15th International Conference on DEVELOPMENT AND APPLICATION SYSTEMS, Suceava, Romania, May 21-23, 2020

Engineering Details of a Smartglasses Application for Users with Visual Impairments

Ovidiu-Andrei Schipor, Adrian Aiordăchioae
MintViz Lab | MANSiD Research Center
Ștefan cel Mare University of Suceava
Suceava 720229, Romania
schipor@eed.usv.ro, adrian.aiordachioae2@student.usv.ro

Abstract—**People with visual impairments encounter many challenges in their everyday life, such as reading text and street signs, shopping, driving, or recognizing facial expressions. Smartglasses applications can assist with all of these tasks. In this paper, we present the technical details of a smartglasses application for users with visual impairments that implements assistive features controlled via voice input commands. We specify the design requirements of our application, which we implement and demonstrate for the Vuzix Blade smartglasses.**

Keywords—visual impairments, smartglasses, prototype, voice input, voice commands, Augmented Reality

I. INTRODUCTION

The term "visual impairment" refers to all disabilities characterized by decreased visual acuity [1]. As of 2020, the number of people with visual impairments exceeds a quarter of a billion globally, from which 76 million are legally blind [1]. There is a vast body of research emphasizing the strong relationship between the severity of the visual disorder and the quality of life in the sense that the lower the visual acuity, the greater the negative impact on well-being [2-6]. Smartglasses applications present the advantage of capturing video in first-person mode and at eye level, and can provide visual, audio, and haptic feedback [7-23]. While visual augmentation addresses only users with low vision, audio and haptic feedback is helpful to both users with low vision [13-23] and users who are blind [7-12]. In this paper, we present the engineering details of a software application for Augmented Reality (AR) glasses and users with visual impairments implementing voice input commands.

II. RELATD WORK

A. Overview and classification of Smartglasses-based Systems, Prototypes, and Applications

Smartglasses-based assistive systems can be classified according to several criteria. From the perspective of the *visual impairment*, such systems address blindness [7-12], generic impairments resulting in low vision [13-21,24-28], or specific impairments, such as colorblindness (inability to distinguish some colors) [22,23], nyctalopia (difficulty to see in dim light or at night) [29], macular degeneration (blurred image in the center of the visual field), or retinitis pigmentosa (decreased peripheral vision) [24,30].

To enable effective interaction between users with visual impairments and smartglasses, touch input is often extended by alternative *input channels*, of which voice input commands [13,14,22,24-26,31-34] and gestures [18,19] are the most frequent. The goal is to maintain a high level of usability even in the absence of visual information. The voice commands are used for all type of tasks from magnification [13,14,24,25] and object recognition [26,32-34] to menu navigation [13,14,24-26,31] and system settings [24-26,31,32,35]. Another criterion refers to the available *output channels* these systems employed to offer feedback and to share results. While people with low vision are usually able to see adapted versions of reality obtained through magnification, contrast enhancement, color transformation etc. [13-18,20-25], the blind users need alternative channels such as audio [7,8,11,12,16,19] or haptic [9]. However, this is not to say that the audio output is not helpful to users with low vision and that is why this modality is usually incorporated in all assistive products.

From the *functionality* point of view, the most encountered features such systems offer are magnification [13-16,18,19,24,25,27,31], contrast and edges enhancing [13-16,18,19,22,25,27], text extraction / text to speech [13,14,16,19,24,26,27] and generic object recognition [15,16,24,26,31]. The majority of these systems are designed to help people with visual impairments during a wide variety of everyday tasks. However, there are products specially designed for specific *use cases*: shopping [15], wayfinding [32-34], reading [18], clothes choosing [35], public sign recognition [12], emotional state identification [10], communication for receiving remote assistance from another person [28]. In this paper we investigated both *prototypes* [13,14,16,22,32-35] and *commercial products* [24-28,31]. While the former are usually presented only through scientific articles, the latter have detailed documentation describing features, interaction techniques, and use cases. However, the prototypes often expose more information about the architectural and implementation strategies.

B. Smartglasses-based Systems and Applications for Users with Visual Impairments

IrisVision [24] is an FDA registered Class-1 medical device that can be used by people with low vision during everyday activities. It consists in a display unit mounted on a headset so that to cover the visual field of the subject. While the former captures and processes the visual information, the latter offer

---

This work was supported by a grant of the Ministry of Research and Innovation, CNCS-UEFISCDI, project no. PN-III-P1-1.1-TE-2016-2173 (TE141/2018), within PNCDI III.

the ability to perform touch commands for zooming (up to 12x) and changes of the view mode. The equipment is powered by a mobile application that uses AR techniques to enhance image quality in accordance with specific medical conditions. NuEyes [25] is a smartglasses-based commercial product developed for helping low-vision users during activities requiring reading, object recognition, magnification (up to 18x), bar code QR scanning, and video recording. From the software perspective, NuEyes is built around ReticleOS, a see-through Android framework, and the list of commands cannot be extended by end users. OrCam MyEye [26] is a small, lightweight device that can be mounted onto the frame of regular glasses. It is used for text and face recognition, identification of banknotes and products. To avoid privacy issues, OrCam MyEye performs audio and video processing locally, without Internet access. Aira [28] is a software application that can be installed on devices powered by Android and iOS. It allows people with visual disabilities to share their smartphone or smartglasses camera with remote assistants/agents in order to receive assistance. Aira can be linked with external services (*e.g.*, Uber) to authorize agents to perform reservations. CyberEyez [31] is an application developed for the Vuzix M100/M300 smartglasses. It employs a video camera for magnification (up to 15x), text reading (100 languages without Internet connection), object recognition (via Google Images), color identification, and recognition of the emotional state of interlocutors. The application can be extended to react to other voice input commands.

Chroma [22] is a prototype built around Google Glass and Glass OS (Google XE) operating system. It is designed for color-blindness, a medical condition characterized by a distortion of the perceived color spectrum. Chroma displays an AR-based view that highlights a specific color, enhances contrast between similar hues and replaces colors. ForeSee [13] and ForeSee++ [14] are HMD vision enhancement systems that offer magnification (up to 35x), text extraction (via the Tesseract OCR engine), contrast and edge enhancement, and black and white reversal. They use VR techniques with the Oculus Rift platform. SeeingVR [16] is a set of fourteen software tools designed to be applied as an overlay on top of a VR application in order to offer real-time visual and audio augmentation to virtual scenes. Nine tools can be plugged into existing VR application without any software changes. The remaining five tools can be integrated by the developers into their applications to offer additional support for people with visual disabilities.

Tian *et al.* [32], designed a system for assisting blind individuals through indoor wayfinding tasks. It combines object identification (doors, elevators, cabinets) with text recognition (text-based signage around doors) to allow people who are blind to navigate unfamiliar environments. Drishti [33] is a prototype for guiding individuals who are blind, which can be used both indoors and outdoors due to integration of a DGPS system. Zhao et al. [34] developed a prototype on top of the Microsoft HoloLens HDM to interface smartphones. A user study showed that participants wished to combine visual and audio features during the wayfinding tasks. Yang *et al.* [35] developed a camera-based prototype for assisting people with visual impairments for choosing clothes by identifying four patterns and eleven colors. The authors used Microsoft Speech SDK for voice command recognition and Support Vector Machines (SVMs) for object classification.

III. THE SMARTGLASSES APPLICATION

In this section, we present the design and implementation details of a smartglasses-based application for people with visual impairments that uses voice input commands; see Fig. 1 for a block diagram highlighting modules and data flows.

A. *Design Criteria*

We adopted five design criteria:

- *Wide addressability*. Our system is designed to be helpful for every person, no matter their visual acuity. It can be a valuable assistant to both low vision and blind users [32], but also for people without visual impairments. The results are presented in different modalities in accordance with the visual impairment.
- *Highly configurable*. Each module of our software architecture has an associated configuration file to easily adjust its functioning. This design requirement assures that the overall architecture is flexible enough to foster a wide variety of use case scenarios.
- *Device independence*. Although our prototype is implemented on smart glasses, it can be easily adapted for another hardware configurations. Due to the separation between the inner business modules and the user interface components, the software classes can be re-used in different implementations.
- *Web orientation*. In order to avoid any dependency with respect to a specific software platform, we decided to implement the prototype only by relying on web technologies. That is why it can run on virtually any device with basic web capabilities (i.e., running JavaScript, performing Ajax requests, etc.).
- *Easy integration*. The prototype should be used either as an independent assistive tool or integrated into wider systems such as smart spaces or smart vehicles [36-40]. In the second case, the user can benefit from information taken from various sources such as motion tracking systems [37,39], EEG helmet [38], or smart wearable [40]. Thus, our prototype needs to be able to exchange data with an ecosystem of devices.

B. *Software Architecture*

Figure 1 presents the modules of the architecture together with the data flows and corresponding configuration options. The modules are represented graphically in the form of colored rectangles, while data and configuration options are shown with dashed rectangles connected via arrows to the main modules. The processing chain starts with the acquisition of voice input. The audio data is transmitted to a module dedicated to STT (Speech-to-Text) conversion. At this level we can customize both the language and the word-level confidence (*i.e.*, the minimum level of confidence the recognized sequence must attain in order to be considered). If there is no sequence that meets this constraint, an informative message is shown to the user.

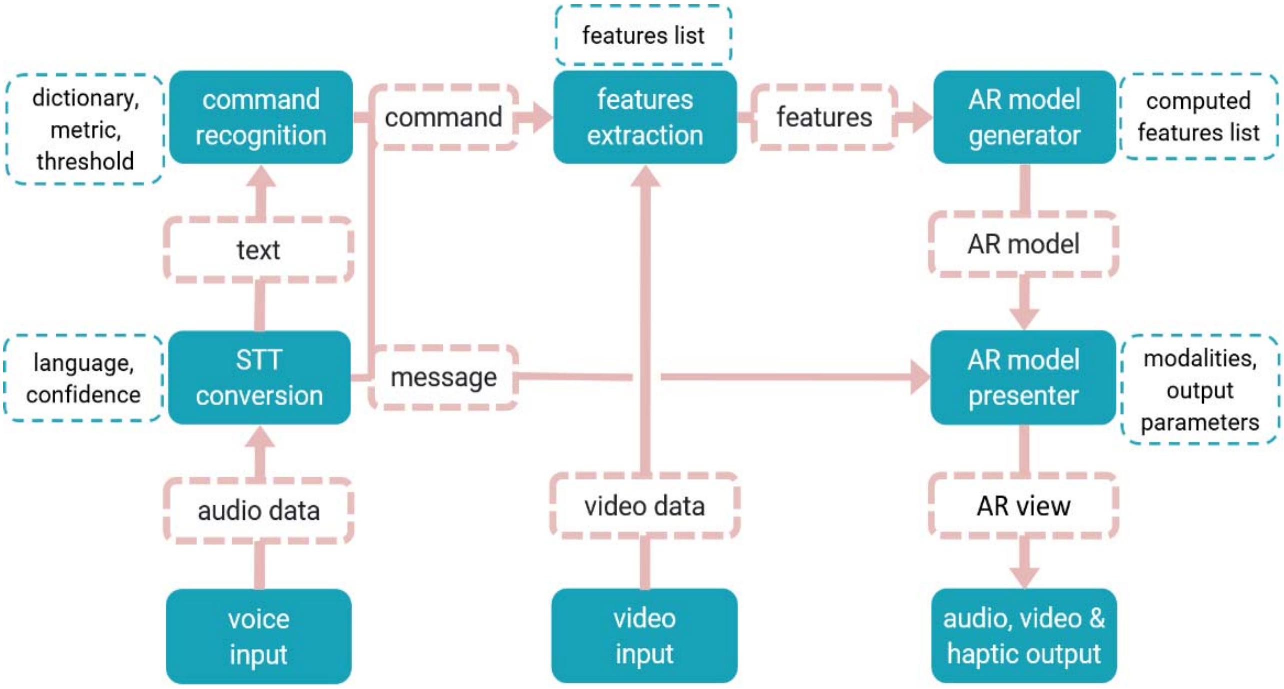


Fig. 1. The software architecture of our prototype outlining software modules (filled blocks), data flows (arrows), and configurations (dashed blocks).

Otherwise, the recognized text is forwarded to the command recognition module. This component is in charge with distinguishing if a text data can be associated with a command. It outputs either an informative message if the command is unknown or a valid reference to the command from the dictionary.

The next module is responsible for video stream features extraction. The information is combined by the AR model generator component to provide a model of the surrounding space. We emphasize an essential difference between the feature extraction and AR model generator modules. While the former processes the video frames one at a time, the latter combines features from successive frames to create a 3D model. However, this model is abstract in the sense that it is not yet associated with specific output channels. It is the role of the AR model presenter to generate a usable form of the model by considering suitable output modalities (audio, video, haptic) and parameters (loudness, size, contrast, colors, intensity, etc.). The processing chain ends with rendering the AR view to the user.

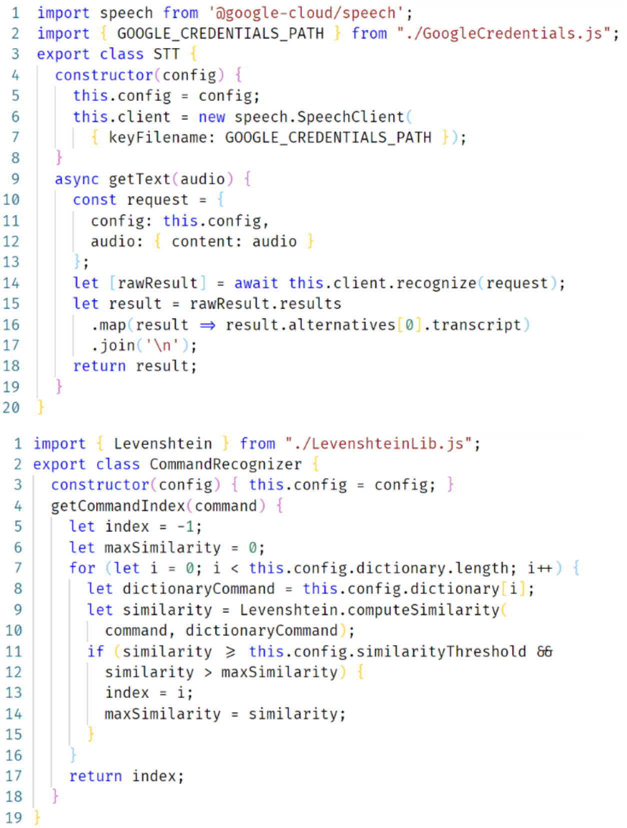
C. *Technical Details*

To recognize voice commands, we performed calls to the Google Cloud Speech API [41]. The command recognition module performs a dictionary search using the Levenshtein distance [42] and a rejection threshold (0.8, derived experimentally); see Figure 2 for an example of JavaScript code implementing speech recognition on the server side and command recognition on the client side. Video frames captured by the video camera are sent to Google Cloud Vision [43]. The feature extraction module returns standardized features independent of the implementation: objects in the scene, labels identifying general objects, locations, and activities, sequences of text, properties describing dominant colors and shapes, and warnings related to ethical issues; see Figure 3 for an instance of results in JSON format for a specific image. Due to strict modularization, different implementations of a component remain transparent for the rest of the system.

Each of the software components is implemented using web compatible technologies: the user interface relies on web pages and the core business modules are powered by the Node.js runtime environment. While the presentation layer is loaded on the smartglasses, the back-end is hosted on a remote server and accessed through a Wi-Fi connection. Since the AR view is represented by a web page embedding the AR content, it can be displayed on all devices that can run a web browser.

We tested out prototype in a setup based on Vuzix Blade Smart Glasses [44]. They employ HD camera (8MP, 30fps@720p or 24fps@1080p), noise canceling microphone, head motion tracking, dual haptic feedback, touchpad, and a full-color see-through DLP display (right eye monocular) to deliver an enhanced Augmented Reality experience. The Vuzix Blade Smart Glasses run on Android 5.1 operating system and support Bluetooth and Wi-Fi connection with other devices such as smartphones, smartwatches, and headphones. We render the AR view in a lightweight version Google Chrome browser.

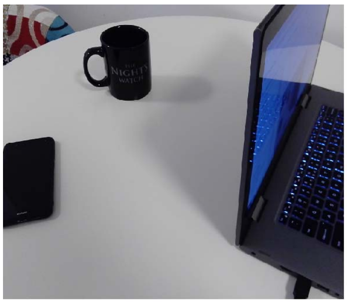
We implemented and tested 9 voice input commands that can be used in a wide variety of scenarios: "recognize objects", "read text", "find <object>", "enable/disable vibration", "disable/enable/increase/decrease volume". The results are presented in accordance with the visual functioning of the user. For example, in the case of low vision, the visual representations of the recognized objects and text are rendered at a custom size and with a custom color scheme to compensate for the visual impairment.



```javascript
import speech from '@google-cloud/speech';
import { GOOGLE_CREDENTIALS_PATH } from "./GoogleCredentials.js";
export class STT {
  constructor(config) {
    this.config = config;
    this.client = new speech.SpeechClient(
      { keyFilename: GOOGLE_CREDENTIALS_PATH });
  }
  async getText(audio) {
    const request = {
      config: this.config,
      audio: { content: audio }
    };
    let [rawResult] = await this.client.recognize(request);
    let result = rawResult.results
      .map(result => result.alternatives[0].transcript)
      .join('\n');
    return result;
  }
}
```

```javascript
import { Levenshtein } from "./LevenshteinLib.js";
export class CommandRecognizer {
  constructor(config) { this.config = config; }
  getCommandIndex(command) {
    let index = -1;
    let maxSimilarity = 0;
    for (let i = 0; i < this.config.dictionary.length; i++) {
      let dictionaryCommand = this.config.dictionary[i];
      let similarity = Levenshtein.computeSimilarity(
        command, dictionaryCommand);
      if (similarity >= this.config.similarityThreshold &&
        similarity > maxSimilarity) {
        index = i;
        maxSimilarity = similarity;
      }
    }
    return index;
  }
}
```

Fig. 2. Code snippet (JavaScript) implementing speech to text conversion on the server side and command recognition on the client side.



```json
[
  { "name": "Laptop",
    "score": 0.92,"bounds":[...]
  },
  { "name": "Mug",
    "score": 0.87,"bounds":[...]
  },
  { "name": "Computer keyboard",
    "score": 0.86,"bounds":[...]
  },
  { "name": "Mobile phone",
    "score": 0.77,"bounds":[...]
  }
]
```

Fig. 3. Example of results delivered by the feature extraction component (JSON format). Each object is described with its name, confidence score, and an array of points for localization in the image.

Results can be also delivered via vibrotactile feedback, a feature is available on the Vuzix Blade smartglasses. Moreover, vibrotactile feedback can be controlled from JavaScript running in the web browser due to the standardization of the Vibration API [46]. This type of feedback can be useful in scenarios involving searching objects, as follows. The user enters the command (e.g., find phone) and the smartglasses vibrate when the object of interest enters the visual field. Vibrations become more intense as the user approaches the object of interest.

IV. CONCLUSION AND FUTURE WORK

We presented a smartglasses-based application to assist people with visual impairments in their visual search tasks, for which we implemented nine voice input commands. We provide the results in a customizable format by using AR techniques and specific settings for each user. Future work will address validation of the application prototype through user studies, but also incorporation of other voice input commands resulted from a survey of the scientific literature and commercial smartglasses [47]. We are also in the process of investigating the integration of our prototype within a smart environment with a dedicated software architecture [37,39]. This way, we can extend the range of the input to other wearable devices such as smart watches, armbands, and smart rings. Other devices can also be integrated to provide additional information about the surrounding environment: panoramic or remote vision (drones), emotion recognition (EEG), infrared or thermal vision, etc. Moreover, we plan to extend our research to address users with both visual and motor impairments [38,48].

ACKNOWLEDGMENT

This work was supported by a grant of the Ministry of Research and Innovation, CNCS-UEFISCDI, project no. PN-III-P1-1.1-TE-2016-2173 (TE141/2018), within PNCDI III.

REFERENCES

[1] WHO, 'Vision 2020 The right to sight', 2020. [Online]. Available: https://www.who.int/blindness/Vision2020_report.pdf.

[2] M. Langelaan, M. R. de Boer, R. M. van Nispen, B. Wouters, A. C. Moll, and G. H. van Rens, “Impact of visual impairment on quality of life: a comparison with quality of life in the general population and with other chronic conditions,” Ophthalmic epidemiology, vol. 14, no. 3, pp. 119–126, 2007.

[3] M. A. Sprangers et al., “Which chronic conditions are associated with better or poorer quality of life?,” Journal of clinical epidemiology, vol. 53, no. 9, pp. 895–907, 2000.

[4] R.-D. Vatavu, B.-F. Gheran, and M. D. Schipor, “The Impact of Low Vision on Touch-Gesture Articulation on Mobile Devices,” IEEE Pervasive Computing, vol. 17, no. 1, pp. 27–37, 2018.

[5] M. D. Schipor and R.-D. Vatavu, “Coping strategies of people with low vision for touch input: A lead-in study,” in 2017 E-Health and Bioengineering Conference (EHB), 2017, pp. 357–360.

[6] R.-D. Vatavu, “Visual impairments and mobile touchscreen interaction: state-of-the-art, causes of visual impairment, and design guidelines,” International Journal of Human.–Computer Interaction, vol. 33, no. 6, pp. 486–509, 2017.

[7] R. Agarwal et al., “Low cost ultrasonic smart glasses for blind,” in 2017 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2017, pp. 210–213.

[8] M. Barney, J. Kilner, G. Brito, A. Araújo, and M. Nogueira, “Sensory Glasses for the Visually Impaired,” in Proceedings of the 14th Web for All Conference on The Future of Accessible Work, 2017, pp. 1–2.

[9] H. C. Stronks, E. B. Mitchell, A. C. Nau, and N. Barnes, “Visual task performance in the blind with the BrainPort V100 Vision Aid,” Expert review of medical devices, vol. 13, no. 10, pp. 919–931, 2016.

[10] C. Wang et al., “Monitoring the Emotional Flow of Blind People by Using Physiological Sensors and Smart Glasses,” in Proceedings of the 2018 ACM Conference Companion Publication on Designing Interactive Systems, 2018, pp. 339–342.

[11] D. Massiceti, S. L. Hicks, and J. J. van Rheede, “Stereosonic vision: Exploring visual-to-auditory sensory substitution mappings in an

immersive virtual reality navigation paradigm," PloS one, vol. 13, no. 7, 2018.

[12] F. Lan, G. Zhai, and W. Lin, "Lightweight smart glass system with audio aid for visually impaired people," in TENCON 2015-2015 IEEE Region 10 Conference, 2015, pp. 1–4.

[13] Y. Zhao, S. Szpiro, and S. Azenkot, "Foresee: A customizable head-mounted vision enhancement system for people with low vision," in Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility, 2015, pp. 239–249.

[14] Y. Zhao, S. Szpiro, L. Shi, and S. Azenkot, "Designing and Evaluating a Customizable Head-mounted Vision Enhancement System for People with Low Vision," *ACM Trans. Access. Comput.*, vol. 12, no. 4, pp. 15:1–15:46, Dec. 2019, doi: 10.1145/3361866.

[15] Y. Zhao, S. Szpiro, J. Knighten, and S. Azenkot, "CueSee: exploring visual cues for people with low vision to facilitate a visual search task," in Proceedings of the 2016 ACM Int. Joint Conference on Pervasive and Ubiquitous Computing, 2016, pp. 73–84.

[16] Y. Zhao, E. Cutrell, C. Holz, M. R. Morris, E. Ofek, and A. D. Wilson, "SeeingVR: A set of tools to make virtual reality more accessible to people with low vision," in Proceedings of the 2019 CHI Conf. on Human Factors in Computing Systems, 2019, pp. 1-14.

[17] A. D. Hwang and E. Peli, "Augmented Edge Enhancement on Google Glass for Vision-Impaired Users," Information Display, vol. 30, no. 3, pp. 16–19, 2014.

[18] L. Stearns, V. DeSouza, J. Yin, L. Findlater, and J. E. Froehlich, "Augmented reality magnification for low vision users with the microsoft hololens and a finger-worn camera," in Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility, 2017, pp. 361–362.

[19] J. Huang, M. Kinateder, M. J. Dunn, W. Jarosz, X.-D. Yang, and E. A. Cooper, "An augmented reality sign-reading assistant for users with reduced vision," PloS one, vol. 14, no. 1, 2019.

[20] S. L. Hicks, et al., "A depth-based head-mounted visual display to aid navigation in partially sighted individuals," PloS one, vol. 8, no. 7, 2013.

[21] J. J. van Rheede, I. R. Wilson, R. I. Qian, S. M. Downes, C. Kennard, and S. L. Hicks, "Improving mobility performance in low vision with a distance-based representation of the visual scene," Investigative ophthalmology & visual science, vol. 56, no. 8, pp. 4802–4809, 2015.

[22] E. Tanuwidjaja et al., "Chroma: a wearable augmented-reality solution for color blindness," in Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, 2014, pp. 799–810.

[23] A. Popleteev, N. Louveton, and R. McCall, "Colorizer: smart glasses aid for the colorblind," in Proceedings of the 2015 workshop on Wearable Systems and Applications, 2015, pp. 7–8.

[24] IrisVision, 'IrisVision Live'! [Online]. Available: https://irisvision.com/wp-content/uploads/2019/03/IrisVisionUserGuide-v3.0-live.pdf. [Accessed: 14- Feb- 2020].

[25] NuEyes, ' NuEyes Pro'. [Online]. Available:https://nueyes.com/downloads/nueyes-pro-user-manual.pdf. [Accessed: 14- Feb- 2020].

[26] OrCam, ' Welcome to OrCam'. [Online]. Available: http://www.voicesystems.it/download/orcam/orcam-myeye-2-user-guide.pdf. [Accessed: 14- Feb- 2020].

[27] Acesight, ' Acesight'. [Online]. Available: https://www.zoomax.com/wp-content/uploads/2019/07/Acesight-User-Guide.pdf..

[28] Aira, 'Manual for Horizon Glasses'. [Online]. Available: https://aira.io/wp/wp-content/uploads/2018/06/Manual-for-Aira-Horizon-Glasses.pdf. [Accessed: 14- Feb- 2020].

[29] C. Hu, G. Zhai, and D. Li, "An Augmented-Reality night vision enhancement application for see-through glasses," in 2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), 2015, pp. 1–6.

[30] C. Iyama et al., "QD laser eyewear as a visual field aid in a visual field defect model," Scientific reports, vol. 9, no. 1, pp. 1–8, 2019.

[31] Cyber Eyez, 'Bring Life Into Focus'. [Online]. Available: https://www.magnifyingaids.com/pdfs/1707.Cybereyez.4pgBrochure%20-%20LARGE%20FONT.pdf. [Accessed: 14- Feb- 2020].

[32] Y. Tian, X. Yang, C. Yi, and A. Arditi, "Toward a computer vision-based wayfinding aid for blind persons to access unfamiliar indoor environments," Machine vision and applications, vol. 24, no. 3, pp. 521–535, 2013.

[33] L. Ran, S. Helal, and S. Moore, "Drishti: an integrated indoor/outdoor blind navigation system and service," in Second IEEE Annual Conference on Pervasive Computing and Communications, 2004. Proceedings of the, 2004, pp. 23–30.

[34] Y. Zhao, E. Kupferstein, H. Rojnirun, L. Findlater, and S. Azenkot, "The Effectiveness of Visual and Audio Wayfinding Guidance on Smartglasses for People with Low Vision.", To appear in Proceedings of ACM CHI Conference on Human Factors in Computing Systems, Hawaii, 2020

[35] X. Yang, S. Yuan, and Y. Tian, "Assistive clothing pattern recognition for visually impaired people," IEEE Transactions on Human-Machine Systems, vol. 44, no. 2, pp. 234–243, 2014.

[36] A. Aiordachioae and R.-D. Vatavu, "Life-Tags: A Smartglasses-based System for Recording and Abstracting Life with Tag Clouds," Proceedings of the ACM on HCI, vol. 3, no. EICS, pp. 1–22, 2019.

[37] O.-A. Schipor, R.-D. Vatavu, and J. Vanderdonckt, "Euphoria: A Scalable, event-driven architecture for designing interactions across heterogeneous devices in smart environments," Information and Software Technology, vol. 109, pp. 43–59, 2019.

[38] O. Gherman, O. Schipor, and B.-F. Gheran, "VErGE: A system for collecting voice, eye gaze, gesture, and EEG data for experimental studies," in 2018 International Conference on Development and Application Systems (DAS), 2018, pp. 150–155.

[39] O.-A. Schipor, R.-D. Vatavu, and W. Wu, "SAPIENS: Towards software architecture to support peripheral interaction in smart environments," Proceedings of the ACM on Human-Computer Interaction, vol. 3, no. EICS, pp. 1–24, 2019.

[40] B.-F. Gheran, R.-D. Vatavu, and J. Vanderdonckt, "Ring x2: Designing gestures for smart rings using temporal calculus," in Proceedings of the 2018 ACM Conference Companion Publication on Designing Interactive Systems, 2018, pp. 117–122.

[41] "Cloud Speech-to-Text - Speech Recognition," Google Cloud. [Online]. Available: https://cloud.google.com/speech-to-text

[42] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," in Soviet physics doklady, 1966, vol. 10, pp. 707–710.

[43] "Vision AI | Derive Image Insights via ML | Cloud Vision API." [Online]. Available: https://cloud.google.com/vision.

[44] "Vuzix Blade® | Augment Reality (AR) Smart Glasses for the Consumer." [Online]. Available: https://www.vuzix.com/products/blade-smart-glasses. [Accessed: 05-Mar-2020].

[45] P.-P. Rusu, M.-D. Schipor, and R.-D. Vatavu, "A Lead-In Study on Well-Being, Visual Functioning, and Desires for Augmented Reality Assisted Vision for People with Visual Impairments," in 2019 E-Health and Bioengineering Conference (EHB), 2019, pp. 1–4.

[46] "Vibration API," W3C. [Online]. Available: https://www.w3.org/TR/vibration/.

[47] A. Aiordăchioae, O.-A. Schipor, R.-D. Vatavu, "An Inventory of Voice Input Commands for Users with Visual Impairments and Assistive Smartglasses Applications," in Proc. of the 15th Int. Conference on Development and Application Systems, 2020

[48] R.-D. Vatavu and O.-C. Ungurean, "Stroke-Gesture Input for People with Motor Impairments: Empirical Results & Research Roadmap," in Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, 2019, pp. 1–14.