Basira: An Intelligent Mobile Application for Real-Time Comprehensive Assistance for Visually Impaired Navigation

Amal Alshahrani, Areej Alqurashi, Nuha Imam, Amjad Alghamdi, Raghad Alzahrani
College of Computing-Computer Science and Artificial Intelligence Department,
Umm Al-Qura University, Makkah, Saudi Arabia

***Abstract*—Individuals with visual impairments face numerous challenges in their daily lives, with navigating streets and public spaces being particularly daunting. The inability to identify safe crossing locations and assess the feasibility of crossing significantly restricts their mobility and independence. The profound impact of visual impairments on daily activities underscores the urgent need for solutions to improve mobility and enhance safety. This study aims to address this pressing issue by leveraging computer vision and deep learning techniques to enhance object detection capabilities. The Basira mobile application was developed using the Flutter platform and integrated with a detection model. The application features voice command functionality to guide users during navigation and assist in identifying daily items. It can recognize a wide range of obstacles and objects in real-time, enabling users to make informed decisions while navigating. Initial testing of the application has shown promising results, with clear improvements in users' ability to navigate safely and confidently in various environments. Basira enhances independence and contributes to improving the quality of life for individuals with visual impairments. This study represents a significant step towards developing innovative technological solutions aimed at enabling all individuals to navigate freely and safely.**

Keywords—Visual impairment; mobility application; computer vision; object detection; obstacle detection

I. Introduction

People with visual impairments suffer from many challenges in their daily lives, and among these basic challenges is the difficulty of crossing streets and moving around in public places. The visually impaired have difficulty identifying safe crossing locations and assessing the possibility of crossing. The street is an unfamiliar and risky environment for these individuals, which affects their freedom of movement and independence. There are about 285 million people who suffer from visual impairment all over the world, including 39 million people whose vision is limited (blind), and 246 million people whose vision is impaired, according to statistics from the World Health Organization [1]. The number of blind people in Saudi Arabia is about 159 thousand blind people, according to undocumented statistics Official.

The impact of visual problems on people's daily activities is profound, with simple tasks such as detecting obstacles and finding their stuff becoming difficult. The issue of mobility for visually impaired people is a major concern to this day.

In Table I, we summarized the limitations of the current systems, namely Envisn [2], Glimpse [3], ChirpAR [4], Be My Eyes [5], and BlindSquare [6]. These drawbacks have been identified through a comparison table between these systems and Basira by carefully examining and evaluating these systems, we have identified specific areas where improvements are necessary. With the introduction of Basira, we aim to address these limitations and provide a comprehensive solution that effectively resolves these issues.

TABLE I. COMPARISON BETWEEN BASIRA AND CURRENT SYSTEMS

Application/function	Glimpse	ChirpAR	Be My Eyes	BlindSquare	Basira
Crossroads safely				●	●
detection the obstacles	●				●
Search for objects with voice feedback		●			●
Arabic language support		●	●	●	●

II. Previous Studies

In recent years, significant advancements have been made in applications and digital devices designed to assist individuals with visual impairments. A study conducted by Senjam et al. [6] in 2021 highlighted that smartphones are now widely accepted and less stigmatized compared to traditional assistive devices. The number of apps tailored for people with visual impairments is also on the rise, including options like VoiceOver, Aipoly Vision, TapTapSee, Be My Eyes, Seeing AI, and Seeing Assistant Move. However, many of these applications are not adequately designed to support safe navigation and road crossing.

In 2022, Mehmood et al. [7] explored the needs and challenges faced by blind and visually impaired individuals in Saudi Arabia concerning the availability and use of digital devices. Their online survey, which included 164 participants, revealed that tools like the White Cane, mobile phones, Envision, Seeing AI, VoiceOver, and Google Maps were commonly used. Mobility was identified as the primary reason for using personal devices, with white canes and mobile phones reported by 49% and 84% of respondents, respectively.

Additionally, Montezuma et al. [8] conducted research in 2021 comparing the performance of Orcam MyEye 1 and Seeing AI. Both applications demonstrated over 95% accuracy in recognizing plain text documents, but their accuracy fell to between 13% and 57% for text on curved surfaces. Participants completed 71% of tasks with Orcam MyEye 1 and 55% with Seeing AI.

In the same year, Salunkhe et al. [9] developed an Android-based object recognition app that leverages the smartphone camera to capture real-time images. This app processes images using TensorFlow's object detection API, specifically the SSD algorithm, achieving an accuracy of around 90% in experimental evaluations.

Furthermore, in 2022, See A et al. [10] proposed a system that combines obstacle detection and object recognition in a single application. Utilizing a deep learning model with the YOLO v3 framework for multi-object detection, this system employs the ARCore Depth Lab API from Google to create a 3D depth map. It identifies obstacles and provides audio alerts while recognizing over 90 different classes of objects.

Lastly, Patil et al. [11] in 2022 discussed a mobile application that integrates multiple functionalities, utilizing artificial intelligence and machine learning techniques. This includes the YOLOv3 algorithm for object recognition, a currency recognition model built with TensorFlow and a dataset from Kaggle, which contains over 1,000 different objects.

III. METHODOLOGY

Basira application has been designed to assist blind and visually impaired individuals in navigating their surroundings. The application aims to develop a portable and flexible navigational solution that meets their needs. To achieve this goal, a smartphone with a depth camera is used. The chosen smartphone for this research is the Honor COR-L29, which operates on the Android V9 operating system with a HiSilicon Kirin 970 processor. The phone features 8 GB of RAM and a large 6.3-inch IPS LCD display. The screen includes a notch at the top housing the front camera, and it offers an FHD+ resolution with a 19.5:9 aspect ratio. The screen-to-body ratio is 83%, indicating very slim bezels and the pixel density is approximately 409 pixels per inch. The rear camera setup is dual, consisting of a 16 MP primary sensor and a 2 MP secondary sensor, with a narrow f/2.2 aperture and LED flash. The camera setup supports depth sensing for portrait mode effects.

The development of the Basira application followed a structured methodology aimed at creating a comprehensive solution to aid individuals with visual impairments in navigating their environments. Central to this methodology was the meticulous design of the user interface, which prioritized flexibility and ease of use. The interface was tailored to accommodate various levels of visual impairment, with a focus on integrating intuitive voice commands for seamless interaction.

A. System Requirements

System requirements are a description of what a system should do, how it should behave, what properties it must have, and what are the limitations of the system, and it is divided into functional requirements and non-functional requirements.

- Functional and Data Requirements

1) The user shall be able to switch between the three functions by swiping the screen, whether it is “detection for obstacles/free Walk ","Cross the road" or "Search for objects".

a) The system shall provide different button for each service or provide swiping the screen.

b) The system shall display the interface of the chosen service.

2) The user should have the ability to search for objects in their surroundings using voice commands.

a) The system should be equipped with voice recognition capabilities to accurately process user voice commands and perform object searches based on the provided voice input.

3) The user should receive voice alerts through the app when using the camera feature to cross the road, providing assistance and guidance.

a) The system should integrate computer vision technology to analyze the live camera feed and provide real-time assistance for street crossing. It should detect objects (traffic light colors, crossing Line existing, whether there are cars or bicycles or not) and generate voice alerts to guide the user safely across the road.

b) The system should have a voice alert feature that can convert text-based information into voice output. It should deliver clear and concise instructions to the user regarding the street crossing process.

4) The user should receive audio feedback that provides information about the names of detected obstacles.

a) The system must utilize appropriate technologies to detect and detection for obstacles (free walk) within the camera's field of view. It should analyze the camera feed and accurately recognize obstacles.

b) The system must provide detection of the obstacles to enhance user safety while moving.

5) the user is presented with a welcome interface along with audio instructions.

a) The system displays three welcome interfaces with audio instructions.

b) The system displays the welcome interfaces only once when the application is opened for the first time.

c) The system allows the user to navigate through the welcome interfaces by swiping the screen.

- Non-Functional Requirements

1) Look and Feel Requirements:

a) Basira App will have an interface that is easy to use and accessible to all users. It will be designed with simplicity in mind and will be compatible with screen readers commonly used by blind people.

2) Usability Requirements:

a) Basira App will be designed to be easy to use for blind and visual impairments, providing clear and concise instructions.

b) The app will be organized to save users' time and effort, with intuitive navigation and a logical flow of tasks.

c) The app will be self-explanatory and not require excessive instructions or guidance.

d) The system will accurately and clearly describe objects to the user using a clear voice.

3) Performance Requirements:

a) Basira App's interface will load quickly, providing a seamless user experience.

4) Portability Requirements:

a) Basira App will be compatible with both Android and iOS platforms, ensuring its availability to a wide range of users.

Fig. 1 shows the relationship between user with different use cases and how they interact with the system. Displays the services provided by system, where the user accesses the following services: Search in the surrounding for Objects, Crossing the street, Obstacles Detecting (free walk) functions.

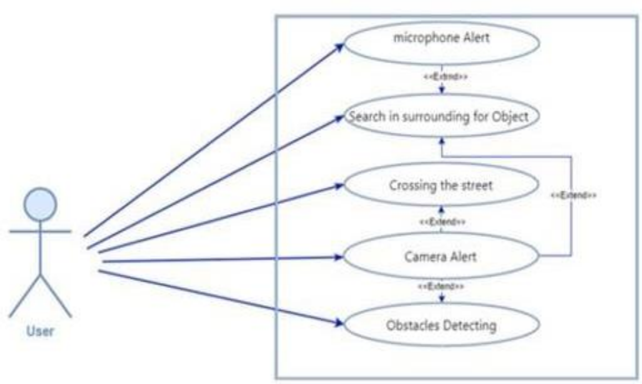

Fig. 1. Use case diagram.

B. Workflow Diagrams

We used the Data Flow Diagrams (DFDs), the data flow diagram is a graphical representation of the data stream within Basira's system. It gives a general overview of the data and system functionality.

Fig. 2 shows context diagram (Level 0) as shown in is a general overview or abstraction view of the whole Basira's system as a single process without any details. It shows the relationship between Basira's system and the external entities User, Text-to-Speech tool, Speech-toText tool and Computer vision tool.

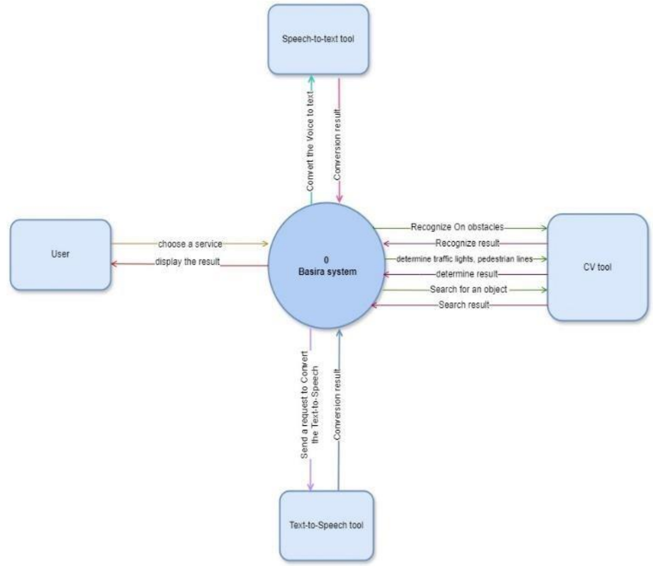

Fig. 2. Data flow diagram Level 0.

Fig. 3 shows the Data Flow Diagram (Level 1), which is a more detailed version of a context diagram. It shows the main processes of a system and the data flows between them. Here, the system has three main processes Obstacles Detecting\free Walk, Crossing the street, Search for objects.

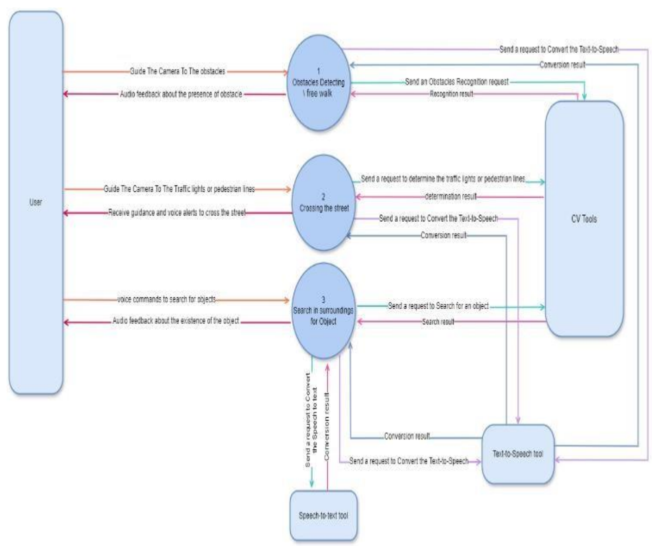

Fig. 3. Data flow diagram level 1.

C. Mobile Development and Implementation

The mobile application was developed utilizing Flutter, a recent and effective mobile framework to build iOS and Android applications from a single code base [12]. The Basira application included the creation of three educational audio interfaces within the application. These interfaces served as informative guides, offering users insights into the features and functionalities of Basira. By providing this educational foundation, users were empowered to navigate the application with confidence and proficiency. The Basira application includes three main functions: Crossing the Road, Free Walking, and Object Searching as shown in Fig. 4.

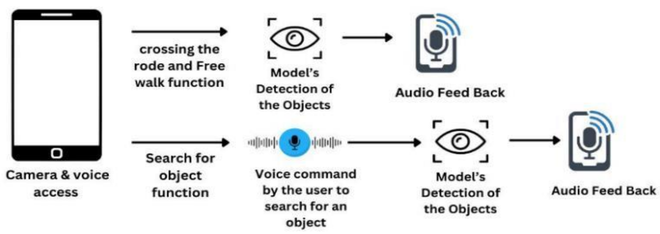

Fig. 4. Shows how the functions of the Basira app work.

A crucial aspect of the Basira application is the implementation of intuitive navigation mechanisms that allow users to easily traverse the app by swiping the screen. This ensures accessibility and enhances the user experience. The user-friendly interface provides quick access to features, offering real-time auditory feedback and voice commands in the object search function. This enhances users' ability to navigate safely and make informed decisions.

D. Development of AI Model

The crossing the road and street obstacle (free walking) dataset was collected from various sources, including Roboflow [13] and the Google Image Search engine. These datasets were carefully categorized to include classes such as pedestrian crossings that alert and guide the visually impaired to the presence of the street, the vehicles and cars encountered during the crossing, and traffic lights with their colours indicating stop, caution, and go. The street obstacle dataset included potholes, vehicles, traffic cones, road barriers, and natural obstacles like branches and trees. Each category in both datasets consisted of 500 carefully selected images to provide a diverse and balanced dataset for model training.

The object detection model was trained on the COCO (Microsoft Common Objects in Context) dataset, which encompasses a wide range of common objects in daily life [14]. The two models were developed and evaluated using the You Only Look Once (YOLOv5) model in study [15].

The crossing the road and free walking model achieved an accuracy of 85.4% and a size of 13.6MB in Pytorch format. The object detection model achieved an accuracy of 94.3% and a size of 14.2 MB in Pytorch format. Both models were built to operate in real-time without any user intervention.

We chose the MVC architecture in Fig. 5 for our system because it provides a clear separation of responsibilities and supports our system's action and reaction approach.

E. Search for Object Function

The 'Object Search function in the Basira application aims to assist blind and visually impaired individuals in finding the items they need. It has been implemented using the YOLOv5s model [16] for real-time object detection. When the user vocally specifies the item they are looking for, the system verifies the detected objects in real time and compares them with the desired item. Upon detecting the desired item, the user is notified via an audio alert. This function enhances the independence of blind users, helping them navigate safely and comfortably by easily identifying items without the need for assistance from sighted individuals.

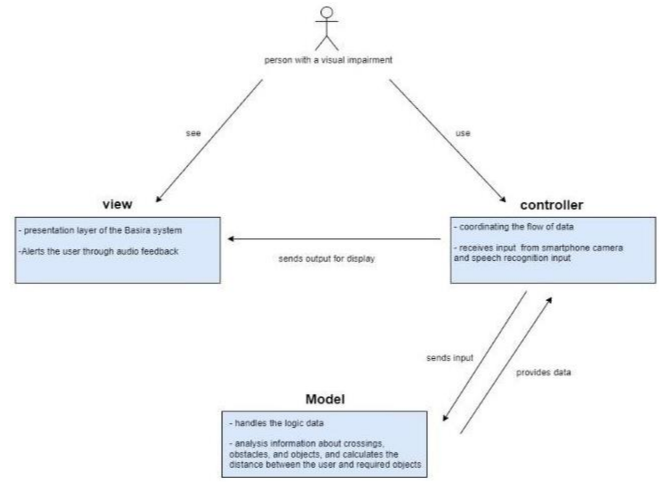

Fig. 5. MVC Diagram.

F. Free Walk / Obstacles Detection Function

The free walk function in Basira application is designed to help visually impaired users navigate their environment by detecting obstacles in their path. This function relies on advanced computer vision algorithm, which is YOLOv5s model, to identify various hazards such as traffic cones, potholes, cars, barriers, and trees through the smartphone's camera feed. When an obstacle is detected, the system provides immediate auditory feedback; alerting the user to the presence of an obstacle and helping them avoid potential hazards.

G. Crossing the Road Function

The crossing the street function in the Basira application is designed to ensure that visually impaired users can safely navigate pedestrian crossings. This function involves detecting traffic signals and crosswalks and providing real-time guidance about when it is safe to cross the street. The system identifies traffic lights and gives auditory guidance, such as indicating when the light is green, and it is safe to cross. Additionally, it detects the presence of vehicles near the crossing area and advises users to wait if necessary.

IV. Results and Discussion

A. User Interface for Basira

Since Basira's target users are visually impaired Arabic native speakers, the language used for both the developed interfaces and audio feedback was Arabic. The interfaces were meticulously designed with a robust emphasis on accessibility and usability for visually impaired users. Drawing upon foundational design principles such as visibility, feedback, constraints, consistency, and affordance [17], these concepts were pivotal in crafting interfaces that are user-friendly and efficient.

As shown in Fig. 6, upon launching the app, users are greeted with the app logo, followed by a welcome interface only if it is their first time using the application. User onboarding is considered important because it helps new users understand how to use the application and navigate its features correctly. It aids in improving the user experience and increasing the likelihood of long-term application usage.

The welcome interface encompasses three distinct screens. Fig. 6(a) showcases the Crossing the Road feature, while Fig. 6(b) presents the Search for Objects interface. Finally, Fig. 6(c) introduces the Free Walking feature. Each interface includes detailed definitions and explanations of the respective application functionalities, accompanied by audio instructions.

Regarding the application's main interface is composed of three straightforward and simple screens. Upon launching the application (excluding the initial launch of the application), the user is presented with the Free Walking interface as the primary screen. This interface displays the camera and includes a sliding visual element that appears as a notification when obstacles are detected, as shown in Fig. 7(a). Swiping left navigates the user to the second interface which is Crossing the Road, It also displays the camera, accompanied by a sliding visual element that appears as a message when a pedestrian lines or traffic light is detected, as depicted in Fig. 7(b). The Search for Objects interface, shown in Fig. 7(c).

(a) (b) (c)

Fig. 6. Basira welcome interfaces: (a) Crossing the Road interface, (b) Search for Objects interfaces, (c) Free Walking interfaces.

(a) (b) (c)

Fig. 7. Basira main interfaces: (a) Free Walking interfaces, (b) Crossing the Road interface, (c) Search for Objects interfaces.

Features a split design: the upper portion displays camera, while the lower part containing a microphone button. By pressing this button, user can voice command to search for specific objects. Each interface provides audio feedback and notifications to the user, enhancing the overall user experience.

B. Testing Results

The Basira application has been tested on a single user from the target audience of individuals with visual impairments (complete or partial blindness), and the individual who has utilized the Basira to undergo testing is a person who is completely blind. Three key functionalities need to be tested by this user: crossing the road, free walking, and searching for objects. We have conducted unit testing, which involves evaluating individual components to ensure their correct operation, to detect and resolve bugs early in the development process.

And it's worth noting that in all the scenarios mentioned below, the user will have their camera open and pointed in the direction they are walking. The application operates in real-time, so the user's gaze will be tracked accordingly.

1) Testing for "crossing the road" function: In the first scenario (Fig. 8) that was tested, a designated crosswalk is present, and a green traffic signal is displayed, indicating that vehicles are required to yield to pedestrians. Additionally, there are no cars or bicycles in the immediate vicinity of the crossing area, it will provide audio feedback in Arabic language stating "***يمكنك العبور***" (you can cross).

Fig. 8. Testing result for "Safe to Cross" condition.

Fig. 9. Testing result for "Red Traffic Light at Crosswalk" condition.

In the second scenario (Fig. 9), both a red traffic light and a designated crosswalk are detected simultaneously. This combination of conditions indicates that it is not safe for pedestrians to cross the road at this time, it will provide audio feedback in Arabic language stating " يوجد خط مشاة الاشارة حمراء لا تعبر"(there is a crosswalk, the light is red, do not cross).

Fig. 10. Testing result for "Cars or Bicycles Present" condition.

In third scenario (Fig. 10), a car is detected, indicating that it is not safe to cross the road. This condition applies regardless of whether the detected vehicles are cars, bicycles, or both. The presence of any moving vehicles near the crossing area poses a significant risk to pedestrians, it will provide audio feedback in Arabic language stating" امامك سيارة لا تعبر " (There is a car ahead, do not cross).

Fig. 11. Testing result for "Safe to Cross (default)" condition.

In fourth scenario (Fig. 11), there are no visible red or yellow traffic signals, and there are no cars or bicycles in the vicinity of the crossing area. Pedestrians are able to cross the crosswalk safely without any anticipated risk, it will provide audio feedback in Arabic language stating" يمكنك عبور الطريق " (you can cross the road).

2) Testing for "free walk / obstacles detection" function: Fig. 12, if Basira detects any obstacles such as: traffic cone, pothole, car, tree or barrier in path of the user, it will provide audio feedback in Arabic language stating ***"امامك معرقل"*** (obstacle ahead).

Fig. 12. Testing results for "free walking (obstacles detection)" function.

Fig. 13. Testing result if no obstacle ahead.

In second scenario in (Fig. 13), if Basira does not detect any obstacles in the user's path, it will not provide any feedback.

3) Testing for "searching for object" function: In the following functionality, the user will utilize their voice. They will press the voice recognition icon and then state the name of the object they are searching for in Arabic.

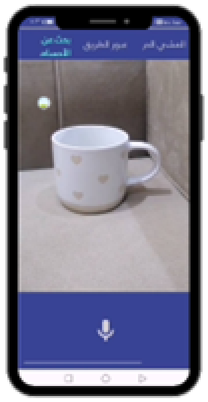

Fig. 14. Testing result for searching about "كأس" cup.

Fig. 14 the user said, **"كأس"** (cup). Then, Basira it will attempt to detect whether the cup is present in front of the camera or not. In this case, the cup is present, so it will provide audio feedback saying **"الشيء الذي تبحث عنه امامك"** (the thing you are looking for is in front of you).

Fig. 15. Testing result for searching about "كأس" cup.

Fig. 15, we will repeat the same experiment but with a different object. The user said, **"كأس"** (cup). Then, Basira will attempt to detect whether the cup is present in front of the camera or not. In this case, the cup is present, so it will provide audio feedback saying **"لم يتم العثور عن الشيء الذي تبحث عنه"** (The object you were looking for was not found). Because the actual object in front of the camera is a book.

V. Limitations and Future Work

Throughout the development phase of our project, we faced several challenges that required innovative solutions. Notably, Flutter lacks built-in support for real-time detection using the YOLO model. To address this, we converted the YOLO model into a TorchScript extension, enabling efficient real-time object detection through frame transmission within our Flutter application. Moreover, time constraints hindered our ability to prepare more data and enhance model accuracy. Initially, we planned to connect Python code with Flutter interfaces and the camera via Flask; however, this setup encountered real-time performance issues, prompting us to rewrite the program entirely in Flutter for better integration.

VI. Conclusion

This study introduces the Basira application, leveraging Deep Learning technology to aid blind and visually impaired individuals. The application offers three primary functions. Firstly, it provides auditory alerts to users regarding obstacles such as trees, barriers, and other impediments on the street. Secondly, it helps users safely cross roads through auditory notifications based on various scenarios, such as pedestrian crossings, traffic light signals (green for crossing, red for waiting), and vehicle presence. Lastly, Basira includes an object search feature, enabling users to vocally inquire about objects in their vicinity. The system responds with auditory alerts confirming the object's presence or absence.

The mobile interface of Basira was designed to look friendly and easy to use. Developed using Flutter, the application is compatible with Android devices. When the application is opened, a welcome interface will appear, showcasing the features offered by Basira. The user is then taken to the free walking (obstacle detection) interface, where they can swipe the screen to smoothly navigate to the road crossing page or the object search page. By seamlessly integrating deep learning models with a user-friendly mobile app, Basira aims to enhance the independence and mobility of blind and visually impaired individuals.

References

[1] World Health Organization. (2010, February 12). Health systems financing: the path to universal coverage. Retrieved from: https://www.who.int/whr/2010/en/

[2] Envision,App,"Envision,[Online].Available:https://www.letsenvision.com/app.

[3] GlimpseAR,Visual,Aid,"AppStore,[Online].Available: https://apps.apple.com/sa/app/glimpse-ar-visual-aid/id1311012359?l=ar.

[4] ChirpAR,"App,Store,[Online].Available:https://apps.apple.com/us/app/chirp-ar/id1439199416.

[5] Be My Eyes Be My Eyes - See the world together. Retrieved from https://www.bemyeyes.com/

[6] Senjam, S. S., Manna, S., & Bascaran, C. (2021). Smartphones- Based Assistive Technology: Accessibility Features and Apps for People with Visual Impairment, and its Usage, Challenges, and Usability Testing. Clinical Optometry, 13, 311-322. DOI: 10.2147/OPTO.S336361. Available at: https://www.tandfonline.com/doi/full/10.2147/OPTO.S336361

[7] Busaeed, S., Mehmood, R., & Katib, I. (2022). Requirements, Challenges, and Use of Digital Devices and Apps for Blind and Visually Impaired. NOT PEER-REVIEWED. Preprints [Online]. Available at: https://www.preprints.org/manuscript/202207.0068/v1

[8] Granquist, C., Sun, S. Y., Montezuma, S. R., Tran, T. M., Gage, R., & Legge, G. E. (2021). Evaluation and Comparison of Artificial Intelligence Vision Aids: Orcam MyEye 1 and Seeing AI. Journal of Visual Impairment & Blindness, 115(4), 277-285. Available at: https://journals.sagepub.com/doi/abs/10.1177/0145482X211027492

[9] Salunkhe, A., Raut, M., Santra, S., & Bhagwat, S. (2021). Android-based object recognition application for visually impaired. *ITM Web of Conferences*, 40, 03001. [Online]. Available: https://doi.org/10.1051/itmconf/20214003001

[10] See, A. R., Sasing, B. G., & Advincula, W. D. (2022). A Smartphone-Based Mobility Assistant Using Depth Imaging for Visually Impaired and Blind. *Applied Sciences*, 12(6), 2802. [Online]. Available: https://doi.org/10.3390/app12062802.

[11] Patil, R., Modi, R., Parandekar, A., & Deone, J. B. (2022). Designing mobile application for Visually Impaired and Blind Persons. SSRN. [Online] Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4108763.

[12] BlindSquare Retrieved from https://www.blindsquare.com/

[13] Flutter-Build apps for any screen." Google, [Online]. https://flutter.dev/

[14] Roboflow. (n.d.). What's New in YOLOv8? Roboflow Blog. Retrieved from https://blog.roboflow.com/whats-new-inyolov8/#yolov8-architecture-a-deep-dive

[15] COCO Consortium. COCO - Common Objects in Context. Retrieved from https://cocodataset.org/#home.

[16] Al-Shahrani, A., Alghamdi, A., Alqurashi, A., Alzahrani, R., & Imam, N. (2024). Real-time comprehensive assistance for visually impaired navigation. International Journal of Computer Science and Network Security, 24(5), 3-4. https://doi.org/10.22937/IJCSNS.2024.24.5.1I. S.

[17] Solawetz, J. (2020). Yolov5 new versionimprovements and evaluation. Roboflow.Seachdate.Retrieved,fromhttps://blog.roboflow.com/yol ov5-improvementsand- evaluation/