Exploiting Vision-Language Models in GUI Reuse

Victoria Niu
Walnut Hills High School
Cincinnati, OH, USA
victoria.j.niu@gmail.com

Walaa Alshammari
University of Cincinnati
Cincinnati, OH, USA
alshamwm@mail.uc.edu

Naga Mamata Iluru
University of Cincinnati
Cincinnati, OH, USA
iluruna@mail.uc.edu

Padmaja Vaishnavi Teeleti
University of Cincinnati
Cincinnati, OH, USA
teeletpi@mail.uc.edu

Nan Niu
University of Cincinnati
Cincinnati, OH, USA
nan.niu@uc.edu

Tanmay Bhowmik
Mississippi State University
Mississippi State, MS, USA
tbhowmik@cse.msstate.edu

Jianzhang Zhang
Hangzhou Normal University
Hangzhou, China
jianzhang.zhang@foxmail.com

***Abstract*—Graphical user interface (GUI) prototyping helps to clarify requirements and keep stakeholders engaged in software development. While contemporary approaches retrieve GUIs relevant to a user's query, little support exists for the actual reuse, i.e., for using an existing GUI to create a new one. To shorten the gap, we investigate GUI-centered reuse via one of the latest artificial intelligence (AI) techniques—vision-language models (VLMs). We report an empirical study involving 73 university students working on ten GUI reuse tasks. Each task is associated with different reuse directions recommended by VLMs and by a natural language (NL) method. In addition, a focused GUI element is provided to offer a starting point for making the actual changes. Our results show that VLMs significantly outperform the NL method in making reuse recommendations, but surprisingly, the focused GUI elements are not consistently modified during reuse. With the assessments made by four experienced designers, we further offer insights into the creativity of human-reuse and AI-reuse results.**

***Index Terms*—vision-language models, creativity in software engineering, multimodal computing**

I. Introduction

Graphical user interface (GUI) prototyping is a crucial technique that enables developers to create an initial version of the design. This technique is highly valuable in the software engineering process. For example, it helps to clarify requirements, spur ideation, and promote stakeholders' engagement [1], [2]. Despite the benefits, GUI prototyping from scratch can be time-consuming and costly [3].

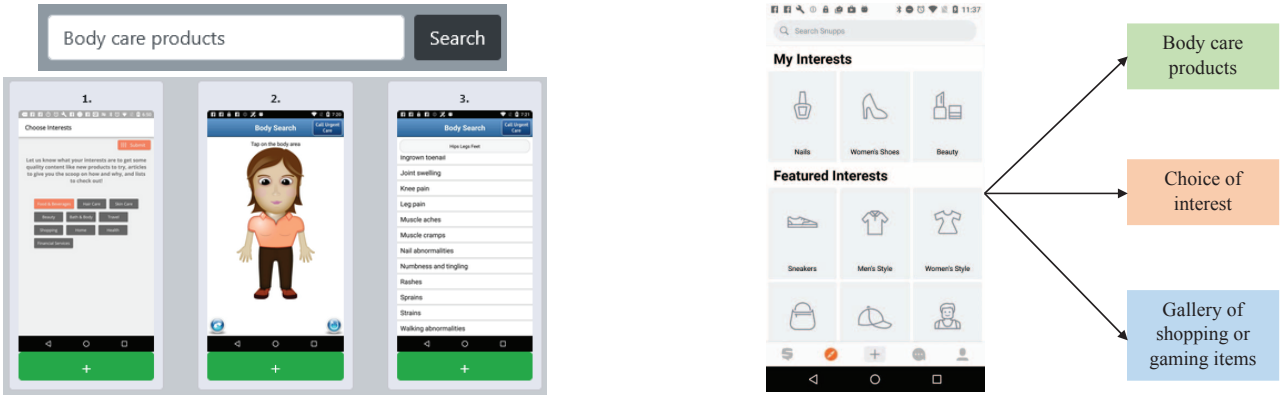
Reusing available GUIs has the potential to reduce prototyping effort [4]. Existing approaches focus mainly on retrieving candidate GUIs relevant to a user's query. For instance, RaWi [5] takes a natural language (NL) query as the input and returns the GUI candidates based on the query's similarity with each GUI's textual profile. Fig. 1-a illustrates RaWi, which further supports reuse by allowing a designer to select and edit a specific GUI.

While retrieval facilitates the identification of reuse candidates, little support exists for the actual reuse, i.e., for using an existing GUI to create a new one [6]. Fig. 1-b depicts GUI-centered reuse that our work addresses where a given GUI corresponds to more than one query (or keyphrase), indicating multiple ways to pursue the reuse. We view Fig. 1-b as a complement to Fig. 1-a, and yet, on its own, GUI-centered reuse is not constrained by a pre-defined query. Rather, the GUI is treated as a first-class citizen, and the keyphrases become their associations, plotting different reuse possibilities.

A major reason why we are interested in ways to revise a GUI is creativity, which hinges critically on clever reuse in software developers' daily work [7]. Besides the reuse directions suggested in the NL keyphrases, we also pay attention to the reuse of a particular element (e.g., a label or an icon) instead of the entire GUI. Kolthoff *et al.* [8] recently showed the importance of requirements validation at the granularity of GUI element. We therefore posit that the specific element would help to offer both explainability [9] and a starting point for making the actual changes [10].

In this paper, we investigate the GUI reuse support provided by one of the latest artificial intelligence (AI) techniques—vision-language models (VLMs). VLMs, such as CLIP [11] and BLIP [12], are trained on large-scale image-caption data using contrastive learning, and have been applied to GUI retrieval [13]. We extend VLMs' uses in recommending reuse options along with focused GUI elements. We compare VLMs by considering NL-based approaches; in particular, we develop a synthesis of RaWi [5] that computes GUI-keyphrase matches and XUI [14] that identifies the salient element in a GUI. Thus, for each GUI, various reuse choices are depicted by the NL-based (RaWi+XUI) method and by the VLMs.

To evaluate which options are viable, we conducted an empirical study by recruiting 73 university students on ten tasks. Each of our study participants carried out five GUI reuse tasks with pen and paper. A task involves one GUI image and four reuse options. We report in this paper how these human designers pursue the reuse directions, and the extent to which they modify the focused elements. Our results show that VLMs significantly outperform the NL-based method in making reuse recommendations, but surprisingly, the focused GUI elements are not consistently modified during reuse. For example, about 40% of our study participants who changed GUIs modified the non-focused elements. To gain additional insights into GUI reuse's creativity, we further performed an assessment with four experienced designers, asking them to



(a) query-based GUI retrieval **(b) GUI-centered reuse**

Fig. 1. **(a)** Retrieving candidate GUIs in response to the query: "Body care products" via RaWi (http://rawi-prototyping.com/ ; query run on November 6, 2024); and **(b)** Multiple potential directions for reusing a single GUI which our work focuses on.

rank the innovative aspects of human-reuse results and AI-reuse results.

This paper makes three main contributions:

- We present a novel GUI-centered reuse task and the first empirical study assessing VLMs' support of the task.
- Coupled with the reuse support, we investigate the role of specific GUI elements in human designers' ideation sketches.
- Extending VLMs' support along the generative dimension, we offer insights into creativity of human-reuse and AI-reuse results.

We share our study materials and results publicly at https://doi.org/10.7945/nkr8-zq38 to facilitate replication. The rest of the paper is organized as follows. Section II provides the background information and also presents prior work related to our research. We then describe our study design in Section III, and analyze the results in Section IV. We discuss the limitations and implications of our research in Section V, and finally conclude the paper in Section VI.

II. Background and Related Work

Software reuse is the use of existing software artifacts or knowledge to create new software [6]. Clever reuse is a primary attribute of creativity in practitioners' daily work [7]. GUIs are important artifacts conveying design ideas and useful for soliciting creative feedback [2]. While prior work helped to develop reusable GUI components [15], [16], contemporary methods focus on efficiently finding relevant GUIs from large-scale repositories such as UXArchive [17] and Rico [18]. These methods take queries in different forms, e.g., an NL keyphrase [5], a hand-drawn sketch [19], or a blueprint wireframe [20]. A query is then matched with the available GUIs in order to retrieve a ranked list of reuse candidates.

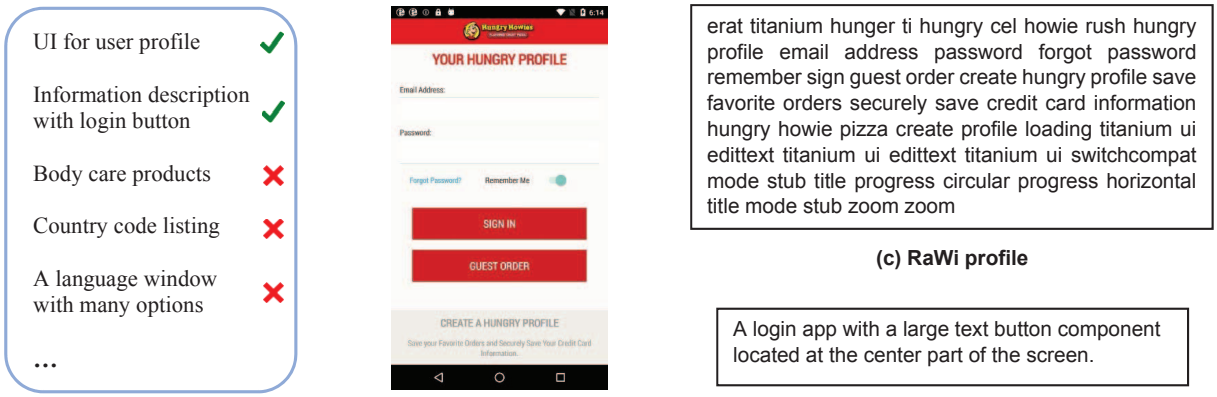
Among the GUI retrieval approaches, RaWi [5] is the most appealing to us for several reasons. First, an answer set of relevant GUIs and NL queries is accessible, which our work is built on. Second, NL is the most common way for stakeholders to express their needs and desires [21]–[23], and hence related to our work, NL is the most pervasive medium for depicting reuse directions. Third, in addition to retrieval, RaWi supports the manual editing of three GUI element types—labels, buttons, and text-inputs—which our work extends. Different from RaWi's aim of using rapid GUI prototyping for requirements elicitation, the task that we address is supporting GUI-centered reuse to catalyze ideation. A comparison is illustrated in Fig. 1.

RaWi builds a GUI's textual profile by extracting activity names, icon labels, and resource identifiers [5]. Fig. 2 shows an example where the GUI screenshot of Fig. 2-b is represented by the textual profile of Fig. 2-c. RaWi further uses a BERT-based ranking model to retrieve GUIs that are textually similar to an NL query. Fig. 2-a lists some queries along with their relevance values defined in RaWi's answer set. The answer set allows for quantifying GUI retrieval's accuracy.

In contrast to retrieving multiple candidate GUIs in response to a single query, we treat the NL keyphrases as different directions for reusing a single GUI. Nevertheless, the gap between Fig. 2-a's keyphrases and Fig. 2-b's GUI is an instance of *cognitive distance*, which according to Krueger [24], is an informal notion that relies on intuition to gauge the amount of intellectual effort that must be expended by software developers in order to take a software system from one stage of development to another. Krueger [24] argued that for a software reuse technique to be effective, it must reduce the cognitive distance.

Thus, we zoom in on a specific GUI element to help reduce the cognitive distance. XUI, an approach recently proposed by Leiva *et al.* [14], leverages deep learning to identify the most salient element in a GUI and then incorporates it into the generation of the GUI's caption. Fig. 2-d provides an XUI illustration where the "SIGN IN" button (i.e., "a large text button component located at the center part of the screen") is considered to be the most important. Such a GUI element, if recommended explicitly to developers, helps reduce the cognitive distance because the developers could focus on modifying a specific element rather than looking for modifications all over the place on the entire GUI.

Interestingly, Kolthoff *et al.* [8] recognized the importance of GUI elements in requirements validation. In particular, they



(a) NL keyphrases **(b) GUI image** **(c) RaWi profile** **(d) XUI caption**

Fig. 2. **(a)** NL keyphrases used as queries in RaWi and as reuse directions in our work; here, a tick (✓) means the keyphrase is relevant to the GUI screenshot of **(b)** and a cross (x) means irrelevance according to RaWi's answer set [5]; **(b)** A GUI screenshot studied by both RaWi [5] and XUI [14]; **(c)** The GUI's textual profile constructed automatically by RaWi; and **(d)** The GUI's caption auto-generated by XUI, highlighting the most salient element.

prompted GPT-4 to check if a user story is implemented in the GUI by any of its elements. Inspired by their work, we identify a focused element to support GUI reuse, and do so by exploiting VLMs. Contemporary VLMs are trained with large-scale image-caption data, so as to have the ability to transform images and texts into a unified embedding. Different from the work of Kolthoff *et al.* [8], we exploit VLMs to first identify a top-matched keyphrase as a reuse direction recommendation, followed by a focused element as a recommendation to make changes so as to achieve the reuse goal.

The impacts of such recommendations, to our surprise, have not been systematically investigated in the software reuse literature. Notably, Frakes and Fox [25] conducted a seminal survey on 16 factors, and showed that five factors affected reuse (e.g., type of industry) whereas the others did not despite the conventional wisdom (e.g., software engineering experience). One reason reuse recommendations were not studied might be few were available in the 29 organizations surveyed in [25]. Assessing how well VLMs recommend reuse directions and focused GUI elements is precisely the focus of our research.

III. Study Design

A. Research Questions (RQs)

Our investigations into the GUI-centered reuse are carried out in two levels: reuse direction and focused GUI element. Moreover, we assess the resulting GUI prototypes mainly from a creativity perspective, i.e., the degree to which the resulting GUI embodies clever reuse [7]. Fig. 3 contextualizes the three RQs in our overall study design.

RQ$_1$: Which methods recommend better reuse directions?

We represent reuse directions in NL keyphrases, and address RQ$_1$ by comparing the top-matched directions recommended by four methods. Two are VLMs: Qwen2-VL-7B [26] and Llama-Vision-11B [27] because they are studied recently in GUI and software engineering contexts [13], [28], [29]. One method is NL-based where we adopt RaWi's BERT ranking

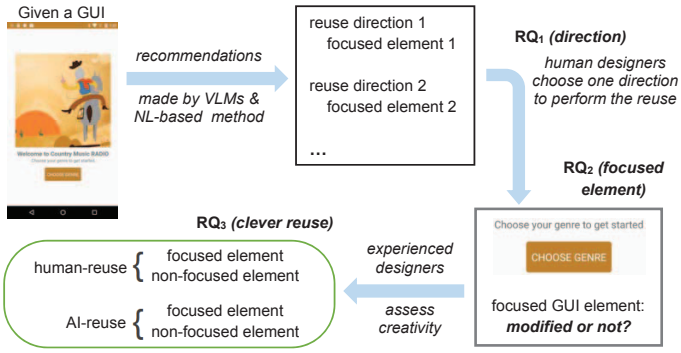


Fig. 3. Overview of our study design.

to retrieve the top-matched keyphrase and use XUI to identify the focused GUI element. To instrument a baseline, we rely on a random method to arbitrarily recommend a reuse direction as well as a focused element. Fig. 4 illustrates a reuse task with the four different recommendations. For RQ$_1$, we are particularly interested in the *directions* that human reusers choose to pursue.

RQ$_2$: Does the focused GUI element help reduce cognitive distance?

As discussed in Section II, our main rationale to recommend a focused GUI element is to help reduce a reuser's cognitive distance [24]. Therefore, we answer RQ$_2$ by analyzing how the original GUI is changed and whether the GUI modification involves the focused element or not.

RQ$_3$: How creative are the GUI reuse results?

The aim of RQ$_3$ is to gain insight of ideation. To that end, we compare human-reuse results—some with the modifications made to the focused elements and others with the modifications made to the non-focused elements—with the AI-reuse counterparts. As will be detailed further in Section III-B, we use a state-of-the-art diffusion model to generate AI-reuse results, and ask experienced designers to rank the resulting GUIs from a creativity angle.

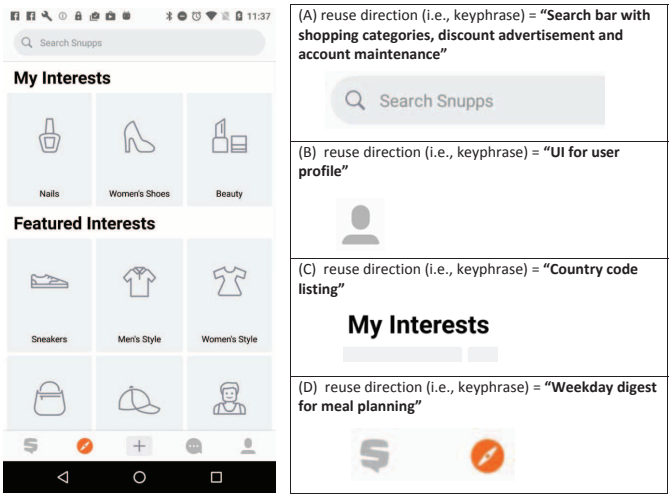


Fig. 4. Illustration of a GUI reuse task where a specific GUI image is presented to the left, and the four reuse options are listed to the right. Here, "Llama-Vision-11B", "Qwen2-VL-7B", "NL-based", and "Random" recommendations are shown in the options (A), (B), (C), and (D) respectively.

B. Datasets and Study Execution

We take advantage of the datasets shared by RaWi [5] and XUI [14]. Their intersection consists of 40 GUI images, from which we randomly select ten to structure our reuse tasks. A task illustration is provided in Fig. 4. The NL keyphrases are adopted from RaWi [5] where the queries linked to the 40 GUI images are selected to form a pool of reuse directions. In total, 27 keyphrases are chosen. We share all the datasets and study materials in our replication package at https://doi.org/10.7945/nkr8-zq38 .

We describe our study execution by referring to Fig. 3's flow. While the NL-based recommendations are taken from RaWi and XUI, we prompt Qwen2-VL-7B and Llama-Vision-11B to, "find the top-matched textual string from the following list: [27 keyphrases]" for a given GUI image. Furthermore, we prompt the two VLMs with: "Which parts of the [image] match '[top-matched keyphrase]'?" in order to obtain the focused GUI element. Together with a keyphrase and an element chosen randomly, these give rise to four different recommendations for a given GUI as shown in Fig. 4.

To explore if these recommendations make any difference, we recruited 73 university students majored in Computer Science or Computer Engineering to perform the GUI reuse tasks. As we wanted to prevent confounding variables as much as possible, we ran the study in a classroom by not allowing the participants to use their laptops, phones, or other digital devices. Rather, the reuse tasks were carried out with pen and paper by the participants working alone. This controlled setting not only helped to block external aids, but also facilitated creativity as the participants could sketch their ideas freely without being constrained by any drawing tools [2]. We observed from our two pre-study pilot trials that each GUI reuse task took about 10 minutes to complete. Thus, to alleviate the fatigue factor, we split the ten tasks randomly in two halves, and asked every participant to perform five tasks. Informed by Frakes and Fox's finding that software engineering experience had no impact on reuse [25], the participant-task assignments were done randomly in our study.

At the start of the study, our participants were notified about the study's approval by an institutional review board. They then provided written consent, giving permissions to use their responses in an anonymized way. Their demographic data were collected, and for each task, they were instructed to choose only one of the (A), (B), (C), and (D) reuse directions. Based on their own choices, they prototyped the GUIs by describing and/or drawing the revisions on paper. A couple of points are worth noting. First, we did not mention whether the revisions should be made in the recommended, focused GUI element. Second, we explicitly mentioned that the revisions could be done on the GUI, the keyphrase, or both. Once a participant completed all the five reuse tasks, we collected the answered paper for further analyses. All participants voluntarily took part in the study; there was no financial or other incentive.

The last step of Fig. 3 compares human-reuse and AI-reuse results. Following RQ_1, we determined the most frequently pursued direction by the human reusers. Conditioned by this direction, we then masked the focused GUI element as shown in Fig. 5. We fed the masked image to a pretrained diffusion model [30] twice: one time to modify the masked part and the other time to modify anything but the masked part. For example, the prompt leading to Fig. 5's top-right result's generation is, "Modify the black box in the bottom of the image toward 'Country code listing'," and that leading to the bottom-right result is, "Modify anything but the black box in the bottom of the image toward 'Country code listing'." In this example, 'Country code listing' represented the participants' most reused option. As a result, our masking was applied to the focused GUI elements associated with that option.

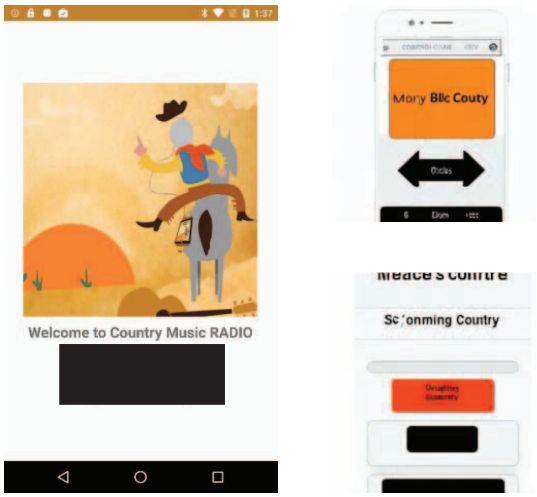


Fig. 5. A masked GUI (left), an AI-reuse result involving the focused element (top-right), and an AI-reuse result involving the non-focused element (bottom-right); AI diffusion run via [30] on November 19, 2024.

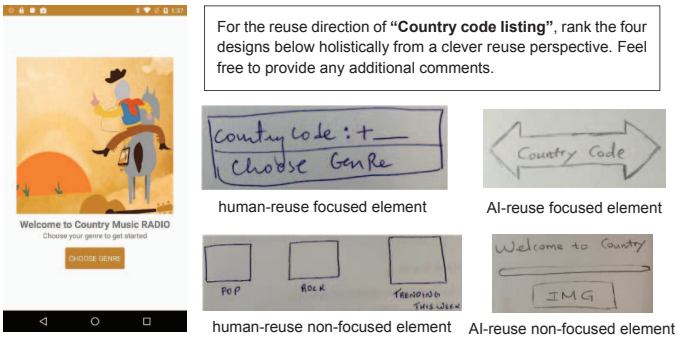


Fig. 6. Creativity assessment illustration where the four textual annotations: "human-reuse focused element", "AI-reuse focused element", etc. are *not* shown to the evaluators.

To help ensure a fair comparison of reuse results, two authors of this paper transformed the diffusion results into hand-drawn sketches. The two researchers worked jointly and sketched only the changed parts by inferring the contents from the context of the entire GUI. Our hand-drawn sketches of the two Fig. 5's results, along with two reuse results produced by our study participants, are shown in Fig. 6. We invited four experienced designers from our professional network to individually rank the creativity aspects of four different reuse results for each of the ten GUIs. These experts have an average of 5.5 years of relevant experience in user experience (UX) and graphical design. Although various facets of creativity have been discussed in the literature (e.g., combinational [31] and transformational [32] creativity), Inman *et al.* [7] pointed out recently that clever reuse was a hallmark that software practitioners valued in their daily creative work. As the illustration of Fig. 6 shows, our instruction to the design experts thus emphasized the reuse direction and the holistic consideration of clever reuse toward that direction.

C. Evaluation Metrics

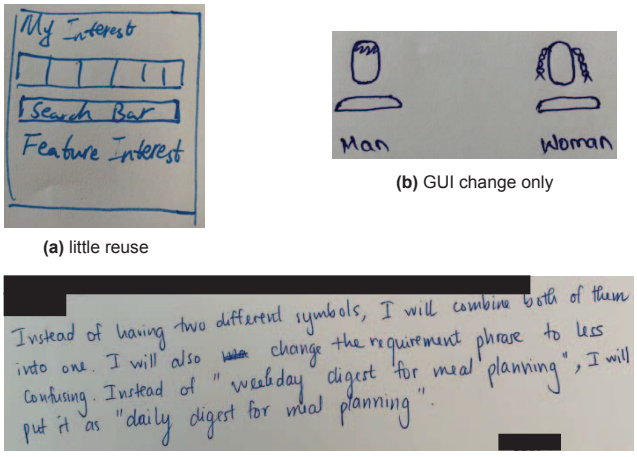
We evaluate RQ_1 by counting the number of times the participants pursued reuse in a recommended direction. To test for statistical discrimination, we apply Wilcoxon rank-sum test to examine whether the distributions of the two groups differ in a meaningful way, without assuming normality. Such a nonparametric test is less sensitive to small sample sizes, though 365 reuse-direction observations were made in our study: 73 participants $\times$ 5 tasks per participant. In addition to p-value for statistical inference, we also report effect size, r, to discern the extent to which the magnitude of the difference is practically significant.

The evaluation of the participants' actual reuse results hinges on the informal gauge of cognitive distance [24]. To that end, we manually classified the human-reuse results into three categories: little reuse, GUI change only, and GUI-keyphrase co-change. Fig. 7 provides some representative results linking to the task of Fig. 4. In Fig. 7-a, the participant chose option (A) and merely rearranged the "Search Bar" from above "My Interests" to below it. In Fig. 7-b, the participant selected option (B) and changed the user profile icon of that option to two icons: Man and Woman. In Fig. 7-c, the participant worked on option (D) and described the change of combining that option's two icons into one; meanwhile, the participant revised the keyphrase from "weekday digest for meal planning" to "daily digest for meal planning".

From the above illustrations, we consider "GUI change only" to exhibit a proper amount of cognitive distance compared to the other two, because "little reuse" does not embrace sufficient new creation and "GUI-keyphrase co-change" requires much intellectual effort to be expended. Our manual classifications of "little reuse", "GUI change only", and "GUI-keyphrase co-change" were first performed by two researchers independently, resulting in a Fleiss's kappa value of .879. We attributed such an almost perfect agreement to the disjoint natures of the three categories. The discrepancies were mainly due to some co-change cases not recognized by either researcher, and were later resolved collaboratively between the two researchers.

Compared to cognitive distance's classification, the modifications made to the focused GUI element or the non-focused ones were much easier to judge. In the rare case that a participant revised both focused and non-focused elements, we regarded it as focused element change. One researcher manually distinguished the focused element modifications from the non-focused element modifications for the human-reuse results in the "GUI change only" and "GUI-keyphrase co-change" categories. We use χ^2 test to analyze the associations of categorical data, e.g., whether the amount of reuse ("GUI change only" or "GUI-keyphrase co-change") correlates with the modifications made to the focused/non-focused elements.

For RQ_3, we not only summarize the rankings received from each of the four design experts, but also calculate Kendall's τ to measure the ordinal associations. The τ value ranges from -1 to 1 where values close to 1 indicate strong agreement between two rankings and values close to -1 indicate strong disagreement. We report pairwise τ values that help assess the significance of the experts' creativity rankings of each GUI's human-reuse and AI-reuse results.



(a) little reuse

(b) GUI change only

(c) GUI-keyphrase co-change

Fig. 7. Representative results classified into different reuse categories.

TABLE I
MOST MENTIONED INDUSTRY SECTORS, INDUSTRY EXPERIENCE, AND TASK COMPLETION TIME OF OUR STUDY PARTICIPANTS

Sector (# of mentions)	Experience (# of participants)		Completed in min	
			median	mean±sd
tech (18)	no experience	(15)	52	48.5±8.3
health (11)	< 1 year	(22)	52.5	52.3±5.1
automotive (5)	1–5 years	(34)	49.5	48.5±9.2
consulting (3)	5+ years	(2)	44.5	44.5±2.1

IV. RESULTS AND ANALYSIS

We present in Table I some demographic data of our study participants as well as the statistics related to their task completion time. The most mentioned industry sector among the participants is tech, including ed tech, big tech, tech startup, etc. Health, which covers nutrition, hospital, medical devices, and so forth comes next. Regarding the industrial and professional experience, 15 participants self-reported having no experience, 22 having less than one year of experience, 34 having one to five years of experience, and 2 having more than 5 years of experience.

To examine whether experience affected reuse, we analyze the completion time of five tasks assigned to each participant. In Table I, the median completion time and the (mean±standard deviation) completion time are reported. Excluding the statistical outliers of the two "5+ years" participants, the completion time statistics of Table I are close across different experience levels. This confirms the little impact that software engineering experience has on reuse [25]. Therefore, we make no distinctions regarding the participant-task assignments in the subsequent analyses.

Turning attention to RQ_1, we show the number of times that the participants pursued the reuse directions recommended by the random method, the NL-based method, and the two VLMs. Fig. 8 plots the numbers. Surprisingly, the worst performer was not the random method but the NL-based one. We offer some discussions on this result in Section V-B. Encouragingly, the two VLMs performed well since over 70% of the total of 365 reuse instances followed VLMs' recommendations. The Wilcoxon rank-sum test results are shown in Table II where the four methods are compared pairwise.

In Table II, the U-statistic represents the degree of difference between the two groups. The null hypothesis is that there is

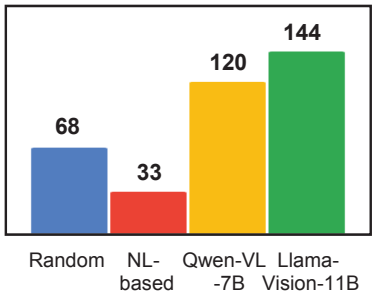


Fig. 8. Number of times that the participants pursued reuse directions recommended by different methods.

TABLE II
WILCOXON RANK-SUM TEST RESULTS ON RECOMMENDED REUSE DIRECTIONS

	Random	Qwen-VL-7B	Llama-Vision-11B
NL-based	U = 70.5 p = 0.128 r = 0.347	U = 96.0 p < 0.001 r = 0.778	U = 94.5 p < 0.001 r = 0.752
Random	– –	U = 73.5 p = 0.082 r = 0.397	U = 83.5 p = 0.012 r = 0.566
Qwen-VL-7B	– –	– –	U = 68.5 p = 0.172 r = 0.313

no significant difference between the groups. We fail to reject the null hypothesis in three comparisons due to the p values being greater than 0.05: "random versus NL-based", "Qwen-VL-7B versus random", and "Llama-Vision-11B versus Qwen-VL-7B". The other three comparisons of Table II lead to the null hypothesis to be rejected. These statistically significant results—Qwen-VL-7B against NL-based, and Llama-Vision-11B against both NL-based and random—also achieve large effect sizes ($r \geq 0.5$), indicating that the differences are highly meaningful and likely to be of practical interest in most contexts. Shedding light on the NL-based and VLMs' support, our answer to RQ_1 is:

> The VLMs significantly outperform the NL-based method in recommending GUI reuse directions, yet the two VLMs investigated in our study are comparable to each other.

Not all reuse is the same. As mentioned in Section III-C and illustrated in Fig. 7, we categorize the human-reuse results in three levels: little reuse, GUI change only, and GUI-keyphrase co-change. The contingency table between the reuse levels and the recommendation methods is shown in Table III. Of all the 365 reuse instances observed in our study, about half were done by making changes to only the GUI—showing a proper amount of cognitive distance as discussed in Section III-C. A third made little reuse whereas about a fifth expended much intellectual effort to traverse rather large cognitive distances.

We perform a χ^2 test of Table III to analyze whether the two categorical variables are independent. The resulting p<0.001

TABLE III
CONTINGENCY TABLE OF REUSE DIRECTIONS PURSUED AND TYPES OF REUSE MADE

	little reuse	GUI change only	GUI-keyphrase co-change
Random	18	26	24
NL-based	4	16	13
Qwen-VL-7B	43	62	15
Llama-Vision-11B	54	76	14
SUM (%)	119 (33%)	180 (49%)	66 (18%)

TABLE IV
CONTINGENCY TABLE OF REUSE AMOUNTS AND CHANGED GUI ELEMENTS

	Change involving	
	focused element	**non-focused element**
GUI change only	116	64
GUI-keyphrase co-change	35	31
SUM (%)	151 (61%)	95 (39%)

and Cramér's V=0.221 suggest that the two variables are statistically correlated but their associations are practically small, due to the resulting effect size of Cramér's $V \in [0.1, 0.3)$. We therefore dive into the GUI element level to address RQ_2.

In Table IV, we break down our participants' reuse results into a finer granularity, except for the ones making little reuse. For either category of "GUI change only" or "GUI-keyphrase co-change", Table IV lists the number of changes involving the focused element and those involving the non-focused element. Around 60% of the GUI changes were made to the focused elements whereas approximately 40% revised the non-focused elements. A χ^2 test of Table IV results in p=0.139 and Cramér's V=0.094, implying that no significant or practical associations exist. The above analyses suggest:

> The focused GUI elements are generally revised more often than the non-focused ones, but the differences are not significant. How to best tackle cognitive distance in the actual change part of GUI reuse thus remains an open challenge.

While we further discuss the change support in Section V-C, we now consider RQ_3 in terms of the cleverness and creativity of reuse. RQ_3 is of particular interest because we view GUI-centered reuse as an important means to ideation. For each of the ten GUIs, we thus chose the most reused direction by our study participants, and then prompted the diffusion model [30] to generate AI-reuse results. We designated the diffused results as "AI-reuse FE" and "AI-reuse NE" where FE refers to the focused element and NE stands for the non-focused element. We also selected a pair of representative human-reuse results, namely "human-reuse FE" and "human-reuse NE". These four results were ranked by experienced designers individually on a task-by-task basis. Fig. 6 illustrates the ranking assessment instruments for one of the ten GUI reuse tasks.

We tally the number of times a type of reuse results was ranked to be the most creative, the second most creative, the third most creative, and the least creative in Fig. 9. We make a few observations. First, human-reuse FE was deemed to be the most creative 17 times out of a total of 40 top-ranked votes (4 experts $\times$ 10 GUIs per expert). Although this share is less than a half (17 / 40 = 42.5%), human-reuse FE almost doubles the next closest one: AI-reuse NE at 9 times. Second, AI-reuse NE received only one least creative rank, implying its

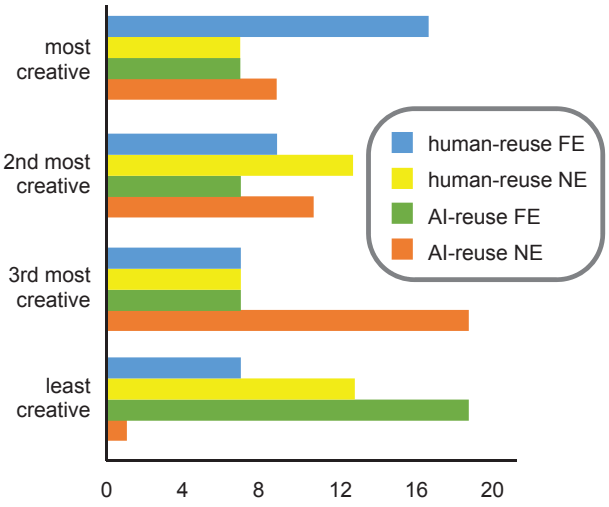


Fig. 9. Number of times a type of results is ranked within a GUI reuse task: human-reuse results involving the modifications of the focused element (human-reuse FE), human-reuse results involving the modifications of the non-focused element (human-reuse NE), AI-reuse results involving the modifications of the focused element (AI-reuse FE), and AI-reuse results involving the modifications of the non-focused element (AI-reuse FE).

reuse results were hardly the worst in the design experts' eyes. Finally, if we aggregate "human-reuse FE" and "human-reuse NE" together, and merge "AI-reuse FE" with "AI-reuse NE", then the human-reuse results were ranked as the most creative 24 times, outperforming AI's 16 times. This may suggest that, currently, compared to generative AI, humans are better at sketching new GUI designs that embrace a high degree of cleverness and creativity.

To evaluate the creativity ranking results from a quantitative angle, we computed Kendall's τ among the four experienced designers. For each GUI task, six pairwise quantifications were made. Out of the 60 comparisons, the average τ is 0.528 and the average p-value is 0.429. These results indicate that the four evaluators tended to agree with each other, and consequently, their rankings were not significantly different. Based on the above results and analyses, we conclude that:

> Compared to generative AI's diffusion-based reuse, humans are currently better at cleverly sketching new GUIs. However, the creativity rankings provided by the four experienced designers in our study were quite homogeneous.

V. DISCUSSION

This section discusses our study's limitations and implications. In particular, we identify some of the most important factors that must be considered when interpreting the results of our empirical study. We then offer insights into why the NL-based method performed the worst in recommending reuse directions, and also shed light on human-AI cooperation toward creative GUI prototyping.

A. Threats to Validity

The construct validity can be affected by the ways that we operationalize *cognitive distance* [24]. On one hand, we classify the amount of reuse into three categories: little reuse, GUI change only, and GUI-keyphrase co-change. On the other hand, we examine the reuse results to check if the actual modifications involve focused elements or non-focused elements. As argued eloquently by Krueger [24], cognitive distance is not a formal measurement that can be expressed with numbers and units. Rather, it is an informal notion that relies on intuition about the relative effort to accomplish various software development tasks. Thus, we use cognitive distance in our work as an intuitive construct to compare the effectiveness of reuse support. As suggested by our answer to RQ_2, cognitive distance remains an intriguing and important gauge for us to understand and improve the efficacy of GUI reuse.

One of the internal validity threats relates to our manually transforming AI diffusion results into hand-drawn sketches. This step is clearly limited by our own hand-drawing abilities, though our intention is to present AI-reuse results in the same common ground as the human-reuse sketches. We share all the AI diffusion images in our replication package to facilitate other researchers' analyses and experimentations.

Another threat to internal validity is our sequential examination of the three RQs. The creativity assessment of RQ_3, for example, is executed by focusing only on the reuse directions mostly pursued by our study participants—answers obtained in RQ_1. While we feel that our execution configurations are logical, caution must be taken in interpreting our findings as a whole rather than separately.

Our results may not generalize to other datasets, tasks, NL-based methods, VLMs, and study participants, all of which are threats to external validity. Because our interests lie mainly in reusing GUIs for ideation, a tradeoff is made to gain in-depth understandings of how different recommendations influence human reusers' choices and actions. Nevertheless, we believe the lessons learned will be valuable for informing better and more scalable support.

B. Underwhelming Performance of NL-Based Method

A puzzling result, as shown in Fig. 8, is that the NL-based method performed the worst in making reuse direction recommendations. Of all the 365 choices made by our study participants, only 9% followed the NL-based recommendations. These recommendations were so ineffective that even the random method's results got followed more than twice as often as them (random: 68 times versus NL: 33 times).

To explore possible reasons, we present in Table V whether the top-matched NL keyphrase is relevant to a GUI. In the table, the first column shows the GUI ID from the Rico repository [18], and hence each row signifies a task in our study. A tick (✓) represents the method's top-matched keyphrase is relevant to the GUI, according to RaWi's answer set [5]. If the keyphrase and the GUI are irrelevant, then the cell is left empty in Table V. The relevance matrix is sparse and is also highly skewed toward the two VLMs. While this indicates that the NL-based method's top-match is not as accurate as the VLMs', it does not explain why the random method outperforms the NL-based one.

TABLE V
RELEVANCE OF TOP-MATCHED KEYPHRASE TO GUI ACCORDING TO RAWI'S ANSWER SET [5]

GUI ID	Random	NL-Based	Qwen-VL-7B	Llama-Vision-11B
31253				✓
39606			✓	
34535				
22082				
43431				✓
13895				
13575				
28001			✓	
70706			✓	✓
7334				✓

TABLE VI
PERCENTAGE OF PARTICIPANTS PURSUED REUSE DIRECTIONS IN SIX TASKS WHERE AT LEAST ONE TOP-MATCHED KEYPHRASE WAS RELEVANT

GUI ID	Random	NL-Based	Qwen-VL-7B	Llama-Vision-11B
31253	3.2%	19.4%	22.6%	54.8%*
39606	3.2%	12.9%	38.7%*	45.2%
43431	9.7%	22.6%	38.7%	29.0%*
28001	21.4%	9.5%	26.2%*	42.9%
70706	33.3%	2.4%	9.5%*	54.8%*
7334	21.4%	0%	28.6%	50.0%*

In Table VI, we show the participant distributions of the six tasks where at least one relevant keyphrase was present in the GUI's four reuse direction options. The * in Table VI indicates the relevant keyphrase. Consistent with the overall trends of Fig. 8, Llama-Vision-11B offered the most pursued recommendations except for one task (43431). Yet, in tasks like 39606 and 28001, the relevant keyphrase does not best support reuse. We posit several hypotheses: (1) relevance is inherently subjective, and therefore justifications must be given in addition to the labels in an answer set; (2) a recommender that is no better than the random support could distract the users and waste their time—for instance, nobody pursued NL-based recommendation in task 7334—in another word, bad recommendations, when made systematically, are worse than no recommendations; and (3) some relevant keyphrase matches the GUI so well that little room for reuse is left. The underwhelming performance of the NL-based method provokes more questions than answers, opening up new avenues for future research.

C. Human-AI Cooperation toward Creative GUI Reuse

Combining the answers to RQ_2 and RQ_3, we offer insights into the situations where human or AI reuse is viewed as creative. Fig. 10 shows four scenarios in which each reuse

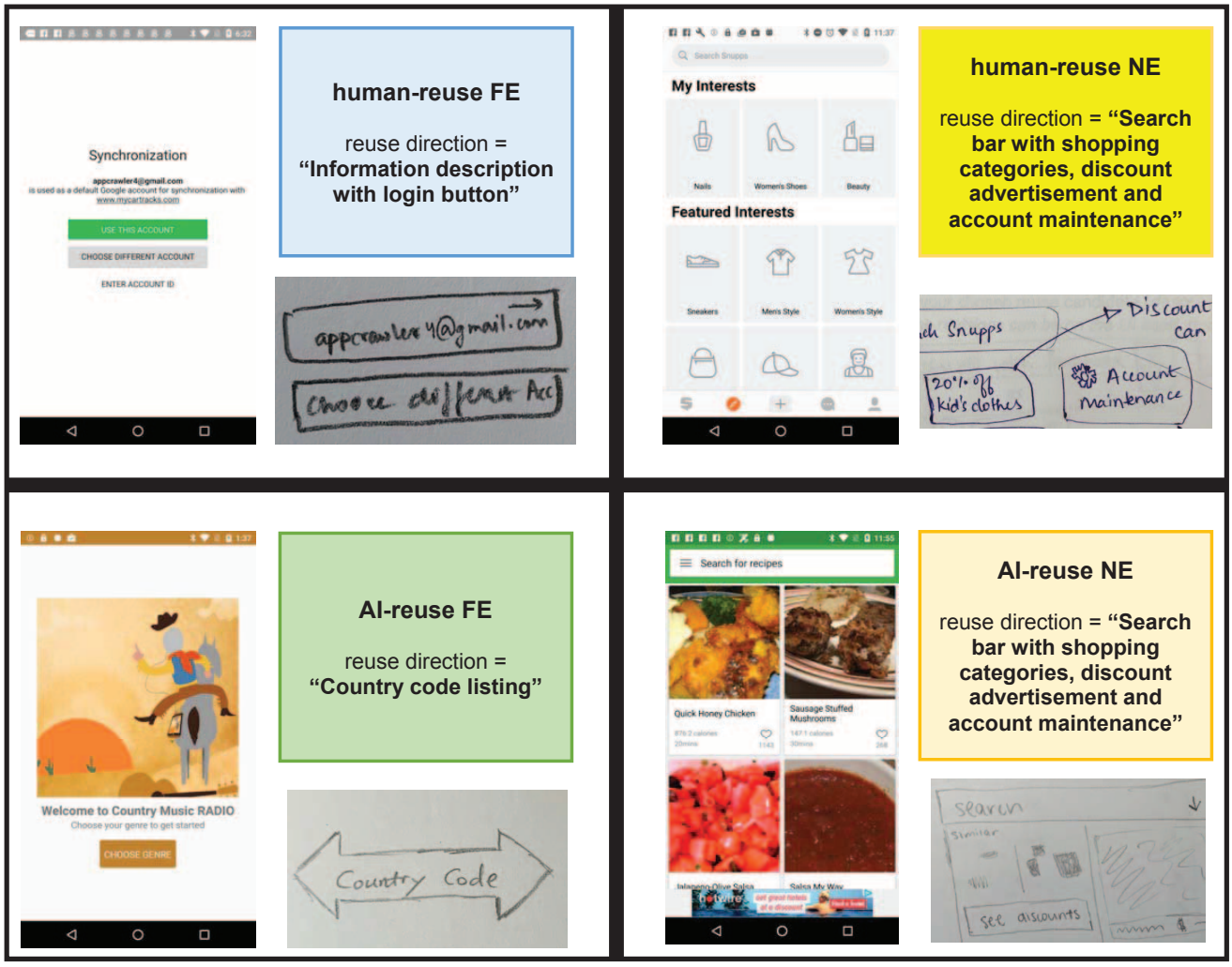


Fig. 10. Samples of the most creative reuse results obtained in answering RQ_3, clockwise: human-reuse involving focused element (top-left), human-reuse involving non-focused element (top-right), AI-reuse involving non-focused element (bottom-right), and AI-reuse involving focused element (bottom-left).

type is rated to be the most creative by at least two of the four design experts.

The top-left example shows human's direct modification of the focused "USE THIS ACCOUNT" button. Here, the reuse direction—especially the "login button"—matches well with the focused GUI element. Apart from that, the keyphrase contains "information description", allowing the human reuser to change the button label from "USE THIS ACCOUNT" to "appcrawler4@gmail.com" accompanied by an arrow (⟶) on top of the email to suggest a login action once the button is pressed. This change is clever because the email-labeled login button not only satisfies the "information description" part of the keyphrase, but also delivers an informative meaning of "THIS ACCOUNT" in the original GUI design. The new design provides useful *information*, and hence becomes more *informative*. Therefore, we argue that "human-reuse FE" can be clever when the keyphrase both matches the original GUI and has subtle meanings.

Advancing clockwise in Fig. 10, we have an example of "human-reuse NE" being ranked as the most creative by two experts. One commented, *"[this design] is functionally correct with respect to the reuse direction."* The reuse direction is given by a long, complicated keyphrase, a part of which (i.e., "search bar") matches well with the focused element (i.e., the "Search Snupps" box on top of the GUI). As a result, changing the FE is unnecessary and may even be undesirable. For a matching keyphrase with a broad scope [33], [34], manually changing NE likely leads to a correct and clever design.

Interestingly, the "AI-reuse NE" case of Fig. 10 has exactly the same reuse direction as the top-right case. The original GUI of the bottom-right of Fig. 10, however, does not match well with the keyphrase. Due to the keyphrase's broad scope, changing NE seems inevitable. The most creative result is produced by AI. One expert evaluator remarked, *"it is really cool even though it doesn't satisfy the account maintenance requirement."* A couple of points are worth noting. First, although functional correctness is important, slight incompleteness could give way to creativity, especially in the ideation stage. Second, if a keyphrase is long, broad, and complicated, implementing it in one GUI may not be ideal. AI-reuse NE can

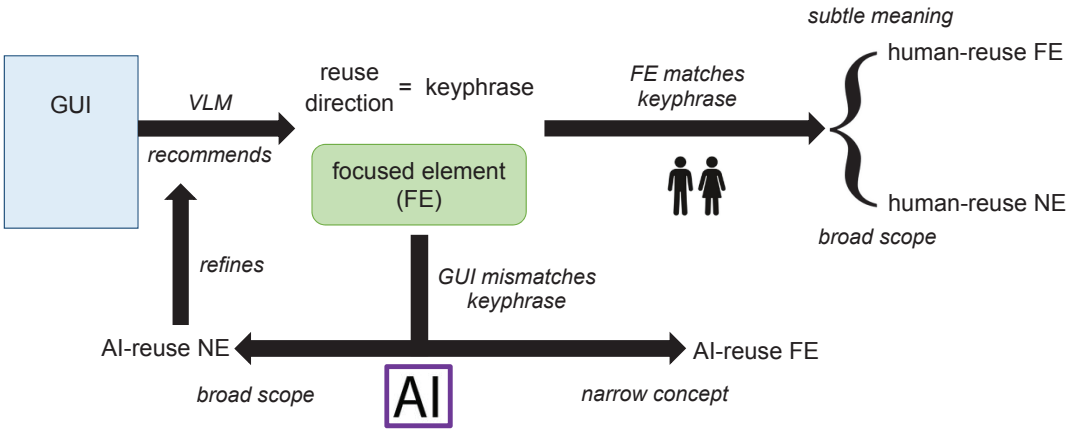


Fig. 11. A GUI ideation framework embracing human-AI cooperation.

potentially reveal this situation, enabling designers to refine, re-organize, or refactor [35]–[37] the reuse direction.

Last but not least, the bottom-left of Fig. 10 is an example showcasing AI-reuse FE's cleverness. Quoting one of the expert evaluators, *"Imagining the arrow means a scroll option with all the country codes, it has an option to view all of them."* The keyphrase, "Country code listing", does not match the original GUI which is a welcome screen allowing the users to choose genre of the Country Music RADIO. Due to the mismatch, the focused GUI elements—"Choose your genre to get started" text and the "CHOOSE GENRE" button—do not anchor the keyphrase. Despite the mismatch, the keyphrase of "Country code listing" is narrowly defined. Consequently, AI-reuse FE in such a situation results in a creative design.

While the above results are preliminary, a framework connecting humans and AI begins to take shape. We depict this framework in Fig. 11. To creatively reuse a GUI for ideation, one can exploit a VLM to recommend a top-matched reuse direction expressed in NL keyphrase. Furthermore, the VLM can be leveraged to identify an FE in the original GUI that is the most salient and relevant to the reuse direction. If the FE and the keyphrase match well, then human-reuse is more preferred: keyphrase with subtle meaning implies an FE change whereas keyphrase with broad scope likely requires the NE to be changed.

When the GUI and the VLM-recommended keyphrase do not match well, Fig. 11 shows that AI-reuse can be explored. While the keyphrase bearing a narrow concept could also narrow AI's change to the FE, a broadly scoped keyphrase could benefit from having AI revise the NE. In the latter case, the AI-reuse NE result may suggest opportunities to refine VLM's reuse direction recommendation. The framework presented in Fig. 11 is derived from our study's empirical results, and we anticipate it to evolve continuously. Nevertheless, this framework represents a unifying step toward human-AI cooperation in the GUI ideation process.

VI. Conclusion

GUI reuse helps reduce cost and effort in design ideation, but the support for the actual reuse in order to promote creativity has been under-explored. In this paper, we have exploited one of the AI advancements—VLMs—to shorten the gap. Our empirical results show that VLMs significantly outperform the NL method in making reuse recommendations, though the actual modifications of a focused GUI element and those of a non-focused element are not practically different. We have analyzed some potential reasons underlying NL method's underwhelming performance, and also derived a framework to connect human and AI in producing clever reuse results.

Our future work includes carrying out more and larger studies to lend strength to the exploratory findings reported here, notably to further evaluate and update the framework proposed in Fig. 11 and to pursue theoretical replications [38], [39]. Also of our interest is instructing the humans and the AI explicitly about creativity in their reuse tasks, e.g., through human-human collaborations [40]–[42] and prompt engineering [43]–[45]. Finally, we will continue seeking ways to reduce cognitive distance (e.g., separating reuse direction recommendation from the FE recommendation) in order to improve the cost-effectiveness of software reuse [24].

Acknowledgment

This work is partially supported by the US National Science Grant (Award: 2236953), and by the Scientific Research Fund of Zhejiang Provincial Education Department (Y202455967) and the Engineering Research Center of Mobile Health Management System, Ministry of Education.

References

[1] T. R. Silva, J.-L. Hak, and M. Winckler, "A review of milestones in the history of GUI prototyping tools," in *Workshop on User Experience and User-Centered Development Processes*, Bamberg, Germany, September 2015.

[2] F. Huang, E. Schoop, D. Ha, J. Nichols, and J. Canny, "Sketch-based creativity support tools using deep learning," *CoRR*, November 2021. [Online]. Available: https://doi.org/10.48550/arXiv.2111.09991

[3] S. Arif, Q. Khan, and S. Gahyyur, "Requirements engineering processes, tools/technologies, & methodologies," *International Journal of Reviews in Computing*, vol. 2, pp. 41–56, 2010.

[4] S. Suleri, Y. Hajimiri, and M. Jarke, "Impact of using UI design patterns on the workload of rapid prototyping of smartphone applications: An experimental study," in *Proceedings of the 22nd International Conference on Human-Computer Interaction with Mobile Devices and Services (MHCI)*, Oldenburg, Germany, October 2020, pp. 4:1–4:5. [Online]. Available: https://doi.org/10.1145/3406324.3410718

[5] K. Kolthoff, C. Bartelt, and S. P. Ponzetto, "Data-driven prototyping via natural-language-based GUI retrieval," *Automated Software Engineering*, vol. 30, no. 1, pp. 13:1–13:34, June 2023. [Online]. Available: https://doi.org/10.1007/s10515-023-00377-x

[6] W. B. Frakes and C. Terry, "Software reuse: Metrics and models," *ACM Computing Surveys*, vol. 28, no. 2, pp. 415–435, June 1996. [Online]. Available: https://doi.org/10.1145/234528.234531

[7] S. Inman, S. D'Angelo, and B. Vasilescu, "Creativity in software engineering," *IEEE Software*, vol. 41, no. 2, pp. 11–16, March/April 2024. [Online]. Available: https://doi.org/10.1109/MS.2023.3340831

[8] K. Kolthoff, F. Kretzer, C. Bartelt, A. Maedche, and S. P. Ponzetto, "Interlinking user stories and GUI prototyping: A semi-automatic LLM-based approach," in *Proceedings of the 32nd IEEE International Requirements Engineering Conference (RE)*, Reykjavik, Iceland, June 2024, pp. 380–388. [Online]. Available: https://doi.org/10.1109/RE59067.2024.00045

[9] N. Maltbie, N. Niu, M. V. Doren, and R. Johnson, "XAI tools in the public sector: A case study on predicting combined sewer overflows," in *Proceedings of the 29th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*, Athens, Greece, August 2021, pp. 1032–1044. [Online]. Available: https://doi.org/10.1145/3468264.3468547

[10] N. Niu, W. Wang, and A. Gupta, "Gray links in the use of requirements traceability," in *Proceedings of the 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE)*, Seattle, WA, USA, November 2016, pp. 384–395. [Online]. Available: https://doi.org/10.1145/2950290.2950354

[11] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," *CoRR*, February 2021. [Online]. Available: https://doi.org/10.48550/arXiv.2103.00020

[12] J. Li, D. Li, C. Xiong, and S. Hoi, "BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation," *CoRR*, February 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2201.12086

[13] J. Wei, A.-L. Courbis, T. Lambolais, B. Xu, P. L. Bernard, G. Dray, and W. Maalej, "GUing: A mobile GUI search engine using a vision-language model," *CoRR*, October 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2405.00145

[14] L. A. Leiva, A. Hota, and A. Oulasvirta, "Describing UI screenshots in natural language," *ACM Transactions on Intelligent Systems and Technology*, vol. 14, no. 1, pp. 19:1–19:28, February 2023. [Online]. Available: https://doi.org/10.1145/3564702

[15] M. Grechanik, D. S. Batory, and D. E. Perry, "Integrating and reusing GUI-driven applications," in *Proceedings of the 7th International Conference on Software Reuse (ICSR)*, Austin, TX, USA, April 2002, pp. 1–16. [Online]. Available: https://doi.org/10.1007/3-540-46020-9_1

[16] S. Stoecklin and C. Allen, "Creating a reusable GUI component," *Software - Practice and Experience*, vol. 32, no. 5, pp. 403–416, April 2002. [Online]. Available: https://doi.org/10.1002/spe.439

[17] Waldo, "UXArchive," https://uxarchive.com/, 2023, Last accessed: January 29, 2025.

[18] 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 *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST)*, Quebec City, QC, Canada, October 2017, pp. 845–854. [Online]. Available: https://doi.org/10.1145/3126594.3126651

[19] F. Huang, J. F. Canny, and J. Nichols, "Swire: Sketch-based user interface retrieval," in *Proceedings of the Conference on Human Factors in Computing Systems (CHI)*, Glasgow, UK, May 2019, pp. 104:1–104:10. [Online]. Available: https://doi.org/10.1145/3290605.3300334

[20] J. Chen, C. Chen, Z. Xing, X. Xia, L. Zhu, J. C. Grundy, and J. Wang, "Wireframe-based UI design search through image autoencoder," *ACM Transactions on Software Engineering and Methodology*, vol. 29, no. 3, pp. 19:1–19:31, July 2020. [Online]. Available: https://doi.org/10.1145/3391613

[21] J. N. och Dag, V. Gervasi, S. Brinkkemper, and B. Regnell, "A linguistic-engineering approach to large-scale requirements management," *IEEE Software*, vol. 11, no. 1, pp. 32–39, January/February 2005. [Online]. Available: https://doi.org/10.1109/MS.2005.1

[22] M. Kassab, C. J. Neill, and P. A. Laplante, "State of practice in requirements engineering: Contemporary data," *Innovations in Systems and Software Engineering*, vol. 10, no. 4, pp. 235–241, December 2014. [Online]. Available: https://doi.org/10.1007/s11334-014-0232-4

[23] N. Niu and S. Easterbrook, "Extracting and modeling product line functional requirements," in *Proceedings of the 16th IEEE International Requirements Engineering Conference (RE)*, Barcelona, Spain, September 2008, pp. 155–164. [Online]. Available: https://doi.org/10.1109/RE.2008.49

[24] C. W. Krueger, "Software reuse," *ACM Computing Surveys*, vol. 24, no. 2, pp. 131–183, June 1992. [Online]. Available: https://doi.org/10.1145/130844.130856

[25] W. B. Frakes and C. J. Fox, "Sixteen questions about software reuse," *Communications of the ACM*, vol. 38, no. 6, pp. 75–87, June 1995. [Online]. Available: https://doi.org/10.1145/203241.203260

[26] Hugging Face, "Qwen2-VL," https://huggingface.co/Qwen, 2024, Last accessed: January 29, 2025.

[27] ——, "Meta Llama," https://huggingface.co/meta-llama, 2024, Last accessed: January 29, 2025.

[28] D. Chen, Y. Huang, S. Wu, J. Tang, L. Chen, Y. Bai, Z. He, C. Wang, H. Zhou, Y. Li, T. Zhou, Y. Yu, C. Gao, Q. Zhang, Y. Gui, Z. Li, Y. Wan, P. Zhou, J. Gao, and L. Sun, "GUI-WORLD: A dataset for GUI-oriented multimodal LLM-based agents," *CoRR*, June 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2406.10819

[29] W. Chen, J. Cui, J. Hu, Y. Qin, J. Fang, Y. Zhao, C. Wang, J. Liu, G. Chen, Y. Huo, Y. Yao, Y. Lin, Z. Liu, and M. Sun, "GUICourse: From general vision language models to versatile GUI agents," *CoRR*, June 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2406.11317

[30] Hugging Face, "Diffuse The Rest," https://huggingface.co/spaces/huggingface-projects/diffuse-the-rest, 2024, Last accessed: January 29, 2025.

[31] H. Gudaparthi, N. Niu, B. Wang, T. Bhowmik, H. Liu, J. Zhang, J. Savolainen, G. Horton, S. Crowe, T. Scherz, and L. Haitz, "Prompting creative requirements via traceable and adversarial examples in deep learning," in *Proceedings of the 31st IEEE International Requirements Engineering Conference (RE)*, Hannover, Germany, September 2023, pp. 134–145. [Online]. Available: https://doi.org/10.1109/RE57278.2023.00022

[32] S. Rayasam and N. Niu, "Using i* for transformational creativity in requirements engineering," in *Proceedings of the 8th International i* Workshop (iStar)*, Ottawa, Canada, August 2015, pp. 67–72. [Online]. Available: https://ceur-ws.org/Vol-1402/paper14.pdf

[33] N. Niu and S. Easterbrook, "Analysis of early aspects in requirements goal models: A concept-driven approach," *Transactions on Aspect-Oriented Software Development*, vol. 3, pp. 40–72, 2007. [Online]. Available: https://doi.org/10.1007/978-3-540-75162-5_3

[34] N. Niu, Y. Yu, B. González-Baixauli, N. Ernst, J. Leite, and J. Mylopoulos, "Aspects across software life cycle: A goal-driven approach," *Transactions on Aspect-Oriented Software Development*, vol. 6, pp. 83–110, 2009. [Online]. Available: https://doi.org/10.1007/978-3-642-03764-1_3

[35] A. Mahmoud and N. Niu, "Supporting requirements traceability through refactoring," in *Proceedings of the 21st IEEE International Requirements Engineering Conference (RE)*, Rio de Janeiro, Brazil, July 2013, pp. 32–41. [Online]. Available: https://doi.org/10.1109/RE.2013.6636703

[36] N. Niu, T. Bhowmik, H. Liu, and Z. Niu, "Traceability-enabled refactoring for managing just-in-time requirements," in *Proceedings of the 22nd IEEE International Requirements Engineering Conference (RE)*, Karlskrona, Sweden, August 2014, pp. 133–142. [Online]. Available: https://doi.org/10.1109/RE.2014.6912255

[37] A. S. Nyamawe, H. Liu, N. Niu, Q. Umer, and Z. Niu, "Automated recommendation of software refactorings based on feature requests," in *Proceedings of the 27th IEEE International Requirements Engineering Conference (RE)*, Jeju Island, South Korea, September 2019, pp. 187–198. [Online]. Available: https://doi.org/10.1109/RE.2019.00029

[38] N. Niu, A. Koshoffer, L. Newman, C. Khatwani, C. Samarasinghe, and J. Savolainen, "Advancing repeated research in requirements engineering: A theoretical replication of viewpoint merging," in *Proceedings of the 24th IEEE International Requirements Engineering*

Conference (RE), Beijing, China, September 2016, pp. 186–195. [Online]. Available: https://doi.org/10.1109/RE.2016.46

[39] C. Khatwani, X. Jin, N. Niu, A. Koshoffer, L. Newman, and J. Savolainen, "Advancing viewpoint merging in requirements engineering: A theoretical replication and explanatory study," *Requirements Engineering*, vol. 22, no. 3, pp. 317–338, September 2017. [Online]. Available: https://doi.org/10.1007/s00766-017-0271-0

[40] X. Jin, C. Khatwani, N. Niu, M. Wagner, and J. Savolainen, "Pragmatic software reuse in bioinformatics: how can social network information help?" in *Proceedings of the 15th International Conference on Software Reues (ICSR)*, Limassol, Cyprus, June 2016, pp. 247–264. [Online]. Available: https://doi.org/10.1007/978-3-319-35122-3_17

[41] T. Bhowmik, N. Niu, W. Wang, J.-R. C. Cheng, L. Li, and X. Cao, "Optimal group size for software change tasks: A social information foraging perspective," *IEEE Transactions on Cybernetics*, vol. 46, no. 8, pp. 1784–1795, August 2016. [Online]. Available: https://doi.org/10.1109/TCYB.2015.2420316

[42] N. Niu, W. Wang, A. Gupta, M. Assarandarban, L. D. Xu, J. Savolainen, and J.-R. C. Cheng, "Requirements socio-technical graphs for managing practitioners' traceability questions," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 4, pp. 1152–1162, December 2018. [Online]. Available: https://doi.org/10.1109/TCSS.2018.2872059

[43] J. Zhang, Y. Chen, N. Niu, and C. Liu, "A preliminary evaluation of ChatGPT in requirements information retrieval," *CoRR*, July 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.12562

[44] M. Dahiya, M. Li, G. Horton, T. Scherz, and N. Niu, "Tracing feature tests to textual requirements," in *Proceedings of the 25th IEEE International Conference on Information Reuse and Integration (IRI)*, San Jose, CA, USA, August 2024, pp. 120–125. [Online]. Available: https://doi.org/10.1109/IRI62200.2024.00035

[45] M. Dahiya, R. Gill, N. Niu, H. Gudaparthi, and Z. Peng, "Leveraging ChatGPT to predict requirements testability with differential in-context learning," in *Proceedings of the 25th IEEE International Conference on Information Reuse and Integration (IRI)*, San Jose, CA, USA, August 2024, pp. 170–175. [Online]. Available: https://doi.org/10.1109/IRI62200.2024.00044