ARLayout: Massive Physical Targets Visual Re-Layout in Mobile Augmented Reality

Category: Research

Paper Type: Technique

Abstract— The increasing popularity of VR/AR/MR devices has attracted a great deal of attention from researchers in the communities of virtual reality, visualization and human-computer interaction. However, enabling the public to create AR-based visualizations is challenging due to the steep learning curve of programming on AR devices. In this paper, we propose *ARLayout*, an authoring tool designed for non-experts to build AR-based visual re-layouts towards massive physical targets, such as books, coffees, drinking, foods, eyeshadows or even some toys, which are captured from the reality world in real-time by the camera of mobile devices such as mobile phones or tablets. All the candidate targets can be segmented and labeled by a convolutional neural network named BOWDA-Net [70], which just requires a small scale of labeled training samples (less than 100 for each case in our experiments). The textual information is recognized by an OCR algorithm. The visual re-layouts of the physical targets include highlighting the search results by fisheye deformation, re-grouping and re-ranking. Specifically, (1) the fuzzy keywords are input by voice to narrow down the number of candidate targets by searching, while the search results will be highlighted in the AR environment to guide them where to find the targets in the reality world. (2) candidate targets can be re-grouped in the AR environment according to one/multiple of their nominal, ordinal or quantitative attributes. (3) candidate targets can be re-ranked in the AR environment according to one of their ordinal or quantitative attributes. We demonstrate the usability, expressiveness and effectiveness of *ARLayout* by user study and case study with three usage scenarios in the evaluation.

Index Terms—Augmented reality, authoring tool, visual layout, physical visualization

1 Introduction

The development and the popularity of augmented reality (AR) devices and the techniques have led to an increasing number of studies designing new authoring tools or personal visualizations (PV). PV is designed to help the public get better insight into the data in a personal context. Unlike the traditional visualizations that are oriented to domain experts, PV is less domain-specific and highly focus on daily life [24,55]. Chen et al. [10] have summarized that mobile AR techniques have great potential to facilitate PV.

In our everyday life, it is probable that it would take us a large amount time to find or choose a target object, e.g., a book, a coffee, a kind of drinking, a cup, a kind of fruit or even an eyeshadow from massive candidates according to some fuzzy information. Similarly, it also would take us too much time to filter, group or sort them by one or multiple additional attributes for better comparison. The used additional attributes can be summarized into the nominal, the ordinal or the quantitative.

For example, finding a book according to the book name with fuzzy keywords or fuzzy author names in a library/bookstore (searching/filtering), and then highlight the search results. Besides, they would browse all the books and filter them to get a smaller number of candidate books (e.g., "Algorithms", nominal) for further comparisons. There are two subsequent actions the users would probably take: (1) Re-grouping. Re-group them according to the publishers ("IEEE" or "Springer", nominal), book series, topics (e.g., "dynamic programming", nominal), or even more additional attributes instead of the unified classification numbers frequently-used in libraries. (2) Re-ranking. sort them according to their ratings (ordinal), prices (quantitative) or even more additional attributes. The illustrations of the library usage scenarios are shown in Fig. 1.

Except for the example of finding/comparing targets from massive candidates, such as the above-mentioned finding/re-grouping/re-ranking books in a library or a bookstore, we also note that many people either are hard to distinguish the names of some massive goods like foods or drinking, or are ambiguous to choose a specific goods from the massive candidates because they could not tell or remember the major differences of them. For example, we find that many people are hard to identify different coffees according to a pre-study investigation of the user study in Section 4, such as Caffe Misto, Blonde Roast, Caffe Mocha, Cappuccino, Caffee Americano, Flat White, Pistachio Latte, Caramel Macchiato, Nitro Cold Brew, Iced Coffee, Irish Cream Cold Brew, White Chocolate Mocha, Iced Pistachio Latte, or even the combinations of coffees with tea and coffees with milk, etc. We also find it difficult for them to choose a coffee in a coffee shop because they cannot remember the major differences of their ingredients (quantitative) e.g., the total caffeine, calorie, protein, fat, milk, etc, the price (quantitative) and rating (ordinal) Is it sweet or sugarless and what the total sugar content (quantitative) or the place of origin (nominal) is. For the users who can identify the coffee names but often are ambiguous to choose a coffee in a coffee shop, he would further filter the coffee to get a small number of target candidates, e.g., "espresso" or "white americano" (searching/filtering), and then re-group them according to the above-mentioned nominal, ordinal or quantitative attributes (re-grouping), or sort them according to the ordinal or quantitative attributes (re-ranking).

We summarize many of such usage scenarios (book, coffee, drinking, fruit, eyeshadow, etc.) towards **massive target objects** in our daily life into three categories according to the tasks and data attributes as follows. We call them **visual re-layout for massive targets** in this paper because almost all of them need to break the original physical layouts in the reality.

- **Searching or filtering**: highlight the searched results to provide visual cues about their physical positions (used attributes: **nominal (including hue), ordinal**).
- **Re-grouping**: re-group the target objects according to one/multiple attributes via breaking the original physical layouts in AR environment (used attributes: **nominal (including hue), ordinal, quantitative**).
- **Re-ranking**: sort the target objects according to one/multiple attributes via changing the original physical layouts in AR environment (used attributes: **ordinal, quantitative**).

The above-mentioned visual re-layout for massive targets can be achieved by AR-based authoring PVs. However, enabling the public especially for the users with little knowledge about programming to create AR-based authoring tools is challenging due to the steep learning curve of programming on AR devices and the inconvenient offline

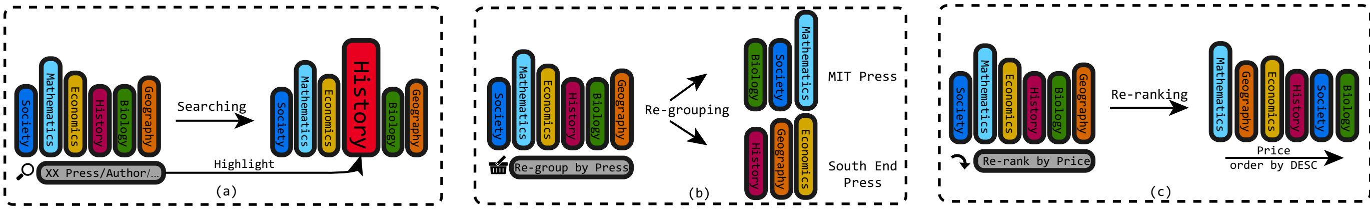


Fig. 1: The task illustrations on searching (a), re-grouping (b), and re-ranking (c) in AR environment. We take the library/bookstore scenario as an example.

workflow where users should work back-and-forth between AR devices and desktop PCs [10].

In this paper, we propose *ARLayout*, an authoring PV tool designed for non-experts to build AR-based visual re-layouts based on real-time videos towards massive physical targets, which are captured from the camera of personal mobile devices such as mobile phones or tablets. All the candidate targets can be segmented and labeled by a convolutional neural network named PaddleSeg [45,46], which just requires a small scale of labeled training samples (less than 100 in our three cases). The textual information about the targets is recognized by an OCR algorithm. The visual re-layouts can be achieved by above-mentioned searching (or filtering), re-grouping and re-ranking, etc. First, searching is used to narrow down the number of candidate targets where the search keywords are input by voice. The search results will be highlighted in the AR environment to guide them where to find the target in the reality world. Secondly, candidate targets can be re-grouped in the AR environment according to one or even multiple of their nominal, ordinal or quantitative attributes. Thirdly, candidate targets can be re-ranked in the AR environment according to one of their ordinal or quantitative attributes. In the experiments, we evaluate the proposed *ARLayout* by user study and three usage scenarios, which demonstrate the usability, effectiveness and expressiveness of the tool.

2 Related Work

2.1 AR Visualizations

Embedded data representations links visualization systems to physical things [66]. Augmented Reality (AR), as an important method to combine digital data and physical world, can visualize data in the physical space to facilitate certain visual explorations and integrate visualization into personal ideas and preferences [8]. When displaying ubiquitous data integrated into the everyday life, spatial immersion issues like depth perception, data localization and object relations become relevant. Works related to AR nowadays can be roughly classified as mobile (or tablets) hand-based systems [?], and head-mounted displays (HMD) systems [22] according to the computing paradigm. The Hololens device includes a depth sensing camera that estimates the distance of each pixel in view and stitches together a mesh or spatial map of the environment [18]. Similar technologies are also available in Google Tango [47] and Intel Realsense [36]. The software development kit (SDK) [7] provides ordinary programmers with more freedom and flexibility to use their own inspiration to design excellent AR applications, such as ARToolkit [35,65], Vuforia [16], ARCore for Android [5], and Mixed for Microsoft Universal Windows platform (UWP) Reality Toolkit [43]. A-Frame [1] which enables create immerse visualization scenes in the browser integrating by WebVR [2] content within HTML.

An inevitable direction of AR development is simultaneous localization and mapping (SLAM) [27] technology, which can build a model simulating the real environment through the background process based on panoramic 3-D reconstruction. Motion tracking based on MonoSLAM [28] has problems such as extensive calculation, long work time, existing scale ambiguity [28], and difficulty detecting feature points when the device is moving fast. The integration of inertial measurement unit IMU [58] to get six degrees of freedom (6DOF) [41] of the device plays a complementary role in improving its refresh rate and accuracy.

AR visualization has been applied to many fields. In outdoor visualization, the CityViewAR [40] provides information about destroyed buildings and historical sites that are affected by the earthquakes. Then, AR in the interpretation of terrain relief [9] shows a great usability, which serves as a motivational tool for the 3-D visualization. Applications in the newly emerging field of Augmented Reality Art show the paradigmatic potential of AR as a new artistic medium [20] .

2.2 Personal Visualization

Personal Visualization fits personal routines, situating visualization in a real-world context, and arousing users' interests. Visualization (Vis) and visual analytics (VA) offer substantial opportunities to help individuals gain insights about themselves. The combination of large interactive displays with personal head-mounted Augmented Reality (AR) [55] for information visualization to facilitate data exploration and analysis. Personal Visualization and Personal Visual Analytics (PV & PVA) are defined in terms of personal context [24]. It broadens the scope of visualization and visual analytics, including casual InfoVis [54], InfoVis for the Masses [29], persuasive computing [19] and personal informatics (PI) [25,26]. Researchers have presented mobile PV systems [12,30,63] to visualize daily information collected from peripheral sensors. Besides, in physical PV systems express abstract data in a physical form [31]. Jansen et al. [32,33] shows physical visualization can improve the users' efficiency in retrieval tasks.

Some researchers have designed physical visualization to visualize personal data. Khot et al. [37] transforms heart rate data into five kinds of 3-D printed material artifacts, and Stusak et al. [62]designs four types of sculptures to visualize running activities and evaluates their usability through a field study. Huang et al. [23] designs a behaviour feedback visualization tool to help people improve their health or personal well-being or to carry out sound environmental sustainability practices. Moreover, understanding and reasoning about personal data is of great importance, e.g., pedometer counts [62], blood pressure readings [21] or home electricity consumption [49], It can be chanllenging while gaining a deeper understanding of one's current practices and learning how to make a change when using data alone.

2.3 Visualization Authoring Tools

The field of immersive visualization makes use of Augmented Reality techniques to successfully support users in visualizing data. However, designing visualizations in immersive environments can be intricate, which needs consideration of issues such as shadows [6,50,52], occlusion, and additional argumentations. Nowadays several techniques are available that allow real-time rendering of shadows in augmented reality scenes [34,64]. Different approaches are used to modify the appearance of occluding objects to uncover the hidden ones. Cut-aways can be found for instance [13,14], and transparencies in combination with masks are used [17].

On one hand, researches have proposed several toolkits that allows interactive authoring and exploration of data visualisation in immersive environments. With MARVisT [10], users without visualization expertise can bind data to real-world objects to create expressive AR glyph-based visualizations rapidly and effortlessly. DXR [61] further provides a GUI for easy and quick edits and previews of visualization designs in-situ, i.e., while immersed in the virtual world. IATK [15] allows for easy assembly of visualisations through a grammar of graphics that a user can configure in a GUI–in addition to a dedicated visualisation API. PapARVis [11] designs an environment that can debug both static and virtual content simultaneously. Automated Window/Icon/Menu/Pointing Device User Interface (WIMP-UI) [3] generation has been considered a promising technology for at least two decades. iVisDesigner [56] achieves high interactive expressiveness through conceptual modularity, covering a broad information visualization design space. A mixed-initiative system Voyager that supports faceted browsing of recommended charts chosen according to statistical and perceptual measures [67].

On the other hand, an API known as OpenGL bridges the gap between piles of raw data and extremely complicated three-dimensional animation in a way [60]. The graphical demands of some of the applications have impeded their successful settlement in Web scenarios, thus people begin to use WebGL [51] to articulates a large portion of the rendering task in the client machine. Declarative visual design language can customizes built-in chart types, such as Echarts [42] and D3 [69]. A high-level grammar Vega-lite [59] provides visual encoding rules and a composition algebra that enables rapid specification of interactive data visualizations. Some features are used to analyze, display and manage the 3-D structure of the molecules by Vega [53].

	Data Scale (Target Number)	Task					Visual Presentation			Work-flow (Single/ Collab)
		Virtual Space	Augmented Infomation	Searching	Re-grouping (multi-attris)	Re-ranking (multi-attris)	Glyph Vis	Small Multiples	Fish Eye Highlight	
DXR	virtual	■	■				■	■		Sin
AVT	virtual	■	■							Collab
SA Vis	<5		■					■		Sin(PV)
VR Visc	virtual	■	■				■	■		Sin(PV)
VR Collab Vis	virtual	■	■							Collab
PapARVis	<5		■					■		Sin(PV)
MarVisT	<30		■				■			Sin(PV)
Our Work	40~1000+		■	■	■	■		■	■	Sin(PV)

Table 1: Comparison to the most related recent work about authoring PV tools towards AR or VR visualizations. The most related approaches include: DXR [61], Augmented Virtual Teleportation (AVT) [57], Situated Analytics (SA Vis) [**?**], Data Visceralization (VR Visc) [38], Shared Surfaces and Spaces (VR Collab Vis) [39], PapARVis [11], MARVisT [10]. The workflow can be categorized into PV (single user in a personal context), single user or collaborative users.

2.4 Relationship with The Most Related Work

We note that there are some most related AR-based tools on authoring visualizations [10, 11], etc. We summarize and discuss the differences between *ARLayout* and the most related ones as shown in Table 1 according to the data scale, tasks (augmented information, searching, re-grouping, re-ranking), visual presentations (glyph, small multiples, fish eye highlight), workflow (personal, single or collaborative).

First, the biggest differences between our work and the existing AR-based PV tools like MARVisT [10] are the data scale and the tasks, we focus on massive targets especially for the number is larger than 40 or even thousands of targets like books in a library/bookstore while most of AR-based PV tools just focus on physical targets with the number smaller than 30 [10], PapARVis [11] (≤ 5) and Situated Analytics [**?**] (≤ 5). The large data scale of this paper poses a new challenge in image segmentation, object labelling, textual information recognition and the AR-based data presentation.

Second, we focus on searching/filtering, re-grouping and re-ranking according to the additional attributes of massive physical targets through the visual re-layouts of them, instead of augmenting the existing static visualizations in PapARVis [11]. The PV tasks are different from the most related work due to the larger data scale of *ARLayout*. On the one hand, our work can process more than a thousand candidate targets at the same time, allowing users to access most of the information in the scenario. On the other hand, this feature on data scale also makes our work adaptive to more scenarios, rather than limited to some specific scenarios with only a few targets.

Third, the visual presentations of *ARLayout* are different. We focus on AR-based fisheye highlighting, virtual word cloud and virtual small multiples for better comparisons for massive targets instead of virtual glyphs [10].

Fourth, there are a series of methods focus on the virtual targets in the virtual space or virtual screen, which are more close to virtual reality (VR) [39, 57, 61] instead of AR.

3 Design Rationale

We illustrate the design goal, design considerations and design details of *ARLayout* in this section.

3.1 Design Goals

The target users of *ARLayout* are the people who are not the experts in a given domain. The "users" or "target users" in this paper refer to this group of users, if they are not specially specified.

We summarize the following four design goals of *ARLayout* according to the requirements collected from the daily life of the target users:

- G1: enable the target users to search/filter massive physical targets, and then highlight the results to guide them where they are in reality.
- G2: enable the users to re-group the physical targets according to their grouping comparison tasks.
- G3: enable the users to re-rank/sort the physical targets according to their ordinal or quantitative comparison tasks.
- G4: provide extra augmented information in AR environments according to their tasks.

3.2 Design Considerations and Design Details

We also summarize the design considerations and design details of *ARLayout* towards the design goal (G1-G4):

First, the tool should be designed to enable the target users to search/filter the massive physical targets by one or multiple fuzzy keywords input by voice, because voice input is simple-to-use in the public's PV context, such as a library, a bookstore, or a coffee shop. The input method by virtual keyboard should be also provided when it is not convenient to input a voice. The search results should be **highlighted by visual cues** to guide the users where they are in reality. Specifically, we use flashing to highlight the search results and further provide an AR-based **fisheye deformation design** to highlight their positions in reality.

Second, the tool should be designed to enable the users to re-group the physical targets according to their grouping comparison tasks (**visual re-layouts**), e.g., re-grouping them according to one/multiple of their nominal, ordinal or quantitative attributes, because grouping can help users better compare target candidates according to the experience in our daily life.

Third, the tool should be designed to enable the users to re-rank the disordered physical targets, or re-rank them according to one or multiple additional attributes (**visual re-layouts**). For example, books in a library are usually sorted by classification number or index number, which can not satisfy users' kinds of ranking needs, e.g., sorting them by the rating, price, publisher or publish year. Similarly, the books in a bookstore are often sorted by user groups, more information like ratings and prices are ignored. As a result, readers may spend a great deal of time finding an ideal book on the bookshelves.

Fourth, in people's daily life, the provided information alongside a target is usually not enough. For example, we can see the title of a book, the name and the price of a cup of coffee. However, the rating of a coffee, a book or a food, and the ingredients or drinks, foods, fruits are often not provided clearly to make comparisons. Therefore, the tool should be designed to display extra information which is often hidden from users or is inconvenient for them to compare. Actually, this kind of augmented information can be provided by **juxtaposition** or using **small multiples** to help them make a better choice. Except for the re-layouts, the sorted information such as the price and the rating can be visualized by **bar charts** or **line charts** in AR.

The extra textual information such as the textual ratings or detailed textual descriptions about the current targets can be visualized by using **word cloud** in the style of **small multiples** to show their keywords, which are generated from online review. Besides, it is significant that the re-layout effects could be integrated into reality without confusion. In addition, users may be non-experts unfamiliar with programming or AR. The inaccurate recognition, rough interaction and tedious interface will add their cognitive burden in a messy situation. Therefore, *ARLayout* will provide an intuitive interface with concise interactions to avoid causing visual clutter and discomfort.

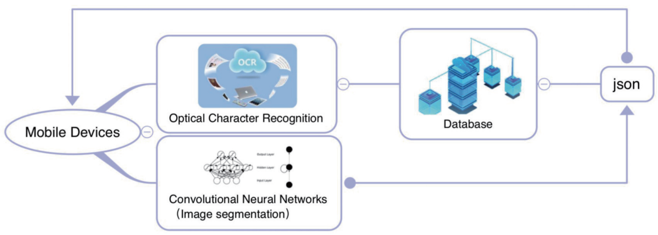


Fig. 2: The structural design of ARLayout, including simplified data processing, data transmission, and data presentation process.

3.3 Design Details: System Workflow Design

As shown in Fig. 2, *ARLayout* consists of two parts. The first part is the mobile client, which is used to take pictures or record a real-time video and then renders objects in AR. The second part is the server, which is employed to process almost all of the data. The overall processing are described as follows: The mobile client constantly takes pictures or record a real-time video of massive targets and sends them to the server. The remote server processes those pictures or key frames, recognizing target objects in them in real-time, and sends targets' data back to the mobile client, which displays them in new layouts. This workflow design has two considerations:

Separation of heavy computing and AR presentation: Unlike traditional applications, *ARLayout* shifts most of the computational intensive tasks to the server. The mobile client only needs to send the requests in multi-thread to ensure real-time target recognition. This enables *ARLayout* to handle a large amount of data without adding a heavy burden to the user's mobile device or influencing user's interaction experience. In the library/bookstore scenario, for example, more than **a thousand** books can be recognized in AR with panoramic pictures.

Separate processing of text and texture: The text and texture in one picture usually contains most of our desired information. We apply different neural network to process these two kinds of data. This makes our model not only suitable for situations where information is expressed more in text, such as library or cafe, but also for situations where texture contains more information, such as eyeshadows in cosmetics shops. For more implementation details, please refer to Section 5.

4 Evaluations

4.1 Case Scenarios

We choose three typical scenarios where people often experience confusion when finding or choosing targets in a messy situation: (1) Find ideal books in the library or book store. (2) Choose coffees that suits one's taste in a coffee shop (3) Choose a suitable eyeshadow palette among many palettes in a cosmetics shop.

We discuss the usage of searching, re-grouping, and re-ranking in the three common scenarios, as well as different dimensions concerning the three information categories mentioned in Sect. 3, as shown in Fig. 3.

In library scenario, the user can search books according to both nominal and quantitative attributes, e.g., searching for books written by a given author, or for books at a given price; The user can also re-group books by ordinal or quantitative attributes, e.g., grouped by

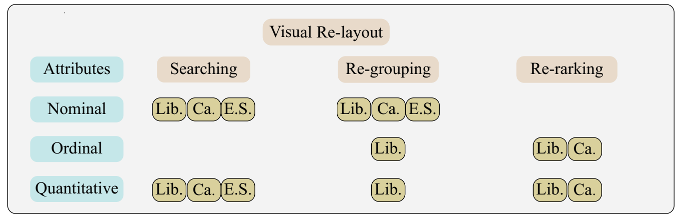


Fig. 3: Building re-layouts with different methods in three scenarios (library, coffee shop and eyeshadow). Targets in different scenarios have different kinds of attributes.

rating or by price intervals. Books can also be re-ranked by ordinal and quantitative attributes e.g., sorting books in ascending order based on the price. Besides, targets in coffee shop and eyeshadow scenarios have less information than the library scenario, so they doesn't involve quantitative or ordinal re-grouping. The eyeshadows cannot be re-ranked as well due to their abstract features.

To verify the usability and effectiveness of *ARLayout*, we design 5 tasks for participants to accomplish: two in the library, two in a simulated coffee shop, and one in a simulated cosmetics shop. The 5 tasks represent most of the possible situations happen in these 3 three places, where people may want to choose a suitable object in a complex scene. In accord with our 3 goals mentioned above, the study aimed to evaluate *ARLayout* regarding three aspects: (a) whether the re-layouting effects well satisfy users' personal requirements; (b) whether the extra information provided is useful and enough to help users; (c) whether *ARLayout* provides vivid and friendly AR interaction;

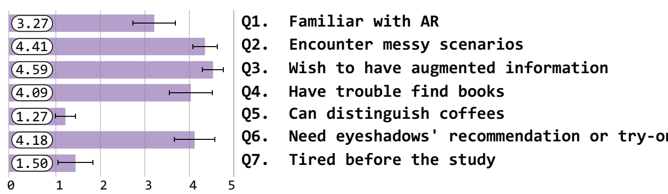


Fig. 4: Pre-study result: Most of participants have experienced these scenarios in their life and require assistance to better obtain and organize information.

4.2 Tasks

We use T*i*-*j* to name the j-th task that happens in the i-th scenario. The first two tasks are set in the library, i.e., **T1-1** and **T1-2**. The next two tasks take place in a simulated coffee shop, i.e., **2-1** and **T2-2**. The last task (**3-1**) take place in a simulated cosmetics shop.

T1-1 required participants to find an algorithm book in front of a bookshelf. The target book is published by Tsinghua University Press, and is suitable for code beginners. They are asked to find it twice (without and with the help of *ARLayout* respectively). For the first time, participants need to find the book one by one, and pick up certain books from the shelf to check the publisher and other details. Then they were required to use *ARLayout* to pick an satisfying book. This task simulates situations where the reader had a specific book to find, or determine a general theme and wants to pick one that suited himself. This task is designed to test whether *ARLayout*'s searching and re-grouping provides significant help in terms of time-saving and effectiveness (**G1,G2**).

T1-2 describes a situation that a casual reader walked in a library or a bookstore. He browses bookshelves and looked for interesting books that suit his taste. Before deciding which book to borrow or buy, he may compare the reviewers' rating and the price of multiple selected books. Participants are required to use *ARLayout* to filter, search, and compare books during the whole process. In this task, the participants are encouraged to freely generate customized layout in AR environment according to their preference. It measures *ARLayout*'s ability to fit various personal needs with customized re-group/re-rank criterions, as well as the helpfulness of the additional information (**G2,G3,G4**).

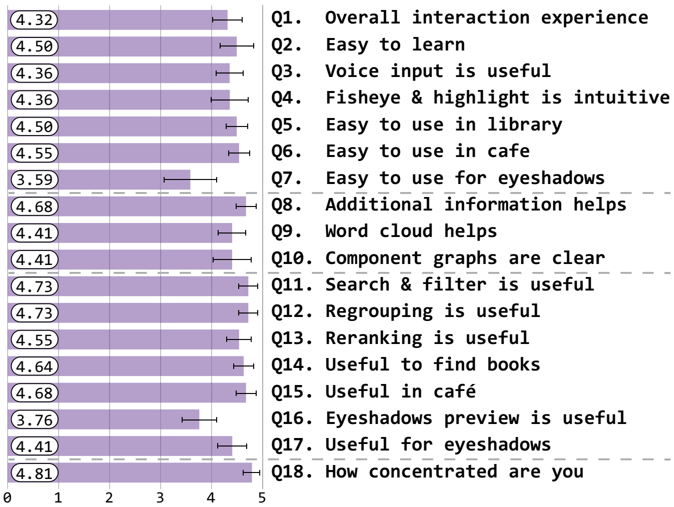


Fig. 5: Post-study result: most of participants react positively to *ARLayout*.

T2-1 requires participants to search for a certain kind of coffee they may heard of before by text or speech in *ARLayout*. They cqn also browse and select other coffees that were unfamiliar to them. They then compare and learn about those selected coffees' features and ingredients in the AR environment. This task tests the effectiveness of searching and highlighting, as well as whether the supplementary information is useful for customers to learn and choose coffees (**G1,G4**).

T2-2 requires participants to choose a coffee that suits their tastes with *ARLayout*, e.g., a coffee with moderate milk but no sugar, or with low calories. This task was similar with **T1-2** in the library, testing *ARLayout*'s capability of generating various customized layout with re-grouping and re-ranking (**G2,G3**).

T3-1 requires participants to browse and learn about several eyeshadow's features, as well as some tutorials with pictures in the AR environment. They can also choose recommended collocation of different colors, and preview the select scheme in a virtual 3-D model. This task tests the helpfulness of searching, re-grouping, and those supplementary information to potential buyers and makeup beginners (**G1,G2,G4**).

4.3 User Study

Questionnaire. The study contains 29 questions which are designed by using five-point Likert scale form, ranging from 1 ('Not at all') to 5 ('Very much'). Besides, there are four short-answers: the first two collects participants' time spent on finding one book with and without *ARLayout* respectively in **T1-1**. The last two record participants' most impressive task and their suggestions respectively.

Participants. There are 22 participants take part in the study (11 males; age: 19 to 23)According to the pre-study, which is illustrated in Fig. 4, many participants are not familiar with AR (Q1: $\mu = 3.27, 95\% CI = [2.80, 3.75]$). However, almost all participants encounter messy scenarios in daily lives (Q2: $\mu = 4.40, 95\% CI = [4.14, 4.68]$). They wish to refer to more information when selecting objects (Q3: $\mu = 4.59, 95\% CI = [4.35, 4.83]$). Specifically, most of them have trouble finding books in the library where books are sorted by index number (Q4: $\mu = 4.09, 95\% CI = [3.60, 4.58]$). Some can't distinguish several coffees clearly (Q5: $\mu = 1.27, 95\% CI = [1.05, 1.50]$). Most female participants wish to have real-time virtual makeup try-on and other people's recommandations. (Q6: $\mu = 4.18, 95\% CI = [3.72, 4.64]$).

Apparatus. Among the five tasks, **T1-1** and **T1-2** were run in the library, others were run in a lab with a poster (width: 1-2m; height: 2m) copied from a local coffee shop, and five different eyeshadows. An 11-inch iPad-Pro 2020 is also provided to participants.

Procedures. Participants are required to first fill in the pre-study questions. Before entering each scene (before **T1-1**, **T2-2** and **T3-1**), we spend 3-5 mins introducing different functions and re-layout effects, and show the operation with an actual example. Participants are then given tasks to complete. Finally, participants fill in the post-study questions.

4.4 User Study Results

The results are shown in Fig. 5. We analyse questionnaires from five aspects: usability, expressiveness, effectiveness, involvement and other suggestions.

Usability. As we designed *ARLayout* to help users reduce complexity, its usage should be simple enough in the first place. According to our study, most participants give high scores to the overall interaction (Q1: $\mu = 4.31, 95\% CI = [4.03, 4.61]$). In particular, in Q2 ($\mu = 4.50, 95\% CI = [4.17, 4.83]$), participants feel UI operation easy to understand, in Q3 ($\mu = 4.36, 95\% CI = [4.10, 4.63]$), voice input is recognized as helpful and efficient, and in Q4 ($\mu = 4.36, 95\% CI = [4.00, 4.73]$), fisheye deformation and result highlighting make the AR interactions clear and intuitive.

Since *ARLayout* has different functions and interactions in three scenarios, it receives high praise in library/bookstore tasks (Q5: $\mu = 4.50, 95\% CI = [4.29, 4.71]$) and coffee shop tasks (Q6: $\mu = 4.54, 95\% CI = [4.34, 4.75]$), while receives lower scores in the eyeshadow task (Q7: $\mu = 3.58, 95\% CI = [3.07, 4.11]$). When asked, participants say its' better to have eyeshadow video tutorials rather than just pictures.

Expressiveness. *ARLayout* provides augmented information for messy objects, so it's necessary that extra information and AR effects is expressive instead of adding to user's cognitive burden. According to Q8 ($\mu = 4.68, 95\% CI = [4.49, 4.88]$) and Q9 ($\mu = 4.40, 95\% CI = [4.14, 4.68]$), bar charts and word clouds concisely help participants obtain a general grasp of objects. Participants also rate that coffee component graphs quickly give them overall impressions about certain coffees in Q10 ($\mu = 4.41, 95\% CI = [4.04, 4.78]$).

Effectiveness. As for the most primary functions of *ARLayout*, participants respond positively and confirm the effectiveness of searching and filtering (Q11: $\mu = 4.73, 95\% CI = [,]$), re-grouping (Q12: $\mu = 4.73, 95\% CI = [4.54, 4.91]$) and re-ranking (Q13: $\mu = 4.55, 95\% CI = [4.30, 4.79]$).

More specifically, compared with finding books traditionally, time cost reduced from average 1.35 minutes to less than 20 seconds after using *ARLayout*. P2 who used to be a temporal librarian, spends 5 seconds traditionally (the fastest), and 12 seconds with *ARLayout*. We revisit him and he say "*ARLayout* is useful for the public, but there's room for improvement with UI tips".

According to Q15 ($\mu = 4.68, 95\% CI = [4.49, 4.88]$), most participants find the browsing and choosing coffee process helpful to them as they don't know much about coffee. P7 says "It helps me especially when I pay attention to fat intake".

In terms of eyeshadows, most participants find re-grouping by rating is effective, and were willing to buy or try the high-rating eyeshadows(Q17: $\mu = 4.41, 95\% CI = [4.13, 4.70]$).

Involvement. As indicated by Q18 ($\mu = 4.65, 95\% CI = [4.66, 4.98]$), almost all participants feel concentrated when doing tasks, and consider the process quite smooth and interesting.

Other Suggestions. In addition, constructive suggestions and other responses are collected in the study, which are mentioned below:

P6 notes: "The fish-eye effects of books make them overlapped and cluttered". Considering the density of books on the shelves in the library/bookstore, replacing the current effects with magnifying as well as pushing away nearby books may be a future improvement.

Besides, in the eyeshadow case, we find female eyeshadow users give lower scores in Q16($\mu = 3.45$) compared to male buyers ($\mu = 4.33$). This is probably because *ARLayout* recognizes eyeshadows merely by color, ignoring detailed texture and eyeshadows' brand which may be considered to be fair important factors. As male buyers just care about eyeshadow ratings, female users like P21 find it "not so useful as detailed texture effects can't be shown correctly on the preview 3-D model". We may consider advancing the recognizing algorithm in the future to distinguish among shimmery ones, matte ones and other kinds of eyeshadows.

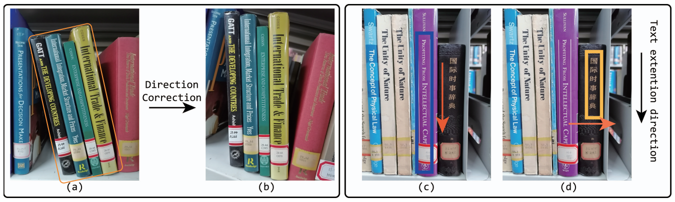


Fig. 6: The books in (a) is slanted, so the OCR module is checked for direction and corrected before it is actually called. The final result is figure (b). (c) and (d) show the difference in the arrangement of text between Chinese and English books. Artistic fonts exist in (e). (f) and (g) define the reference regions required for a custom template. Reference Fields are identified in (f) and Identification Areas are identified in (g).

5 Implementation and Performance

An implementation overview of the implementation workflow and the details are described in this section.

5.1 Database

We create a large database on the server for three application scenarios to serve these scenarios that require real-time information feedback [**?**, 48]. Considering the additional information, the tool can better achieve design goal G4, providing visual presentations or personalized recommendations by accessing the database in realtime.

5.2 Optical Character Recognition

In the actual applications, we found that the traditional OCR methods are not available for some complex scenarios, e.g. slanted texts (Fig. 6 (a)), different text alignments in different languages (Fig. 6 (c) and (d)). The problems are that the English book texts (Fig. 6 (c)) are often aligned in a vertical direction, while the Chinese book texts (Fig. 6 (d)) are often aligned in a horizontal direction.

In our experiments, we improve the traditional OCR method, following a language adaptive design [44], to achieve a large amount of the text characters over massive targets in reality. In the approach, the texts of a book can be calibrated first if they are slanted, and the language of a book will be identified first then the corresponding OCR directions will be estimated, which make the OCR recognition much more robust and scalable in the applications in reality.

5.3 Image Segmentation and Labelling

AR can effectively help users to migrate augmented information scenario to real-world scenarios for observation. Therefore, we use the method of image segmentation, to help us segment and label the components from the image, and then visualize the individual components in AR devices.

Convolutional Neural Network. To get a better result for various scenarios in real applications [70], we plan to use a convolutional neural network (CNN)-based approach to do image segmentation and labelling. We adopt an CNN-based open-source platform named PaddleSeg [45, 46] to do image segmentation and labelling, which just requires to manually annotate labels for a small amount of samples for training (less than 100 for each usage scenario in this paper). PaddleSeg is one of the state-of-the-art deep learning models for semantic image segmentation, where the goal is to assign semantic labels (e.g., person, dog, cat and so on) to every pixel in the input image. In PaddleSeg, DeepLab [**?**] is one of its key modules. Therefore, we take DeepLab as an example, illustrating how PaddleSeg is integrated into *ARLayout*, as shown in Fig. 7. The encoder module encodes multi-scale contextual information by applying atrous convolution at multiple scales, while the simple yet effective decoder module refines the segmentation results along object boundaries.

5.4 Coordinates Consistency between Virtuality and Reality

To guarantee that the positions of the targets in AR space are consistent with their real-world positions, we designed the coordinate transformation algorithm. On the client-side, pictures taken by users are sent back to the server for recognition. The server sends back JSON data indicating the 2-D coordinates of targets in each picture. If given an image with resolution of 3840*1880 as shown in Fig. 8 (c), a red book is recognized at (-640,1280) with width 192 pixels and height 1344 pixels. Whenever a picture is taken, we use ARKit to detect a possible virtual plane in front of the camera, get the current transformation matrix M_{plane}, and then compute the distance d between the virtual plane and the camera with LiDAR Scanner [4].

5.5 Small Multiples and Virtual Models

Regarding the visualizations for augmented information, the related data will be sent to the server and the client will receive the processed data from the server. We design some visual presentation components like bar charts, line charts, word cloud and the ingredients graphs, etc., which can be selected and composed for different usage scenarios.

We also create a virtual translucent screen in the AR environment to show those augmented information. Besides, in the eyeshadow scenario, we create a virtual head model, stylize the model's eyes with the selected eyeshadow color to show the 3-D preview presentations for users to help users get a better fitting.

6 Usage Scenarios

To illustrate how *ARLayout* searches, re-groups and re-ranks massive physical targets in AR environments, we describe three usage scenarios where users use *ARLayout* to search and filter targets, obtain their information, and locate ideal ones among massive targets, i.e., the usage scenarios in a library/bookstore, a coffee shop and an eyeshadow fitting.

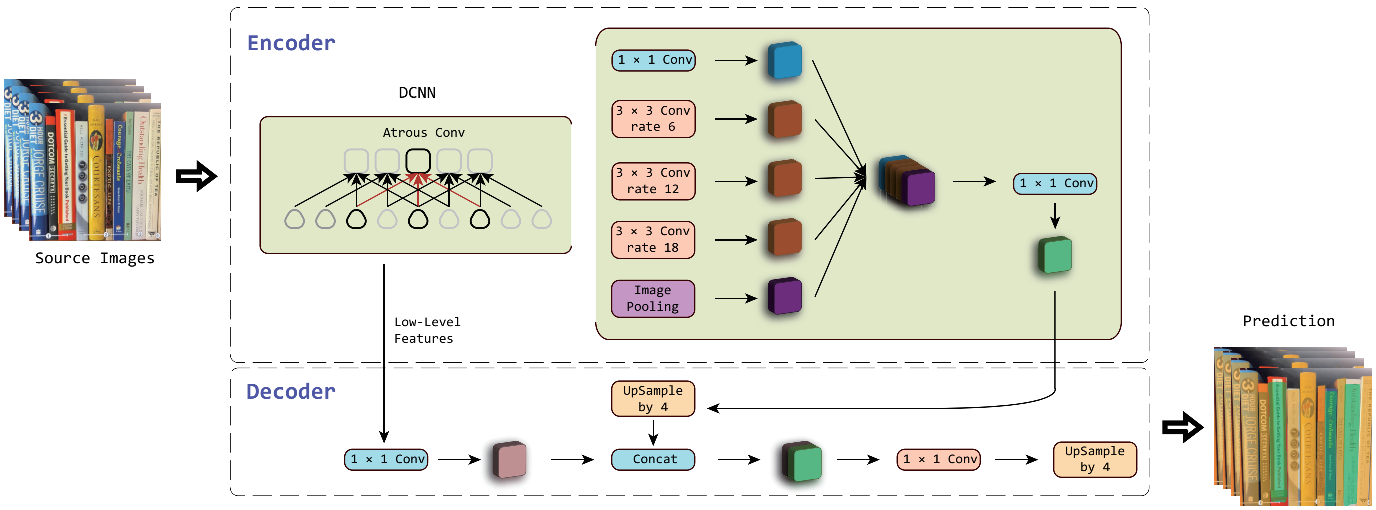
6.1 Library/Bookstore Scenario

Suppose Zelda is a student major in economy. She wants to buy some economic books to broaden her horizons. She prefers books from the "University of Chicago Press", which is recognized as having been publishing high-quality books. She comes to the social science area in a library/bookstore, facing a large row of bookshelves with around 1000 books. Instead of looking for books in the bookstore's searching system (which is often just open for bookstore clerks), by which she may have dizzy and tedious choosing experience.

(1) **Fuzzy searching/filtering**: she decides to quickly scan the whole bookshelves by the panoramic camera by *ARLayout*. Soon 778 books are recognized in total. She then filters unrelated books by saying "economic". *ARLayout* deals with the voice input, and filters those books by fuzzy search. Seeing that only 118 economic books remains, Zelda chooses to visualize those books in the AR space and browses them as shown in Fig. 9 (b). She finds that only one book nearby is from "University of Chicago Press", then she want to find more books on "economic' and published by "University of Chicago Press".

(2) **Re-grouping**: she re-group those 118 books by publisher and searches by saying "Chicago" or inputting by the virtual keyboard of her mobile phone. This time, seven books from the "University of Chicago Press" are highlighted and placed on a bookshelf in front of her with fisheye effect (Fig. 9 (c)). Books from other presses are also grouped and placed on the other layers of the shelf (Fig. 9 (d)), so she choose a book from them.

(3) **Fuzzy re-searching**: she wants to search all books with fuzzy keyword "social" , there are 21 books highlighted in red (Fig. 9 (d)). She uses fisheye effect to view each book's details like author and reviewers' words (Fig. 9 (c)). But she finds these social books are not highly rated or the authors are not in her favorite author list, So she changes idea: she re-ranks those books by ratings or further filter them by author names.




```
Image caption:
DeepLab is a state-of-art deep learning model for semantic image segmentation,
where the goal is to assign semantic labels (e.g., person, dog, cat and so on)
to every pixel in the input image.
The above image describe the work flow of DeepLabv3+.~\cite{deeplabv3plus2018}
DeepLabv3+ extends DeepLabv3 by employing a encoderdecoder structure.
The encoder module encodes multi-scale contextual information by
applying atrous convolution at multiple scales, while the simple yet effective decoder
module refines the segmentation results along object boundaries.

cite:
@inproceedings{deeplabv3plus2018,
  title={Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation},
  author={Liang-Chieh Chen and Yukun Zhu and George Papandreou and Florian Schroff and Hartwig Adam},
  booktitle={ECCV},
  year={2018}
}

paper url:
https://arxiv.org/pdf/1802.02611.pdf
```

Fig. 7: The network illustration about how PaddleSeg is integrated into *ARLayout*. We take DeepLab as an example, one of the key modules of PaddleSeg. The encoder module encodes multi-scale contextual information by applying atrous convolution at multiple scales, while the simple yet effective decoder module refines the segmentation results along object boundaries.

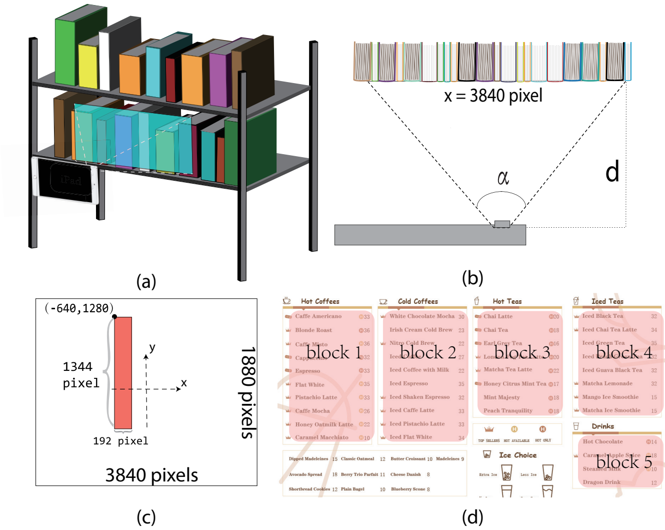


Fig. 8: (a) shows a mobile device is scanning the bookshelves. (b) is the top view of (a) showing the horizontal view angle α, where d is the distance between the mobile device and the target books estimated by *ARLayout*, and x is the number of pixels horizontally. (c) shows a segmented book sample with the extracted position and size. (d) there are five blocks with recognized coffee names on a coffee menu.

(4) **Re-ranking**: she sort all of the books which are placed from left to right on the same layer of the shelf by descending order (Fig. 9 (e)). Then she selects four books that seems suitable, those selected books are moved to a lower layer of the virtual bookshelf where are designed to place the candidate books (Fig. 9 (f)), just like a shopping cart.

(5) **Comparing by word cloud visualizations in small multiples**: she views and compares the word cloud of each book's keywords. Among those four books, one book has keywords "story" and "understand", other books' keywords are "city", "environmentalist" and "empire" (Fig. 9 (g)). Zelda is interested in the "story" one and the "empire" one, but she's also concerned with the prices if she is going to buy in a bookstore.

(6) **Comparing with kinds of diagrams in small multiples**: so she compares these books' ratings and prices together in the bar chart (Fig. 9 (h)). She finds that the "story" (the first) rated as high as the "empire" one (the fourth), but is a little cheaper than the "empire" one. So she choose the "story" one and restores those books to their original layout.

(7) **Title Precise Searching**: finally, she voice input the title "human resources management". The book is magnified and highlighted on the left upper side (Fig. 9 (i)) by flashing. She walks by and locates the book in the reality space according to its position shown in the screen (Fig. 9 (j)).

6.2 Coffee Shop Scenario

A new coffee shop opens in Zelda's campus. she don't know much about coffee, but is willing to try some in the new coffee shop. She walks in the coffee shop, takes a picture of the coffee menu by *ARLayout*. Soon it recognizes 40 different drinks, and creates a menu upon the original one.

She remembers that she once ordered a cup of espresso, which she thinks is rather bitter, so she wants to see the ingredients. So she firstly voice inputs "Espresso" and finds that it's highlighted in the "Hot Coffees" group(Fig. 10 (b)). She views the details of Espresso, learns that it's surely bitter, as no sugar is added in it (Fig. 10 (c)).

She then re-groups those drinks according to milk or sugar (Fig. 10 (d-e)). She browses and selects some drinks with high ratings in the "medium sweet" and "sweet" group, as shown in Fig. 10 (f). Then she compares those drinks' ingredients in small multiples, and finds that Cappuccino has a balance between sugar, milk and caffine, which may suit to her taste, as shown in Fig. 10 (g). However, her fitness coach's advice cross her mind that she needs to limit the calorie intake to 1300 calories everyday, while the coffee summary shows that Cappuccino has 140 calories per cup. So she re-ranks all the drinks by calorie content. This time, coffees are sorted from left to right by calorie, as shown in Fig. 10 (h). She begins browsing on the right side, where coffees with relatively low calorie are located. She find some of coffees that she haven't drunk. To have a quick grasp of them, she views their word cloud (Fig. 10 (i)). She learns that Blonde Roast is regarded to be "mellow" in the word cloud, Iced Coffee is "rich", and Caffee Americano has keyword "espresso", which may be too bitter for her. She browses Blonde Roast's summary, which confirms that it only contains five calories per cup (Fig. 10 (j)). Finally, she chooses Blonde Roast and enjoys its "soft and mellow flavor" described in the summary.

In addition, *ARLayout* can also handle larger menus like the poster hanging on the wall outside the coffee shop, as shown in Fig. 11 (a). The user can view details var fisheye, re-group, re-rank, select and compare drinks on the poster, which is similar to the smaller menu, as shown in Fig. 11 (b-d).

6.3 Eyeshadow Scenario

One day Zelda is shopping in a cosmetics shop. She is not good at making up, especially eyeshadow, because different eyeshadows may have unique effects, and sometimes several eyeshadows may be applied to different places to form a colour combination. So she uses *ARLayout* and scans those eyeshadows displayed on the desk, as shown in Fig. 12 (a). Soon 15 different eyeshadows with 96 different colors (or textures) are recognized. She then re-groups them by eyetypes, and use fisheye effect to view the details of the "protruding eye" group, as shown in Fig. 12 (b). A graph pops up on the side of the selected eyeshadow, showing the ideal position for users to apply it on. Zelda chooses a kind of golden brown eyeshadow, and previews its 3-D effect on a virtual model, as shown in Fig. 12 (c). Zelda still finds it hard to choose several eyeshadows that matches each other, so she re-groups them by high rated schemes, this time, three recommended color schemes are lined up in front of her, as shown in Fig. 12 (d). Zelda views the details about each eyeshadow's effects and features, learns that eyeshadows in "Scheme 13" is suitable for simple day look. So she restores those colors to their original layout and search for "Scheme 13" by voice. Those eyeshadows in "Scheme 13" are flashing in red, as shown in Fig. 12 (e). As a result, she chooses an eyeshadow palette that contains some colors in "Scheme 13".

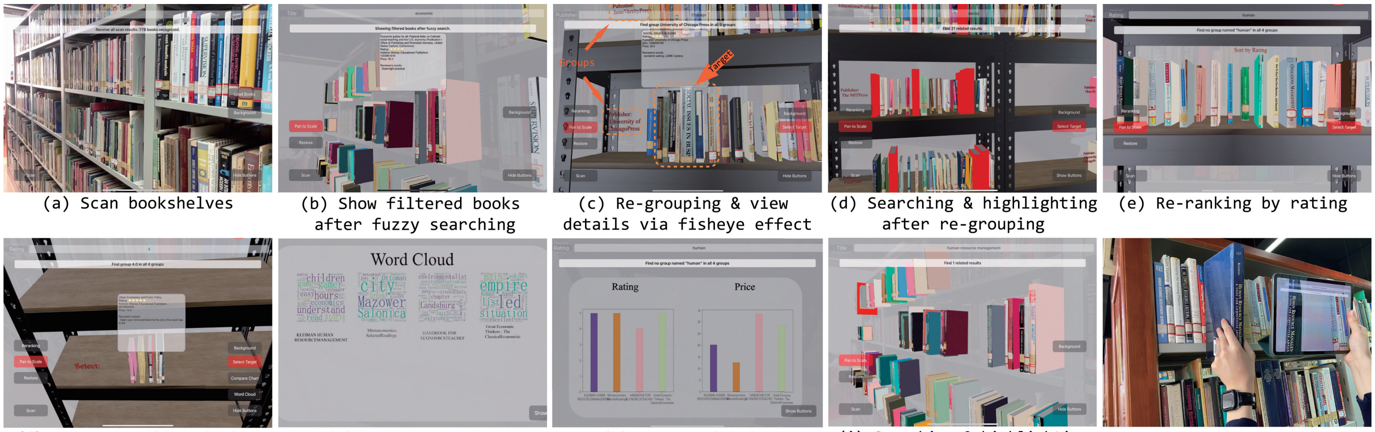
7 Discussion and Future Work

We discuss the limitations and some potential extension work of this paper in this section.

We design a personal visualization tool named *ARLayout* for non-experts without programming knowledge to build AR-based visual re-layouts towards massive physical targets. We summarize the salabilities and some limitations of *ARLayout* as follows:

The scalability of application scenarios. Except the usage scenarios presented in the paper, the current version of *ARLayout* supports various application scenarios where: (1) the targets with textual information such as menus of coffee, drinks, foods, etc., goods in supermarkets with name-price labels, or even a large number of cars in a parking lot (license plates), etc. (2) the targets with colors such as eyeshadow, colored balls or balloons in a large amusement park, the colored goods in supermarkets, etc. (3) the targets with 2-D shapes and some simple 3-D shapes like tables, chairs, windows, etc., however, it is not applicable for complex 3-D shapes due to the limitations of image labelling algorithms towards various 3-D objects.

Rendering limitation of AR devices. We used the latest 11-inch 2021 iPad-Pro with A12Z processor, which is one of the top end configuration for the nowadays iPad with ARKit. The framework could visualize the AR targets (actually are virtual targets) in AR environment well when the number is below 200. However, as the target number increases, the frame rate drops. Though several experiments, we found that the frame rate stabilizes at 60 per second when the number of targets (e.g., books in the library scenario) is below 200, and decreases



(a) Scan bookshelves (b) Show filtered books after fuzzy searching (c) Re-grouping & view details via fisheye effect (d) Searching & highlighting after re-grouping (e) Re-ranking by rating

(f) Select candidate books (g) Augmented info: word cloud (h) Augmented info: bar chart of rating & price (i) Searching & highlighting in originial layout (j) Find the target book

Fig. 9: The user builds re-layouts and find books in a library. (a): The user scans the original bookshelves, 778 books recognized. (b): *ARLayout* visualize the remaining books after fuzzy searching "econimic". (c): The user chooses to re-group by publisher, and searches "Chicago". Books from "University of Chicago Press" and other presses are placed on different layers. The user browses those books with fisheye effect. (d): The user searches "social" in books' names, several books are highlighted in red. (e): The user re-ranks those books by ratings. Books sorted in descending order are placed from the left to the right. (f): The user selects several candidate books, they are moved to a layer below. (g): *ARLayout* shows books' word clouds. (h): The user compares books' ratings and prices via bar chart. (i): The user restores books to their original layout, and searches a book's name. The target book is highlighted in red. (j): The user finds and picks the target book according to its loation on the screen.

to 30 frames per second if it exceeds 230. Furthermore, if AR books' number increases to 320, the frame rate will drop significantly to below 15.

Nevertheless, this limitation (low FPS when the target number grows larger than 320) mainly restricted by the current mobile device would **not appear** if users follow a rountine task workflow, i.e., searching the targets first to narrow down the number of candidates to 320 or less. The more unrelated targets are filtered out, the more efficient and the more effective the users can fulfill their tasks. For example, users can search books with a fuzzy keyword "data structure" to decrease the data space significantly, and then the subsequent tasks including re-grouping, re-ranking or further searching will be visualized fluently enough in AR environment. The major issue of this limitation is that the AR device can not show too many AR objects (e.g., larger than 320).

It is worth noting that the image segmentation and labelling components of *ARLayout* are **scalable** and **not limited** by the target number, because the convolutional neural network and the OCR algorithm are run on the server which can even handle thousands of books in the library usage scenario in our experiments. More importantly, unlike the mobile device, the computation resources of the server are **scalable enough** and which could be upgraded easily. As a result, while *ARLayout* recognizes almost all the books scanned by the user, we recommend the user to first filter out unrelated books by fuzzy searching before actually visualizing those books in the AR space in order to narrow down the data space.

Image resolution limitation. *ARLayout* recognizes objects by images taken from the mobile devices. Ideally, the user only need to take one panoramic picture that contains all the target objects. However, targets' details may not be recognized if they are too small in the picture, that is the user is stand too far away from the massive targets. For example, in the library/bookstore scenario, instead of scanning all layers of the bookshelves, the user may walk closer to the bookshelves and scan one layer at one time by panoramic stitching due to lack of light or limited imaging quality.

We plan to optimize the following aspects of *ARLayout* in future work:

More complex application scenarios support. Currently *ARLayout* is capable of recognizing books, coffee names, drinking menu, food menu, or eyeshadows. In the future, we plan to expand its usage scenarios to other similar ones like choosing cups, fruits or flowers, etc. Because targets with text on it, or in different colors and shapes can be well recognized by trained neural networks. During targets recognition, targets' additional information is acquired from the database, so an interface for fast data import can be implemented in future work to meet the data requirements of more scenarios.

Detailed Texture Recognition in AR. Some of user study participants consider it could be better to use the same texture to show the AR effects, especially in the eyeshadow scenario. At present, the preview effect of eyeshadow only stays at the transfer of RGB value. Further study with more focus on detailed texture recognition is therefore suggested, e.g., the eyeshadow texture on glossiness or oiliness, etc. A Deep Texture Encoding Network (DeepTEN) with a novel Encoding Layer integrated on top of convolutional layers offers possibilities to solve this problem [68]. Besides, we are trying to use a better architecture, such as abandoning the server if the mobile device is affordable and building a lightweight neural network to move them to run on the mobile devices.

8 Conclusion

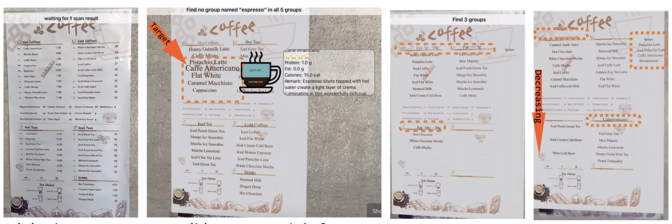
In this paper, we present *ARLayout*, a personal visualization tool designed for non-experts to build AR-based visual re-layouts towards massive physical targets, such as books in a library/bookstore, coffees in a coffee shop, eyeshadows in a shop, etc. The real-time video are captured from the reality world by the camera of mobile devices. All the candidate targets can be segmented and labeled by a convolutional neural network PaddleSeg [45,46]. The network just requires a small scale of labeled training samples, i.e., less than 100 for each case in our experiments, while the textual information is recognized by an OCR algorithm.

The visual re-layouts of the physical targets include (1) highlighting the search results by AR-based fisheye deformation, (2) re-grouping them according to their additional nominal, ordinal or quantitative attributes, and (3) re-ranking them according to their additional ordinal or quantitative attributes. In the searching scenario, the fuzzy keywords are input by voice to narrow down the number of candidate targets, and then the search results will be highlighted by flashing, transparency or AR-based fisheye deformation in the AR environment to guide them where to find the targets in the reality world. In the re-grouping task, candidate targets can be re-grouped in the AR environment according to one or multiple of their nominal, ordinal or quantitative attributes. In the re-ranking task, candidate targets can be sorted in the AR en-

(a) The original coffee menu (b) Searching & highlighting (c) Augmented info: fisheye & details (d) Re-grouping by milk (e) Re-grouping by sugar

(f) Select coffees (g) Augmented info: ingredients small multiples (h) Re-ranking by calories (i) Augmented info: word cloud (j) Browsing after re-ranking

Fig. 10: The user buinds re-layouts for a coffee menu. (a): The user scans the coffee menu(b): The user search "espresso". Three related coffees are highlighted.(c): The user browses the menu with fisheye. The focused coffee will be magnified, with its summary shown besides it. (d): The user re-groupps all the drinks by sugar. (e): The user selects four candidate coffees. They are moved to the right side of the menu(f): The user compares candidate coffees by their ingredients graphs in small multiples. (g): The user re-groupps drinks by fat. (h): The user re-ranks drinks by calories. Drinks with more calories are moved to the left side. (i): The user compares the word cloud of the candidate coffees. (j): The user browses drinks on the right side of the menu to choose one with less calories.



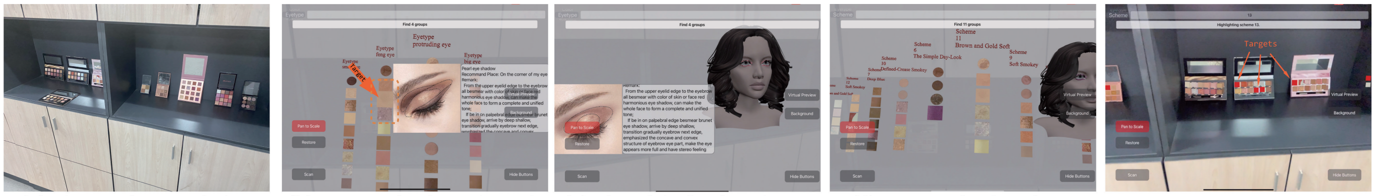
(a) The poster on the wall (b) Augmented info: fisheye & details (c) Re-grouping by fat (d): Re-ranking & selecting

Fig. 11: (a): The orignial poster hanging on the wall outside a coffee shop. (b): The user browses coffees' details with fisheye effect. (c): The user re-groups drinks by fat content intervals. (d): The user re-ranks drinks by calorie content.

vironment according to one of their ordinal or quantitative attributes. In the experiments, we demonstrate the usability, expressiveness and effectiveness of *ARLayout* by a user study and three case studies.

References

[1] `https://aframe.io/`.
[2] `https://webvr.info/`.
[3] `https://www.interaction-design.org/literature/book/the-glossary-of-human-computer-interaction/wimp`.
[4] `https://developer.apple.com/cn/augmented-reality/arkit/`.
[5] Arcore. `https://developers.google.cn/ar/`.
[6] Y. A., G. LT., and H. JL. Development of sensitivity to information provided by cast shadows in pictures. *Perception*, 7(3):333–341, 1978.
[7] D. Amin and S. Govilkar. Comparative study of augmented reality sdks. *International Journal on Computational Science & Applications*, 5(1):11–26, 2015.
[8] S. Butscher, S. Hubenschmid, J. Müller, J. Fuchs, and H. Reiterer. Clusters, trends, and outliers. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, apr 2018. doi: 10.1145/3173574.3173664
[9] C. C. Carrera, J. L. S. Perez, and J. de la Torre Cantero. Teaching with ar as a tool for relief visualization: usability and motivation study. *International Research in Geographical and Environmental Education*, 27(1):69–84, 2018. doi: 10.1080/10382046.2017.1285135
[10] Z. Chen, Y. Su, Y. Wang, Q. Wang, H. Qu, and Y. Wu. Marvist: Authoring glyph-based visualization in mobile augmented reality. *IEEE Transactions on Visualization and Computer Graphics*, 26(8):2645–2658, 2020. doi: 10.1109/TVCG.2019.2892415
[11] Z. Chen, W. Tong, Q. Wang, B. Bach, and H. Qu. Augmenting static visualizations with paparvis designer. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2020.
[12] M.-C. Chiu, S.-P. Chang, Y.-C. Chang, H.-H. Chu, C.-H. C. Cheryl, F.-H. Hsiao, and J.-C. Ko. Playful bottle: a mobile social persuasion system to motivate healthy water intake. In *Proceedings of the 11th international conference on Ubiquitous computing*, pp. 185–194, 2009.
[13] F. Chris, A. Ronald, and D. Mike. Augmented-reality visualizations guided by cognition: Perceptual heuristics for combining visible and obscured information. In *Proceedings. International Symposium on Mixed and Augmented Reality*, pp. 215–320. IEEE, 2002.
[14] B. Christoph, S. Tobias, H. S. M., and N. Nassir. Improving depth perception in medical ar. In *Bildverarbeitung für die Medizin 2007*, pp. 217–221. Springer, 2007.
[15] M. Cordeil, A. Cunningham, B. Bach, C. Hurter, B. H. Thomas, K. Marriott, and T. Dwyer. Iatk: An immersive analytics toolkit. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 200–209, 2019. doi: 10.1109/VR.2019.8797978
[16] D. Cushnan and H. E. Habbak. *Developing ar games for ios and android*. Packt Publishing Ltd, 2013.
[17] M. Erick and S. Dieter. Importance masks for revealing occluded objects in augmented reality. In *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*, pp. 247–248, 2009.
[18] G. Evans, J. Miller, M. I. Pena, A. MacAllister, and E. Winer. Evaluating the microsoft hololens through an augmented reality assembly application. In *Degraded environments: sensing, processing, and display 2017*, vol. 10197, p. 101970V. International Society for Optics and Photonics, 2017.
[19] B. J. Fogg. Persuasive computers: perspectives and research directions. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 225–232, Jan. 1998.
[20] V. Geroimenko. Augmented reality technology and art: The analysis and visualization of evolving conceptual models. In *2012 16th International Conference on Information Visualisation*, pp. 445–453, 2012. doi: 10.1109/IV.2012.77
[21] C.-L. Goh and S. Nakatake. A sensor-based data visualization system for training blood pressure measurement by auscultatory method. *IEICE TRANSACTIONS on Information and Systems*, 99(4):936–943, 2016.
[22] D. Herr, J. Reinhardt, R. Krüger, G. Reina, and T. Ertl. Immersive visual analytics for modular factory layout planning. In *Proc. IEEE VIS Workshop Immersive Analytics*, 2017.
[23] D. Huang. *Visualizing personal data in context: an on-calendar design strategy for behaviour feedback*. PhD thesis, 2016.
[24] D. Huang, M. Tory, B. A. Aseniero, L. Bartram, S. Bateman, S. Carpendale,



(a): Scan various eyeshadows (b): Re-grouping by eyetypes & fisheye & details (c): 3-D virtual makeup try-on (d): Re-grouping by schemes (e): Searching & highlighting

Fig. 12: The user builds re-layouts for eyeshadows. (a): The user scans the various eyeshadows displayed on a cosmetics shop's desk. (b): The user re-groupps those eyeshadows by eyetypes, and views certain eyeshadow's details as well as a graph showing places to apply it on. (c): The user views the candidate eyeshadow's 3-D virtual makeup try-on. (d): The user re-groupps eyeshadows by high-rated scheme. Different schemes have features like "Deep Blue" or "Soft Smokey". (e): The user searches "Scheme 13" in their original layout, eyeshaodws contained in this scheme are highlighted.

A. Tang, and R. Woodbury. Personal visualization and personal visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 21(3):420–433, 2014.

[25] L. Ian, D. Anind, and F. Jodi. A stage-based model of personal informatics systems. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 557–566, 2010.

[26] L. Ian, F. Jodi, and D. Anind. Know thyself: monitoring and reflecting on facets of one's life. In *CHI'10 Extended Abstracts on Human Factors in Computing Systems*, pp. 4489–4492, 2010.

[27] D. A. J. Real-time simultaneous localisation and mapping with a single camera. In *Computer Vision, IEEE International Conference on*, vol. 3, pp. 1403–1403. IEEE Computer Society, Nice, France, 2003.

[28] D. A. J., R. I. D., M. N. D., and S. Olivier. Monoslam: Real-time single camera slam. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1052–1067, 2007.

[29] D. M. J. *Information visualization for the people*. PhD thesis, Massachusetts Institute of Technology, Department of Comparative Media Studies, 2008.

[30] K. J., M. B. Skov, P. Jeni, and P. Rahuvaran. Using mobile phones to support sustainability: a field study of residential electricity consumption. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2347–2356, 2012.

[31] Y. Jansen, P. Dragicevic, P. Isenberg, J. Alexander, A. Karnik, J. Kildal, S. Subramanian, and K. Hornbæk. Opportunities and challenges for data physicalization. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 3227–3236, 2015.

[32] Y. Jansen and P. D. J.-D. Fekete. Evaluating the efficiency of physical visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2593–2602, 2013.

[33] Y. Jansen and K. Hornbæk. A psychophysical investigation of size as a physical variable. *IEEE transactions on visualization and computer graphics*, 22(1):479–488, 2015.

[34] W. Jason and H. Tobias. Pictorial depth cues for outdoor augmented reality. In *Ninth IEEE International Symposium on Wearable Computers (ISWC'05)*, pp. 92–99. IEEE, 2005.

[35] H. K. Artoolkit: Library for vision-based augmented reality. *Technical Report of Ieice Prmu*, 101:79–86, 2002.

[36] L. Keselman, J. I. Woodfill, A. Grunnet-Jepsen, and A. Bhowmik. Intel realsense stereoscopic depth cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–10, 2017.

[37] R. A. Khot, L. Hjorth, and F. Mueller. Understanding physical activity through 3d printed material artifacts. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 3835–3844, 2014.

[38] B. Lee, D. Brown, B. Lee, C. Hurter, S. Drucker, and T. Dwyer. Data visceralization: Enabling deeper understanding of data using virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1095–1105, feb 2021. doi: 10.1109/tvcg.2020.3030435

[39] B. Lee, X. Hu, M. Cordeil, A. Prouzeau, B. Jenny, and T. Dwyer. Shared surfaces and spaces: Collaborative data visualisation in a co-located immersive environment. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1171–1181, feb 2021. doi: 10.1109/tvcg.2020.3030450

[40] G. A. Lee, A. Dünser, Seungwon Kim, and M. Billinghurst. Cityviewar: A mobile outdoor ar application for city visualization. In *2012 IEEE International Symposium on Mixed and Augmented Reality - Arts, Media, and Humanities (ISMAR-AMH)*, pp. 57–64, 2012. doi: 10.1109/ISMAR-AMH.2012.6483989

[41] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale. Keyframe-based visual–inertial odometry using nonlinear optimization. *The International Journal of Robotics Research*, 34(3):314–334, 2015.

[42] D. Li, H. Mei, Y. Shen, S. Su, W. Zhang, J. Wang, M. Zu, and W. Chen. Echarts: a declarative framework for rapid construction of web-based visualization. *Visual Informatics*, 2(2):136–146, 2018.

[43] J. Liberty, J. Gallowayy, P. Japikse, and J. Hartwell. *Building Your First Windows 10 UWP App*. Windows 10 Development with XAML and C# 7, 2018.

[44] O. Y. Ling, L. B. Theng, A. Chai, and C. McCarthy. A model for automatic recognition of vertical texts in natural scene images. In *2018 8th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, pp. 170–175, 2018. doi: 10.1109/ICCSCE.2018.8685019

[45] Y. Liu, L. Chu, G. Chen, Z. Wu, Z. Chen, B. Lai, and Y. Hao. Paddleseg, end-to-end image segmentation kit based on paddlepaddle. `https://github.com/PaddlePaddle/PaddleSeg`, 2019.

[46] Y. Liu, L. Chu, G. Chen, Z. Wu, Z. Chen, B. Lai, and Y. Hao. Paddleseg: A high-efficient development toolkit for image segmentation, 2021.

[47] Marder-Eppstein and Eitan. Project tango. In *ACM SIGGRAPH 2016 Real-Time Live!*, pp. 25–25. 2016.

[48] J. McAuley, C. Targett, Q. Shi, and A. van den Hengel. Image-based recommendations on styles and substitutes. *ACM*, pp. 43–52, 2015.

[49] P. Monigatti, M. Apperley, and B. Rogers. Power and energy visualization for the micro-management of household electricity consumption. In *Proceedings of the International Conference on Advanced Visual Interfaces*, pp. 325–328. ACM, New York, USA, 2010.

[50] A. B. P. Shadows as sources of cues for distance of shadow-casting objects. *Perceptual and motor skills*, 89(2):571–584, 1999.

[51] T. Parisi. *WebGL: up and running*. " O'Reilly Media, Inc.", 2012.

[52] M. Pascal, K. D. C., and K. Daniel. The perception of cast shadows. *Trends in cognitive sciences*, 2(8):288–295, 1998.

[53] A. Pedretti, L. Villa, and G. Vistoli. Vega: a versatile program to convert, handle and visualize molecular structure on windows-based pcs. *Journal of Molecular Graphics and Modelling*, 21(1):47–49, 2002.

[54] Z. Pousman, J. Stasko, and M. Mateas. Casual information visualization: Depictions of data in everyday life. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1145–1152, 2007. doi: 10.1109/TVCG.2007.70541

[55] P. Reipschlager, T. Flemisch, and R. Dachselt. Personal augmented reality for information visualization on large interactive displays. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1182–1192, Feb 2021. doi: 10.1109/tvcg.2020.3030460

[56] D. Ren, T. Höllerer, and X. Yuan. ivisdesigner: Expressive interactive design of information visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2092–2101, 2014. doi: 10.1109/TVCG.2014.2346291

[57] T. Rhee, S. Thompson, D. Medeiros, R. dos Anjos, and A. Chalmers. Augmented virtual teleportation for high-fidelity telecollaboration. *IEEE Transactions on Visualization and Computer Graphics*, 26(5):1923–1933, 2020. doi: 10.1109/TVCG.2020.2973065

[58] J. E. S. and S. Soatto. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *The International Journal of Robotics Research*, 30(4):407–430, 2011.

[59] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-lite: A grammar of interactive graphics. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):341–350, 2017. doi: 10.1109/TVCG.2016.2599030

[60] D. Shreiner, G. Sellers, J. Kessenich, and B. Licea-Kane. *OpenGL programming guide: The Official guide to learning OpenGL, version 4.3*. Addison-Wesley, 2013.

[61] R. Sicat, J. Li, J. Choi, M. Cordeil, W.-K. Jeong, B. Bach, and H. Pfister. Dxr: A toolkit for building immersive data visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):715–725, 2019. doi: 10.1109/TVCG.2018.2865152

[62] S. Stusak, A. Tabard, F. Sauka, R. A. Khot, and A. Butz. Activity sculptures: Exploring the impact of physical visualizations on running activity. *IEEE transactions on visualization and computer graphics*, 20(12):2201–2210, 2014.

[63] K. Sunyoung and P. Eric. Inair: sharing indoor air quality measurements and visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1861–1870, 2010.

[64] T. Takahiro, Y. Haruyoshi, K. Yoshinari, and O. Yuichi. Visualization methods for outdoor see-through vision. *IEICE transactions on information and systems*, 89(6):1781–1789, 2006.

[65] D. Wagner and D. Schmalstieg. Artoolkit on the pocketpc platform. In *2003 IEEE International Augmented Reality Toolkit Workshop*, pp. 14–15, 2003. doi: 10.1109/ART.2003.1320418

[66] W. Willett, Y. Jansen, and P. Dragicevic. Embedded data representations. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):461–470, jan 2017. doi: 10.1109/tvcg.2016.2598608

[67] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):649–658, 2016. doi: 10.1109/TVCG.2015.2467191

[68] H. Zhang, J. Xue, and K. Dana. Deep ten: Texture encoding network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2896–2905, 2017. doi: 10.1109/CVPR.2017.309

[69] N. Q. Zhu. *Data visualization with D3. js cookbook*. Packt Publishing Ltd, 2013.

[70] Q. Zhu, B. Du, and P. Yan. Boundary-weighted domain adaptive neural network for prostate mr image segmentation. *IEEE Transactions on Medical Imaging*, 39(3):753–763, 2020. doi: 10.1109/TMI.2019.2935018