Transactions on Energy Systems and Engineering Applications

Article

Development of an accessible system to enhance driving instruction for individuals with hearing impairments

Diana Peralta[1], Giovanni Caicedo[1] and Adolfo Duarte[1,*]

[1] Department of Electronic Engineering, Universidad Popular del Cesar, Valledupar, Colombia.
[*] Correspondence: adolfoduarte@unicesar.edu.co

Received: 20 January 2025; Accepted: 21 October 2024; Published: 24 April 2025

Abstract: This development took place within the framework of Resolution 20223040030355 from the Ministry of Transportation of Colombia, which states that individuals with hearing disabilities have the right to obtain driver's licenses. Consequently, driving schools must adapt their methodologies to facilitate the participation of these individuals in classes. With the aim of improving these processes for deaf individuals during practical driving lessons, a prototype was created. This prototype incorporates an algorithm to process voice commands and route them to a speech recognition model called VOSK, along with a graphical interface implemented in Python using the PyQt5 library. Tests were conducted during practical sessions in vehicles at a driving school in the city, and a socialization session with members of an association of individuals with hearing disabilities resulted in mostly positive outcomes. The project's approach introduces innovative methodologies compared to those used in other countries, contributing to educational and inclusive advancement through technology.

1. Introduction

The absence of knowledge and widespread misinformation about hearing disabilities in Colombia has resulted in decades of prejudice against the affected community [1]. Among the challenges faced by deaf individuals is the erroneous belief that they are unable to participate in activities that do not strictly necessitate the sense of hearing, such as driving vehicles [2].

Various studies have examined the relationship between hearing loss and traffic accidents. Research indicates that no direct correlation has been found between the hearing impairment of drivers and an increased frequency of road accidents. On the contrary, in some cases, the deficiency in hearing ability has been linked to a higher level of caution and attention while driving, resulting in a decrease in road incidents [3, 4]. Based on these conclusions and referring to a report conducted by the Universidad de los

How to cite this article: Peralta, Diana; Caicedo, Giovanni; Duarte, Adolfo. Development of an accessible system to enhance driving instruction for individuals with hearing impairments. *Transactions on Energy Systems and Engineering Applications*, 6(1): 617, 2025. DOI:10.32397/tesea.vol6.n1.617

Andes for the National Road Safety Agency, the Ministry of Transportation of Colombia issued Resolution 20223040030355 on May 31, 2022, authorizing individuals with various levels of hearing loss to obtain driving licenses for personal vehicles [2].

In this context, the Ministry of Transportation's resolution mandates that driving schools must make accommodations to instruct individuals with hearing disabilities who wish to obtain their driving licenses. These accommodations include the provision of sign language interpreters for theoretical sessions or the use of deferred video formats to facilitate communication between the learner and the instructor. For practical classes, reasonable adjustments must be made to ensure that the student with a disability can receive instructions from the instructor without significant distractions, utilizing written or audiovisual methods for this purpose [2].

At present, the inclusion of learners with varying degrees of deafness in the country's driving schools is insufficient. Moreover, many instructors lack the necessary training to teach individuals with hearing impairments, necessitating the use of sign language interpreters. While effective in a classroom setting, this alternative can create distractions within the vehicle during practical classes, as the involvement of more than two people could increase sensory stimuli and potentially delay the learner's comprehension of instructions [2]. In contrast, in countries like the United Kingdom, instructors are provided with explanatory material on driving education for deaf individuals. Their approach involves using British Sign Language during the class, with the instructor positioning their hand in a specific area in front of the vehicle dashboard, within the driver's field of vision, allowing the driver to visualize instructions through signs [5].

In terms of technological developments, in Saudi Arabia, there is currently research underway to facilitate communication between deaf drivers and hearing passengers through the application of machine learning. At present, the focus of this research is on building and testing an Arabic Sign Language Corpus to later integrate it into a model [6]. On the other hand, in Jordan, the development of an assistance device for deaf drivers was carried out, which recognizes voice navigation instructions from GPS applications and translates them into understandable vibration patterns [7]. Consequently, the implementation of an initial prototype seeks to address the inclusion challenges faced by individuals with hearing disabilities in obtaining a driving license. The primary focus is on improving communication between the instructor and the learner with disabilities, aiming to minimize distractions during practical sessions. Additionally, the initiative strives to align driving schools with the new guidelines set by the Ministry of Transportation. Ultimately, this effort aims to benefit an estimated population of 314,320 to 483,219 individuals with varying degrees of hearing impairment in the country [8].

The system developed in this work differs significantly from the previously mentioned solutions. Our prototype uses a voice recognition algorithm (VOSK) to convert voice commands into text, which is then displayed on a graphical interface implemented in Python using the PyQt5 library. This approach not only facilitates practical driving instruction by translating voice commands into visual instructions accessible to students with hearing disabilities, but it is also adaptable to different languages and accents, making it more versatile compared to systems based on sign language or vibrations. In Colombia, the inclusion of individuals with hearing disabilities in driving schools is regulated by Resolution 20223040030355 from the Ministry of Transportation, which requires methodological adaptations to facilitate their participation [2]. However, the implementation of these adaptations is limited, and many teaching centers are neither adequately equipped nor trained to serve students with hearing disabilities. In other Latin American countries, the situation is similar, with inclusion efforts in their initial stages and a lack of specialized infrastructure and training.

This document is structured as follows: the second section explains the methodology applied in the hardware implementation, the third section covers the design and architecture of the software, the fourth section elaborates on the design of the graphical interface and the distribution of its views, the fifth section details the tests conducted on the prototype in a practical environment, and the sixth section presents the collected results. Finally, in the last section, conclusions and future work in the development of the prototype are discussed.

2. Hardware

In designing the hardware of the device, careful consideration was given to the necessary requirements. Consequently, four essential components were chosen, as illustrated in the block diagram of the prototype in Figure 1.

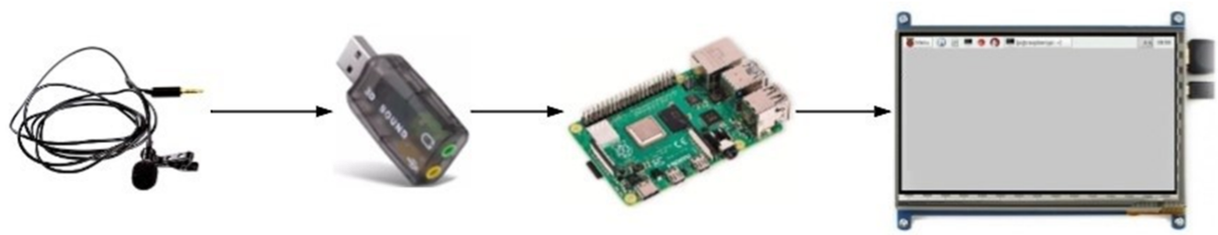

Figure 1. Block diagram of prototype.

This configuration consists of the following components:

- **Microphone:** serving as the input device for the prototype, it captures and transmits audio signals for subsequent processing.
- **USB Audio Card:** responsible for receiving signals from the microphone and transmitting them to the computer via a USB 2.0 port.
- **Single Board Computer (SBC):** where the voice command processing and comparison algorithm is executed. Additionally, it runs the Python application designed for this purpose.
- **Touchscreen Display:** acting as the prototype's output device, it is connected to the SBC.

Regarding the design of the prototype's casing, considerations included the peripherals and their power requirements, as well as the material and color. Black acrylic was selected for the casing material due to its cost-effectiveness and its color, which helps users focus on the screen rather than the casing design. The front of the casing features a seven-inch diagonal perforation corresponding to the touchscreen, as shown in Figure 2. User interaction is required due to the hybrid nature of this peripheral.

Figure 2. Front view of the device casing.

Laterally, the casing features two perforations (Figure 3); one for an input peripheral (sound card) and another for the power port. The latter is facilitated by a USB-C cable, while the sound card is equipped with two jack inputs, color-coded as yellow for the microphone and green for headphones. If the user employs headphones, the use of a splitter would be appropriate.

Figure 3. Side view of the prototype casing.

Several mechanisms were considered for attaching the prototype to the vehicle. Initially, velcro strips were considered, but this idea was discarded due to the adhesive residue they leave upon removal, potentially causing discomfort and mess in the car. The goal was to develop an attachment system that is minimally invasive and does not affect the vehicle.

Subsequently, a base with three suction cups for windows, coated with vaseline, was suggested, but the prototype remained unstable. Efforts to address this issue included adding an acrylic plate shaped like an "L" with an additional suction cup for balance, but the material proved to be prone to breaking. The final solution involved using velcro strips with suction cups on the ends to attach the prototype to the windshield of the vehicle. Additionally, a base with two suction cups on the bottom provided stability, as shown in Figure 4. Power supply was managed with a USB charging adapter connected to the vehicle's 12V power socket, ensuring a practical and affordable solution for the user.

Figure 4. Vehicle Attachment Mechanism.

3. Algorithm and architecture

In a general sense, the program aims to capture the instructions given by the instructor. It processes these instructions through a voice recognition model that transforms the audio input into text. Subsequently, the text is compared against a database of commands in search of matches, and upon finding one, the associated view for the recognized instruction is displayed. The flowchart describing this process is illustrated in Figure 5.

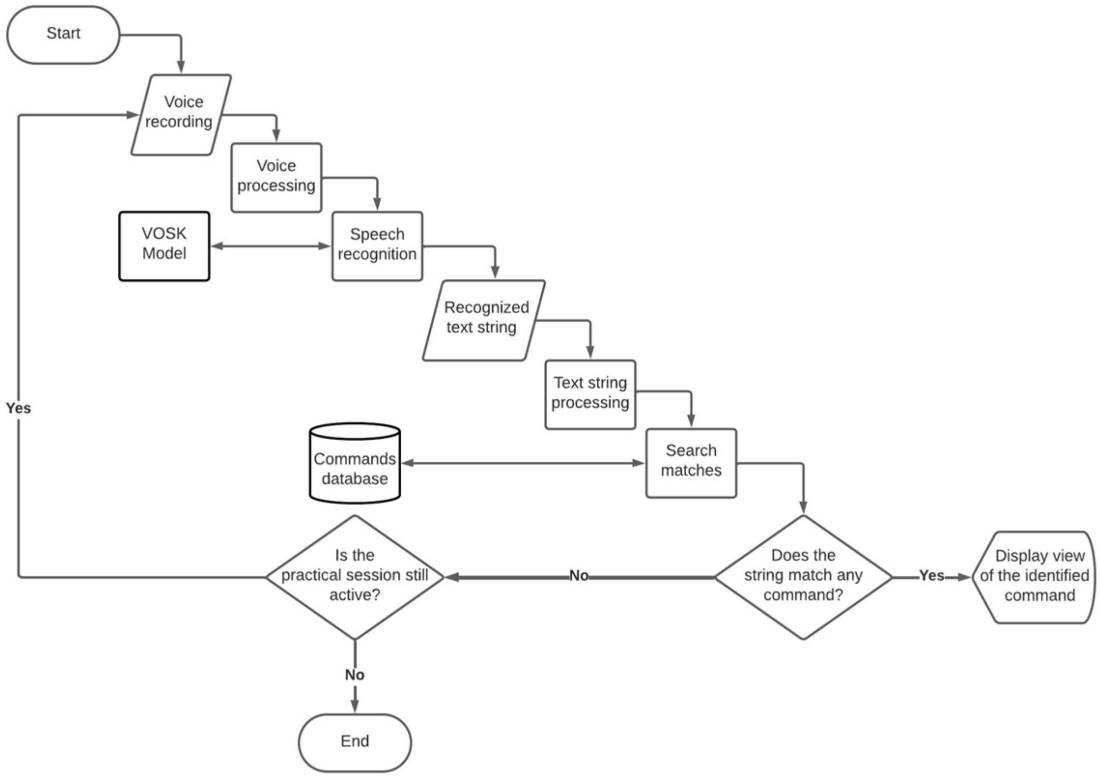

Figure 5. Program flowchart.

The program was designed following the Model-View-Controller pattern. This architecture was implemented with the purpose of facilitating software development and maintenance. Controllers serve as intermediaries between the views and the application models, responsible for receiving and processing data. Views, on the other hand, represent the user interface of the system, where users can interact with the application. Additionally, models handle and utilize project data. The distribution of the different parts of the program is illustrated in Figure 6.

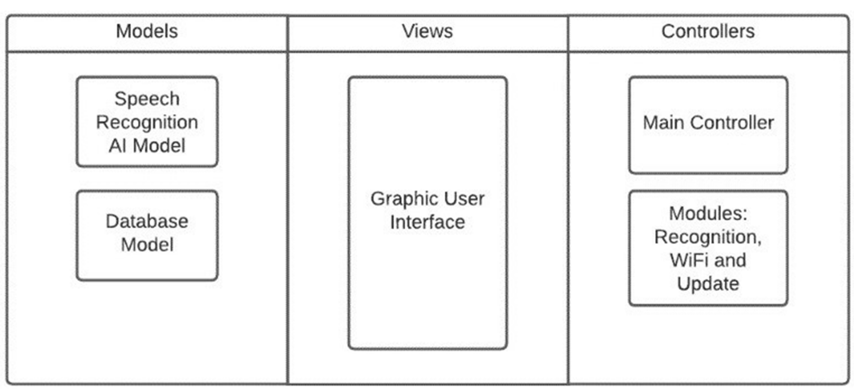

Figure 6. General scheme of the organization of the application.

3.1. Application controllers

The controllers have been distributed into two groups, the first being the main controller and the modules. These modules are responsible for carrying out specific functions in the software. The application is implemented to utilize threads, enabling the execution of different processes concurrently, thereby improving responsiveness and performance. The main controller runs in the primary thread, which is responsible for managing the interaction between the application views and overseeing the flow of data. Additionally, it activates secondary threads in which modules are executed based on the required functionality.

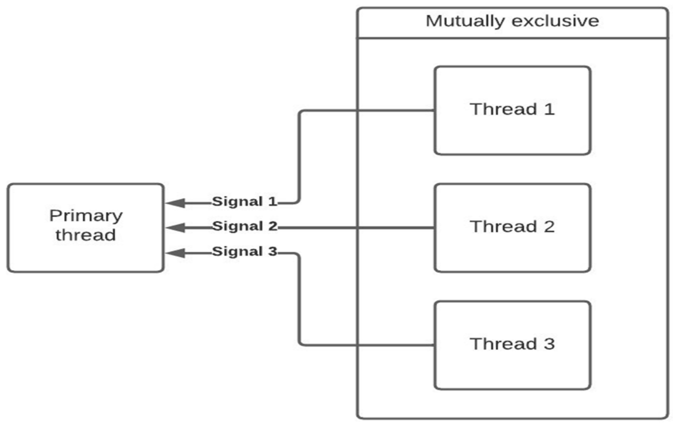

Figure 7. Representation of the operation of the application threads.

As observed in Figure 7, communication between threads is accomplished through signals, which are emitted from a secondary thread to the primary thread. Additionally, a design was implemented for the application in which only one secondary thread is active at a time. This was done to preserve system stability and prevent the accumulation of signals sent to the primary thread.

Recognition Module: This module was implemented to maintain continuous audio recording, carry out processing, and subsequent recognition of instructions. It is executed in a secondary thread, pragmatically named the recognition thread. The module is divided into two stages.

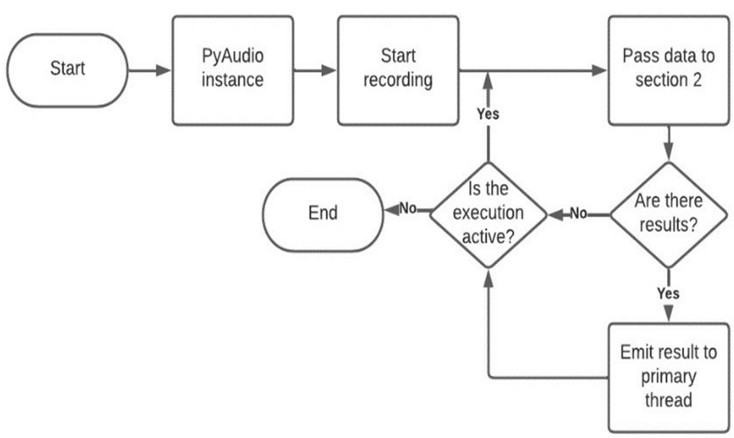

Figure 8. Flowchart of the first stage of the capture and recognition algorithm.

The first stage is responsible for audio recording using a Python library called PyAudio, designed to manage device audio drivers. Through an instance of the PyAudio class, the necessary parameters for audio recording are configured: a recording format of 16 bits, 1 recording channel, a sampling frequency of 16 kHz, and a buffer size of 8192 frames. With this configuration, the program is ready to perform audio recording in a loop, capturing audio data continuously and distributing it in each frame. This data is used as input parameters for the next stage, which, if it finds a matching command, returns a list containing the image and name of that command, or a null value if no match is found. Subsequently, a signal is emitted to the primary thread with the necessary information to display the corresponding view. The operation of this stage can be summarized as illustrated in Figure 8.

The second stage acts as an intermediary between the first stage of the recognition module and the VOSK speech recognition model. Upon launching the main application, the VOSK model is initialized along with its dependencies. Additionally, a list of commands is obtained from the model in the database when the first part of the module starts, which is subsequently utilized in this stage. One of the key criteria for choosing the VOSK models was their ability to work offline and their lightweight nature, ensuring their functionality in areas with low or no coverage and on devices with limited resources, such as Raspberry Pi or mobile phones. Furthermore, in various conducted studies, the system demonstrated an acceptable Word Error Rate compared to other speech recognition systems [9, 10].

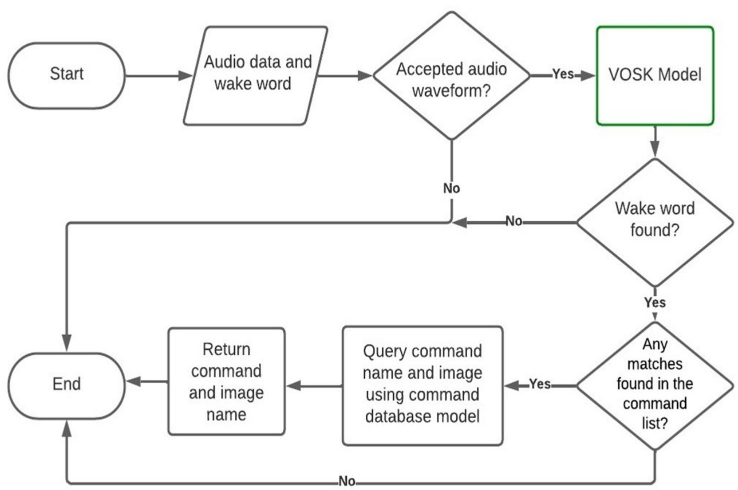

Figure 9. Flowchart of the second stage of the capture and recognition algorithm.

The main function of this stage receives audio data and the activation command as input parameters. These data are supplied by the first stage in each loop cycle, as depicted in the flowchart shown in Figure 9. Initially, it is verified that the audio data represents a complete audio wave, evaluated in real-time through partial interpretation of each supplied frame. If this condition is met, the model returns a text string representing the recognized audio.

Subsequently, it is checked if the activation command is present in the result; if so, the rest of the string is processed. Using the difflib library, recognition accuracy is enhanced by a function that checks similarities between the string and the list of commands. If commands exceeding the established threshold are found, the program queries the database model to obtain the name of the image corresponding to the recognized command, returning a list with the command name and the corresponding image. If no matches are found, a null value is returned.

WiFi Connection Module: This module manages the device's internet connection to connect with cloud services for updates. Development involves encoding and implementing commands in the machine's Shell, allowing retrieval of information related to the device's network. Integration was performed with a Linux system's proprietary controller called Network Manager. Upon accessing the connection view, the module queries information about networks within the WiFi card's range, capturing the response as a text string to be processed, organized, and emitted to the primary thread as a list of dictionaries. This list provides data to the respective views. Additionally, if the user selects a network to connect to, the module attempts to establish the connection and subsequently notifies the result.

Update Module: This module handles the entire process of updating the device's command database. Upon accessing the module, updates are checked by consulting the version of the online database commands and comparing it with the offline database. If updates are available, necessary data, such as commands and image identifiers, is obtained, initiating the update process. It is noteworthy that the update module has been coded to allow users to update commands on the device without the need for developer intervention. Additionally, associated images for these commands can be stored on any cloud service. For downloading, it is sufficient to link the URL or ID generated by the used server in the database. The update status is continuously emitted to the primary thread during the process, and the result is communicated upon completion.

3.2. Command Database Model

The model functions are defined to manage information stored in offline and online databases, along with their structure. When an application controller needs access to information, these model functions serve as the first layer of interaction, acting as intermediaries between controllers and the database. The database is implemented in SQL, using SQLite as the engine for the offline database and MySQL for the online one. It features a table called "commands" with five columns. The "id" column represents a unique integer, functioning as the primary key for each entry. The "command" column is of text type, storing the textual string of the command. The "img" column is a text type that holds the name of the image associated with the command, while "imgId" is an integer related directly to each image. The last column, "ver," is an integer type describing the version of each command.

4. Graphic User Interface

The graphical user interface design was conceived to be both functional and aesthetically pleasing. For this purpose, the Python library PyQt was chosen, specifically its fifth version (PyQt5). This library is linked to the Qt framework, which offers various widgets and also provides a connection to the QtDesigner tool. QtDesigner facilitates the design of graphical user interfaces through a "drag and drop" approach, where components are dragged and arranged in a layout as needed. Given that the primary purpose of the user interface is to visually display voice-issued commands, the background is set to white [11]. The main

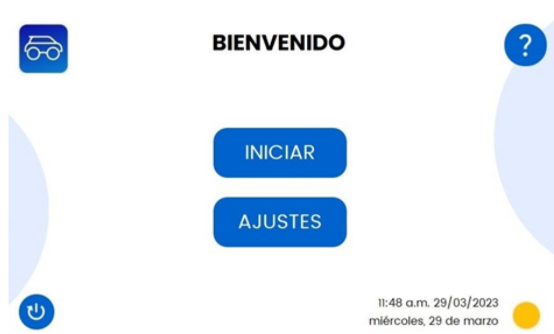

Figure 10. Main view of the interface.

view of the interface is depicted in Figure 10, showcasing four buttons: start, settings and the power/update button located in the bottom left corner.

Upon clicking the "INICIAR" button, the simulation of the practical driving class commences, as illustrated in Figure 11, displaying a GIF while awaiting a voice command. In this view, there are two buttons: one for returning (which navigates to the previous view) and another for pause, serving to pause the simulation. Additionally, there is a section at the bottom displaying "subtitles," which are essentially the detected commands.

Figure 11. Practical class view in waiting state.

When the instructor issues a command and it is detected, the associated GIF or image linked to that command in the database is displayed (Figure 12). After a certain amount of time has elapsed, the GIF for the command transitions to the waiting GIF. This change aims to prevent the learner from getting confused or thinking that the command has been given twice, or that they still need to execute the previous command. Furthermore, this design follows audiovisual accessibility guidelines, particularly those concerning Subtitles for Deaf and Hard-of-Hearing Individuals (SPS). In line with these guidelines, the subtitles feature a gray background to provide contrast against the white application background, and the font used is Tahoma [12].

Figure 12. Practical class view with GIF of an instruction.

In the "AJUSTES" section, the view depicted in Figure 13 is presented. This section is designed to allow users to customize certain features of the prototype. Initially, this section allows the modification of the command voice used to activate command recognition. Secondly, it enables the adjustment of the waiting time for GIFs before transitioning to the command waiting GIF shown in Figure 11. Additionally, users can choose to display or hide the date and time, as well as disable the confirmation sound indicating that the system detected the command and adjust its volume. This view also includes a "back" button, returning to the previous view, and an "Apply" button to save the changes made to the commands.

Figure 13. Interface settings view.

Furthermore, the interface features a view called index, as shown in Figure 14, which serves as an index of commands. This allows the instructor or student to locate the GIFs linked to voice commands and vice versa. Searches can be performed manually or using the search bar for enhanced accessibility.

Figure 14. Commands index view.

To update the prototype's database, it needs to connect to the internet. The initial view for this process is the Wi-Fi connection view (Figure 15), where the user must input the password for the corresponding private network.

Figure 15. Wi-Fi connection view for updates.

If the password is incorrect, an error will appear when attempting to connect. If the network is public and doesn't require authentication, no password entry is necessary. After successfully connecting, the view transitions to the updates view, as depicted in Figure 16. In the event of new commands, the database is updated.

Figure 16. Updates view.

5. Testing

Upon completion of the development phase, tests were conducted to assess the effectiveness and performance of the prototype. The initial prototype database comprises a total of 76 commands representing instructions frequently used in practical driving lessons, allowing for expansion based on user needs. To ensure the relevance and comfort of the instructions for individuals with hearing disabilities, a meeting was held with leaders from the Valledupar Deaf Association (ASORDUPAR), during which the prototype concept was presented, and opinions and suggestions were gathered.

Given that this project is framed by Resolution 20223040030355 and targeted at Driving Schools, it underwent testing in the instructional vehicles of a Driving School Center (CEA) in the city of Valledupar. In Figure 17, the device is observed on the dashboard of a teaching vehicle during a practical session.

Figure 17. Practical class conducted using the Prototype.

At the end of the practical class sessions, a satisfaction survey was conducted with each instructor using Google Forms. The aim was to collect their feedback based on their experience applying driving skill instruction with the prototype.

The survey included the following six questions, which were rated on a linear scale from 1 to 5, as outlined below:

1. Strongly Disagree,
2. Disagree,
3. Neutral,
4. Agree,
5. Strongly Agree.

6. Results

The questions and their corresponding answers in the survey conducted with the Driving School instructors were as follows:

- Do you think the prototype facilitates the learning of driving skills for individuals with hearing disabilities? Answer: 5 (Strongly Agree).
- Do you think the prototype enhances communication and understanding between the instructor and the student with hearing disabilities during driving lessons? Answer: 5 (Strongly Agree).
- Do you think the prototype provides adequate feedback to students with hearing disabilities? Answer: 5 (Strongly Agree).
- Do you recommend the use of this prototype to other instructors who teach individuals with hearing disabilities? Answer: 5 (Strongly Agree).
- Do you think the prototype fulfills the objectives of teaching driving skills to individuals with hearing disabilities? Answer: 3 (Neutral).

Based on the feedback from instructors at the Valledupar Driving School, several shortcomings were identified, and future enhancements for the prototype were planned. Overall, the results were positive, with instructors expressing "Strongly Agree" on 5 out of 6 satisfaction survey questions, and only one "Neutral" response.

Figure 18. Type of representative images for speed variations.

During the presentation of the prototype to ASORDUPAR, the association's leaders provided suggestions to enhance information transmission. They recommended using simple and direct images to represent commands, such as showing an image of the desired speed for gear changes or using finger gestures to indicate "first gear" or "second gear," as illustrated in Figure 18 and Figure 19. Additionally, they proposed training instructors to teach individuals with hearing disabilities using visual hand signals, similar to methodologies employed in countries like the United Kingdom. These suggestions aimed to improve the reception of commands by individuals with hearing disabilities.

7. Future Work

To address potential improvements in the technology and hardware of the developed system, it is suggested to continue working on the optimization of the VOSK voice recognition algorithm, ensuring its accuracy and reliability under various noise conditions and with different accents. This would include extensive testing in controlled and real environments with a larger and more diverse sample of users, as well as the implementation of advanced signal processing techniques to enhance the system's robustness in the presence of background noise and variability in pronunciation. Regarding the hardware, it is

Figure 19. Type of representative images for gear changes.

crucial to evaluate and improve its efficiency and robustness, considering alternatives that offer better performance or lower cost, such as using more advanced devices or more economical and efficient components. Additionally, the integration of the prototype with GPS navigation systems and driving assistance applications could provide more comprehensive and accurate instructions during practical lessons, and the implementation of automatic system update capabilities through the cloud would allow the prototype to stay updated with the latest software improvements and command databases. Furthermore, investigating the integration of the system with teaching methodologies adapted to the specific needs of students with hearing disabilities, collaborating with experts in special education, and implementing innovative pedagogical techniques could significantly improve the learning experience.

8. Conclusions

A prototype was developed as an auxiliary tool for teaching driving skills to individuals with hearing disabilities. This prototype introduces an innovative and inclusive technological approach, enabling their participation in everyday activities alongside hearing individuals. This method presents new methodologies, differing from traditional driver training techniques used in other countries.

When comparing the techniques used in driving instruction, such as those in the United Kingdom, it is evident that the learning curve for mastering the device is shorter than the learning curve required to master the set of signs needed to communicate instructions to a student with hearing impairment. This is because the commands integrated into the prototype are similar to those typically used in practical driving lessons with hearing individuals. However, it is not possible to compare the efficiency of the prototype with the ongoing research in Saudi Arabia, as it is still in the laboratory phase. As for the vibrotactile device developed in Jordan, its target population is individuals with hearing disabilities. Nonetheless, its implementation and functionality differ from those of this prototype, although they could potentially complement each other to create a more robust device

This prototype has the potential to advance the country's development by promoting the creation of supportive tools for the education of people with disabilities. It establishes a foundation for future research aimed at optimizing both software and hardware to enhance efficiency and autonomy. Plans are in place to conduct trials in more driving schools across the region and the country, enabling the evaluation of its effectiveness among a significant population of individuals with hearing disabilities who are in the process of obtaining a driver's license.

Funding: This research received no external funding.

Author contributions: Conceptualization by D.P., G.C and A.D.; Methodology by D.P., G.C and A.D.; Software by D.P. and G.C; Investigation by D.P., G.C and A.D.; Resources by A.D.; Supervision by A.D; Writing – original draft by D.P. and G.C.

Disclosure statement: The authors declare no conflict of interest.

References

[1] Lizeth Macklauth Delgado Amaya, Maryi Betzabeth Herreño Vargas, et al. *Revisión Documental: El estado actual de las investigaciones desarrolladas sobre discriminación hacia personas con discapacidad auditiva en países Latinoamericanos de habla hispana entre los años 2009 al primer trimestre de 2018.* PhD thesis, Corporación Universitaria Minuto de Dios, 2018.

[2] Ministerio de Transporte de Colombia. Por lo cual se modifica el numeral dos del anexo i de la resolución 207 de 2014. https://www.mintransporte.gov.co, April 2022. Resolución 20223040030355.

[3] Jamie Dow, Lucie Boucher, David Carr, Judith Charlton, Linda Hill, Sjaan Koppel, Roy Lilly, Richard Marottoli, Desmond O'Neill, Mark Rapoport, Christine Roy, Bernard Sévigny, Neil Swirsky, Emmanuelle Gagné, Claude Giroux, and Tamara Rader. Does hearing loss affect the risk of involvement in a motor vehicle crash? *Journal of Transport amp; Health*, 26:101387, September 2022.

[4] Admira Beha. Self-assessment of driving abilities of deaf and hearing drivers. *European Journal of Humanities and Social Sciences*, 2(6):70–75, November 2022.

[5] J. Rogers. Teaching deaf people to drive, 2022.

[6] Samah Abbas, Hassanin Al-Barhamtoshy, and Fahad Alotaibi. Towards an arabic sign language (arsl) corpus for deaf drivers. *PeerJ Computer Science*, 7:e741, November 2021.

[7] Mwaffaq Otoom, Mohammad A. Alzubaidi, and Rama Aloufee. Novel navigation assistive device for deaf drivers. *Assistive Technology*, 34(2):129–139, January 2020.

[8] INSOR. Caracterización de ciudadanos, usuarios y grupos de interés, 2020.

[9] Jaeyoung Huh, Sangjoon Park, Jeong Eun Lee, and Jong Chul Ye. Improving medical speech-to-text accuracy using vision-language pre-training models. *IEEE Journal of Biomedical and Health Informatics*, 28(3):1692–1703, 2023.

[10] Rostislav Kolobov, Olga Okhapkina, Olga Omelchishina, Andrey Platunov, Roman Bedyakin, Vyacheslav Moshkin, Dmitry Menshikov, and Nikolay Mikhaylovskiy. Mediaspeech: Multilanguage asr benchmark and dataset. *arXiv preprint arXiv:2103.16193*, 2021.

[11] Joshua M Willman. *Getting Started with PyQt*, page 1–11. Apress, 2022.

[12] Departamento de Cooperación y Asistencia Técnica and Servicio Nacional de la Discapacidad. Recomendaciones sobre accesibilidad audiovisual, 2019.