A2W: Context-Aware Recommendation System for Mobile Augmented Reality Web Browser

Kit Yung Lam*
kylambd@cse.ust.hk
HKUST
Hong Kong SAR

Lik Hang Lee
likhang.lee@kaist.ac.kr
KAIST and University of Oulu
Republic of Korea

Pan Hui
panhui@cse.ust.hk
HKUST and University of Helsinki
Hong Kong SAR

ABSTRACT

Augmented Reality (AR) offers new capabilities for blurring the boundaries between physical reality and digital media. However, the capabilities of integrating web contents and AR remain under-explored. This paper presents an AR web browser with an integrated context-aware AR-to-Web content recommendation service named as A2W browser, to provide continuously user-centric web browsing experiences driven by AR headsets. We implement the A2W browser on an AR headset as our demonstration application, demonstrating the features and performance of A2W framework. The A2W browser visualizes the AR-driven web contents to the user, which is suggested by the content-based filtering model in our recommendation system. In our experiments, 20 participants with the adaptive UIs and recommendation system in A2W browser achieve up to 30.69% time saving compared to smartphone conditions. Accordingly, A2W-supported web browsing on workstations facilitates the recommended information leading to 41.67% faster reaches to the target information than typical web browsing.

CCS CONCEPTS

• **Human-centered computing** → *Mixed / augmented reality*; **Web-based interaction**.

KEYWORDS

Augmented Reality, Mixed Reality, Web Browser, Interaction in-the-wild, Multi-device collaboration, Recommendation systems.

*K.Y. Lam and L.H. Lee contributed equally to this work (co-first author).

1 INTRODUCTION

In the context of emerging mobile augmented reality (MAR), our web contents (e.g. WebAR, WebXR) can superimpose on the physical world through mobile headsets [8, 11, 35, 40], with multitudinous applications including education and training [3, 12, 15] and entertainment [30, 39]. However, the current augmented reality (AR) experiences cannot proceed after taking off the mobile headsets. The applications are single-purpose service with sporadic usages, and eventually, the users give up the AR applications as the AR experience disconnects from their daily activities [14]. Specifically, we consider a common scenario as follows: *"A user with an AR mobile headset is on the way of one's daily commuting. The user encounters a multiple physical entities and nearby events (e.g. shops and restaurants) and results in related interaction with such entities with AR views. The interactions have been recorded, and further bring convenience to AR journey through appropriate system recommendations. However, the interaction and the corresponding system recommendations in AR are not inherited to the sedentary workstation at home(s) or office(s). In other words, the convenience discontinue once the AR being dismissed."*

As shown in the above scenario the supports of the practicality and continuity of AR towards our daily lives even after taking off the AR device are missing. Hence, the follow-up influence by AR experience is worthwhile for further investigation. How can we take advantage of AR towards our daily life is the key question of this paper. Collaboration between MAR interfaces and sedentary workstation could become a common scenario in future. Therefore, this paper aims to bridge the MAR experiences with the daily activities even after taking off the mobile headsets, by considering the outdoor AR-driven journey before reaching the web browser on a sedentary workstation perhaps at the user's workbench (e.g. the daily commute). We describe the following in-use challenges.During an outdoor journey, users with mobile headsets could encounter some virtual overlays on the way until reaching the workbench. However, the constrain-sized screen on mobile headsets can display limited information [28, 29], while the full version of web contents will be delegated to sedentary workstations. Afterwards, the web experiences on workstations leverage the cues from the user interaction with the virtual overlays in MAR perhaps outdoors.

An enabler is required to extend the web experience based on the interaction in MAR, where context-aware physical environments can facilitate the readiness of the users' web experience in the post-AR tasks. Thus, we propose AR-to-Web (A2W), a cross-device framework for AR-driven context-aware web browsing, which specifically bridges the human-web interaction in both outdoor AR journeys and sedentary workstations. It features with user-centric

web browsing on mobile headsets, and accordingly, a personalized web experiences driven by MAR. A2W allows concise web contents to display on the constrained-size screen on mobile headsets for ubiquitous web browsing with seamless user interaction, in which the web content is adaptive to the physical surroundings and the contexts. A context-aware recommender system in A2W offers extended AR benefits by connecting the MAR in outdoors to the web browsing on workstations.

To the best of our knowledge, A2W is the first work focusing on context-aware web browsing experience extended beyond mobile headsets. Our contribution is threefold: First, A2W framework serves as the backbone for illustrates the interaction paradigm between outdoor AR experience, context-aware web content, and workstation web browsing experience. We present the interaction model of context-aware web browsing minimizing interaction, as shown in Figure 2. Second, the A2W recommender system has taken the user interaction on mobile headsets into the account of context-aware web browsing. This integration enables the AR web browsing experience with better usability. Third, user evaluations show that A2W context-aware web browsing experience outperforms the smartphone experience significantly. A2W can create seamless benefits across post-AR stages.

2 RELATED WORK

A2W is established on the basis of the literature on AR Web applications and content management, Recommender System, as well as client-server architecture for task offloading.

AR Web Browsers: A User Perspective. AR applications can be categorised into two types: (1) Leveraging the web technologies for the implementation of platform-specific AR browsers [34, 43], and (2) solely web browsers [20, 36]. Although the existing works facilitate the AR content generation and rendering of static animated contents [32], the optimized design of AR layout, including visual content placement and user experience is neglected [24]. A2W parses a fully-functional webpage and achieves context-aware content placement in mobile headsets, and brings web contents in AR with user-centric perspective.

A survey [13] demands for user-centric AR browsers considering both the information display on AR interfaces and user affordance in a holistic manner [10, 31]. The optimization of information placement [2] with user cognitive loads [33], as well as content management [7] under HTML 5 framework through connecting the sensor data and web content, improves the usability of AR web browser and hence achieves easier user interaction. The recent works build a hierarchy to organise and visualize high volume of information in AR displays [41], and adopt a flattened hierarchy for easier content acquisition [17]. Although the latest works address the information display on AR web browser and the gestural interaction, the user interaction among these framework are not thoroughly studied. In contrast, the generic context-aware framework of A2W further considers the user interaction in outdoors before reaching sedentary workstations, implying the continued benefits of multi-device AR.

AR Web Content Recommender System. Recommender systems (RS) provide personalized recommendations on certain items or
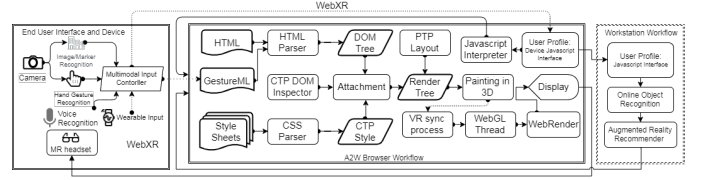

Figure 1: The workflow for the proposed 3-tier A2W AR Web Browser: the leftmost column is the End User Interface and Device Tier (Section 3.1); the middle column is the workflow of the rendering engine in A2W Web Browser Tier (Section 3.2), the rightmost column is the workflow of the recommendation system in Workstation Tier (Section 3.3).

products to different users. Most AR applications use distance-based filtering and visibility-based filtering [46]. However, these filters are insufficient to the dynamic change of environmental contexts in outdoors [41]. Thus, spatial, visibility and knowledge-based filters are suggested in Hybrid RS that employs two or more above techniques to overcome the above inadequacy [37], and more suitable for such AR scenarios. ReadMe [5] is the first RS with the real-time hybrid algorithm, but neglects the user-centric interaction with digital overlays. Considering the minimal interaction cost (e.g. click no.) [21] in outdoors, A2W addresses the interaction cost of MAR with RS, and reinforces the reachability of AR web contents.

Constrained Design for AR Web Browsing. Many AR frameworks rely on client-server architectures to offload computationally intensive tasks to a distal server, and provide a light-weight client for mobile headsets [16] to alleviate the constrained computational resource. Existing AR client-server architectures [10, 24, 45] effectively render rich AR content after detecting markers or object features on smartphones but they neglect the key constraints on AR smartglasses such as small screen real estate and content management. In contrast, the A2W architecture leverages geo-localized sensitive information and displays content on the basis of computed 'region of interest'. In the A2W display, we avoid putting the AR content in the 'region of interest'[22] and the idle space is used to show the web content for AR browsing scenarios. This design intentionally reserves user's interaction with the physical environment in mobile scenarios. During the outdoor walk, the users with the optimized content display have better experience of immersive AR web browsing than the content directly overlaid on the marker object, in terms of mental and physical loading and efficient assessment of information[22].

3 SYSTEM IMPLEMENTATION

A2W is a 3-tier architecture: (1) End User Interface and Device, (2) A2W Web Browser, and (3) Recommendation System tier, enabling user interaction with existing web content (Figure 1).

3.1 End User Interface and Device Tier

The End User Interface and Device tier enables the user interaction in mobile scenarios. We follow the WebXR API standard [35, 38] for providing a uniform abstraction layer for real-time rendering and access to interaction devices under the A2W architecture. Therefore,
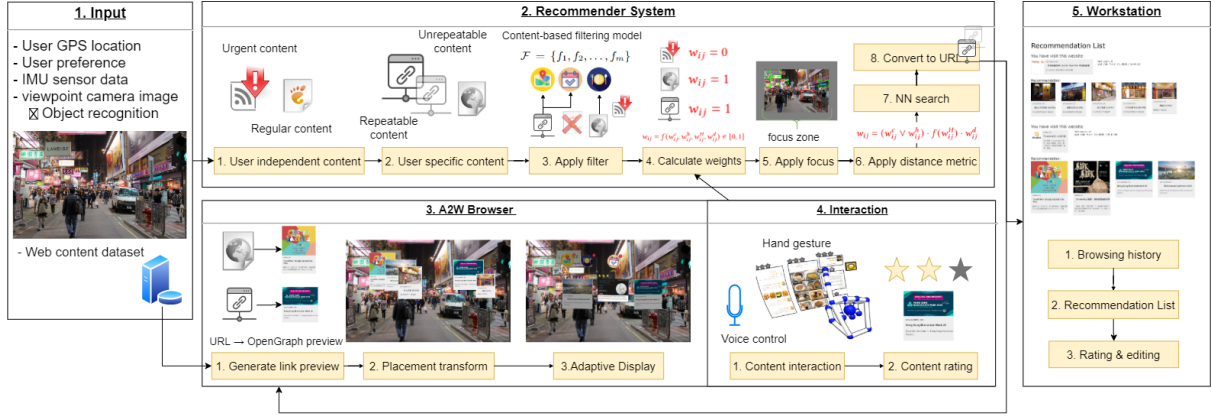
Figure 2: The workflow in A2W framework: PTP (algorithm 1) and CTP in (3) A2W Browser encourages mobile (4) Interaction with web contents, while the user interaction updates the (2) RS for seamless AR benefits to the (5) Workstation.

common input devices such as embedded camera and wearable sensors are supported when the data format following the WebXR standard [19, 38]. Multi-modal input manages the input devices then converts the input data into action and command(*acts*) to interact with the HTML elements and web browsing.

3.2 A2W Web Browser Tier

A2W web browser is built on WebKit render engine with several modifications, and supports the standard HTML protocol, which is responsible for rendering all supported DOM elements on the mobile headset. We list out the major modifications in our A2W web browser: (1) The rendering process of HTML DOM elements follows A2W browser workflow after updating the style properties of the elements . The default WebKit workflow maintains the style processing. The CSS Parser governs the downloaded style information and aggregates the external CSS files with the markup style. (2) Device JavaScript Interface manages the sensor data from multi-modal input controller. It manipulates the selected DOM elements with the supporting hand gestural interaction methods in 3D by GestureML [23](drag and drop, resize, delete, duplicate, translation, rotation, click and page navigation) when *acts* are received from a multi-modal input controller. (3) We adopt the Layout Cascading Style Sheets (LCSS) template [22] which overrides the original web page CSS and provides 3D spatial style properties for the DOM element. The Content Transformation Parser (CTP) [22] converts the DOM element with LCSS style into an appropriate WebGL format, after the original WebKit CSS is parsed. (4) **PTP** Layout [22] overrides the original WebKit Layout component to achieve context-aware layouting for the DOM elements. Placement Transform in the workflow handling the context-aware position of each content block, which considers the distance between the user and the related physical object. algorithm 1 describes the details of the 3D DOM element placement in context environments.

To display content in AR, we adopt Layout CSS (LCSS), which builds on top of the CSS language to provide AR specific display information. The LCSS instructions are stored as a separated style sheet. Several DOM elements, including $< header >$, $< nav >$, and

---

ALGORITHM 1. Placement Transform Algo. (PTA)

Input: A_C: Set of unplaced web contents from RS, g is the list of group for all web contents in A_C;

    cam: Vector position of camera view space in virtual scene; *gps*: current user GPS position;

    G: contents group by category; V_{top}: Vector position of Top Menu from *cam*;

w: default width of every content;

while A_C *not* $\emptyset$ **do**

    $c \leftarrow A_C.pop()$;

    $c.p \leftarrow c.position$;

    if $c.gps$ *not* $\emptyset$ **then**

        set $c_{Distance} \leftarrow d(gps, c.gps)$;

        $c.p \leftarrow c_{Distance}$ in cam+ c.offset;

        foreach $g \in A_C$ **do**

            if $c.p == g.p + g.radius$ **then**

                if $c.subject == g.subject$ **then**

                    hide $c\&g$, show icon of $c.subject$ on $c.p$;

                $c.p.x \leftarrow c.p.x + c.width/2$;

                $g.p.x \leftarrow g.p.x - g.width/2$;

                $c.scale \leftarrow c.scale/2*(c.weight)$;

                $g.scale \leftarrow c.scale/2*(g.weight)$;

    else

        $G[c.subject]$.append(c);

        sort G by $G[c.subject].weight$

    foreach $g \in G$ **do**

        g.position $\leftarrow V_{top} + (G$.indexOf(g) $*w)$;

Output: Transformation of each link preview in the screen view of AR/MR headset

---

$< img >$, have default LCSS attributes. Each website can override the default LCSS style and create its own AR layout. The rendering procedures from the HTML/CSS code to the AR display. comply the following steps: (1) The A2W web browser receives and parses the HTTP response. If the Content-Type is HTML, the **CTP** passes it to step 2. Otherwise, the content is ignored. (2) A2W rendering engine parses the HTML document and convert to a DOM tree. (3) New requests are issued for each additional resource in the HTML source including images, style sheets, and JavaScript files. (4) External scripts such as JavaScript are parsed and executed, the DOM elements are updated accordingly. (5) The CSS Parser parses the downloaded style information and aggregates the external CSS files with the markup style elements in the HTML source. (6) The **CTP**

DOM Inspector traverses and inspects the DOM tree to identify the web page structure.

3.3 Recommendation System Tier

The recommendation system in A2W framework provides web content recommendation service for both AR devices (mobile) and workstations (sedentary). A2W recommender system (RS) is designated for web browsing on mobile AR or MR headsets, and workstations. A2W RS aims at minimizing the user's interaction cost [21] of reaching the item of interest during AR journeys, while maximizing the information according to the context environment.

A2W RS is adapted from the content-based and spatial filters, instead of the social network factor and collaborative filter in ReadMe [5], with below details. The web content and its corresponding URL are the only type of virtual objects recommended by RS. The website URL is classified by the primary subject of the content [1], and this object categorisation [5] further applies to each content categories in A2W RS. Also, we introduce the user interaction with web contents as the expanded factors in the weight function for each web content. The user's short-term preference (w_{ij}^h) records the browsing action such as clicking and scrolling in the webpage, while the user's long-term preference (w_{ij}^H) considers the rating of web contents. In both cases, the weights are directly proportional to the number of user interaction but inversely to the times it was in the recommendation list (RL). The weight will be updated for all objects in the same content category (Eq. 1).

$$w_{ij} = f(w_{ij}^c, w_{ij}^h, w_{ij}^H, w_{ij}^d) \in [0, 1] \tag{1}$$

When the user location or detected object from viewpoint camera is changing, the weight of user focus w_{ij}^c will be updated as Eq. 2.

$$w_{ij}^c = \begin{cases} 1, & \text{if } Obj_{Distance} < \text{maximum distance, or} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

In specific, the weight is proportional to the physical distance between the user and the object, and its value is zero if the user has a recent check-in to an object that belongs to the same non-repeatable category (Eq. 3). The recent check-in object refers to any web contents that have been opened and reviewed in a period. For example, all the restaurant website can be classified as the non-repeatable category by the user, once the user has access a restaurant website, the weights of all restaurant web contents will be updated to zero.

$$w_{ij} = (w_{ij}^c \vee w_{ij}^h) \cdot f(w_{ij}^H) \cdot w_{ij}^d \tag{3}$$

A2W scores all the contents in the database relative to each other, such that the score for each content is between 0 and 1 inclusive, and such that the total scores of all contents equal to 1. The scores for each user-item pair (u,i) are computed as follows (Eq. 4). In the exponential function (exp), $\bar{w}_u$ and $w_{i/j}$ refer to the weight of the user and the item (i) embeddings in the list of result (j), respectively, while the denominator represents the summation over all contents in the database.

$$score(u, i) = \frac{exp(\bar{w}_u \top w_i)}{\sum_j exp(\bar{w}_u \top w_j)} \tag{4}$$

The algorithm of A2W RS starts with the given data including the user profile, real-time location, results of logo/object detected from the viewpoint camera, and all the user preference (e.g. previous check-ins). Then, the algorithm follows the abstraction of weighting virtual objects until the Nearest Neighbour Search [5]. The content-based filters search for the metadata of web contents, and remove those containing the prohibited keywords. Next, the user preferences and context data convert to URL parameters for each web content URL address, with three types of URL request formats: GET request, POST request and RESTful API. As the recommended results generated, the user preferences and parameters should be attached to the URL in the formats of **GET request** and **RESTful API**. These parameters include real-time user locations, user preference and web contents rating from the A2W web browser. The GET request is formatted as *domain.com/index.html?name = val&name2 = value2*. Finally, the results are classified by the content category, and deliver to A2W browser. The RL synchronizes with workstations, and allows the user's reviews.

Web content dataset. Users on our web platform can create and modify the dataset of web contents through A2W RS (see Figure 3), i.e. the user can add specific Uniform Resource Identifier (URI) to their own RS. We construct A2W RS dataset by concatenating
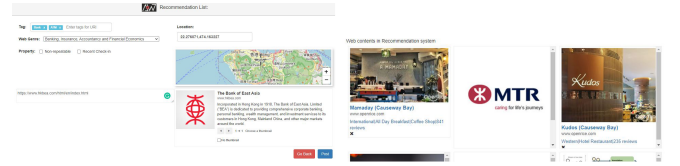

Figure 3: The URI editor for the recommendation system. A user can input the GPS coordinates for labeling the location-based web contents. The web genre of the URI is useful to group the rendering objects for both the A2W browser and recommendation system.

placemark (i.e. GPS locations), street view images [44], and the *place items* (PI) representing individual business entities. Figure **??** (Appendix) shows the information provided by each PI. A2W RS converts the PIs into web contents by adding the default weighting values (Eq. 1). To evaluate the performance A2W RS dataset with existing Google Map API unfiltered data (i.e. Place API), we count the number of place items provided by Google Place API with the GPS positions of each placemark (Figure 4). The key difference of A2W RS from the default Place API is the user's view point being mapped with dynamic user preference (Eq. 3), leading to different results in Table 1.

The radius of the place search limits to 500 meters , while the focus is 90° Field Of View for each direction of the placemark (Applied for both A2W RS (Eq. 2) and Google Map Place API). Also, we filtered out the web contents with categories not belonging to the 96 categories of place types from Google Place API [18]. Since portal/event/platform-based web contents such as Eventbrite/Youtube are excluded in the Google Map Place API, our dataset follows the exclusion. We further classified the types of web contents from Google Map Place API into nine main categories, and matched with the web contents suggested by A2W RS to get the coverage percentage of location-based information covered in five specific regions.

We ran a simulation of A2W RS on the Google Street View dataset [44] containing 10,343 placemarks with six pictures per each placemark (62,058 Google Street View images). By default, we collect the data from the Google Map Place API and convert each location resource and commercial entity to the dataset in A2W RS.

Accordingly, Table 1 presents the total number of web contents suggested by Google Map Place API and A2W RS grouped by 9 categories. We view Place API as the reference standard, i.e. the higher the better. The total number of place items from Google Map Place API is 6,132 (943 + 808 + 1145 + 987 +2249, Table 1) in the regions, while the total number of web contents suggested by A2W RS with the Google Map Place API dataset is 4,046 (65.98% of Google Map Place API). Divided by the categories, *Computers*, *Business* and *Recreation* are the top-3 highest ratio of web contents, indicating that A2W RS dataset achieved sufficiently high coverage of location-based information for these categories comparing with Google Map Place API. Nevertheless, A2W RS still suffers from lower ratios, due to insufficient dataset (e.g. Home, Health, and Arts). Even though A2W RS current has discounted performance compared with Google Place API, A2W emphasizes on the user-centric interaction (Figure 4) with mobile headsets and its extended benefits on workstations– that is, the compensation from the lower interaction cost, to be examined in the next section.
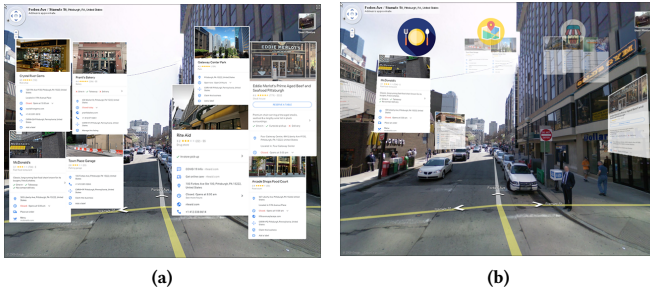

(a)   (b)

Figure 4: An example of A2W RS results in the simulation of Google Street Map images dataset[44], including 8 (6) restaurants, 12 (3) shops and 8 (2) commercial buildings within the radius of 500m, respectively. (a) The raw 2D web content results directly placed on the screen without PTP transformation. (b) The results presented the recommended web contents in CTP and PTP (algorithm 1) transformed cluster view, in which icons in groups are on the top menu.

4 EXPERIMENTAL PARADIGM

In the experimental setting (Figure 5a), users with A2W web browser receives the recommendation lists (RL). The physical binding web contents are placed on top of that object according to PTA in Table 1, which is presented as OpenGraph link preview button format before the user clicks on link preview buttons. For other non-physical binding web contents, the category icons show on top of the AR screen order by the score of the top-5 contents in the same category. In the AR interface with category icons, the results from the recommendation are listed in OpenGraph link preview format. In
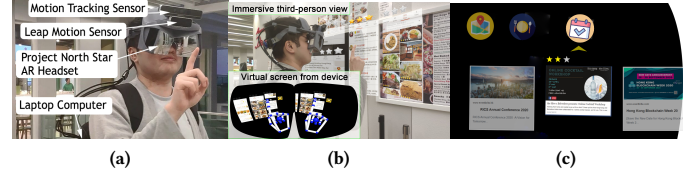

(a)   (b)   (c)

Figure 5: (a) Outdoor configurations with 4G network; (b) illustrating A2W in the experimental tasks (e.g. restaurant), plus hand gestural interaction; (c) a screenshot of A2W AR web browser in observer mode, showing the initial interface of A2W AR web browser with categorised web contents.

case of the cluttered display caused by large amounts of physical object binding contents, or more than two categories of the content in a radius region A2W web browser adopts the hierarchical clustering [41] of web contents based on their categories. A hierarchical representation of the web contents is created for showing all category icons and the number of the web contents in the cluster. The ordering of web content affects the efficiency of accessing web content and completion time. Therefore, the web contents with higher scores being displayed in the first place.

The implemented A2W is evaluated to show that 1) the user can access the target information ubiquitously, and 2) A2W RS enhances both MAR and desktop browsing experiences. Twenty participants (Gender: 14M /6F; Age: $\bar{M}$ = 24.9, σ = 3.1) were asked to complete three tasks of two-stage web browsing in outdoors (headsets) and afterwards on workstations. To address the context information related to the location of the experiments, we collected 50 different types of websites configured with categories and URL parameters, as the additional dataset for the local experiments: 1. Productivity Web App, 2. Entertainment Contents such as YouTube and Vimeo. 3. Location-based web contents such as Yelp and Meetup.

4.1 Stage I: Outdoor Tasks

The outdoor tasks primarily examine the user interaction with the information supported by PTA in A2W, throughout the daily scenarios of commuting between home and office. The participants accomplish their tasks in a 4G network under two alternative conditions: 1) **A2W web browsing with adaptive UI**: PTP layout shows web contents in dynamic position on mobile headsets (Figure 5a and 5b); 2) **Web browsing on smartphones (baseline)**: a smartphone touchscreen with a typical web browser.

The participants perform three tasks [9] in random orders and repeat three times for each task (i.e. 2 conditions * 3 tasks * 3 trials): Find a nearby 1) restaurant, 2) a public facility, and 3) an upcoming event. We record the completion time in each trial to understand the A2W easiness in mobile scenarios. A 15-min briefing session enables the participants to get familiar with the A2W interfaces and the gesture set.

Figure 5c illustrates the user interaction with the web contents on the mobile headset. The rating interface shows rating buttons on top of each web contents. The score of the contents will update once the user presses on the button. The unrated web contents are contributed to a score of zero to the weight w_{ij}^H. The leftmost

Table 1: The comparison for total number of web contents suggested by Google Map Place API and A2W RS, in five regions from the Google Street View dataset [44]. In each region, *Place API* column presents the number of items from Google Map API, while *A2W* column presents the number of web contents from A2W RS with Google Map API. The 2nd column adjacent to the *category* is the ratio of web contents by A2W RS to the number of items from Google Map API. A higher ratio indicates that A2W RS with the dataset is more suitable as the location-based RS.

GPS Coord.	Regions From To	Orlando, FL (28.537, -81.380) (28.558, -81.377)		Pittsburgh, PA (40.434, -80.016) (40.449, -79.98)		South Manhattan (40.706, -74.014) (40.733, -73.975)		Mid Manhattan (40.734, -74.01) (40.749, -73.971)		North Manhattan (40.75, -74.009) (40.764, -73.956)	
	Placemark count	1324		3078		1765		2036		1999	
Category (Ratio to Place API)		Place API	A2W	Place API	A2W	Place API	A2W	Place API	A2W	Place API	A2W
Arts	58.56%	39	12	97	57	92	67	79	61	137	63
Business	75.32%	227	171	173	124	109	81	313	236	572	438
Computers	76.54%	275	236	50	38	127	72	69	38	195	164
Health	57.72%	101	64	158	74	240	116	188	79	427	310
Home	45.59%	66	25	56	21	35	14	57	20	194	106
Recreation	69.01%	170	136	99	73	167	83	51	42	197	138
Shopping	66.88%	45	38	88	63	174	92	125	69	193	156
Society	64.31%	20	13	26	15	52	37	64	48	149	87
Sports	60.59%	8	3	61	26	149	97	41	19	185	124
Total	65.98%	943	698	808	491	1145	659	987	612	2249	1586

rating button means "Neutral" to the contents, and contributes the score of 0.1 to w_{ij}^H. The middle rating button means "Positive" to the contents and contributes the score of 0.5 to w_{ij}^H. The rightmost rating button means "Like" and contributes the score of 1 to w_{ij}^H.

4.2 Stage II: Workstation Tasks

The follow-up task (i.e. 2 conditions * 3 tasks, no repetition) on Workstations (WS) investigates the extended AR benefits in web browsing in post-headset interactions. According to the outdoor tasks, the participant works on the respective contents: 1) gives food review; 2) rating on google map; 3) sign up the event. A2W RS responds to the participants' interaction of web contents in outdoors and updates the recommendation list (RL) such as the examples from Figure 6. A2W computes the utility rate of the recommendation list by counting the webpage visit in outdoors. The utility rate indicates the complexity of the task and the quality of the RL. A sophisticated web browsing task involves multiple webpage visits (i.e. lower utility rate), or vice versa. Also, a high-quality RL presents web content links, and the user shows interests in visiting the web content, achieving a higher utility rate. A webpage, opened from the Recommendation List, by typing URL, or from search engine and contributed to the completion of task, is counted as one effective webpage visit; the first HTTP request (both GET and POST which redirect to another webpage or update of web content) and URL forward from that webpage is not counted as an effective visit, e.g. a web browsing is accomplished throughout one webpage from the Recommendation List plus another webpage from the search engine, where A2W counts one effective visit and two webpage visit. Thus, the utility rate of link in the A2W Recommedation List is 50%. Furthermore, we adopt the Top-N accuracy metrics to evaluate the accuracy of the top recommendations, comparing the items the participants have actually interacted on workstations. RS produces the ranked list of recommended items based on the context of environment information in the outdoor task. A2W provides the updated list, and asks participants to click on the web contents

relevant to the task. Then, we compute and aggregate the global Top-N accuracy metrics using Recall@N [4, 6, 47].

Figure 6: The procedures of sample cases in three tasks: 1st row: an input image from viewpoint camera; 2nd row: the results from RS; 3rd row: the result of adaptive content placement by CTP & PTP; 4th row: the result of A2W RL on Workstations, showing the visited web contents aggregated in A2W AR Web browser, and recommended web contents.
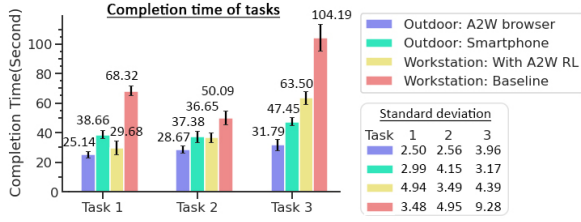
Figure 7: Mean completion time (Top ends of the bars) of outdoor and workstation tasks, with standard deviation values.

4.3 Results: Outdoors, Workstations and RS

Two-way RM-ANOVA analysis shows the statistical significance in the effects of **Condition** ($F_{(1,114)}$=443.7 ,p<0.01) and **Task** ($F_{(2,114)}$ = 64.58, p<0.01) to the completion time in seconds (s), with significant **Interaction** between Tasks and Conditions ($F_{(2,114)}$ = 11.76, p<0.01). Figure 7 shows the task completion times of three tasks, in which the outdoors and workstation activities are separated by the conditions. Generally, A2W browsers with adaptive UIs consistently outperforms the baseline of smartphone interactions in all three tasks, achieving 30.69% shorter averaged completion time ($\bar{M}_{C1}$ = 28.53 s, σ_{C1} = 3.33 s) than the baseline of typical browsing on smartphones ($\bar{M}_{C2}$ = 41.16 s, σ_{C2} = 5.49 s).

The **Condition** effect is represented by two distinctive platforms. The headsets with A2W leverages the content display and management to compensate the feature lost in touchscreen interaction, while smartphones own dominant interaction capability in terms of direct manipulations on touch-sensitive surface [26, 27]. It is important to note that the CTP and PTP in A2W browser display the contents in the right way, and their backbone of A2W RS selects the right contents. Under the similar tasks of finding a website in three contexts, the user interaction with the typical browser on smartphones involves a number of taps to direct into the target context. Even though touch-based interaction (e.g. direct taps) on tangible interfaces (i.e. smartphones) poses advantages of responsiveness [25], the information retrieval in A2W RS, supporting the content managements in CTP and PTP, serves as a critical enabler to assist the participants reaching the target information and reducing the interaction burdens on mobile headsets [29]. The users with A2W achieves significant time improvements since A2W makes the context ready for the user interaction, while the intuitive clicks with hand gesture enable the participants to quickly reach the target web contents. With the adaptive UIs, the optimized content placement leads to lesser visual demands and hence the shorten browsing time. Figure 7 illustrates the mean task time and standard deviation for A2W with mobile headsets (purple) and typical browser on smartphones (green) during the outdoor tasks.

Also, we recorded the average number of contents recommended in the outdoor experiments: 15 restaurant contents, 8 public facilities, 3 future events, and 9 other contents were recommended. The number of contents is driven by the web content database and algorithm, in which the static web contents such as the information of restaurants and public facilities contribute a significant proportion in our database. Thus, it is easier to reach the information about

Table 2: The participant counts with effective visits in post-AR, i.e. transition from a webpage to another, with increasing task complexity from Task 1 to 3.

Task	1		2		3	
Effective visit no.	1	>=2	1	>=2	1	>=2
With RL	20	0	14	6	20	0
Without RL	20	0	12	8	1	19

the static web contents, and the significant **Task** effect makes distinguishable completion times among the three tasks. It can be said that the difficulties of accessing the web contents are directly proportional to the task completion times. The primary reason is that the restaurant contents own the highest priority, while the future events have the lowest priority; these priorities affect the content displays.

In the follow-up experiments on workstations, web browsing tasks with A2W RL ($\bar{M}_{C1}$ = 43.275 s, σ_{C1} = 15.3 s) outperforms the baseline by 41.67% across all three tasks ($\bar{M}_{C2}$ = 74.20 s, σ_{C2} = 23.52 s). It is important to note that Task 3 results in longer completion time than Task 1 and 2 as the task of signing up an event involves text entry on the workstation. The complicated tasks (e.g. Task 3) leads to a more obvious performance gap between the two conditions. The participants without A2W RL have to spend extra efforts to search the event on workstations. In contrast, A2W leverages the weighting updates in mobile headsets, and subsequently reacts to the users on workstation responsively. Figure 7 depicts the values of means and standard deviations for A2W (yellow) and smartphones (red) by tasks on workstations.

We defined the target number of webpage is 60 (20 participants * 3 repetitions), and recorded 87 webpage visited in total. The number of the visited pages can serve as an auxiliary indicator to reflect the difficultly of the tasks or the easiness of reaching the web contents, but not necessarily the efficiency of a recommender system. Thus, the task completion time serves as the key indicator for the efficiency of A2W RS. In the study, one effective visit is counted by a transition from a webpage to another. For instance, some participants type the URL to visit Google Map, and subsequently type the place in Task 2 (i.e. 2 effective visits). We report the effective visits of the participants in Table 2. With RL, the overall utility rate is 90% ((20+14+20)/60) as six trials involving the link from search engine take more than one effective visit. Most trials are completed with the web link suggested by A2W RS, resulting in the improved completion time. In contrast, the utility rate drops significantly if the target web contents are not available at RL, e.g., in the more sophisticated Task 3, most participants without RL did >=2 effective visits, causing performance degradation.

Table 4 shows Hits@5/@10 Count (H@5/@10 Count CT) and the global Top-5/-10 (Recall@5/@10) accuracy metrics. A2W achieves the Recall@5 of 0.321 and the Recall@10 of 0.506, indicating that 32.1% and 50.6% of interacted items (Inter. CT) were ranked among the top-5 and top-10 items, respectively. That is, the users are interested in 1.605 out of 5 items for Recall@5, and 5.06 out of 10 items for Recall@10. Although more suggested items (i.e. Recall@10) demonstrates higher chances of hitting the target items, we have to consider the practicality by balancing the hit rates and the limited screen real estate on mobile headsets. Thus, Recall@5 becomes

Table 3: Technology Acceptance Survey showing positive user feedback in three metrics of PU, PEOU and IU.

Questions	AVG	MED	MIN	MAX	SD
Perceived Usefulness (PU), Cronbach Alpha = 0.9804 (>0.7)					
1)A2W would enable AR web browsing in outdoor conveniently.	3.80	4	3	5	0.523
2)A2W would improve the way I look for web contents in mobile AR.	3.85	4	2	5	0.745
3)A2W framework is useful for the outdoor web browsing.	3.65	4	2	5	0.671
4)A2W browser is useful for displaying context-aware web contents in mobile scenarios.	3.60	4	3	5	0.598
Perceived Ease of Use (PEOU), Cronbach Alpha = 0.9629 (>0.7)					
5)Learning to use A2W AR browser would be easy.	3.70	4	2	5	1.081
6)It would be easy for me to become skilful at using A2W.	2.78	3.5	2	5	0.759
7)I find A2W web browser easy to use.	3.65	3.5	2	5	1.089
Intention Of Use (IOU), Cronbach Alpha = 0.9962 (>0.7)					
8)When A2W becomes available, I intend to use it for browsing the AR web.	2.89	3	2	5	0.827
9)When A2W becomes available, I will use it with mobile headsets.	3.65	3.5	3	5	0.745
10)When A2W becomes available, I would use it frequently.	3.90	4	3	5	0.641

Table 4: Global Top-N accuracy metrics using Recall@N, separated by Outdoors (O) and Workstations (WS).

Task	1		2		3		Total
Stage	O	WS	O	WS	O	WS	
H@5 CT	21	31	15	18	8	13	106
H@10 CT	32	49	26	32	13	15	167
Inter. CT	60	90	55	65	30	30	330
Recall@5	0.35	0.34	0.27	0.28	0.27	0.43	0.32
Recall@10	0.53	0.54	0.47	0.49	0.43	0.50	0.51
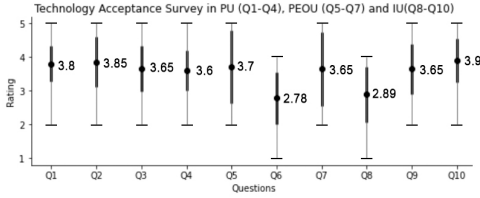
Figure 8: Technology Acceptance Model [42] from Table 3 (Appendix), where 1 – disagreement and 5 – agreement.

a compromised choice for mobile headsets by introducing more interaction cost, i.e. a scroll gesture to the next 5 items, while Recall@10 serves as a more favorable choice on workstations as more spacious displays are available. We acknowledge that the small number of trials, due to the limited participants, and the cold-start effect, possibly introduce bias to the metrics. It is suggested that the weighting in our dataset for context environments can obtain a higher precision by long-term interacted counts.

4.4 User Acceptance to the A2W Browsing

After completing the five tasks, the participants fill in a questionnaire about the technology acceptance. The participants have high technology literacy (4.2 out of 5.0). Overall, the participants are particularly positive (5pt-scale in Figure 8). They found the A2W browser to be easy to use ($\bar{M}$ = 3.65). Indeed, web browsing without context awareness was reported to be less efficient. Participants also considered the experimental browser easy to learn ($\bar{M}$ = 3.70), and

to become skillful at ($\bar{M}$ = 3.55). The participants also considered the A2W browser to be a convenient solution to the outdoor walk ($\bar{M}$ = 3.80). Although they considered that using the A2W would slightly improve the way they control than the smartphone control ($\bar{M}$ = 3.85), they found the prototype useful to outdoor web browsing ($\bar{M}$ = 3.65), especially for displaying context-aware web information in mobile scenario ($\bar{M}$ = 3.60). Finally, the participants generally agree with using the A2W in the future and use it frequently.

5 CONCLUSION

This paper presents a 3-tier AR web browsing framework named as A2W, serving as responsive web design for AR. A2W sheds light to the incrementally connection of the AR web browsing experience with the users' daily routines, primarily supported by the context-aware recommender system. The experimental A2W on AR headsets primarily serves as an evidence demonstrating the novel paradigm of user interaction with mobile AR in our urban cities. Users inside such immersive urban environments achieves an overall enhanced usability of 37.75% in-the-wild.

REFERENCES

[1] UK Web Archive. [n.d.]. UK Selective Web Archive Classification Dataset. 1996 - 2010. TSV. https://doi.org/10.5259/ukwa.ds.1/classification/1 Open dataset comprising a manually-generated selective archive in .TSV format.

[2] Ronald Azuma and Chris Furmanski. 2003. Evaluating Label Placement for Augmented Reality View Management. In *Proceedings of the 2Nd IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)*. IEEE Computer Society, Washington, DC, USA, 66–. http://dl.acm.org/citation.cfm?id=946248.946790

[3] Jorge Bacca, Silvia Baldiris, Ramon Fabregat, Sabine Graf, et al. 2015. Mobile augmented reality in vocational education and training. *Procedia Computer Science* 75 (2015), 49–58.

[4] R Bambini, P Cremonesi, and R Turrin. 2010. Recommender Systems Handbook, chapter Recommender Systems for a IPTV Service Provider: A Real Production Environment.

[5] Dimitris Chatzopoulos and Pan Hui. 2016. ReadMe: A Real-Time Recommendation System for Mobile Augmented Reality Ecosystems. In *Proceedings of the 24th ACM International Conference on Multimedia* (Amsterdam, The Netherlands) (*MM '16*). ACM, New York, NY, USA, 312–316. https://doi.org/10.1145/2964284.2967233

[6] Paolo Cremonesi and Roberto Turrin. 2010. Controlling consistency in top-N recommender systems. In *2010 IEEE International Conference on Data Mining Workshops*. Ieee, 919–926.

[7] P. Dangkham. 2018. Mobile augmented reality on web-based for the tourism using HTML5. In *2018 International Conference on Information Networking (ICOIN)*. 482–485. https://doi.org/10.1109/ICOIN.2018.8343165

[8] Moria Engberg, Jay David Bolter, and Blair MacIntyre. 2018. RealityMedia: an experimental digital book in WebXR. In *2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 324–327.

[9] E Enge. 2019. Where is the Mobile vs. Desktop Story Going. *Perficient Digital* (2019). https://www.perficientdigital.com/insights/our-research/mobile-vs-desktop-usage-study

[10] Timo Engelke, Mario Becker, Harald Wuest, Jens Keil, and Arjan Kuijper. 2013. MobileAR Browser–A generic architecture for rapid AR-multi-level development. *Expert Systems with Applications* 40, 7 (2013), 2704–2714.

[11] Irene Gironacci, Kim Vincs, and John McCormick. 2020. A Recommender System of Extended Reality Experiences. In *Proceedings of the 2020 3rd International Conference on Image and Graphics Processing* (Singapore, Singapore) (*ICIGP 2020*). Association for Computing Machinery, New York, NY, USA, 96–100. https://doi.org/10.1145/3383812.3383839

[12] Raphael Grasset, Andreas Duenser, Hartmut Seichter, and Mark Billinghurst. 2007. The mixed reality book: a new multimedia reading experience. In *CHI'07 extended abstracts on Human factors in computing systems*. 1953–1958.

[13] Jens Grubert, Tobias Langlotz, and Raphael Grasset. 2013. Augmented reality browser survey. *Commun. ACM* 56 (11 2013). https://doi.org/10.1145/2527190

[14] Jens Grubert, Tobias Langlotz, Stefanie Zollmann, and Holger Regenbrecht. 2016. Towards pervasive augmented reality: Context-awareness in augmented reality. *IEEE transactions on visualization and computer graphics* 23, 6 (2016), 1706–1724.

[15] Jennifer Herron. 2016. Augmented reality in medical education and training. *Journal of Electronic Resources in Medical Libraries* 13, 2 (2016), 51–55.

[16] Zhanpeng Huang, Weikai Li, Pan Hui, and Christoph Peylo. 2014. CloudRidAR: A Cloud-based Architecture for Mobile Augmented Reality. In *Proceedings of the 2014 Workshop on Mobile Augmented Reality and Robotic Technology-based Systems* (Bretton Woods, New Hampshire, USA) (*MARS '14*). ACM, New York, NY, USA, 29–34. https://doi.org/10.1145/2609829.2609832

[17] F. Husain, S. M. Hussein, T. A. Mastor, A. R. A. Rasam, A. M. Samad, I. Ma'arof, and K. A. Hashim. 2013. A review of WEB - based GIS for visualization of George Town, Penang sustainability as world heritage site. In *2013 IEEE International Conference on Control System, Computing and Engineering*. 435–440. https://doi.org/10.1109/ICCSCE.2013.6720004

[18] Google Inc. 2018. *Google Places API*. https://developers.google.com/places/supported_types

[19] B Jones and N Waliczek. 2019. WebXR device API. *W3C Working Draft* 10 (2019).

[20] Felix Klein, Dmitri Rubinstein, Kristian Sons, Farshad Einabadi, Stephan Herhut, and Philipp Slusallek. 2013. Declarative AR and Image Processing on the Web with Xflow. In *Proceedings of the 18th International Conference on 3D Web Technology* (San Sebastian, Spain) (*Web3D '13*). ACM, New York, NY, USA, 157–165. https://doi.org/10.1145/2466533.2466544

[21] Branislav Kveton and Shlomo Berkovsky. 2016. Minimal interaction content discovery in recommender systems. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 6, 2 (2016), 1–25.

[22] K. Y. Lam, L. H. Lee, T. Braud, and P. Hui. 2019. M2A: A Framework for Visualizing Information from Mobile Web to Mobile Augmented Reality. In *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom*. 1–10. https://doi.org/10.1109/PERCOM.2019.8767388

[23] Gesture Markup Langauge. 2013. GestureML Wiki. Extensible markup language used to define gestures that describe interactive object behavior and the relationships between objects in an application. *Accessed on* 3 (2013), 15.

[24] T. Langlotz, T. Nguyen, D. Schmalstieg, and R. Grasset. 2014. Next-Generation Augmented Reality Browsers: Rich, Seamless, and Adaptive. *Proc. IEEE* 102, 2, 155–169. https://doi.org/10.1109/JPROC.2013.2294255

[25] L. Lee and P. Hui. 2018. Interaction Methods for Smart Glasses: A Survey. *IEEE Access* 6 (2018), 28712–28732. https://doi.org/10.1109/ACCESS.2018.2831081

[26] Lik Hang Lee, Tristan Braud, Farshid Hassani Bijarbooneh, and Pan Hui. 2019. TiPoint: Detecting Fingertip for Mid-air Interaction on Computational Resource Constrained Smartglasses. In *Proceedings of the 23rd International Symposium on Wearable Computers* (London, United Kingdom) (*ISWC '19*). ACM, New York, NY, USA, 118–122. https://doi.org/10.1145/3341163.3347723

[27] Lik Hang Lee, Tristan Braud, Farshid Hassani Bijarbooneh, and Pan Hui. 2020. UbiPoint: Towards Non-Intrusive Mid-Air Interaction for Hardware Constrained Smart Glasses. In *Proceedings of the 11th ACM Multimedia Systems Conference* (Istanbul, Turkey) (*MMSys '20*). Association for Computing Machinery, New York, NY, USA, 190–201. https://doi.org/10.1145/3339825.3391870

[28] Lik Hang Lee, Tristan Braud, Kit Yung Lam, Yui Pan Yau, and Pan Hui. 2020. From seen to unseen: Designing keyboard-less interfaces for text entry on the constrained screen real estate of Augmented Reality headsets. *Pervasive and Mobile Computing* 64 (2020), 101148. https://doi.org/10.1016/j.pmcj.2020.101148

[29] L. H. Lee, K.Y. Lam, Y. P. Yau, T. Braud, and P. Hui. 2019. HIBEY: Hide the Keyboard in Augmented Reality. In *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom*. 1–10. https://doi.org/10.1109/PERCOM.2019.8767420

[30] Jakob Leitner, Michael Haller, Kyungdahm Yun, Woontack Woo, Maki Sugimoto, and Masahiko Inami. 2008. IncreTable, a mixed reality tabletop game experience. In *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*. 9–16.

[31] T. Leppänen, A. Heikkinen, A. Karhu, E. Harjula, J. Riekki, and T. Koskela. 2014. Augmented Reality Web Applications with Mobile Agents in the Internet of Things. In *2014 Eighth International Conference on Next Generation Mobile Apps, Services and Technologies*. 54–59. https://doi.org/10.1109/NGMAST.2014.24

[32] L. Li, X. Qiao, Q. Lu, P. Ren, and R. Lin. 2020. Rendering Optimization for Mobile Web 3D Based on Animation Data Separation and On-demand Loading. *IEEE Access* (2020), 1–1.

[33] 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* (New Orleans, LA, USA) (*UIST '19*). Association for Computing Machinery, New York, NY, USA, 147–160. https://doi.org/10.1145/3332165.3347945

[34] Blair MacIntyre, Alex Hill, Hafez Rouzati, Maribeth Gandy, and Brian Davidson. 2011. The Argon AR Web Browser and Standards-based AR Application Environment. In *Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality (ISMAR '11)*. IEEE Computer Society, Washington, DC, USA, 65–74. https://doi.org/10.1109/ISMAR.2011.6092371

[35] Blair MacIntyre and Trevor F Smith. 2018. Thoughts on the Future of WebXR and the Immersive Web. In *2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 338–342.

[36] Christoph Oberhofer, Jens Grubert, and Gerhard Reitmayr. 2012. Natural Feature Tracking in JavaScript. In *Proceedings of the 2012 IEEE Virtual Reality (VR '12)*. IEEE Computer Society, Washington, DC, USA, 113–114. https://doi.org/10.1109/VR.2012.6180908

[37] Carlos Plaza de Miguel. 2014. ARLodge: context-aware recommender system based on augmented reality to assist on the accommodation search process. (2014).

[38] Olle Renius. 2019. A Technical Evaluation of the WebXR Device API for Developing Augmented Reality Web Applications.

[39] Christopher Stapleton, Charles Hughes, Michael Moshell, Paulius Micikevicius, and Marty Altman. 2002. Applying mixed reality to entertainment. *Computer* 35, 12 (2002), 122–124.

[40] Nikolai Suslov. 2019. LiveCoding. space: Towards P2P Collaborative Live Programming Environment for WebXR. In *Proceedings of the Fourth International Conference on Live Coding (ICLC 2019), Medialab Prado, Madrid, Spain, http://iclc.livecodenetwork. org/2019/papers/paper133. pdf*.

[41] M. Tatzgern, V. Orso, D. Kalkofen, G. Jacucci, L. Gamberini, and D. Schmalstieg. 2016. Adaptive information density for augmented reality displays. In *2016 IEEE Virtual Reality (VR)*. 83–92. https://doi.org/10.1109/VR.2016.7504691

[42] Yui-Pan Yau, Lik Hang Lee, Zheng Li, Tristan Braud, Yi-Hsuan Ho, and Pan Hui. 2020. How Subtle Can It Get? A Trimodal Study of Ring-Sized Interfaces for One-Handed Drone Control. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 2, Article 63 (June 2020), 29 pages. https://doi.org/10.1145/3397319

[43] Yu You and Ville-Veikko Mattila. 2013. Visualizing Web Mash-ups for In-situ Vision-based Mobile AR Applications. In *Proceedings of the 21st ACM International Conference on Multimedia* (Barcelona, Spain) (*MM '13*). ACM, New York, NY, USA, 413–414. https://doi.org/10.1145/2502081.2502246

[44] A.R. Zamir and M. Shah. 2014. Image Geo-localization Based on Multiple Nearest Neighbor Feature Matching using Generalized Graphs. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* PP, 99, 1–1. https://doi.org/10.1109/TPAMI.2014.2299799

[45] Wenxiao Zhang, Bo Han, and Pan Hui. 2018. Jaguar: Low latency mobile augmented reality with flexible tracking. In *Proceedings of the 26th ACM international conference on Multimedia*. 355–363.

[46] Zhuo Zhang, Shang Shang, Sanjeev R Kulkarni, and Pan Hui. 2013. Improving augmented reality using recommender systems. In *Proceedings of the 7th ACM conference on Recommender systems*. ACM, 173–176.

[47] Qian Zhao, Martijn C. Willemsen, Gediminas Adomavicius, F. Maxwell Harper, and Joseph A. Konstan. 2019. From Preference into Decision Making: Modeling User Interactions in Recommender Systems. In *Proceedings of the 13th ACM Conference on Recommender Systems* (Copenhagen, Denmark) (*RecSys '19*). Association for Computing Machinery, New York, NY, USA, 29–33. https://doi.org/10.1145/3298689.3347065