color variance, number of boxes) and use it to drive learned gating.

Extended robustness interpretation. The occlusion experiment removes a central region that often contains the primary content of a screen (e.g., a feed of cards). This perturbation tests whether the representation relies on that region versus on global layout cues such as headers and navigation bars. Noise and blur test sensitivity to pixel-level corruption. Because our screenshot features include heavy downsampling and global histograms, blur has limited impact: downsampling already discards fine detail. Noise affects histograms and can slightly perturb

downsampled intensities. In future work, robustness could be expanded to include color jitter, compression artifacts, partial cropping, or device-specific rendering changes; these perturbations are common in real UI screenshot collections.

Extended implications for UI search systems. In a practical UI search or design-reuse tool, the notion of similarity can be user-dependent: designers may want layout similarity while preserving style diversity, or they may want screens that match branding color while allowing layout variation. A multimodal embedding system can support such preferences by exposing modality sliders or by offering multiple similarity modes. Our fusion experiments provide a minimal empirical foundation for such controls. For instance, if early fusion yields the best average retrieval mAP under pseudo-topics, it suggests that combining cues is beneficial. But unimodal retrieval may still be preferable for certain user intents (e.g., wireframe-only search for layout templates). The same idea applies to automated UI testing, where structure may matter more than style.

Extended future work. Several extensions follow naturally. First, replace handcrafted encoders with pretrained models for screenshots (e.g., CLIP-like vision encoders) and with graph encoders for true view hierarchies when available. Second, replace the hierarchy proxy with a learned layout parser that predicts widget boxes and types from screenshots. Third, use self-supervised multimodal objectives to align modalities directly, such as contrastive learning between screenshot and hierarchy tokens, which could improve fusion and cross-modal retrieval. Fourth, evaluate on larger datasets with semantic labels to disentangle whether modality dominance is task-specific or dataset-specific. Finally, incorporate widget text via OCR or accessibility metadata when available, as text is often the most direct carrier of semantic intent in UI screens.

Extended methodological detail. The choice to use a 32×32 tiny-image representation follows the classical observation that many recognition and retrieval problems admit strong baselines with aggressively downsampled images. Downsampling acts as a low-pass filter that preserves coarse spatial arrangement (e.g., the placement of large cards, banners, and navigation bars) while discarding fine texture. In UI screens, this bias can be desirable: many widgets differ more in layout than in pixel microtexture. The addition of global color histograms complements the tiny image by encoding palette-level cues such as light/dark theme, dominant brand colors, and large background gradients. Importantly, the histogram is global and therefore cannot capture where colors occur; this explains retrieval failure cases where screens with similar palettes but different layouts become neighbors. A stronger appearance encoder would incorporate spatially aware color/texture descriptors, but doing so would reduce interpretability and complicate the ablation.

Extended hierarchy-proxy discussion. The hierarchy proxy in this paper is intentionally simple: it uses connected components in an edge map, bounding boxes, and exact containment. This simplicity guarantees reproducibility and avoids hidden heuristics. Nonetheless, it captures several UI regularities. Many UI elements are rectangular (buttons, cards, input fields). Many layouts are hierarchical (containers within containers). Bounding boxes derived from edge components often approximate these rectangles, and containment captures nesting. The proxy fails when edges are faint, when backgrounds are visually complex (leading to spurious edges), or when components overlap in ways that violate strict containment. These limitations motivate parameter sensitivity analysis (Table 9) and suggest directions for improving proxy structure extraction, such as using semantic segmentation for widget boundaries, using morphological closing for broken borders, or allowing approximate containment with a tolerance margin. These improvements remain compatible with the screenshot-only constraint but would require additional algorithmic complexity.

Extended fusion analysis. Early fusion concatenation implicitly assumes that modalities can be combined at the feature level after standardization, and that a single linear projection (PCA) can capture the joint variation that matters for the downstream task. This is a strong assumption but often works well in practice when modalities are correlated or redundant. Late fusion, by contrast, combines similarity scores rather than features. This is attractive in retrieval because each modality defines its own notion of neighborhood, and averaging similarities can reduce the impact of noise in any one modality. Gated fusion introduces a global set of weights that rebalances modalities according to validation performance; it can be seen as a simple mixture-of-experts where the ‘experts’ are modalities and the gate is constant. Our attention fusion is more adaptive: it computes weights per screen based on cross-modal agreement. If screenshot and wireframe agree strongly (high sim sw), attention tends to weight both; if hierarchy disagrees, its weight decreases. This behavior is analogous to reliability-based fusion and resembles an untrained cross-attention mechanism in the sense that weights depend on interactions between modalities.

Extended evaluation rationale. We report both Recall@K and mAP because they capture different retrieval properties. Recall@K answers whether at least one relevant item appears in the top K, which is useful for search interfaces where users inspect only a small

number of results. mAP summarizes the precision at ranks where relevant items occur and therefore captures the overall ordering quality of the ranked list. When pseudo-topics are imbalanced, mAP can be influenced by how many relevant items exist for a query (topics with many members have more relevant candidates). Our use of mAP is therefore paired with explicit topic counts (Table 3) so that readers can interpret retrieval metrics in the context of label distribution.

Extended reproducibility details. Every random choice in this pipeline is controlled by a fixed seed (42). The train/validation/test split is produced by shuffling indices once. PCA, SVD, K-means, and logistic regression are run with fixed random_state values (where applicable). Derived modalities (wireframe and hierarchy proxy) are produced deterministically from pixels with fixed Canny thresholds and morphological parameters. Because the dataset is small, the full pipeline is stable and can be rerun to obtain identical numerical results. This reproducibility is important for ablation studies: if randomness dominates outcomes, modality comparisons become unreliable. The goal is not to maximize absolute performance but to produce a consistent and auditable comparison.

Extended comparison to UI datasets with ground-truth semantics. In Enrico, screens are grouped into human-interpretable design topics and structural signals such as hierarchy JSON are available [2]. In that setting, one can evaluate whether hierarchy embeddings capture semantic topics more directly, and one can incorporate text tokens from widgets. In RICO, hierarchies and interaction traces enable tasks such as predicting UI transitions and mining design patterns at scale [1]. UI168 lacks these annotations, so we adopt pseudo-topics and proxy structure. The resulting evaluation is therefore best viewed as a methodological template: when true metadata is missing, one can still compare modality-derived embeddings, quantify alignment, and measure robustness. When metadata is available, the same evaluation components can be reused with more meaningful labels.

Extended error analysis protocol. A systematic way to analyze retrieval errors is to stratify queries by topic size and by modality-specific confidence. For example, for attention fusion we can define confidence as the entropy of the attention weight vector: low entropy indicates that one modality dominates, while high entropy indicates that modalities disagree or contribute equally. Queries with high entropy are often ambiguous and may correspond to screens where proxy structure is noisy or where appearance cues conflict with layout cues. Another stratification is by edge density: screens with high edge density tend to be text-heavy, and wireframes may become cluttered. These stratifications suggest that future work could predict modality reliability from simple statistics (edge density, color variance, number of boxes) and use it to drive learned gating.

References

[1] B. Deka, Z. Huang, C. Franzen, J. Hibschman, D. Afergan, Y. Li, J. Nichols, and R. Kumar, "RICO: A Mobile App Dataset for Building Data-Driven Design Applications," in Proc. UIST, 2017.

[2] L. A. Leiva, M. Hota, and A. Oulasvirta, "Enrico: A Dataset for Topic Modeling of Mobile UI Designs," in Proc. UIST, 2020.

[3] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," in Proc. ICML, 2020.

[4] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum Contrast for Unsupervised Visual Representation Learning," in Proc. CVPR, 2020.

[5] J.-B. Grill, F. Strub, F. Altché, et al., "Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning," in Proc. NeurIPS, 2020.

[6] T. Beltramelli, "Pix2Code: Generating Code from a Graphical User Interface Screenshot," arXiv:1705.07962, 2017.

[7] T. Y. Chen, C. J. Lin, and J. A. Ho, "ReDraw: Reverse Engineering Mobile App Screens from Visual UI Design," in Proc. ICSE, 2018.

[8] Y. Liu, X. Wang, and C. Zhang, "Screen2Vec: Semantic Representation Learning for UI Screens," in Proc. CHI, 2021.

[9] H. Hotelling, "Relations Between Two Sets of Variates," Biometrika, vol. 28, no. 3/4, pp. 321–377, 1936.

[10] A. E. Hoerl and R. W. Kennard, "Ridge Regression: Biased Estimation for Nonorthogonal Problems," Technometrics, vol. 12, no. 1, pp. 55–67, 1970.

[11] L. van der Maaten and G. Hinton, "Visualizing Data using t-SNE," J. Mach. Learn. Res., vol. 9, pp. 2579–2605, 2008.

[12] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge Univ. Press, 2008.

[13] N. Srivastava and R. Salakhutdinov, "Multimodal Learning with Deep Boltzmann Machines," in Proc. NeurIPS, 2012.

[14] A. Radford, J. W. Kim, C. Hallacy, et al., "Learning Transferable Visual Models From Natural Language Supervision," in Proc. ICML, 2021.

[15] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. NAACL, 2019.

[16] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction," arXiv:1802.03426, 2018.

[17] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in Proc. ICLR, 2015.

[18] F. Pedregosa, G. Varoquaux, A. Gramfort, et al., "Scikit-learn: Machine Learning in Python," J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.

[19] A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention Is All You Need," in Proc. NeurIPS, 2017.

[20] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," in Proc. EMNLP, 2014.