

# MineXR: Mining Personalized Extended Reality Interfaces

Hyunsung Cho  
Carnegie Mellon University  
Pittsburgh, PA, USA  
hyunsung@cs.cmu.edu

Mark Parent  
Reality Labs Research, Meta Inc.  
Toronto, ON, USA  
mrkprnt@meta.com

Hrvoje Benko  
Reality Labs Research, Meta Inc.  
Redmond, WA, USA  
benko@meta.com

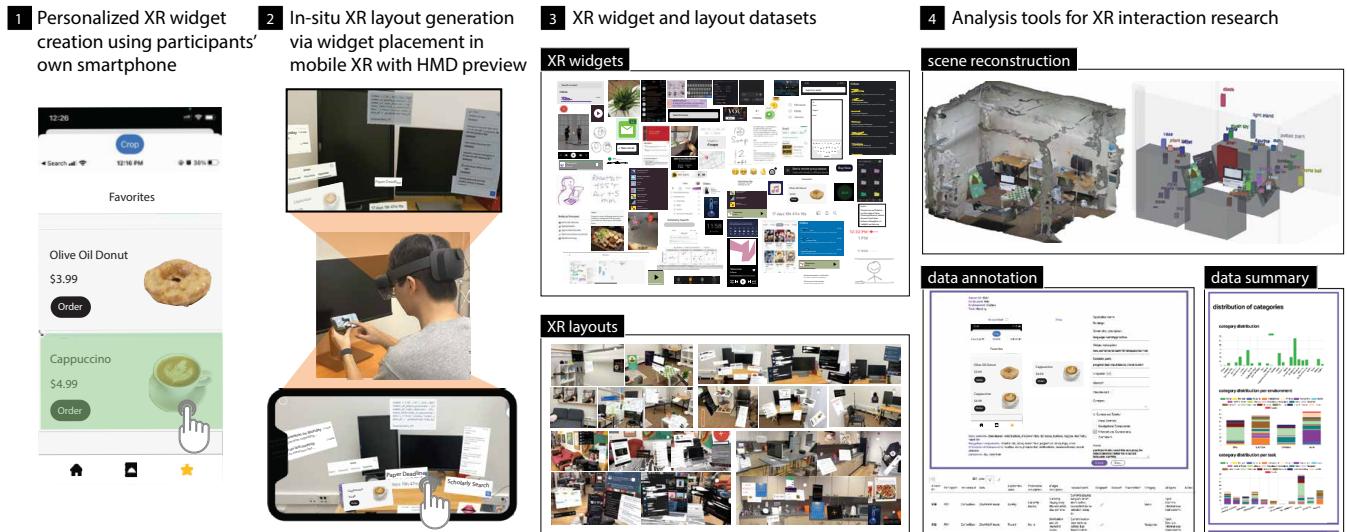
Yukang Yan  
Carnegie Mellon University  
Pittsburgh, PA, USA  
yanyukanglw@gmail.com

Missie Smith  
Reality Labs Research, Meta Inc.  
Redmond, WA, USA  
missie.smith@gmail.com

David Lindlbauer  
Carnegie Mellon University  
Pittsburgh, PA, USA  
davidlindlbauer@cmu.edu

Kashyap Todi  
Reality Labs Research, Meta Inc.  
Redmond, WA, USA  
kashyap.todi@gmail.com

Tanya Jonker  
Reality Labs Research, Meta Inc.  
Redmond, WA, USA  
tanya.jonker@meta.com



**Figure 1:** MineXR’s data mining workflow enables researchers to collect and analyze users’ personalized XR interfaces in various contexts. We release our MineXR dataset, consisting of 695 XR widgets in 109 unique XR layouts.

## ABSTRACT

Extended Reality (XR) interfaces offer engaging user experiences, but their effective design requires a nuanced understanding of user behavior and preferences. This knowledge is challenging to obtain without the widespread adoption of XR devices. We introduce MineXR, a design mining workflow and data analysis platform for collecting and analyzing personalized XR user interaction and

experience data. MineXR enables elicitation of personalized interfaces from participants of a data collection: for any particular context, participants create interface elements using application screenshots from their own smartphone, place them in the environment, and simultaneously preview the resulting XR layout on a headset. Using MineXR, we contribute a dataset of personalized XR interfaces collected from 31 participants, consisting of 695 XR widgets created from 178 unique applications. We provide insights for XR widget functionalities, categories, clusters, UI element types, and placement. Our open-source tools and data support researchers and designers in developing future XR interfaces.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

CHI ’24, May 11–16, 2024, Honolulu, HI, USA

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0330-0/24/05.

<https://doi.org/10.1145/3613904.3642394>

## CCS CONCEPTS

- Human-centered computing → Mixed / augmented reality; Systems and tools for interaction design.

## KEYWORDS

Extended Reality, Personalized UI, Datasets

### ACM Reference Format:

Hyunsung Cho, Yukang Yan, Kashyap Todi, Mark Parent, Missie Smith, Tanya Jonker, Hrvoje Benko, and David Lindlbauer. 2024. MineXR: Mining Personalized Extended Reality Interfaces. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24), May 11–16, 2024, Honolulu, HI, USA*. ACM, New York, NY, USA, 18 pages. <https://doi.org/10.1145/3613904.3642394>

## 1 INTRODUCTION

Extended Reality (XR) has the potential to transform how users interact with digital content and user interfaces. Within the context of everyday always-on XR, we envision users wearing next-generation head-mounted displays (e.g., XR glasses or contact lenses) that enable ubiquitous access to digital information, complementing or replacing their current mobile devices such as smartphones and smartwatches. This will enable users to unobtrusively and ubiquitously access information for tasks such as navigation, messaging, cooking, or web browsing.

Research in XR has explored a wide range of factors for improving user experience and performance, from advanced interaction techniques [64] and representation of content [13, 25, 38, 46, 55, 71], to view management [2, 4, 26, 35, 57] and adaptive layout systems [7, 55]. It is still unclear, however, *what* virtual content users would find most useful, or like to see and interact with, in everyday XR. Specifically, we lack an understanding of what kind of applications and digital functionalities users will require, and how contextual aspects such as activity and environment impact this choice. In addition, it is still undetermined whether an application-based interface approach that dominates current desktops and smartphones is desirable for XR or whether a more fine-grained functionality-based approach based on individual widgets is preferred by users. The former encapsulates functionality in a clear, hierarchical manner, but might not make use of flexible presentation opportunities in XR, where interfaces can be placed nearly anywhere with arbitrary amounts of information. The latter allows flexible presentation but might lead to XR layouts that are perceived as disconnected or tedious.

To address this challenge, we contribute the (1) MineXR data collection and analysis platform and (2) MineXR dataset of personalized XR layouts. The data collection tool (Figure 1) provides a novel workflow that enables researchers to collect custom XR layouts *in situ* using personal digital content. Participants of a data collection use their personal smartphones to extract content, specifically screenshots of their apps and frequently used websites in different environments and for various activities. Leveraging our platform, participants create fully personalized XR widgets and compose XR layouts freely in space using their smartphone. They concurrently view the layout through an optical see-through head-mounted display. We refer to *widgets* as individual or compound virtual UI elements containing specific functionalities of an application, which participants place in the XR environment, e.g., a play/pause button for music (individual) or a lyrics+play control panel (compound) from a music-playing app. Other examples of

widgets include real-time bus location tracker, favorite coffee order, today's schedule, etc. Researchers can review and annotate the collected data in the web-based MineXR data exploration and annotation tool and a 3D layout viewer.

Leveraging our platform, we contribute a dataset of personalized XR layouts to answer questions regarding which functionalities are desirable in what context, and how widgets are composed. The MineXR dataset consists of 109 XR layouts comprising of a total of 695 widgets, which were created by 31 participants for activities in everyday lives (e.g., focus work, meeting with colleagues, relaxing, cooking, and making coffee), located in four different real-world environments (office, living room, kitchen, coffee shop). We refer to a combination of activity and environment as a *scenario* (e.g., *working in office, relaxing in living room*). Our dataset is annotated with information about what widgets and functionalities were extracted from screenshots for XR, what categories of applications were used (e.g., productivity, entertainment), which widgets were clustered in space, and what types of UI elements were used (informational component, navigational component, input control). We connect this data with information about activity and environment to gather more detailed insights on XR layouts in different contexts. We present analysis results focused on the categories, functionalities, clusters, and UI elements of the collected data. We contribute a set of design recommendations for the creation of XR interfaces based on the analysis. For example, future XR interfaces should provide *functionality-based* XR interfaces beyond the application-based paradigm.

Our dataset complements current approaches for gathering data in XR scenarios that rely on presenting users with predetermined applications [9, 57]. As participants created individualized XR layouts using their personal digital content, our collected data better reflects their realistic preferences and requirements for layouts. The dataset enables researchers to draw insights into content selection, layout consideration, and context-aware usage of future XR applications, beyond what is presented in this paper. Furthermore, we believe it can be a valuable resource for developing novel computational approaches for XR UIs that address challenging problems such as context-aware functionality recommendation and assistive placement. We believe that our dataset is a step towards a future where everyday XR interfaces become feasible and beneficial for users, without requiring researchers to wait for the release of future headsets that can be worn in an always-on manner. Our dataset, data collection workflow, and analysis tools are available at <https://augmented-perception.org/publications/2024-minxr.html><sup>1</sup>.

## 2 RELATED WORK

MineXR enables interactive mining and analysis of personalized XR interfaces using existing UIs. Insights drawn from data collected with MineXR can be used to develop adaptive user interfaces for XR. Here, we discuss prior literature on design and adaptation of XR user interfaces as well as the use of existing UIs to inform design.

<sup>1</sup>Some figures in the paper include mockups to de-identify the original application, while retaining essential design elements and functionalities of the original screenshots and widgets.

## 2.1 Adaptive User Interfaces for XR

Existing approaches to building adaptive user interfaces for XR focus on the presentation of virtual elements, i.e., *where* and *how* to place virtual elements in various environments.

As summarized by Qian et al. [66], there exists a wide variety of approaches to anchoring XR contents in the virtual world, from tracking-based approaches [43, 69] to marker-based approaches [33, 37, 39, 67, 68, 72] that anchor XR contents using fiducial markers or specific objects. Geometry-based and semantic-based approaches employ more adaptive strategies that are based on the understanding of a target scene. Geometry-based approaches align XR contents with edges, planes, and meshes [15, 18, 19, 23, 30, 31, 64]. Semantic-based approaches leverage the semantic relationships between the physical entities in the scene and virtual elements [7, 14, 29, 45, 54, 70], or the current context of the user [20], including physical activity, e.g., walking [44] and cognitive load [55].

Another line of research investigates the interaction behaviors with different layouts of XR user interfaces. Lu et al. [56, 57] and Davari et al. [9] evaluated the Glanceable AR paradigm for accessing information in AR HMDs, in which secondary information resides at the periphery of vision, staying unobtrusive yet accessible through a quick glance.

All of these approaches consider where to place *pre-defined* virtual elements in the real-world scene and how to position them in an adaptive layout. Despite the efforts on adaptive presentation of virtual elements, it is not clear what *content* should be presented in XR interfaces. Specifically, it is unclear what kinds of virtual elements users would want to use in XR for everyday uses, and how much of the content users would like to see in an XR environment at a time. The lack of knowledge is partly due to the yet-immature XR devices for everyday uses. However, this is a chicken-and-egg problem, where understanding the potential usability can provide insights and motivation to develop the XR devices further to support these use cases. The MineXR platform aims to alleviate this problem by enabling researchers to collect data on these aspects of XR content creation.

## 2.2 Leveraging Existing UIs for Design

In website and mobile app designs, data-driven models have been introduced to help designers understand current practices and trends through example designs [10, 41, 42]. In web design, Kumar et al. introduced an example-based web design retargeting model [42] and a design mining technique [41] for a large-scale collection of web designs that is used for understanding design demographics, design curation, and supporting data-driven design tools.

In mobile app design, Deka et al. collected a repository of user flows [12] and mobile app designs [10] through a combination of crowdsourcing and automated design and interaction mining from Android apps. This large-scale mobile app design dataset supports design search, UI layout generation, UI code generation, user interaction modeling, and user perception prediction. Annotated datasets of mobile UI screens can serve as a basis for generating accessibility metadata labels for UI components [81], generating screen summaries [74], and performance testing [11]. Additionally, these datasets can be leveraged for semantic embeddings [3, 52] in mobile UI to decode into other forms of UIs such

as automatic UI generation [21, 22], smartphone shortcuts [51], custom UI mashup [40, 47], and conversational UIs [53, 73].

Unlike website and mobile app designs, such data-driven model approaches in XR for everyday uses cannot be directly adopted primarily because XR as a platform is not yet as prevalent as web or mobile apps and thus has a small user base. More importantly, XR user interfaces are not constrained inside a screen, exponentially enlarging the possible interaction space, contexts, and settings, which complicates the collection of interface designs with significant coverage of the design space. The in-situ interactive mining workflow of MineXR enables researchers to elicit contextual, personalized XR interfaces through in-situ creation of XR widgets and 3D spatial placement in diverse real-world contexts.

## 2.3 Prototyping for XR Interfaces

Most research on XR interfaces has focused on enhancing designers' prototyping and authoring process when designing specific XR experiences. These designer tools support the creation of 3D AR content or rapid prototyping of interactive AR experiences [24, 27, 28, 59, 61]. For example, Pronto [50] and other works by Leiva et al. [48, 49] enable rapid prototyping for XR interfaces using sketches and enactment. They ground their design based on the recurring issues in XR design they identified: creating and positioning 3D assets, handling the changing user position, and orchestrating multiple animations, which echo the major challenges identified by MacIntyre et al. [59]. These prototyping and authoring tools assist XR experience designers in imagining their creations in action as a process of design iteration.

Researchers have also explored in-situ and ad-hoc authoring experiences through immersive authoring [34, 75, 79]. VR has been a popular platform to support immersive authoring [60, 80] with the advantage of allowing designers to explore scenarios free from temporal and spatial limitations [36, 78]. To remedy the hardships in conducting XR prototyping in the real world, creating experiences in VR based on the virtual replica of the physical environment has been also widely adopted [17, 65, 76]. For example, Qian et al. proposed an integrated workflow, ScalAR [66], which provides a VR authoring environment where a designer can create semantic-level associations of the XR contents in each virtual scene synthesized from real-world sample scenes. Its model learns the semantic associations from the designer's demonstrations and adaptively renders AR content in different target environments.

These works have a different goal of streamlining the prototyping and authoring process of XR interface creation through VR with a focus on *where* and *how* a given XR content would be presented. Our work, on the other hand, focuses on *what content* will be used in XR and *in which format* the content will be presented. We achieve this through a bottom-up approach to take advantage of the rich array of personalized interfaces in mobile applications and websites.

## 3 MINEXR

We present MineXR, a novel workflow that enables researchers to collect, or ‘mine’, rich data on personalized XR layouts for gaining insights into future usage of XR and for adaptive XR systems.

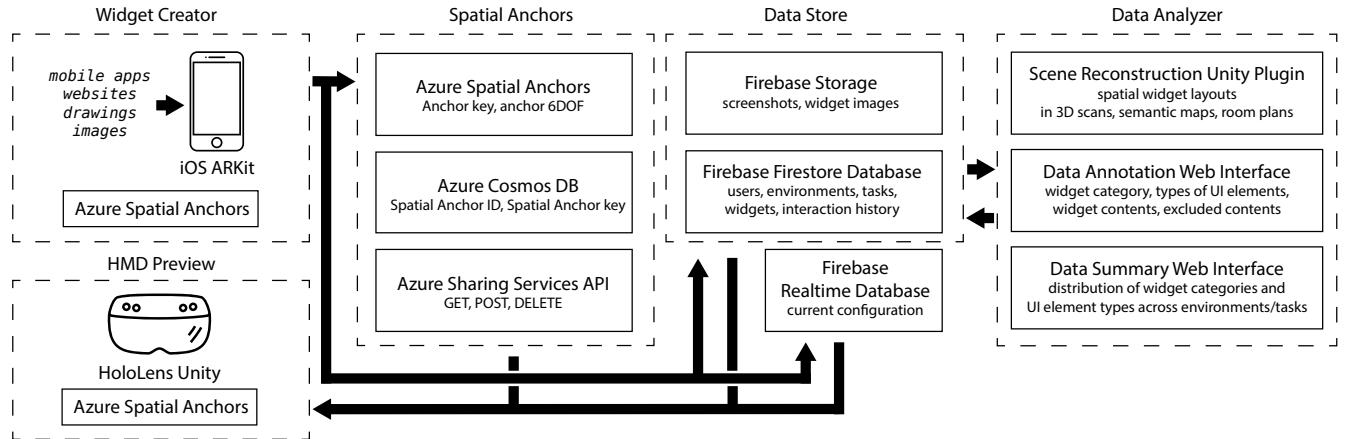


Figure 2: Architecture of the MineXR data collection and analysis platform.

### 3.1 Design Principles

We took inspiration from XR authoring tools [62], adaptive interfaces [5, 7], cross-device interaction (cf. Brady et al. [6]), and prototyping toolkits [32, 63]. Our platform is designed to enable researchers to collect rich data for future everyday XR experiences. This, in turn, means that we aim to provide a fast, flexible, and well-suited experience for participants who create the XR layouts. Our design goals are targeted toward participants of a data collection.

*Personalized Prototyping.* Using predefined widgets introduces designer bias and can limit our understanding of what content and UIs may be desirable in the future. To circumvent this issue, we use a *bottom-up approach* by taking advantage of the rich app ecosystem available on smartphones, inspired by work on designing personalized websites [42]. Specifically, participants create their personalized XR UIs from apps and content available within their own device.

*In-situ Prototyping.* Typically, prototyping is restricted to lab settings, which limits participants' abilities to contextualize and understand their varying needs. We enable in-situ prototyping, where participants are immersed in diverse real-world scenarios and environments, to enable rich and individualized data collection and understanding. We develop an untethered setup, with a smartphone and HMD, connected via cloud infrastructure, and use spatial anchors, synchronized between the two devices, so that participants can create XR layouts in near-arbitrary uninstrumented locations.

*Decomposing Apps to Functionalities.* We believe that personalized experiences require a fine-grained selection of *functionalities* rather than just the selection of apps, similar to work on personalized UI mashup [40, 47]. We enable flexible composition of widgets into XR layouts through cropping and free-form mockups.

*Zero Programming.* We aim to enable participants with no prior experience in programming or interface authoring to create personalized XR layouts. There exists a rich set of tools for content creation and layout adaptation in XR. To the best of our knowledge, however, no previous approach has enabled participants to create

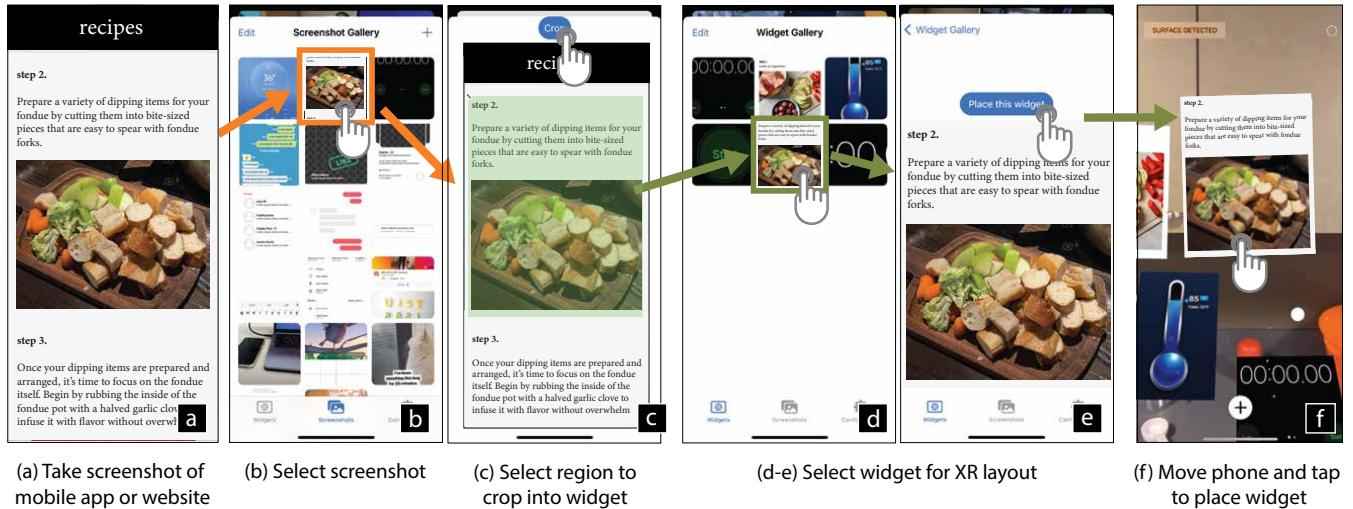
personalized interfaces based on their own existing digital content (i.e., apps). We thus adopt well-known paradigms of smartphone interactions (e.g., built-in functionalities for screenshots or image annotation) and use advanced XR interactions only for display purposes. This simplifies the creation of XR layouts drastically.

### 3.2 Overview of Data Collection

We explain the workflow, assuming that researchers have set up the MineXR data collection platform, and guided participants to a real-world environment or lab. In a given environment and task, participants first take screenshots of mobile applications and websites, create sketched mockups, or choose existing media from their phone (Figure 2. Widget Creator). Using this media, participants create *widgets* for XR usage. Widgets are the individual or compound virtual UI elements containing specific functionalities of an application, which participants place in the XR environment. For example, in Figure 3, the cropped step-by-step recipe element is a widget, which serves the functionality of showing a step-by-step recipe.

To create an XR layout, participants place each widget in the current environment by moving a smartphone running the MineXR application to the desired physical location and tapping on the screen. MineXR continuously shows a preview of the composed XR layout in the head-mounted display (HMD) (Figure 2. HMD Preview). This facilitates the process of envisioning the actual usage of this XR layout.

MineXR consists of five integrated components: widget creator, widget placement, cloud storage, HMD preview, and data analyzer. The widget creator, widget placement, and the HMD preview provide participants with the ability to create and view XR layouts, enabled by the MineXR cloud storage component (Figure 2. Spatial Anchors and Data Store). The data analyzer enables researchers to work with the collected data (Figure 2. Data Analyzer). Before or after data collection, researchers can create virtual replica of the data collection environments using a 3D scan and a semantic 3D map via commercial software for later use in data analysis.



**Figure 3: Widget creation and placement workflow.** MineXR provides a smartphone-based widget creation and XR placement workflow. A participant can create new widgets from screenshots of mobile apps or websites. Then, they can create XR widget layouts by moving the phone to the desired widget location and tapping on the screen to place.

### 3.3 Widget Creation

The widget creator (Figure 2) enables participants to specify different widgets for everyday XR usage based on their own mobile apps, websites, photos, and drawings. It captures *what* content and interfaces participants might want to typically use in XR by collecting visual representations of desired functionalities in a given context.

Figure 3 summarizes the workflow for creating a personalized XR layout from participants' perspective. The creator functions in three steps. The participant first takes screenshots of mobile apps and websites or drafts a new mockup drawing using the iOS screenshot and markup features (Figure 3a). They can redact any sensitive or private information from screenshots using the in-built markup functionality. Our custom iOS app, written in Swift, enables participants to import screenshots or mockup drawings of their choice (Figure 3b). Additionally, MineXR provides placeholders for different apps (e.g., email, messenger, dating) that can be used instead of participants' actual app.

Next, the app provides a cropping functionality where participants can crop different parts of the imported screenshots into widgets (Figure 3c). The cropped widgets are stored as separate files and displayed in the widget gallery (Figure 3d). All screenshots and widgets are uploaded to the MineXR cloud storage (Figure 2. Data Store) along with their metadata, including information on the hierarchical relationship between screenshots and widgets.

### 3.4 In-situ Widget Placement

Once the participant selects a widget to include in their XR layout (Figure 3d-e), they place the widget as an anchor in the XR scene through a mobile view implemented with iOS ARKit. Participants specify the 3D position of the widget in the scene by using the phone as a proxy of the widget. When the participant brings the phone to the desired location for the widget and taps on the screen, the app creates a spatial anchor at the phone's position in

the world coordinate, tracked by ARKit (Figure 3f). The position of the last placed widget can be adjusted multiple times following the same procedure. As an alternative, the widget placement component provides a raycasting selection method, enabling participants to select and re-place previously positioned widgets.

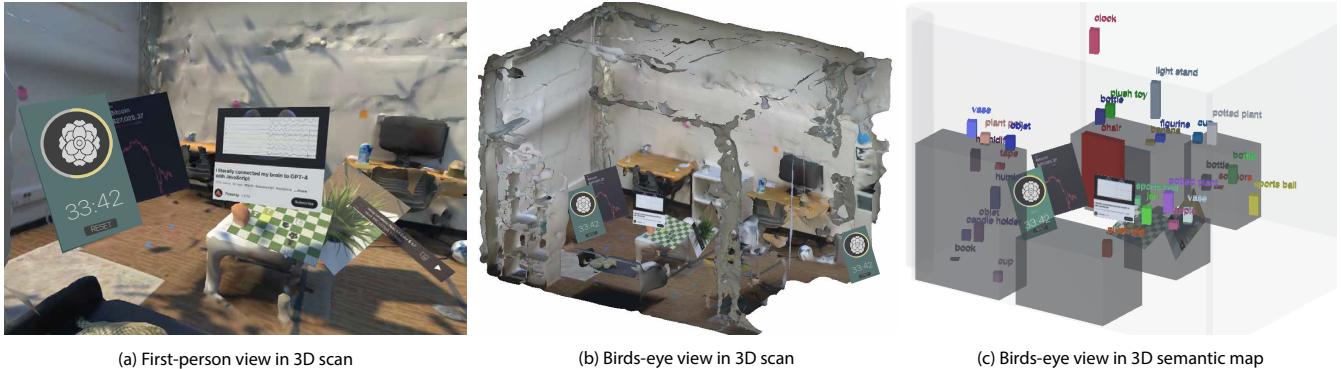
### 3.5 Cloud Data Storage

MineXR's server consists of Azure Spatial Anchors (ASA) manager (Figure 2. Spatial Anchors) and Firebase data store (Figure 2. Data Store). Once the local anchor is successfully created by placing an XR widget, the iOS app creates a cloud anchor using ASA SDK<sup>2</sup>. MineXR stores the position and orientation of the created anchors. The information is retrieved using a client-side SDK for widget placement on the phone, and for the XR preview shown through the HMD. On the server side, MineXR stores the anchor including metadata via ASA services that can be accessed via Azure Cosmos DB and Azure Sharing Services.

MineXR employs another data store using Google Firebase<sup>3</sup> Storage and Firestore Database for screenshots, widget images and other widget-related information. Each widget instance stores information about its participant, environment, task, container screenshot, interaction history (add/update), and anchor ID. This anchor ID is used to query the anchor's position and rotation. The participant, environment, and task information are entered as text input in the iOS app by the experimenter at the beginning of data collection in every scenario. A scenario refers to an environment-task pair. For example, working in the office is a unique scenario from working in a coffee shop (different environment) or having a meeting in the office (different task).

<sup>2</sup><https://azure.microsoft.com/en-us/products/spatial-anchors>

<sup>3</sup><https://firebase.google.com/>



**Figure 4: Example screenshots taken from MineXR’s interactive scene reconstruction Unity plugin, including a participant-generated layout. Researchers can reconstruct the spatial layout of each scenario with an optional overlay of the environment’s 3D scan or semantic map. The 3D scan (a,b) in this example was generated using Polycam, the semantic map (c) using Apple RoomPlan.**

### 3.6 Head-mounted Display (HMD) Preview

MineXR enables cross-platform sharing and provides participants with an XR preview through an HMD (Figure 2. HMD Preview). The HMD application was implemented in Unity, also using the ASA SDK, and deployed to a Microsoft HoloLens 2 headset. The Unity application continuously queries our service for widget updates. When it detects a new widget, it creates a new object at the anchor position and rotation (obtained from ASA) and applies the texture retrieved from the widget image in Firebase Storage. In the current implementation, the environment and task information are input as metadata in the iOS app. This real-time synchronization approach, however, considers potential extensions in future implementations, e.g., automatically detecting a participant’s dynamically changing environment and task through human activity recognition based on continuous inertial measurement unit (IMU) and audio sensing.

### 3.7 Data Analyzer

The data analyzer (Figure 2. Data Analyzer) supports researchers by providing an interface to view and annotate the collected data. It offers a 3D spatial layout and scene reconstruction tool in a Unity plugin (Figure 4) and a web-based data annotation and analysis interface (Figure 5).

MineXR collects the following raw information, which is available in the dataset:

- Real-time participant position and rotation in world coordinates
- World coordinates of widgets
- Layout placements of widgets per scenario
- Screenshots and cropped widget images along with a mapping between screenshot and created widget
- Interaction history (add/update) per widget

On top of these raw data, researchers can provide semantic annotations of the cropped widgets for richer analysis using the data analyzer, for examples included in our dataset:

- App names
- Screenshot descriptions

- Widget descriptions and functionalities
- Widget categories
- UI components in the widget
- Widget clusters

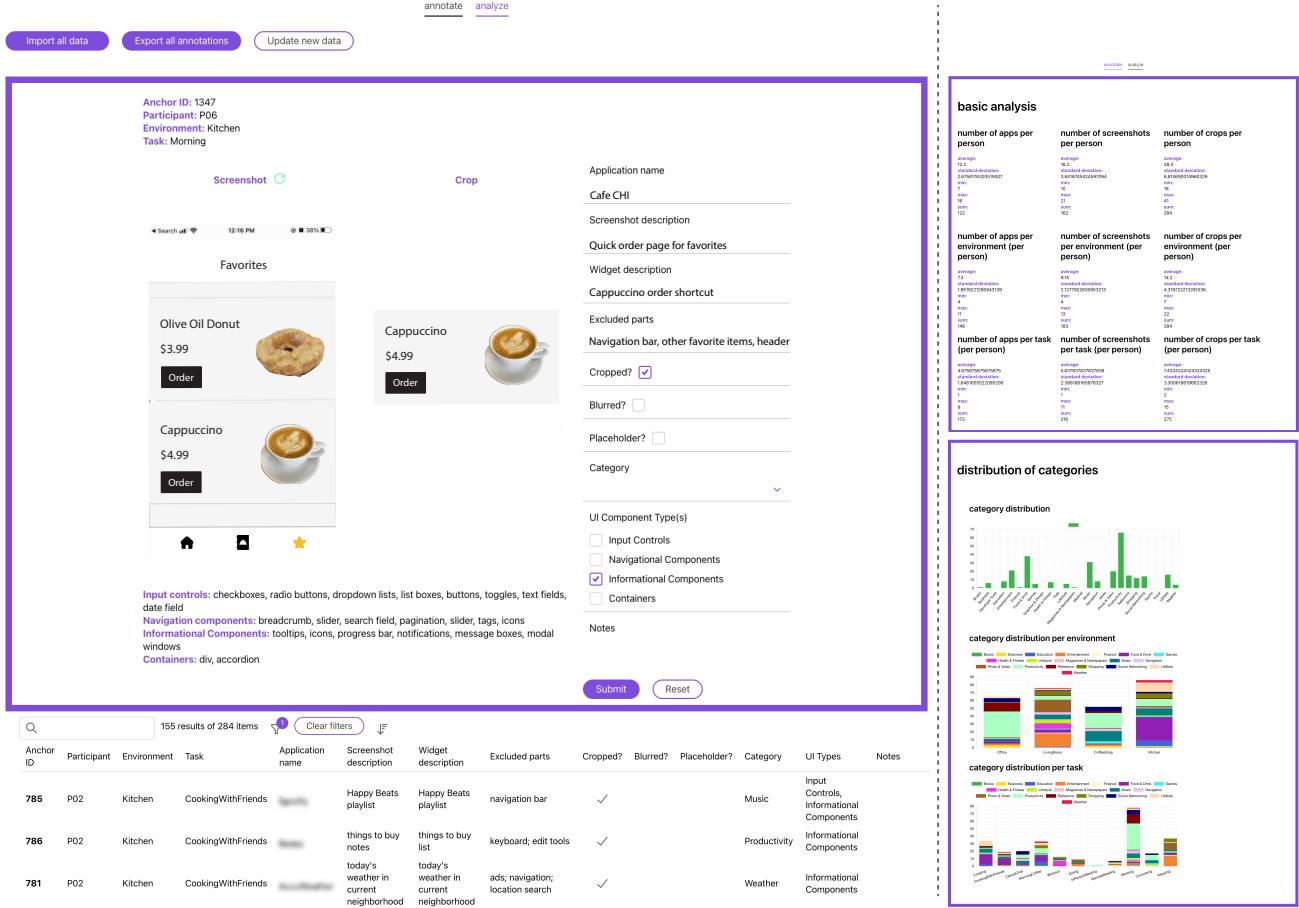
**3.7.1 Scene Reconstruction.** MineXR includes a Unity plugin that enables researchers to reconstruct the participant-generated spatial layout (Figure 4). The plugin takes the participant ID and environment as inputs, and retrieves all the widget anchors and their interaction history from our server to reconstruct the scene including all widgets. The plugin provides two options for reconstruction: showing all widgets at once and step-by-step interaction history browsing using arrow keys. By default, all widgets and their spatial layouts can be rendered within an empty 3D space in the custom Unity plugin. Researchers using MineXR may obtain a 3D scan and a semantic 3D map via commercial software, such as Polycam<sup>4</sup> and Apple RoomPlan<sup>5</sup>, respectively. If 3D scans of the environment are available, the reconstructed widgets and layouts can be viewed within this 3D space (Figure 4a-b) or a semantic 3D map (Figure 4c).

**3.7.2 Data Annotation and Summary.** MineXR contains an integrated web interface for data annotation and analysis, implemented in React JS, shown in Figure 5. Researchers can load all widget data stored in the server, e.g., screenshots, widget images, environment, and scenario information. Researchers annotate the data using a list of form fields for each attribute, e.g., text descriptions of the screenshot and cropped image, types of UI elements, and category of the widget from a list of 27 possible categories of the Apple App Store<sup>6</sup>, such as Business, Entertainment, or Finance. This step could be further automated in the future, e.g., by leveraging information from the RICO dataset [10]. MineXR includes an autocomplete suggestion feature to expedite the annotation process of repetitive contents. Researchers can browse through the annotated data using an interactive data table that supports data search, filtering, and sorting. MineXR also provides an analysis dashboard showing a

<sup>4</sup><https://poly.cam/>

<sup>5</sup><https://developer.apple.com/augmented-reality/roomplan/>

<sup>6</sup><https://developer.apple.com/app-store/categories/>, retrieved March 16 2023



**Figure 5: Web-based data annotation and summary interface.** The annotation interface (*left*) loads each instance data of widget placement and displays the corresponding anchor ID, participant ID, environment, task, screenshot image, and widget image of the widget. The interface provides a form interface including fields for data annotation, e.g., application name, screenshot description, widget description, excluded parts, type of UI element(s), application category, etc. Below the purple annotation box is an interactive data table where researchers can browse through the annotations by search, filters, and sorts. The summary interface (*right*) provides basic statistical summary and visualizations of the annotated data.

real-time overview of the annotated data. The information includes the average, minimum, and maximum number of screenshots created per participant for each environment or task, and visualizes the distribution of categories and types of UI elements too. Finally, researchers can annotate groups and clusters of widgets in combination with the scene reconstruction Unity plugin. This provides valuable insights into potential combinations of functionalities for a given context, especially when viewed in the 3D scan, or in the environment where participants created the layout by leveraging our preview functionality. We use our tool for the analysis of our dataset.

## 4 MINEXR DATA COLLECTION

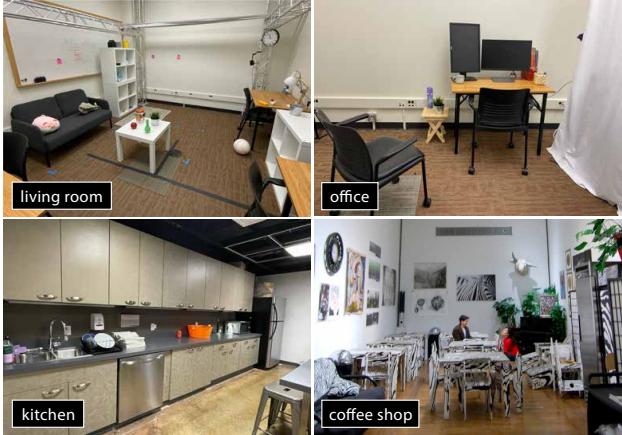
MineXR enables researchers to collect data on what contents participants would place in an XR environment, and gather insights about layout, grouping, and desired functionalities. In the following,

we report on the design of the data collection. Our method and later analysis serve as a blueprint for future research using MineXR.

### 4.1 Scenarios and Setup

We collected data in four separate environments. In each environment, participants created XR layouts for two separate tasks (e.g., working, chatting with friends). A *scenario* is a combination of environment and task (e.g., working in kitchen, relaxing in coffee shop). Each participant was asked to create a total of four layouts for different scenarios, i.e., two environments  $\times$  two tasks. The protocol was approved by the local university institutional review board.

**4.1.1 Environments.** We collected data in an office and a living room, both set up in an experimental room. Additionally, we collected data in a kitchen and a coffee shop. Figure 6 shows all environments in our data collection.



**Figure 6: The four environments in which we collected data. Each participant created XR layouts in two of the four environments.**

**4.1.2 Scenarios.** In each environment, the experimenter inquired participants about different activities they perform on a daily basis. Participants were presented with two scenarios per environment based on their personal routines for which they should create XR layouts. Those scenarios covered typical interactions, such as relaxing and workout in the living room; focused work and discussion in the office; and making coffee and cooking with friends in the kitchen. They created one XR layout for each scenario.

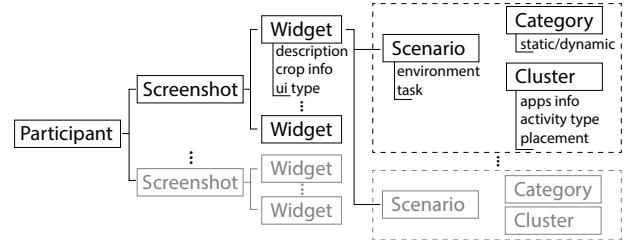
**4.1.3 MineXR Data Collection Tasks.** For interacting with MineXR, participants completed two different tasks, specifically (1) creating screenshots and (2) generating XR layouts.

*MineXR Task 1: Generating app data through screenshots.* Participants collected screenshots of applications using MineXR. The experimenter instructed them to “take screenshots of applications that you would find useful for XR within the current environment.” Previously taken screenshots (e.g., in other environments) were stored and could be reused. Participants could always update the screenshots if they missed an app or functionality. They were able to redact personal information, or utilize any of the placeholder applications.

*MineXR Task 2: Generating XR widgets and layouts.* For each scenario, participants were asked to create XR widgets with the screenshots they had taken with MineXR. Participants were instructed that they could “crop the parts of the app screenshots that they would find useful in XR for the given environment and scenario.” They placed the widgets to create an XR layout tailored for the scenario.

## 4.2 Procedure

After completing the consent form and demographic questionnaire, participants were guided to the first environment. The selection and order of the environments was counterbalanced. Participants then received a brief training for iOS study app and the Microsoft HoloLens 2, and the overall system. After the training, participants were presented with the first scenario, collected screenshots, and



**Figure 7: Reference of data and connections that can be found in the MineXR dataset. As indicated by the gray text, participants take multiple screenshots. Participants create multiple widgets by cropping screenshots and placing them. Widgets can be used in multiple scenarios and clusters.**

created the XR layout. Participants were encouraged to think aloud during the process. After participants indicated completion, the experimenter introduced the second scenario, and, after completion, guided the participant to the second environment to repeat the data collection for two additional scenarios. Data collection took between 90 and 120 minutes per participant. Few participants only had completed three scenarios after 120 minutes, mostly because they created complex layouts, and engaged in extensive think-aloud procedures. For those participants, we stopped data collection and only completed three scenarios. We audio-recorded the full sessions for qualitative analysis.

**4.2.1 Participants and Apparatus.** We recruited 31 paid participants (14 male, 16 female, 1 undisclosed) from a local University through convenience sampling (23) and external participants through a recruiting platform (8). Participants were aged between 18 and 35 years ( $M = 25.9$ ,  $SD = 4.04$ ), and were students (undergrad and graduate), researchers, or business professionals. Participants had an average experience with Augmented Reality of  $M = 2.16$  ( $SD = 1.10$ ) and Virtual Reality of  $M = 2.7$  ( $SD = 0.95$ ), on a scale from 1 (None) to 5 (Expert). All participants had normal or corrected-to-normal vision based on self-reports.

The first ten participants used the custom MineXR iOS study app on their personal devices (minimum requirement iPhone 10 and iOS 16.0), installed by the experimenter. The remaining 21 participants took the screenshots on their personal phones, and then transferred them to a dedicated study phone (iPhone 14 Pro) using AirDrop. This transfer process turned out to be simpler than installing the custom iOS app on their phone, and could be easily repeated on-demand during the study if participants desired other screenshots. A Microsoft HoloLens 2 was used for delivering the see-through XR experience. The application platform was used as described in Section 3.

## 5 MINEXR DATASET

In the following, we describe the dataset and our annotations. The annotations were distributed among four annotators, and cleaned up after completion (e.g., unify app names). The dataset consists of images, raw annotation data stored in the database, and 3D scans of the environments. Additional analysis was performed in spreadsheets, as all annotations are also available as CSV files.

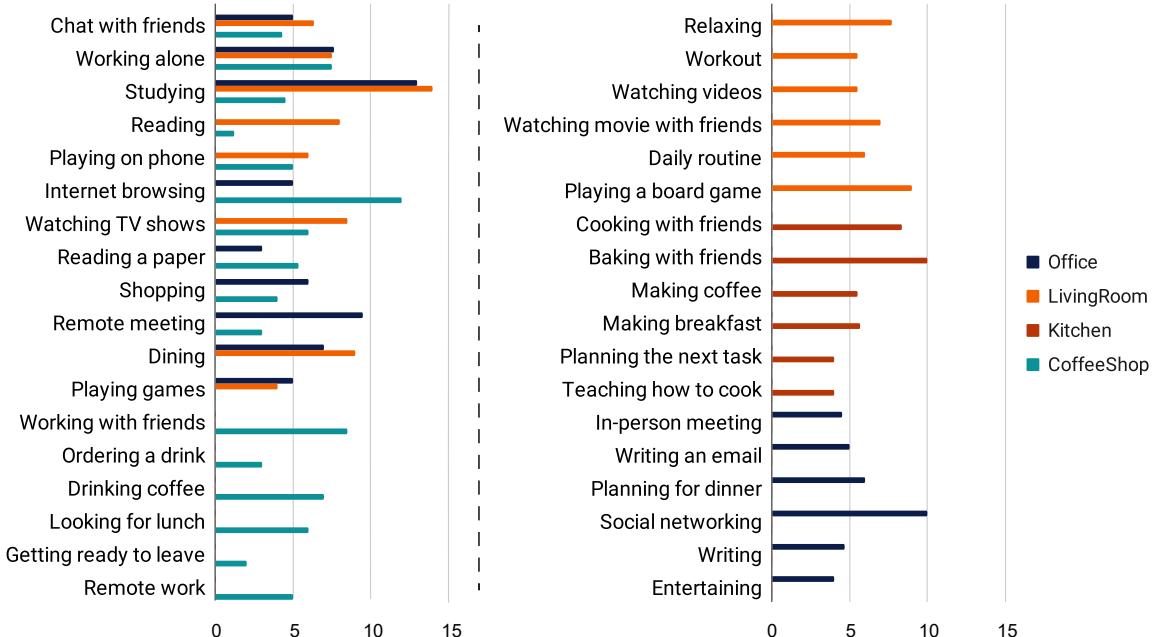


Figure 8: Average number of widgets placed in each scenario (environment × task).

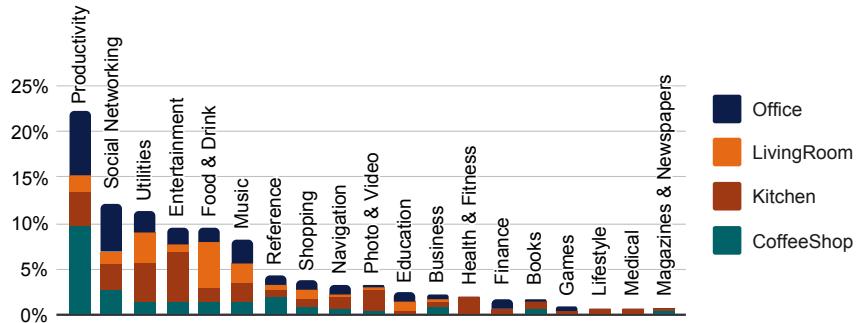


Figure 9: Distribution of the widgets categories, and across the four different environments.

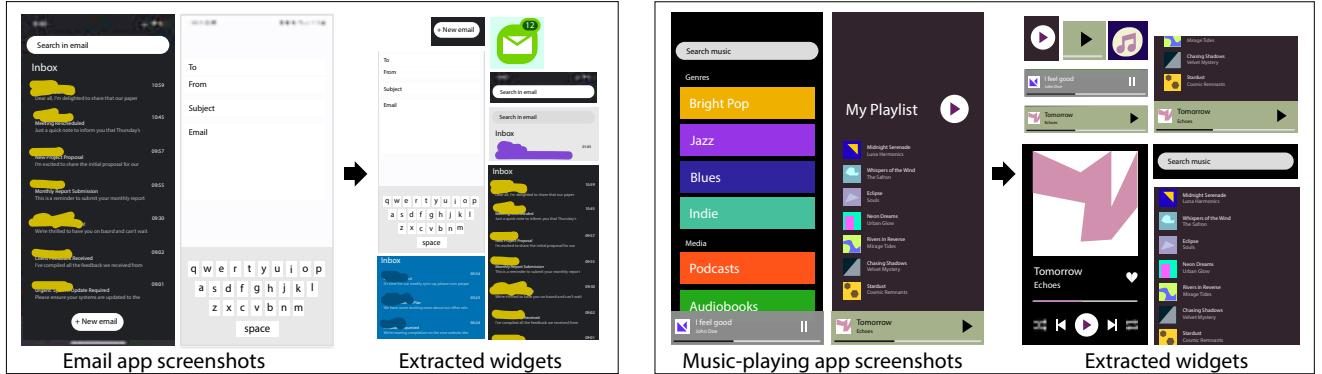
## 5.1 Base Data

A summary of the annotated data is shown in Figure 7. Participants took a total of 502 screenshots, on average 16.19 ( $SD = 5.90$ ) per participant, from a total of 178 unique applications and websites. The MineXR dataset contains 109 unique XR layouts. Participants created and placed a total of 695 widgets, on average 22.4 widgets ( $SD = 8.29$ ) per person, and on average 6.38 widgets ( $SD = 3.0$ ) per scenario. The distribution of widgets across scenarios is shown in Figure 8. Note that the scenarios were selected based on participants' actual daily tasks in the given environment.

The widgets were from 19 different categories (see Figure 9), including Productivity (22.2%, e.g., email, work messenger, notes, todo apps), Social networking (12.1%, e.g., social media, messaging apps), Utilities (11.3%, e.g., clock, Internet browsers, search

engine), Entertainment (9.6%, e.g., TV shows, movies, video streaming, video-based social media), Food & Drink (9.4%, e.g., food delivery, mobile food order apps), and Music (8.3%, e.g., music-playing, music-streaming apps).

Among the 695 widgets, 52.81% were crops of screenshots into smaller UI element(s) while 47.19% were used as the whole screenshot. Examples of an email and music-playing app from various participants are shown in Figure 10. Using the MineXR annotation tools, we annotated the collected widgets. The high number of crops highlights that participants appreciate using *individual specialized functionalities*, rather than full applications, and that this, in turn, changes their usage pattern. This is in contrast to the traditional application-based operating system in desktops and smartphones, which has been adopted in current XR systems as well, i.e., users access a digital functionality by opening an application first and navigating inside the app. Participants found this navigation step unnecessary in spatial computing and wanted direct access to specific functionalities.



**Figure 10: Examples of screenshots and the extracted widgets.**

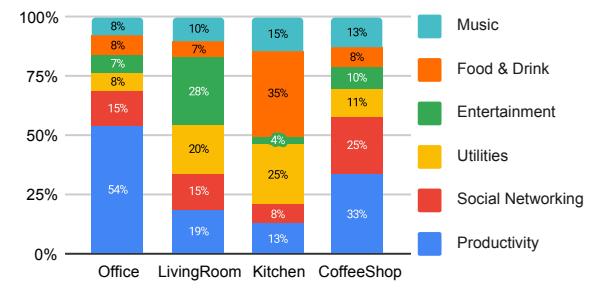
## 5.2 Categories of Widgets by Context

The dataset enables us to provide insights into the usage patterns of XR widgets. Specifically, we were interested which widgets participants want to use in XR for a given environment, and which widgets they want to use for a given scenario.

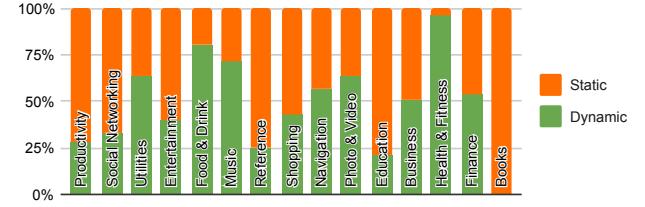
**5.2.1 Categories per Environment.** Figure 9 shows the different usages for each widget category. Figure 11 provides a more detailed view of the four environments, and how the top-used categories were employed. All environments yield different utilization in terms of primary category: office with 54% productivity, living room with 28% entertainment, kitchen with 35% food & drinks (recipes, food order apps as inspiration), and coffee shop with 33% productivity. All those align well with an intuitive understanding of which apps are useful in particular contexts.

Looking into individual environments also reveals stark differences in general usage patterns, beyond the widgets associated with the primary activity. In the office environment, the usage of widgets from other categories is relatively equal between 7% and 15%, and mostly related to peripheral activities such as social networking and music listening. In the living room, participants relied more heavily on widgets from utilities (20%), productivity (19%), and social networking (15%). Utilities were largely ambient interfaces such as clocks and weather apps. The split between productivity and social networking can be associated with the different activities that participants imagined performing in the environment. In the kitchen environment, participants similarly leveraged utilities (25%), as well as music apps (15%) as ambient displays and peripheral activities (music listening), respectively. Additionally, participants leveraged productivity apps (13%) such as notes, email, and calendars. Finally, social networking (22%) and music (13%) widgets were used frequently for the coffee shop environment. This shows that environments typically yield one primary widget category based on the current task; but that the composition of the full layout is diverse between contexts. The data from the dataset can be used to further automate app suggestions for future XR approaches, for example.

**5.2.2 Categories per Scenarios.** We categorize the activities provided by participants (e.g., studying, working) into static (22) and dynamic (30) scenarios. Static scenarios are typically single-user



**Figure 11: Relative distribution of most frequently used widgets with respect to the individual environments.**



**Figure 12: Distribution of most frequently used widgets with respect to whether an environment was static or dynamic.**

scenarios that do not involve a lot of changes in position such as working on a desk. Dynamic scenarios can involve movement and other objects, e.g., cooking, or multiple users. Figure 12 shows the distribution of the top categories between static and dynamic scenarios. Books, finance, productivity, entertainment, reference, or education are typically predominantly associated with static environments. Food & drinks are largely used in dynamic environments. Photo & video, shipping, navigation and business are equally leveraged in both types of scenarios.

## 5.3 Functionality-based Widgets

We further broke down the data into functionalities to understand what functionalities of the categories participants want to use in each environment. We annotated descriptions of all 695 widgets



**Figure 13: Examples of clusters (orange border) for different environments and tasks.**



**Figure 14: Screenshots of participants' view when creating widgets with annotations of clusters (orange border).**

using MineXR’s data annotator and grouped widgets by functionalities based on the annotated descriptions. We grouped similar functionalities within- and between-applications, e.g., ‘email inbox’ in different email apps. Appendix A.1 shows a snippet of these widget annotations—top 10 functionalities from each environment and counts of their occurrences in the environment. The app icon and UI navigation are popular functionalities acting as a shortcut to another functionality across all environments. For other ranked functionalities, different environments exhibit a notably different list of functionalities, which reflects the popular category of each environment. Our annotated descriptions of the widgets and their functionalities offer a rich understanding of *what* users want to use in XR in different contexts. We envision that these data will serve as a valuable building block for the future of computational XR, e.g., building context-aware XR widget recommendation models.

#### 5.4 Clusters of XR Widgets

Data collected through MineXR also reveals naturally occurring clusters of XR widgets. We manually annotated the widgets to identify clusters based on spatial distribution. An annotator viewed each layout, and then clustered the widgets based on proximity and semantics. Participants created a total of 214 clusters, with an average of 6.90 clusters ( $SD = 3.7$ ) per participant, and an average of 1.96 clusters ( $SD = 1.1$ ) per scenario. Figure 13 shows examples of layouts and annotated clusters based on scenarios. Figure 14 shows examples of clusters in different environments. On average, clusters consisted of 2.98 widgets ( $SD = 2.14$ ). Participants created new groups of widgets by seamlessly placing individual widgets from different applications together, as exemplified in Figure 15. They also reused the same widgets when forming clusters in various environments for different tasks. Figure 16 is an example that demonstrates the needs to re-combine functionalities from different applications considering the specific contexts in XR interactions.

Roughly 30% of widgets were placed individually, i.e., outside of any cluster. 23% of widgets were placed in pairs. Of those, roughly

half were newly composed layouts from a single application, such as a recipe app that was re-composed of two screens (grocery list, textual description), or a decomposition of the video conferencing application (controls, caller view). Finally, the remaining 47% consisted of three or more widgets, with 14% containing more than 5 widgets.

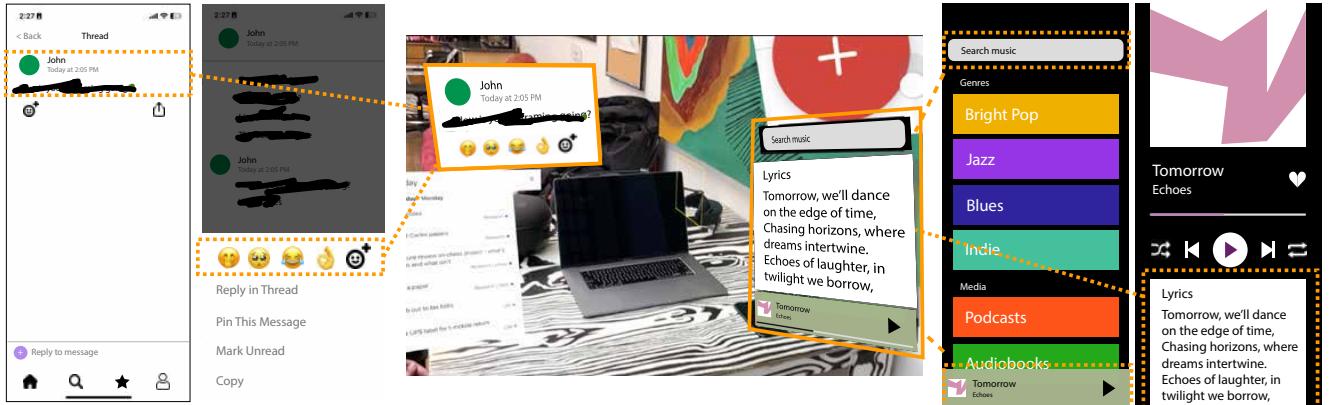
**5.4.1 Activity Types.** We annotated each cluster to determine whether it is part of the primary activity (e.g., email for work; recipe for cooking), peripheral activity (e.g., social network for work), or ambient (e.g., photos, music, clock). 39.72% were related to a primary activity, 34.58% related to a peripheral activity, and 25.7% related to an ambient display. Our data shows that participants typically compose their interfaces with one primary application in mind, and enrich their workspace with distributed peripheral information.

The distribution by cluster size is shown in Figure 17. Individual elements were oftentimes used as ambient interfaces, whereas larger clusters were oftentimes composed to resemble primary workspaces (e.g., work messenger, email, social media for task messaging). The distribution of peripheral interfaces is roughly constant across the different cluster sizes, at around 30%. This outlines a general opportunity for the design of XR interfaces: while the primary activity determines what widgets are used, peripheral interfaces are used across contexts, thus should be available to users at all times.

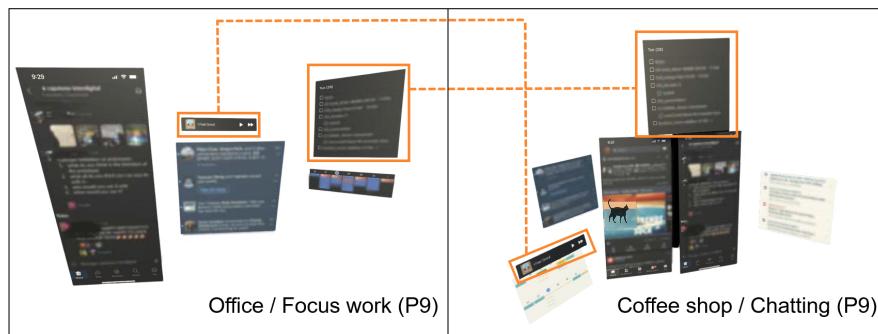
#### 5.5 UI Element Types in XR Widgets

We annotated the type(s) of UI elements contained in all widgets based on the UI element types<sup>7</sup>, namely *Input Controls*, *Navigational Contents*, and *Informational Components*. This manual annotation process could be further improved by leveraging approaches on smartphone UI understanding [10, 77], depending on the requirements of researchers. 46.74% contained *Informational Components* that display static or dynamic information, e.g., unread emails, currently playing music, food order status, or a virtual plant. *Input Controls* were present in 32.38% widgets, e.g., search box, emoji reactions to a message, or 3D chess game. Lastly, 20.88% of widgets were *Navigational Components*, and mostly shortcuts to informational components, e.g., a work messenger’s workspace icon, a button to browse food menu, or a list of shortcuts to project-related documents. Figure 18 shows the distribution of different types of UI elements in all widgets for the different environments. The figure reveals slight differences in usage patterns (e.g., more informational

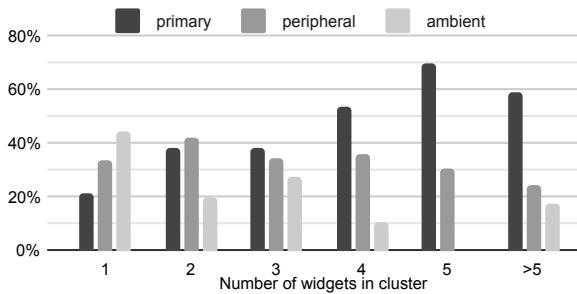
<sup>7</sup><https://www.usability.gov/how-to-and-tools/methods/user-interface-elements.html>



**Figure 15:** Examples of new widget layouts (solid orange border) composed from different UI components (dotted orange border) of the same app.



**Figure 16:** Examples of the same widgets (orange border) reused in clusters for different environments and tasks.

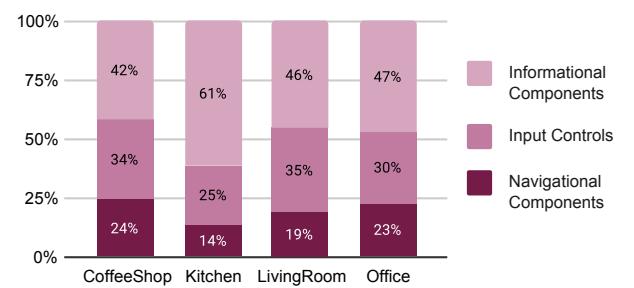


**Figure 17:** Distribution of widget clusters for each activity type, split by the size of each cluster.

components for the kitchen), but are largely comparable across environments.

## 5.6 Widget Placement

In our data collection, we found that participants have different anchoring strategies for different widgets. Examples are shown in Figure 14. For example, in work settings, many *Productivity* and *Reference* category widgets were placed with respect to the main workstation. P10 wished to save and “carry” their workspace layout



**Figure 18:** UI element types of widgets per environment.

around the laptop, regardless of the environment, i.e., office or coffee shop. This goes in line with findings from SemanticAdapt [7] and Lu et al. [58]. Some widgets were anchored with respect to fixtures in the room, for example, by the entrance of kitchen, by the window, and on the fridge space. Existing adaptive XR interfaces focus on changing the placement of XR widgets or applications (cf. Section 2.1). Adaptation algorithms are typically developed based on preliminary data collected in VR environments with predefined virtual interface elements [7, 55]. Our approach supplies richer ground truth data for these approaches with a larger set of user-defined widgets.

## 6 DISCUSSION

MineXR enables research to collect rich data on the usage of XR layouts and to draw conclusions about users' preferences, as well as challenges, and opportunities for future XR approaches. Our dataset is a first step in this direction. Besides gathering initial insights, we envision future usage for recommender algorithms and adaptive systems. MineXR enables more detailed and personalized findings, that are essential for designing general and adaptive XR interfaces.

### 6.1 Design Guidelines

In the following, we distill findings regarding user preferences for XR systems from our dataset into design guidelines with relevant anecdotal data. We hope that researchers leverage our dataset and data collection toolset to extend those findings in the future.

- **Design modular UI elements beyond apps for functionality-based XR.** Participants created their layouts by cropping and re-composing existing apps for focused digital access. For example, P5 commented that extracting specific functionalities avoids possible distractions that happen when navigating through their phone. Future XR interfaces should provide widget-based UI elements to facilitate recomposition and clustering by functionality, in addition to following an application-based paradigm.
- **Develop XR interfaces that adapt to users' space and activity** We confirm the hypothesis of context-aware adaptive XR systems [7, 55] that the usage of XR interfaces heavily depends on users' current space and activity, and whether an activity is static or dynamic. P7, for example, chose "to selectively show and hide information in everyday AR in a way that doesn't feel overwhelming and that complements an existing physical space aesthetically". XR systems should provide context-sensitive information and adapt their usage accordingly.
- **Provide continuous access to peripheral functionalities.** More than a third of clusters of XR widgets were used for peripheral tasks such as social media and communication; and were used throughout all environments and scenarios. XR layouts should provide users continuous access to functionalities that might not be relevant to users' primary tasks. Especially in work contexts, P2 liked to use the XR space as an "infinite extended monitor" (Figure 14 left) where they could add widgets without having to switch between tabs.
- **Provide seamless integration of ambient information.** Users frequently leverage widgets such as clocks or weather as ambient information displays, which should be reflected in any XR layout. This helped P7, for example, to "not have to switch between apps" and "get information from them at a glance without breaking my main task focus", which supports the Glanceable AR paradigm explored in prior art [9, 56, 57]. Our dataset informs the design for ambient, glanceable interface with grounded data on what widgets were utilized for the ambient use and how they were composed in a layout. Ambient XR interface can be further integrated into existing spaces, like decorative physical objects as proposed by Han et al. [30].
- **Accommodate both informational components and user input for navigation.** Users leveraged XR for both informational components and input (e.g., search, scrolling through emails). XR layouts should provide both functionalities, and not just focus on

serving as a flexible display surface. Furthermore, providing users with shortcuts to navigate between functionalities is desirable.

- **Cluster widgets by functionality for efficient layout composition.** Users frequently created clusters of widgets since they aided in completing a specific task (e.g., helpers for cooking), or because they were semantically similar (e.g., social media). This insight can be leveraged when assisting users in composing their layouts, for example by providing suggestions, or for algorithms that assist in the placement of virtual elements.
- **Enable personalization mechanisms for various aspects of XR layouts.** While our dataset reveals multiple generalizable patterns, the final designs of participants, unsurprisingly, were highly personal. As with most novel technology, there is no one-size-fits-all interface. Given the flexibility of XR interfaces, and the large diversity in context-dependent factors, efficient mechanisms that enable users to personalize their interfaces are key for building efficient experiences. This personalization should cover multiple aspects of an XR layout, including the selection of functionalities, their placement, and grouping.

### 6.2 Applications vs. Functionalities for XR

MineXR enabled participants of the data collection to not only place full applications but to extract widgets that contain specific functionalities through cropping. This gave them significantly more freedom to create XR layouts that are not bound to current applications. Participants used this cropping functionality for more than 50% of the placed widgets, from just cropping unimportant parts, to extracting search bars, email lists, to app icons as shortcuts; or hedonistically to place photos and videos that serve as decorative elements. We believe that future XR approaches should enable a more integrated experience, and move away from app-based layouts towards more fine-grained functionality-based approaches to provide focused digital access with less distraction, comparable to widgets in modern smartphone operating systems or task-oriented intents in voice user interfaces. Prior work on computational understanding of semantic components in mobile UIs [8, 74, 77] can be used in future works to automate the extraction of functionalities from existing UIs and creation of XR widgets. Some works showcase how these extracted functionalities can be used to generate new interfaces for conversational bots [53], virtual assistants [1], and customized mashups [40, 47]. Data and insights obtained with MineXR serve as groundings for functionality-oriented XR interface design.

### 6.3 Different Modes of Data Collection

We leverage users' personal smartphones in combination with a head-mounted display. We believe this combination enables users to utilize their personal apps and information, while providing the context of future always-on XR. Through the headset, users are able to see virtual widgets in space with natural head rotations, without having to point their smartphone at them, which we believe provides higher external validity. A natural extension to expand the scope of data collection involves enabling users to create widgets not only from smartphones but also from a wider array of personal devices, such as laptops, smartwatches, and tablets.

We decided against a smartphone-only approach, as we believe that the combination with an HMD allows users to fully leverage

their environment for creating XR layouts, and limits challenges such as arm fatigue. Additionally, we decided against performing the data collection in fully immersive VR, since it is unclear whether user behavior is equivalent in the real world. We believe MineXR leads to more realistic data, since users are in diverse real settings. We hope to compare VR data collection with our approach in the future, and investigate questions of perceptual and behavioral differences.

#### 6.4 Privacy-preserving Data Collection

We provide users with two privacy-preserving mechanisms to share data: they can blur personal information, and we provide placeholder apps (e.g., email, dating). While those mitigate some privacy concerns, the nature of the data collection (i.e., with an experimenter) poses challenges. Users might, for example, not be comfortable sharing specific apps they would like to use in an environment. Our dataset required manual cleanup to blur text as an additional step. A combination with crowd-sourcing might provide a balance in terms of privacy and data. Our platform collects the 3D position of the placed widgets and does not collect photos or videos. We extend our data with static 3D scans or abstracted 3D semantic maps. This approach enables researchers to gather insights into the placement and grouping of specific functionalities without sacrificing privacy.

#### 6.5 Limitations & Future Work

**6.5.1 Dynamic UI Layouts.** Our data collection tool currently supports only static, world-centric placement. The placed widgets are affixed to the world. The tool supports updating the widget position but does not support a dynamic layout. In our current data collection, participants created static UI layouts to complete their tasks, and they found these layouts to be sufficient, for example, to simulate a user-centric widget by placing the widget around themselves. However, the current dataset does not annotate whether the widgets were anchored to the world or the user. We regard it as interesting future work to explore the design of temporally dynamic UI layouts, where users can specify user-centric widget behaviors. These dynamic layouts can be especially helpful for tasks involving multiple steps that need to be completed sequentially. For example, when cooking in the kitchen, the user may need different functionalities at different steps (choosing from recipes, following tutorials). They should be able to adjust the layout dynamically to suit their needs. With actionable adaptation of MineXR, for example letting users define both the location and the temporal order of widgets when placing them, we believe we can facilitate the exploration on dynamic UI layouts in the future.

**6.5.2 Interactivity.** Our current platform leverages screenshots rather than functioning virtual elements. Therefore, it remains unknown how users would actually interact with these interfaces, and how the users would modify and adapt the interfaces in turn through the course of interactions. Ideally, a personalized layout would provide users with fully interactive versions to study long-term use and deployment. Although our dataset and initial analyses offer a warm start for the future of adaptive XR interfaces, it is still an open question how users would interact with these UIs and how the interactions would affect the UIs in turn.

**6.5.3 In-situ data collection using MineXR.** MineXR's in-situ approach enables collecting data for in-depth and comprehensive understandings of XR experiences, as showcased in our dataset. Our data collection took approximately two hours for participants to take screenshots and create four XR layouts in two environments under a think-aloud protocol, including introduction and debriefing. This constitutes considerable effort, especially compared to online studies [16, 46]. We believe the development of an in-the-wild, interactive version of MineXR that enables more longitudinal data collection provides an interesting direction for future work.

**6.5.4 MineXR for Computational XR.** We see a core usage of our dataset and MineXR in adaptive XR systems. We envision a wide range of usages, from providing interface recommendations to users based on previously selected functionalities, to automatically extracting widgets and functionalities, to automatic layout approaches. Our data is the first that comes close to providing ground-truth data for XR systems. For example, our data can be used to validate previous findings on XR layout placement strategies, e.g., placing near objects with semantic relevance [7], and reveal emerging placement strategies from a more in-depth analysis in relation to the environment. These newly discovered strategies can be in turn incorporated as additional factors or constraints in optimization objectives used in adaptive XR interface works (e.g., [7, 19, 55]). Furthermore, findings from previous works can be incorporated in MineXR data collection process as well. For example, widget extraction can be facilitated with automated UI structure extraction [77]. Widget placement can be facilitated by snapping to planes [64], snapping to semantically-related objects [7], or integrating ambient information with physical objects [30]. We hope that by open-sourcing our dataset as well as our data collection and analysis platform, researchers will further expand and utilize our dataset.

### 7 CONCLUSION

We present MineXR, a novel data collection platform that enables researchers to collect data on users' preferences in terms of layout and desired functionality of future XR interfaces and XR experiences. We leverage participants' personal smartphones for the selection and placement of widgets, and a head-mounted display for presenting XR content. Our platform enables in-situ data collection and provides researchers with the opportunity to create rich datasets. We contribute a dataset of 109 unique XR layouts from 31 users composed of 695 widgets, as well as an analysis that led to a set of design guidelines for XR interfaces. We believe that MineXR is a main step towards designing future XR interfaces that are personalized and beneficial for users. Our insights benefit manual design approaches as well as adaptive XR systems. MineXR tools and dataset are available at <https://augmented-perception.org/publications/2024-minexr.html>.

### ACKNOWLEDGMENTS

We thank Kaitlyn Ng and Yixuan (Sherry) Wang for their assistance with running user studies and annotating data. Funding to attend this conference was provided by the CMU GSA/Provost Conference Funding.

## REFERENCES

- [1] Deniz Arsan, Ali Zaidi, Aravind Sagar, and Ranjitha Kumar. 2021. App-Based Task Shortcuts for Virtual Assistants. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. ACM, Virtual Event USA, 1089–1099. <https://doi.org/10.1145/3472749.3474808>
- [2] Ronald Azuma and Chris Furmanski. 2003. Evaluating label placement for augmented reality view management. In *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings*. IEEE, 66–75.
- [3] Chongyang Bai, Xiaoxue Zang, Ying Xu, Srinivas Sunkara, Abhinav Rastogi, Jindong Chen, and Blaise Agüera y Arcas. 2021. UBERT: Learning Generic Multimodal Representations for UI Understanding. *arXiv*:2107.13731 (Aug 2021). <http://arxiv.org/abs/2107.13731> arXiv:2107.13731 [cs].
- [4] Blaine Bell, Steven Feiner, and Tobias Höllerer. 2001. View management for virtual and augmented reality. In *Proceedings of the 14th annual ACM symposium on User interface software and technology*. 101–110.
- [5] João Belo and Mathias N Lystb. 2022. AUIT – the Adaptive User Interfaces Toolkit for Designing XR Applications. *CoRR*, abs/2202.17.
- [6] Frederik Brudy, Christian Holz, Roman Rädle, Chi-Jui Wu, Steven Houben, Clemens Nylandsted Klokmose, and Nicolai Marquardt. 2019. Cross-device taxonomy: Survey, opportunities and challenges of interactions spanning across multiple devices. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow Scotland UK). ACM, New York, NY, USA. <https://doi.org/10.1145/3290605.3300792>
- [7] Yifei Cheng, Yukang Yan, Xin Yi, Yuanchun Shi, and David Lindlbauer. 2021. SemanticAdapt: Optimization-based Adaptation of Mixed Reality Layouts Leveraging Virtual-Physical Semantic Connections. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. ACM, Virtual Event USA, 282–297. <https://doi.org/10.1145/3472749.3474750>
- [8] Hyunsung Cho, DaEun Choi, Donghwi Kim, Wan Ju Kang, Eun Kyoung Choe, and Sung-Ju Lee. 2021. Reflect, not regret: Understanding regretful smartphone use with app feature-level analysis. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW2 (2021), 1–36.
- [9] Shakiba Davari, Feiyu Lu, and Doug A. Bowman. 2022. Validating the Benefits of Glanceable and Context-Aware Augmented Reality for Everyday Information Access Tasks. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, Christchurch, New Zealand, 436–444. <https://doi.org/10.1109/VR51125.2022.00063>
- [10] Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. 2017. Rico: A Mobile App Dataset for Building Data-Driven Design Applications. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM, Québec City QC Canada, 845–854. <https://doi.org/10.1145/3126594.3126651>
- [11] Biplab Deka, Zifeng Huang, Chad Franzen, Jeffrey Nichols, Yang Li, and Ranjitha Kumar. 2017. ZIPT: Zero-Integration Performance Testing of Mobile App Designs. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM, Québec City QC Canada, 727–736. <https://doi.org/10.1145/3126594.3126647>
- [12] Biplab Deka, Zifeng Huang, and Ranjitha Kumar. 2016. ERICA: Interaction Mining Mobile Apps. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, Tokyo Japan, 767–776. <https://doi.org/10.1145/2984511.2984581>
- [13] Stephen DiVerdi, Tobias Hollerer, and Richard Schreyer. 2004. Level of detail interfaces. In *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE, 300–301.
- [14] Zhi-Chao Dong, Wenming Wu, Zenghao Xu, Qi Sun, Guanjie Yuan, Ligang Liu, and Xiao-Ming Fu. 2021. Tailored reality: Perception-aware scene restructuring for adaptive vr navigation. *ACM Transactions on Graphics (TOG)* 40, 5 (2021), 1–15.
- [15] Ruofei Du, Eric Turner, Maksym Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Cruces, et al. 2020. DepthLab: Real-time 3D interaction with depth maps for mobile augmented reality. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 829–843.
- [16] John J Dudley, Jason T Jacques, and Per Ola Kristensson. 2021. Crowdsourcing design guidance for contextual adaptation of text content in augmented reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–14.
- [17] Barrett Ens, Fraser Anderson, Tovi Grossman, Michelle Annett, Pourang Irani, and George Fitzmaurice. 2017. Ivy: Exploring Spatially Situated Visual Programming for Authoring and Understanding Intelligent Environments. In *Proceedings of the 43rd Graphics Interface Conference* (Edmonton, Alberta, Canada) (*GI '17*). Canadian Human-Computer Communications Society, Waterloo, CAN, 156–162.
- [18] Barrett Ens, Eyal Ofek, Neil Bruce, and Pourang Irani. 2015. Spatial Constancy of Surface-Embedded Layouts across Multiple Environments. In *Proceedings of the 3rd ACM Symposium on Spatial User Interaction*. ACM, Los Angeles California USA, 65–68. <https://doi.org/10.1145/2788940.2788954>
- [19] Andreas Fender, Philipp Herholz, Marc Alexa, and Jörg Müller. 2018. Optispace: Automated placement of interactive 3d projection mapping content. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–11.
- [20] Andreas Fender, David Lindlbauer, Philipp Herholz, Marc Alexa, and Jörg Müller. 2017. Heatspace: Automatic placement of displays by empirical analysis of user behavior. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. 611–621.
- [21] Michael Fischer, Giovanni Campagna, Silei Xu, and Monica S. Lam. 2018. Brasau: automatic generation of graphical user interfaces for virtual assistants. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*. ACM, Barcelona Spain, 1–12. <https://doi.org/10.1145/3229434.3229481>
- [22] Krzysztof Gajos and Daniel S Weld. 2004. SUPPLE: automatically generating user interfaces. In *Proceedings of the 9th international conference on Intelligent user interfaces*. 93–100.
- [23] Ran Gal, Lior Shapira, Eyal Ofek, and Pushmeet Kohli. 2014. FLARE: Fast layout for augmented reality applications. In *2014 IEEE international symposium on mixed and augmented reality (ISMAR)*. IEEE, 207–212.
- [24] Danilo Gasques, Janet G. Johnson, Tommy Sharkey, and Nadir Weibel. 2019. What You Sketch Is What You Get: Quick and Easy Augmented Reality Prototyping with PintAR. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, Glasgow Scotland UK, 1–6. <https://doi.org/10.1145/3290607.3312847>
- [25] Nehla Ghouaiel, Jean-Marc Cieutat, and Jean-Pierre Jessel. 2014. Adaptive augmented reality: plasticity of augmentations. In *Proceedings of the 2014 Virtual Reality International Conference*. 1–4.
- [26] Raphael Grasset, Tobias Langlotz, Denis Kalkofen, Markus Tatzgern, and Dieter Schmalstieg. 2012. Image-driven view management for augmented reality browsers. In *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 177–186.
- [27] Sinem Guven and Steven Feiner. 2003. Authoring 3D Hypermedia for Wearable Augmented and Virtual Reality. In *Seventh IEEE International Symposium on Wearable Computers, 2003. Proceedings*. IEEE, White Plains, NY, USA, 118–126. <https://doi.org/10.1109/ISWC.2003.1241401>
- [28] Sinem Guven, Steven Feiner, and Ohan Oda. 2006. Mobile Augmented Reality Interaction Techniques for Authoring Situated Media On-Site. In *2006 IEEE/ACM International Symposium on Mixed and Augmented Reality*. IEEE, Santa Barbara, CA, USA, 235–236. <https://doi.org/10.1109/ISMAR.2006.297821>
- [29] Lei Han, Tian Zheng, Yinhe Zhu, Lan Xu, and Lu Fang. 2020. Live semantic 3d perception for immersive augmented reality. *IEEE transactions on visualization and computer graphics* 26, 5 (2020), 2012–2022.
- [30] Violet Yinuo Han, Hyunsung Cho, Kiyoshi Maeda, Alexandra Ion, and David Lindlbauer. 2023. BlendMR: A Computational Method to Create Ambient Mixed Reality Interfaces. *Proceedings of the ACM on Human-Computer Interaction* 7, ISS (2023), 217–241.
- [31] Dominik Herr, Jan Reinhardt, Guido Reina, Robert Krüger, Rafael V Ferrari, and Thomas Ertl. 2018. Immersive modular factory layout planning using augmented reality. *Procedia CIRP* 72 (2018), 1112–1117.
- [32] Jaylin Herskovitz, Yi Fei Cheng, Anhong Guo, Alanson P Sample, and Michael Nebeling. 2022. XSpace: An Augmented Reality Toolkit for Enabling Spatially-Aware Distributed Collaboration. *Proceedings of the ACM on Human-Computer Interaction* 6, ISS (2022), 277–302.
- [33] Ke Huo, Yuanzhi Cao, Sang Ho Yoon, Zhuangying Xu, Guiming Chen, and Karthik Ramani. 2018. Scenario! Spatially mapping smart things within augmented reality scenes. In *Proceedings of the 2018 CHI Conference on human factors in computing systems*. 1–13.
- [34] Ke Huo, Vinayak, and Karthik Ramani. 2017. Window-Shaping: 3D Design Ideation by Creating on, Borrowing from, and Looking at the Physical World. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction*. ACM, Yokohama Japan, 37–45. <https://doi.org/10.1145/3024969.3024995>
- [35] Brandon Huynh, Abby Wysepal, Vivian Ross, Jason Orlosky, and Tobias Höllerer. 2022. Layerable Apps: Comparing Concurrent and Exclusive Display of Augmented Reality Applications. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 857–863.
- [36] Hans-Christian Jetter, Roman Rädle, Tiare Feuchtner, Christoph Anthes, Judith Friedl, and Clemens Nylandsted Klokmose. 2020. “In VR, everything is possible!”: Sketching and Simulating Spatially-Aware Interactive Spaces in Virtual Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, Honolulu HI USA, 1–16. <https://doi.org/10.1145/3313831.3376652>
- [37] Dongsik Jo and Gerard Joungyun Kim. 2016. ARIoT: scalable augmented reality framework for interacting with Internet of Things appliances everywhere. *IEEE Transactions on Consumer Electronics* 62, 3 (2016), 334–340.
- [38] Simon Julier, Marco Lanzagorta, Yohan Baillot, Lawrence Rosenblum, Steven Feiner, Tobias Höllerer, and Sabrina Sestito. 2000. Information filtering for mobile augmented reality. In *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*. IEEE, 3–11.

- [39] Hirokazu Kato and Mark Billinghurst. 1999. Marker Tracking and HMD Calibration for a Video-Based Augmented Reality Conferencing System. In *Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*. IEEE Comput. Soc, San Francisco, CA, USA, 85–94. <https://doi.org/10.1109/IWAR.1999.803809>
- [40] Donghwi Kim, Sooyoung Park, Jhhoon Ko, Steven Y Ko, and Sung-Ju Lee. 2019. X-droid: a quick and easy android prototyping framework with a single-app illusion. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. 95–108.
- [41] Ranjitha Kumar, Arvind Satyanarayan, Cesar Torres, Maxine Lim, Salman Ahmad, Scott R. Klemmer, and Jerry O. Talton. 2013. Webzeitgeist: design mining the web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, Paris France, 3083–3092. <https://doi.org/10.1145/2470654.2466420>
- [42] Ranjitha Kumar, Jerry O. Talton, Salman Ahmad, and Scott R. Klemmer. 2011. Bricolage: example-based retargeting for web design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, Vancouver BC Canada, 2197–2206. <https://doi.org/10.1145/1978942.1979262>
- [43] Kin Chung Kwan and Hongbo Fu. 2019. Mobi3dsketch: 3d sketching in mobile ar. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–11.
- [44] Wallace S. Lages and Doug A. Bowman. 2019. Walking with adaptive augmented reality workspaces: design and usage patterns. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*. ACM, Marina del Ray California, 356–366. <https://doi.org/10.1145/3301275.3302278>
- [45] Yining Lang, Wei Liang, and Lap-Fai Yu. 2019. Virtual agent positioning driven by scene semantics in mixed reality. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 767–775.
- [46] Jaewook Lee, Fanjie Jin, Younsou Kim, and David Lindlbauer. 2022. User Preference for Navigation Instructions in Mixed Reality. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 802–811. <https://doi.org/10.1109/VR51125.2022.00102>
- [47] Sunjae Lee, Hoyoung Kim, Sijung Kim, Sangwook Lee, Hyosu Kim, Jean Young Song, Steven Y. Ko, Sangeun Oh, and Insik Shin. 2022. A-mash: providing single-app illusion for multi-app use through user-centric UI mashup. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. ACM, Sydney NSW Australia, 690–702. <https://doi.org/10.1145/3495243.3560522>
- [48] Germán Leiva, Jens Emil Grønbæk, Clemens Nylandsted Klomkose, Cuong Nguyen, Rubaiat Habib Kazi, and Paul Asente. 2021. Rapido: Prototyping Interactive AR Experiences through Programming by Demonstration. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. ACM, Virtual Event USA, 626–637. <https://doi.org/10.1145/3472749.3474774>
- [49] Germán Leiva, Nolwenn Maudet, Wendy Mackay, and Michel Beaudouin-Lafon. 2019. Enact: Reducing Designer–Developer Breakdowns When Prototyping Custom Interactions. *ACM Transactions on Computer-Human Interaction* 26, 3 (June 2019), 1–48. <https://doi.org/10.1145/3310276>
- [50] Germán Leiva, Cuong Nguyen, Rubaiat Habib Kazi, and Paul Asente. 2020. Pronto: Rapid Augmented Reality Video Prototyping Using Sketches and Enaction. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, Honolulu HI USA, 1–13. <https://doi.org/10.1145/3313831.3376160>
- [51] Toby Jia-Jun Li, Amos Azaria, and Brad A. Myers. 2017. SUGILITE: Creating Multimodal Smartphone Automation by Demonstration. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, Denver Colorado USA, 6038–6049. <https://doi.org/10.1145/3025453.3025483>
- [52] Toby Jia-Jun Li, Lindsay Popowski, Tom Mitchell, and Brad A Myers. 2021. Screen2Vec: Semantic Embedding of GUI Screens and GUI Components. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM, Yokohama Japan, 1–15. <https://doi.org/10.1145/3411764.3445049>
- [53] Toby Jia-Jun Li and Oriana Riva. 2018. Kite: Building Conversational Bots from Mobile Apps. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, Munich Germany, 96–109. <https://doi.org/10.1145/3210240.3210339>
- [54] Wei Liang, Xinzhe Yu, Rawan Alghofaili, Yining Lang, and Lap-Fai Yu. 2021. Scene-aware behavior synthesis for virtual pets in mixed reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [55] David Lindlbauer, Anna Maria Feit, and Otmar Hilliges. 2019. Context-Aware Online Adaptation of Mixed Reality Interfaces. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. ACM, New Orleans LA USA, 147–160. <https://doi.org/10.1145/3332165.3347945>
- [56] Feiyu Lu and Doug A. Bowman. 2021. Evaluating the Potential of Glanceable AR Interfaces for Authentic Everyday Uses. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. IEEE, Lisboa, Portugal, 768–777. <https://doi.org/10.1109/VR50410.2021.00104>
- [57] Feiyu Lu, Shakiba Davari, Lee Lisle, Yuan Li, and Doug A. Bowman. 2020. Glanceable AR: Evaluating Information Access Methods for Head-Worn Augmented Reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, Atlanta, GA, USA, 930–939. <https://doi.org/10.1109/VR46266.2020.00113>
- [58] Feiyu Lu and Yan Xu. 2022. Exploring Spatial UI Transition Mechanisms with Head-Worn Augmented Reality. In *CHI Conference on Human Factors in Computing Systems*. ACM, New Orleans LA USA, 1–16. <https://doi.org/10.1145/3491102.3517723>
- [59] Blair MacIntyre, Maribeth Gandy, Steven Dow, and Jay David Bolter. 2004. DART: a toolkit for rapid design exploration of augmented reality experiences. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*. 197–206.
- [60] Mark R Mine. 1995. ISAAC: A Virtual Environment Tool for the Interactive Construction of Virtual Worlds. (1995).
- [61] Michael Nebeling, Janet Nebeling, Ao Yu, and Rob Rumble. 2018. ProtoAR: Rapid Physical-Digital Prototyping of Mobile Augmented Reality Applications. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, Montreal QC Canada, 1–12. <https://doi.org/10.1145/3173574.3173927>
- [62] Michael Nebeling and Maximilian Speicher. 2018. The trouble with augmented reality/virtual reality authoring tools. In *2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE. <https://doi.org/10.1109/ismar-adjunct.2018.00098>
- [63] Michael Nebeling, Maximilian Speicher, Xizi Wang, Shwetha Rajaram, Brian D Hall, Zijian Xie, Alexander RE Raistrick, Michelle Aebersold, Edward G Happ, Jiayin Wang, et al. 2020. MRAT: The mixed reality analytics toolkit. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [64] Benjamin Nuernberger, Eyal Ofek, Hrvoje Benko, and Andrew D. Wilson. 2016. SnapToReality: Aligning Augmented Reality to the Real World. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, San Jose California USA, 1233–1244. <https://doi.org/10.1145/2858036.2858250>
- [65] Arnaud Prouzeau, Yuchen Wang, Barrett Ens, Wesley Willett, and Tim Dwyer. 2020. Corsican Twin: Authoring In Situ Augmented Reality Visualisations in Virtual Reality. In *Proceedings of the International Conference on Advanced Visual Interfaces*. ACM, Salerno Italy, 1–9. <https://doi.org/10.1145/3399715.3399743>
- [66] Xun Qian, Fengming He, Xiyun Hu, Tianyi Wang, Ananya Ipsita, and Karthik Ramani. 2022. ScalAR: Authoring Semantically Adaptive Augmented Reality Experiences in Virtual Reality. In *CHI Conference on Human Factors in Computing Systems*. ACM, New Orleans LA USA, 1–18. <https://doi.org/10.1145/3491102.3517665>
- [67] Francisco J Romero-Ramirez, Rafael Muñoz-Salinas, and Rafael Medina-Carnicer. 2018. Speeded up detection of squared fiducial markers. *Image and vision Computing* 76 (2018), 38–47.
- [68] Hartmut Seichter, Julian Looser, and Mark Billinghurst. 2008. ComposAR: An intuitive tool for authoring AR applications. In *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*. IEEE, Cambridge, UK, 177–178. <https://doi.org/10.1109/ISMAR.2008.4637354>
- [69] Spatial. 2023. Create, share, and experience your creativity in 3D. <http://www.spatial.io/>
- [70] Tomu Tahara, Takashi Seno, Gaku Narita, and Tomoya Ishikawa. 2020. Retargetable AR: Context-aware augmented reality in indoor scenes based on 3D scene graph. In *2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 249–255.
- [71] Markus Tatzgern, Valeria Orso, Deniz Kalkofen, Giulio Jacucci, Luciano Gamborini, and Dieter Schmalstieg. 2016. Adaptive information density for augmented reality displays. In *2016 IEEE Virtual Reality (VR)*. IEEE, 83–92.
- [72] Daniel Wagner and Dieter Schmalstieg. 2007. Artoolkitplus for pose tracking on mobile devices. (2007).
- [73] Bryan Wang, Gang Li, and Yang Li. 2023. Enabling Conversational Interaction with Mobile UI Using Large Language Models. *arXiv:arXiv:2209.08655*
- [74] Bryan Wang, Gang Li, Xin Zhou, Zhourong Chen, Tovi Grossman, and Yang Li. 2021. Screen2Words: Automatic Mobile UI Summarization with Multimodal Learning. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. ACM, Virtual Event USA, 498–510. <https://doi.org/10.1145/3472749.3474765>
- [75] Tianyi Wang, Xun Qian, Fengming He, Xiyun Hu, Ke Huo, Yuanzhi Cao, and Karthik Ramani. 2020. CAPturAR: An Augmented Reality Tool for Authoring Human-Involved Context-Aware Applications. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. ACM, Virtual Event USA, 328–341. <https://doi.org/10.1145/3379337.3415815>
- [76] Zeyu Wang, Cuong Nguyen, Paul Asente, and Julie Dorsey. 2021. DistanciAR: Authoring Site-Specific Augmented Reality Experiences for Remote Environments. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM, Yokohama Japan, 1–12. <https://doi.org/10.1145/3411764.3445552>
- [77] Jason Wu, Xiaoyi Zhang, Jeff Nichols, and Jeffrey P Bigham. 2021. Screen Parsing: Towards Reverse Engineering of UI Models from Screenshots. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. ACM, Virtual Event USA, 470–483. <https://doi.org/10.1145/3472749.3474763>
- [78] Haijun Xia, Sebastian Herscher, Ken Perlin, and Daniel Wigdor. 2018. Spacetime: Enabling Fluid Individual and Collaborative Editing in Virtual Reality. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. ACM, Berlin Germany, 853–866. <https://doi.org/10.1145/3242587.3242597>

- [79] Ya-Ting Yue, Yong-Liang Yang, Gang Ren, and Wenping Wang. 2017. SceneCtrl: Mixed Reality Enhancement via Efficient Scene Editing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM, Quebec City QC Canada, 427–436. <https://doi.org/10.1145/3126594.3126601>
- [80] Lei Zhang and Steve Oney. 2020. FlowMatic: An Immersive Authoring Tool for Creating Interactive Scenes in Virtual Reality. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. ACM, Virtual Event USA, 342–353. <https://doi.org/10.1145/3379337.3415824>
- [81] Xiaoyi Zhang, Lilian de Greef, Amanda Sweeny, Samuel White, Kyle Murray, Lisa Yu, Qi Shan, Jeffrey Nichols, Jason Wu, Chris Fleizach, et al. 2021. Screen recognition: Creating accessibility metadata for mobile applications from pixels. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–15.

## A APPENDIX

### A.1 Top 10 functionalities by environment

**Table 1: Top 10 functionalities by environment**

<b>Living Room</b>		<b>Office</b>	
(App/Group) Functionality	Count	(App/Group) Functionality	Count
(Group) App icon	52	(Group) App icon	25
(Group) UI navigation	24	(Group) Message conversation list	19
(Group) Video	18	(Group) UI navigation	19
(Social Media App) Stories	17	(Files) Reference figure	18
(Group) Music play/pause control	17	(Group) Social media post	12
(Group) Current weather information	16	(Group) Email inbox	7
(Group) Social media post	16	(Video Conferencing App) Video call screen	7
(Video Sharing App) Search	14	(Group) Music play/pause control	7
(Group) Search engine search	11	(Group) Video	7
(Group) Music playlist	9	(Group) Song title and artist	7

<b>Coffee Shop</b>		<b>Kitchen</b>	
(App/Group) Functionality	Count	(App/Group) Functionality	Count
(Group) UI navigation	45	(Group) Instructions	15
(Group) Message conversation list	20	(Group) App icon	14
(Group) Social media post	19	(Group) Video	13
(Group) App icon	18	(Group) Music playlist	7
(Group) Music play/pause control	17	(Learning App) Language learning practice	6
(Group) Email inbox	10	(Group) Email inbox	5
(Group) Search engine search	10	(Group) To-do list	4
(Group) To-do list	9	(Group) Music play/pause control	4
(Group) Daily schedule	9	(Online Shopping App) product image	4
(Social Media App) Post caption & comments	8	(Group) Daily schedule	4

<b>All environments</b>	
(App/Group) Functionality	Count
(Group) App icon	110
(Group) UI navigation	92
(Group) Message conversation list	46
(Group) Video	45
(Group) Music play/pause control	45
(Group) Music playlist	26
(Social Media App) Stories	26
(Group) Email inbox	25
(Group) Current weather information	23
(Group) Today's schedule	21