2-2017

Empath-D: Empathetic design for accessibility

Kenny Tsu Wei CHOO
Singapore Management University, kenny.choo.2012@phdis.smu.edu.sg

Rajesh Krishna BALAN
Singapore Management University, rajesh@smu.edu.sg

Kiat Wee TAN
Singapore Management University, williamtan@smu.edu.sg

Archan MISRA
Singapore Management University, archanm@smu.edu.sg

Youngki LEE
Singapore Management University, YOUNGKILEE@smu.edu.sg

Citation
CHOO, Kenny Tsu Wei; BALAN, Rajesh Krishna; TAN, Kiat Wee; MISRA, Archan; and LEE, Youngki. Empath-D: Empathetic design for accessibility. (2017). *HotMobile '17: Proceedings of the 18th International Workshop on Mobile Computing Systems and Applications, Sonoma, CA, USA, 2017 February 21-22*. 55-60.
Available at: https://ink.library.smu.edu.sg/sis_research/3634

Empath-D: Empathetic Design for Accessibility

Kenny Tsu Wei Choo, Rajesh Krishna Balan, Tan Kiat Wee,
Jagmohan Chauhan, Archan Misra, Youngki Lee
Singapore Management University

ABSTRACT

We describe our vision for *Empath-D*, our system to enable *Empathetic User Interface Design*. Our key idea is to leverage Virtual and Augmented Reality (VR / AR) displays to provide an *Immersive Reality* environment, where developers/designers can emulate impaired interactions by elderly or disabled users while testing the usability of their applications. Our early experiences with the *Empath-D* prototype show that *Empath-D* can emulate a cataract vision impairment of the elderly and guide designers to create accessible web pages with less mental workload.

CCS Concepts

•**Human-centered computing → Systems and tools for interaction design; Accessibility design and evaluation methods;**

Keywords

Accessibility design; empathetic design; virtual reality; augmented reality.

1. INTRODUCTION

It has been said that *You can't really understand another person's experience until you've walked a mile in their shoes*. This is especially true when designing applications meant to be used by the elderly or disabled who have visual, auditory, or motor limitations. The number of current technology users with disabilities is already very high; for example, the US Census Bureau estimates that 19% of US population has a disability in 2010 [20]. 8.5% of today's world population is also above the age of 65 (where impairments are commonly observed) and by 2050, the global elderly population is projected to swell to 16.7% [13]. Moreover, application usage continues to become increasingly pervasive (e.g., at work, home or commuting). We thus need to significantly enhance the development process for mobile & wearable applications to be more sensitive to the accessibility needs of such challenged users, and to ensure that they function well across diverse contexts.

In this paper, we pitch a vision to use cutting edge Virtual and Augmented Reality (VR/AR) displays to provide an *Immersive Reality* environment where developers can *step into the shoes* of elderly or disabled users and test the usability of their applications under a variety of real-world contexts. We call this an *Empathetic User Interface Design*, where an application's usability evaluation is embedded in the application's iterative prototyping phase and is performed iteratively by the developers. While such *user-centric* design principles are not new, our approach has two key distinguishing characteristics: (a) usability evaluation is not performed at the end of app development cycle and does not necessarily involve testing by impaired users, but is performed *continually* by the developer team and (b) such evaluations utilize *Immersive Reality* enabled by VR/AR devices to *realistically emulate* the interactions of impaired users with the application. This significantly expedites the usability evaluation and also reduces the ethical issues that may arise while recruiting impaired users.

As a proof of concept, we propose *Empath-D*, a prototype that allows an app developer to use *immersive reality* to rapidly discern and evaluate usability/accessibility issues that people with different types of disability are likely to experience, for both input and output interfaces. Key to such *immersive* evaluation is the development of "Impairment Profiles", where each profile specifies a set of unique perturbations to idealized interactions by able-bodied users. These perturbations are then applied as a Pre-Rendering Filter to the application's user interactions, as a means to emulate the user experience of an appropriately impaired user. Two canonical examples would be: (1) to model a user with cataracts, a form of visual impairment, *Empath-D* modifies the view of the screen shown on a VR device by blurring the content of the display, while (2) to model a user with Parkinson's, a form of motor impairment, *Empath-D* adds an appropriate perturbation to the touch coordinates reported when a user presses on a smartwatch's touch-sensitive screen.

We outline the *Empath-D* architecture and discuss technical challenges in achieving our *Immersive Reality* vision. Also, we use an experiment to demonstrate the possible applicability and usefulness of *Empath-D* for accessibility-conscious mobile & wearable development. Our initial user study with six web designers shows that *Empath-D* makes it easier for them to design web pages for accessibility guidelines [23], reducing mental demands (NASA TLX [7] during the design process, and compared against using guidelines). We emphasize that *Empath-D* is intended to explore usability issues that go beyond traditional content-presentation guidelines (mostly applied to the design of Web pages and Web applications) and enable realistic evaluation of other forms of impairments that are especially relevant for the rapidly expanding class of immersive, wearable/mobile applications.

In summary, we make the following contributions in this paper:

- We describe our vision for *Empath-D*: a system designed to help developers to rapidly and iteratively prototype user in-

HotMobile '17, February 21-22, 2017, Sonoma, CA, USA

DOI: `http://dx.doi.org/10.1145/3032970.3032981`

terface designs for disabled and elderly people in an empathetic manner. We also present the technical challenges (providing many rich and interesting avenues of future work), that must be addressed to realize such a system.

- We present a proof-of-concept VR-driven user study that shows that *Empath-D* helps generate more *actionable* feedback for accessibility-aware design, and improves the usability of an existing guidelines-driven approach.

2. MOTIVATING SCENARIOS

The following scenarios showcase our vision for *Empath-D*.

Scenario 1: Designing for motor problems: *Alice is designing a mobile application for users with Parkinson's (which causes tremors in the hand). She configures the impairment model in* Empath-D *for Parkinson's and straps on electromyographic sensor bands (e.g., Myo* [1]*) on both arms. Alice then interacts with the application prototype in* Empath-D's *immersive reality environment – which simulates the home environment where the app is meant to run.* Empath-D *presents Alice with a 1st person perspective situating her as an avatar in the virtual environment and creates a virtualised phone (running Alice's app) that is placed in the simulated home environment.* Empath-D *modifies the sensor and touch inputs of the sensor bands and the output of the Avatar's limbs to produce a jittered output that accurately simulates hand motion with tremors. Alice notices that the app's buttons are not large enough for her to press accurately and adjusts her design.*

Scenario 2: Designing for visual impairment: *Charlie is designing an AR-type application that automatically overlays visible information on top of unclear images on web pages for people who suffer from cataracts – a condition where the lens of the eyes gets clouded resulting in blurred vision. Charlie has followed the Web Content Accessibility Guidelines that suggest the best practices for visual disability-friendly webpages. However, he is uncertain if he has done enough for the realistic conditions where his AR application would be used. Using* Empath-D*, Charlie creates an immersive reality environment which allows him to use his simulated AR application in various environments with different images.* Empath-D *modifies the visibility of these images based on multiple parameters (e.g., size of cataract, lighting conditions). Using the simulation, Charlie fixes his AR application to work better in varied environments and conditions; something he could not achieve using conventional testing methods.*

Each of these scenarios focuses on a single modality of impairment. In *Empath-D* , we envision scenarios where multiple impairments are combined and tested together. This highlights the true power of *Empath-D* as it is designed to allow (a) model multimodal impairments, (b) recreate rich environmental contexts, that may include multiple sensors and input/output devices (e.g., phones, watches, tablets, TVs). *Empath-D* also allows designers to "experience" for themselves the possibly non-additive effects that result from such multiple impairments.

3. IMMERSIVE EVALUATION APPROACH

Existing approaches to designing for accessibility rely heavily on standards and guidelines. For example, the World Wide Web Consortium (W3C) has published the Web Content Accessibility Guidelines (WCAG) 2.0 [3], which provides recommendations for making web content more accessible. When applying WCAG 2.0 to design the web pages for the elderly, one possible success criterion is to ensure that *"text can be resized without assistive technology up to 200 percent without the loss of content or functionality"*. There are multiple ways to fulfil this, and depend on the designer's interpretation. A designer may use buttons to allow for the increase in the size of the text, but will consequently have to address the interaction problems when a page becomes long and requires scrolling. Designers may also have difficulties situating the problems hindering accessibility. Newell et al. [15] report that only when designers met their elderly users and had them use paper prototypes did they understand the challenges that the elderly faced. Ultimately, guidelines do not provide a concrete means to ensure usable design for a target user group.

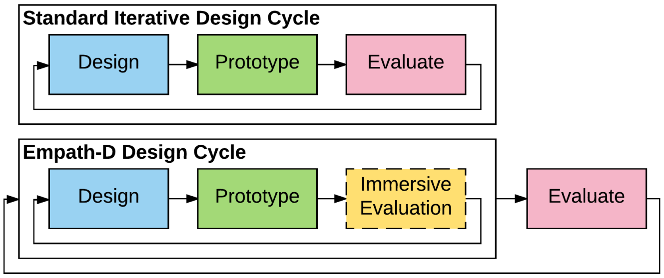


Figure 1: Guideline-based vs. *Empath-D's* design cycles

Different from conventional *guideline-based approaches*, *Empath-D* adopts a *immersive evaluation approach*. Figure 1 depicts the key difference between the two approaches. In the guideline-based approaches, the developers rely solely on a set of guidelines and usability evaluation is performed with the actual users with impairments. However, the immersive evaluation approach includes a *immersive evaluation* phase that allows developers to feel the same way that the users feel through immersive reality, which in turn can quickly feed forward to the next design and prototyping iteration.

The *immersive evaluation approach* speeds up the process of design, and allows the designer to experience the problems that their target users experience. This provides a significant advantage on two folds. First, the designer can *rapidly* create usable interfaces without needing to involve the actual disabled users early in the design cycle. This is especially important for the rare types of disabled users who are harder to find and get access to. Second, we argue that even with actual users, a designer may not be able to fully experience the disabilities they may have and have to rely on observations. *Empath-D* allows designers to directly experience these disabilities and empathise for the disabled users.

In addition, the *immersive evaluation approach* situates the problems that disabled users face, so as to augment both existing standards and guidelines-based approaches and *iterative user-centered design*. User-centered design is a common approach to the design of user interfaces [17]. Designers, developers work together with users to articulate their needs and limitations. Ethnographic studies are applied early in the design process to ground designs, and an *iterative* approach is commonly adopted to quickly work through problems that may occur in early designs. *Empath-D* supports and enhances this process.

4. EMPATH-D SYSTEM OVERVIEW

We now present the overview of the *Empath-D* system. Figure 2 shows the system architecture.

Using *Empath-D* in a VR. To immersively evaluate the application, the designer starts by compiling the application binaries to run the application instance within a virtualised device (or a physical running instance for AR). The designer then adjusts the profile settings for the impairment and selects a context of use. She puts

---

[1] `https://www.myo.com/`

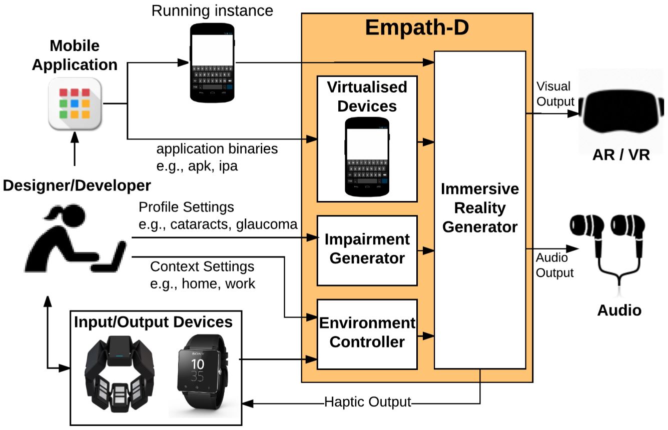


Figure 2: Overview of *Empath-D*

on her haptic-enabled input/output devices, VR set, earphones and experiences the immersive reality that *Empath-D* generates. She then tests out her application in the VR immersive reality.

Components of *Empath-D* . *Empath-D* consists of four main components (see Figure 2 in orange). *Virtualised Devices* are virtual machines of the target testing device. These are akin to the emulation capabilities provided by Android virtual devices, which allow for a virtual running instance to test applications. Virtualised devices run within the VR, allowing users to interact with them as in the real world. Extra input/output devices may be used to support physical sensing or feedback that are hard to be emulated.

The *Impairment Generator* uses the profiles set by the user to generate immersive experiences that are authentic to the elderly or disabled. For instance, it is responsible for creating the visual models (e.g., blurred vision from cataracts) that will be applied to the first person visual display. If an audio impairment (e.g., High-Frequency Hearing Loss) is to be generated, the Impairment Generator would modulate all audio output to the user such as to filter out high-frequency sound.

The *Environment Controller* generates virtual surroundings and activities to emulate testing environments (e.g., walking scenes to test a navigation app). It takes the inputs from external devices to emulates the appropriate user behaviours within the target scenario.

Lastly, the *Immersive Reality Generator* combines the outputs from all the components to generate the corresponding visual, audio and haptic outputs.

5. DESIGN & SYSTEM CHALLENGES

In this section, we present some of the key challenges that we anticipate in building *Empath-D* .

5.1 Split-Emulator Operation of App

For realistic emulation of a user's interaction with the mobile or wearable-based app, *Empath-D* will need to support a "split-UI" mode of operation of the mobile or wearable device: (a) for realistic *immersion*, the visual and audio output of the test app will have to be faithfully rendered in the virtual world (i.e., via the VR device's display and speakers), while (b) to enable realistic *interaction* with the app, the user should be able to interact "naturally" with the original devices. As an exemplar of this "split-UI" operation mode, the user will see the current screen of the app on the VR display, but presses a button on a real smartphone (held in her hand), which in turn causes the output on the VR display to reflect the actual change in the UI of the app. Here, the key challenge is to achieve this mode of dual interaction seamlessly, with an *unmodified* app.

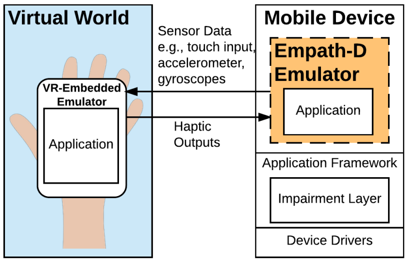


Figure 3: Split-Emulator Mode of Operation

To achieve this goal, the app effectively runs in a *distributed emulator* (see Figure 3). In one mode, the app's core logic runs in the VR-embedded emulator. However, the mobile device runs an emulator as well–this emulator (a) intercepts all input events (such as button presses, screen touch) on the mobile device and transmits it to the VR-embedded emulator, (b) handles selected output events (e.g., vibration alerts) that may not be reproducible by the VR device, and (c) optionally, transmits and receive network traffic (e.g., to capture network interactions via a 4G network interface that may be absent on the VR device). The app's output is thus primarily directed to the VR's interfaces (similar to Rio [1]): the screen display is generated and rendered on the VR display, the audio output is piped to the VR speakers. While such split-processing of an app has been proposed before [4], such emulation has well-known limitations in terms of inability to use native hardware (e.g., if the app utilizes the mobile's GPU for sensor processing). An alternative model is to run the app entirely on the mobile device, albeit within an Emulator layer that redirects its visual/audio output to the VR-embedded emulator, for appropriate rendering on the VR device.

We suspect that the preferred emulator choice will depend on the specific characteristics of the app–e.g., network-bound apps (e.g., speech recognition systems) will probably benefit from having the app logic run on the mobile device, while the VR-based App mode is better for apps with continually changing display content.

5.2 Efficient Implementation of Impairments

Empath-D's goal requires the testing of the usability of the app under various impairment conditions. Such impairments are typically manifested as either perturbations in input sensor streams (e.g., perturbations to the exact screen coordinates pressed upon by the user) or degradations in output streams (e.g., blurriness in the output display or distortions in the audio output). To support immersive operations, these perturbations must not only be representative of the impairment being studied, but must also be applied to the input/output streams without compromising *latency* and processing *throughput*.

We observe that these disability-related distortions are *independent of the specific app artifacts*, but instead occur directly on the raw input/output streams. For example, audio distortions are not related to the language of the audio output (or whether the output is music or conversations); instead, the distortions may be viewed as a non-uniform mask/filter over the entire audio spectrum.

Accordingly, for application-independent incorporation of such impairments, the Impairment Profiles will be directly implemented between the raw sensor "device drivers" and the App Emulator layer. More specifically, in Android, to introduce to on-screen touch and press events, the Impairment Layer is implemented as a "shim" between the Linux device driver and the Android Application Framework's native *Activity* class: the shim can modify the

Parameter	VR	AR
Context-Fidelity	High	Variable
Context-Diversity	High	Limited
Complexity	High	Moderate
Cognitive Consistency	High	Moderate

Table 1: AR vs. VR Considerations

(x,y) coordinates of the event, as well as alter other parameters such as ABS_MT_PRESSURE (to reflect possible impairments in the pressure applied). Similarly, to emulate visual impairments while viewing the display, the Impairment Layer code is embedded within the Application Framework's View class, and modifies the master 'View" object before it is rendered via the framebuffer.

5.3 Tradeoffs between VR and AR devices

Empath-D's key idea is to use immersive devices, together with simulated impairments, to recreate a realistic test environment for rapid, iterative usability testing by app designers. While it is possible to emulate impairments on a desktop development environment, this likely limits context-fidelity. For example, the use of AR/VR would allow us perform head-tracking, giving us a reasonable estimation of where visual impairments (e.g., blur in central vision from glaucoma) might occur. A desktop development-based emulation would either be static, or require users to move the impairment models around. We hence focus on AR/VR technologies.

One of the open questions is: will AR or VR devices be more suitable for recreating both the environmental context and the experience of impaired operations? The answer is not immediately obvious: at a high-level, the choice of VR vs. AR platforms may depend on the app being tested, as this involves tradeoffs on multiple dimensions (Table 1 summarizes the key considerations and issues that we anticipate):

- *Context Fidelity & Diversity:* This refers to the ability to accurately recreate the context in which the app's usage is being tested. In general, we expect that a VR-based usage will offer higher context fidelity, as the virtual environment is entirely programmable–it can be modified to capture a wider variety of contexts (e.g., walking around at home, eating in a foodcourt, different lighting/noise conditions). Naturally, an AR-based implementation of *Empath-D* would be limited in terms of the contexts being evaluated, because the tester would now need to be physically immersive (in the real world) in each such context. On the other hand, in the VR-based scenario, the credibility of the test results would depend on the quality of the virtual world being generated.
- *Implementation Complexity:* From an implementation standpoint, an AR-based solution is simpler, as the external context does not need to be recreated but would be directly observable by the user. In this case, the audiovisual rendering of the app, running on the mobile or wearable device, would simply be overlaid on this field-of-view display. In contrast, a VR-based implementation would need to recreate the entire immersive world and its context (e.g., multiple medical devices in an elderly patient's home, with some of these devices generating alarms that might confuse the elderly user during the testing phase). Given that either platform would need to support the split-emulator mode described above, it is possible that the processing overload on a VR-display may prove to be prohibitively high, leading to observable lags.
- *Cognitive Consistency:* VR devices have the potentially advantage that they *isolate* the user from their current physical environment. To understand the potential advantages of such isolation, consider a use case where the designer is trying to mimic the interaction of a Parkinson's afflicted person (with observable hand tremors) with his smartphone. Under VR-based operation, the smartphone (and the user's hand motion) would be viewed solely via the VR display; thus, any perturbations to the real hand movement would be observed by the user within the VR world. In contrast, with an AR-based implementation, the user would notice the absence of tremors in his hand movement and the fact that he correctly pressed an icon on the phone's display, while the emulated interaction presented via the AR display would potentially show the results of such tremors and "false press" events on the display. Such inconsistency might generate unacceptable cognitive dissonance, invalidating the evaluation process.

Figure 4: An emulated vision with cataracts.

6. EARLY EXPERIENCES OF EMPATH-D

We built a proof-of-concept prototype of *Empath-D* on the Samsung Gear VR, and used it to simulate a cataract vision impairment of the elderly to guide the design for more accessible webpages. We report our early experiences of using the *Empath-D* prototype with 6 users (Video available at https://is.gd/empathd).

6.1 Experiment Design

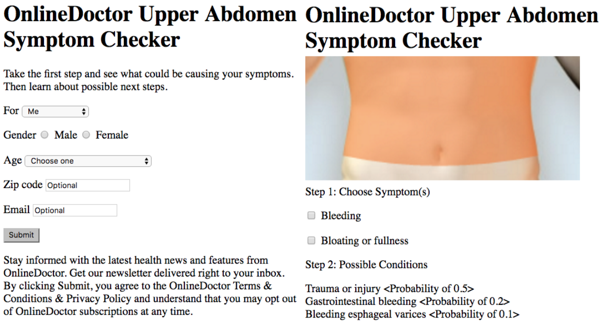
We adopted a 2x2 counter-balanced design, having 6 participants (6 Males, ages 27-41, with good eyesight) perform the design of 2 stripped down webpages twice (see Figure 5), each time taking up to 45 mins. All participants have experience in designing webpages in the past, and have working knowledge of HTML/CSS.

Participants were split into two groups. The first group designed using guidelines only (Condition 1; C1), then AR+guidelines (Condition 2; C2). The second group designed using guidelines+AR first, then guidelines only. Participants were exposed to each condition with at least one night's rest in between to reduce the effects of fatigue and learning. They are given the NASA TLX [7] after the completion of each condition, and asked to complete them in reference to the tasks of designing with guidelines, or guidelines+AR. We conducted a short semi-structured interview with participants during C2, to understand their experiences using the AR interface.

Participants were asked to design for elderly users suffered from cataracts, a vision impairment that is experienced as blurred vision (see Figure 4). We ask participants to focus on supporting vision in their designs. Participants were free to use the Internet to research methods for implementation. We also provided help on implementation to fulfil the designs that the participants came up with.

The two webpages chosen were a common use case: elderly searching for health-related information [22]. The first page asks users for common information like age and gender (see Figure 5, left). The second page is where users select the symptoms that they are experiencing and which returns the possible conditions and their associated probabilities (see Figure 5, right).

WCAG 2.0 Design Guidelines for the Elderly. The W3C provides the Web Content Accessibility Guidelines 2.0, a set of organising principles and guidelines. Each guideline is associated with



OnlineDoctor Upper Abdomen Symptom Checker

Take the first step and see what could be causing your symptoms. Then learn about possible next steps.

For Me

Gender ○ Male ○ Female

Age Choose one

Zip code Optional

Email Optional

Submit

Stay informed with the latest health news and features from OnlineDoctor. Get our newsletter delivered right to your inbox. By clicking Submit, you agree to the OnlineDoctor Terms & Conditions & Privacy Policy and understand that you may opt out of OnlineDoctor subscriptions at any time.

OnlineDoctor Upper Abdomen Symptom Checker

Step 1: Choose Symptom(s)

☐ Bleeding

☐ Bloating or fullness

Step 2: Possible Conditions

Trauma or injury <Probability of 0.5>
Gastrointestinal bleeding <Probability of 0.2>
Bleeding esphageal varices <Probability of 0.1>

Figure 5: Sample webpages used for the experiments

a level of importance: A, AA, AAA (in decreasing order of importance). The W3C Web Accessibility Initiative (WAI) recommends meeting all the guidelines associated with A and AA. The W3C WAI further specifies guidelines that are relevant towards elderly design [23]. We adopt a portion of these guidelines (i.e., Perceivable information and user interface) as the guidelines for our experiment. These guidelines relate to text size, text style and text layout, and colour and contrast.

Cataract Simulation. We created a simple AR cataract simulation by implementing Unity's Gaussian Blur component (on the lowest setting) on top of Vuforia's camera see-through mode on the Samsung Galaxy Note 4. The Note 4 was inserted into the Samsung Gear VR to allow the user to attain AR-simulated impaired vision by holding the Gear VR over the eyes.

6.2 Experimental Results

6.2.1 *Workload during Webpage Design*

We examined the overall workload and six different dimensions of the NASA TLX between the two conditions: C1 and C2. Figure 6 shows the results. For the overall workload, the two conditions differed significantly (Mann-Whitney U = 4, $n_1 = n_2 = 6$, $p < 0.05$, two-tailed), demonstrating that the use of AR interface reduced the design effort. Almost all participants in C2 (P1 - P5) consistently used the AR interface to evaluate their designs.

Among the six more detailed dimensions, we found that *Mental Demand* was the only dimension that presented a significant difference (Mann-Whitney U = 4, $n_1 = n_2 = 6$, $p < 0.05$, two-tailed). There are two main reasons for the increased Mental Workload. First, the AR interface allowed users to quickly evaluate their designs, which gave them the confidence that their designs would work (P3, P5, P6). Comparatively, designing with only guidelines required participants to construct a mental model of how an elderly person with cataracts may perceive the design. Second, participants found that the design guidelines are vague, and often had difficulty knowing what they should implement for accessibility for elderly people with cataracts (P1, P3).

The mean scores for Temporal Demand, Performance, Effort and Frustration are comparatively better, however without statistical significance. This is likely due to the differences in participants' prior experiences with AR and guidelines along with their intuitions. For example, P2 relied heavily on the AR interface to quickly perform evaluations to do his designs, which led to lower temporal demands. P5, on the other hand, paid more attention to the guidelines to come up with paper designs before implementation, resulting in stronger time pressure.

6.2.2 *Attaining WCAG 2.0 Success Criteria*

We examined how well guidelines and guidelines+AR conform to the accessibility guidelines. We ran *aChecker* [5] (an automated accessibility checker) on the second page of the design tasks for all conditions. A Wilcoxon Signed-Rank test (W = -2 > -21, p > 0.05, two-tailed) found no significant differences between the two conditions. This suggests that *Empath-D* is no worse than existing methods (i.e., guidelines) for designing webpages for accessibility. We note that this initial result may be an artefact of our experimental design, where participants are asked to design webpages in a short fixed period of time, and may not be able to fully express their designs. This is evident when we note that 5 participants wanted to - but failed - to increase the size of radio buttons or checkboxes within time. Participants added HTML comments to indicate their intent instead, and as such would not be picked up by *aChecker*.

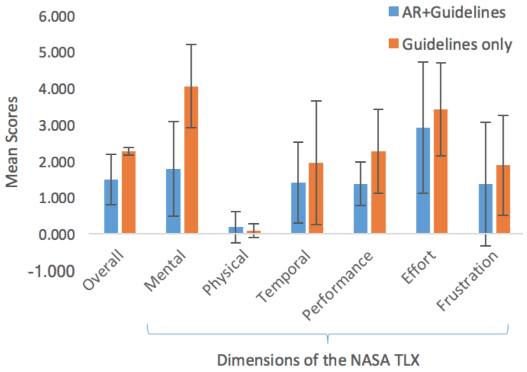


Figure 6: Mean scores of the NASA-TLX, showing the overall score and six specific dimensions (lower is better, see [7])

6.2.3 *Subjective Feedback*

All participants were positive about using the AR interface for design. Participants noted that guidelines are often unclear, and the AR interface offers a concrete means to situate design (P1, P3, P5). Participants also indicated that they preferred if there were means to allow them to use the AR interface, without having to interrupt the flow of webpage design (P1, P2, P3) – as they needed to wear on and take off the device repeatedly. Accordingly, most participants (P1, P3-P6) used the AR interface for the evaluation, whereas only one used it more frequently even during modifying the webpage. Interestingly, participants noted (P1, P3-P6) despite its usefulness, that prolonged usage would likely cause physical discomfort. Lastly, almost all participants (P1-P4, P6) indicated that they felt empathy for the elderly with cataracts. P1 and P3 in particular remarked that the"vast majority of web pages" or signages are not designed for accessibility.

7. RELATED WORK

Simulated Design. There is prior work to assist user interface designers in designing better interfaces for the elderly and people suffering from vision impairments. Higuchi et al. [8] proposed a tool to simulate the visual capabilities of the elderly while Mankoff et al. [12] developed a tool to simulate a user with visual and motor impairments using a desktop screen. *Empath-D* extends prior ideas and uses VR/AR as the medium for immersive evaluation.

SIMVIZ [2, 21] uses the Oculus Rift VR display to simulate six different types of visual impairments for the task of reading text on a smartphone. Later work [21] simulats hearing ailments by using a pair of microphones with equalized headphones. While previous work has focus on simulation in single modality (visual or auditory), *Empath-D* aims to combine modalities to support any application type, ailment (visual, auditory, motor) and environment. Additionally, *Empath-D* also aims to assist AR/VR development by providing application binary emulation, and GUI-based specification of impairment profiles and environments.

Designing for inclusiveness. *Empath-D* is inspired by prior researchers, such as Newell [16], who have argued that traditional User Centred Design techniques provides little guidance for designing interfaces for elderly and disabled users due to the large variation amongst the type and degree of impairments. They further argued that the standard guidelines for designing disabled-friendly UIs are too general [14] and lacked empathy for users.

Testing of Mobile Applications. Recently there have been many systems, such as Vanarsena [19], AMC [9], Puma [6], DynoDroid [11], DECAF [10], AppsPlayground [18], for automatically testing and identifying various types of UI and systems bugs in mobile applications. *Empath-D* takes a different approach in that we do not detect bugs after the application is finished. Instead we allow the designer to rapidly test early iterations of the designs. In this way, we hope to reduce the pain of having to make significant UI changes at the end of the design cycle – or worse, end with an application that cannot be used effectively by the target demographic.

8. DISCUSSION

Coverage and accuracy of impairment emulation. Impairments vary in nature (i.e., visual, aural, motor, speech, and cognitive), severity, and may also present simultaneously. While further study is needed to understand what impairments can be realistically emulated, *Empath-D* can likely emulate impairments that are visual or aural in nature. For instance, dyslexia may be emulated using a camera to capture and recognise text input. The display could then show text with fonts that make reading difficult to represent the cognitive difficulties that dyslexic users may face. Conversely, given that haptics interface touch and motor abilities, have a wider range of positions on the body for expression and are much more specific in nature (e.g., emulating a finger tap, or arthritis). This limits the possibilities for simulating many motor impairments.

Accuracy of impairment emulation is also important to consider, but close to 100% accuracy may not be neceesary. For instance, to achieve full fidelity in emulating the cataract, an eye tracker will have to accurately translate a region of blur to the appropriate parts of the display in a VR. However, even without such high-fidelity emulation, a web designer using *Empath-D* (which provides *functional fidelity*) would likely increase the size of the text such that a user with cataracts can see it, achieving the similar outcome as if one had a full-fidelity representation.

Developer Burden. There would be an initial learning curve for developers (in particular those who are not familiar with VR/AR development) to use *Empath-D* . *Empath-D* aims to simplify this process by providing a UI to allow developers to easily specify the location of their application binaries, impairment profiles, and environmental context settings. *Empath-D* then generates the required resources and settings files that are appropriate for a specific VR development environment (e.g., Unity). For further customisation (such as for new environments), developers will still have to develop them in the VR development environment. Despite the initial learning curve, we expect *Empath-D* is likely to reduce the overall work required through quick and continuous immersive evaluation.

9. CONCLUSIONS

In this paper, we presented our vision toward *Empathetic User Interface Design*, and proposed *Empath-D*, to achieve the vision. Running over AR/VR devices, *Empath-D* provides developers or designers with *Immersive Reality* environment, where they can empathise impairments of disabled users while testing the usability of applications. We discussed various use case scenarios of *Empath-D* along with its system architecture and foreseeable technical challenges. Our initial user study with 6 web designers shows that *Empath-D* makes it easier for them to design web pages to meet accessibility guidelines for elderly with cataracts, while reducing mental demands during the design process compared to simply using guidelines. We aim to extend *Empath-D* to emulate a variety of impairments and usage scenarios and conduct an extensive user study with a large number of developers and impaired users.

10. ACKNOWLEDGMENTS

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its IDM Futures Funding Initiative and partially by the Singapore Ministry of Education Academic Research Fund Tier 2 under research grant MOE2014-T2-1063.

11. REFERENCES

[1] A. Amiri Sani, K. Boos, M. H. Yun, and L. Zhong. Rio: A system solution for sharing i/o between mobile systems. In *Proc. of MobiSys*, 2014.

[2] H. C. Ates, A. Fiannaca, and E. Folmer. Immersive simulation of visual impairments using a wearable see-through display. In *Proc. of TEI*, 2015.

[3] B. Caldwell, M. Cooper, L. G. Reid, and G. Vanderheiden. Web Content Accessibility Guidelines (WCAG) 2.0, Dec. 2008.

[4] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti. Clonecloud: Elastic execution between mobile device and cloud. In *Proc. of EuroSys*, 2011.

[5] G. Gay and C. Q. Li. Achecker: Open, interactive, customizable, web accessibility checking. In *Proc. of W4A*, 2010.

[6] S. Hao, B. Liu, S. Nath, W. G. Halfond, and R. Govindan. Puma: programmable ui-automation for large-scale dynamic analysis of mobile apps. In *Proc. of MobiSys*, 2014.

[7] S. G. Hart and L. E. Stavenland. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In *Human Mental Workload*, chapter 7, pages 139–183. 1988.

[8] K. Higuchi, Y. Sakaguchi, K. Sugiyama, and T. Nakano. Simulating the human vision of elderly for designing control panels. In *Proc. of IEEE SMC*, 1999.

[9] K. Lee, J. Flinn, T. Giuli, B. Noble, and C. Peplin. Amc: Verifying user interface properties for vehicular applications. In *Proc. of MobiSys*, 2013.

[10] B. Liu, S. Nath, R. Govindan, and J. Liu. Decaf: detecting and characterizing ad fraud in mobile apps. In *Proc. of NSDI*, 2014.

[11] A. Machiry, R. Tahiliani, and M. Naik. Dynodroid: An input generation system for android apps. In *Proc. of FSE*, 2013.

[12] J. Mankoff, H. Fait, and R. Juang. Evaluating accessibility by simulating the experiences of users with vision or motor impairments. *IBM Systems Journal*, 44(3):505–517, 2005.

[13] National Institute on Aging. World's older population grows dramatically. https://www.nia.nih.gov/newsroom/2016/03/worlds-older-population-grows-dramatically.

[14] A. Newell and P. Gregor. Human computer interaction for people with disabilities. 1988.

[15] A. F. Newell, A. Dickinson, M. J. Smith, and P. Gregor. Designing a portal for older users: A case study of an industrial/academic collaboration. *ACM TOCHI*, 13(3):347–375, 2006.

[16] A. F. Newell, P. Gregor, M. Morgan, G. Pullin, and C. Macaulay. User-sensitive inclusive design. *Universal Access in the Information Society*, 10(3):235–243, 2011.

[17] J. Nielsen. Iterative user-interface design. *Computer*, 26(11), 1993.

[18] V. Rastogi, Y. Chen, and W. Enck. Appsplayground: automatic security analysis of smartphone applications. In *Proc. of CODASPY*, 2013.

[19] L. Ravindranath, S. Nath, J. Padhye, and H. Balakrishnan. Automatic and scalable fault detection for mobile applications. In *Proc. of MobiSys*, 2014.

[20] US Census Bureau. Americans with disabilities: 2010. https://www.census.gov/newsroom/releases/archives/miscellaneous/cb12-134.html.

[21] F. Werfel, R. Wiche, J. Feitsch, and C. Geiger. Empathizing audiovisual sense impairments: Interactive real-time illustration of diminished sense perception. In *Proc. of AH*, 2016.

[22] Worldwide Web Consortium. How older adults use the web. https://www.w3.org/TR/wai-age-literature/#how.

[23] Worldwide Web Consortium Web Accessibility Initiative. How WCAG 2.0 applies to older people. https://www.w3.org/WAI/older-users/developing.html.