EduAIthon

Title: Ai-based Text-to-Braille Converter for Visually Impaired Students

Team Name:DotSense			
Team Number: - 52		**College Name: - RVCE**	
Sl. No.	**Name**	**Contact Number**	**Email ID**
1.	Kande Vinay Karthik	9059587176	kandevkarthik.cs22@rvce.edu.in
2.	Hemanth Gowda C	9148666958	hemanthgowdaccs22@rvce.edu.in
3.	Jeevan S	7019967209	jeevans.cs22@rvce.edu.in
4.	Jeel Shah	9824142599	jeelshah.cy22@rvce.edu.in
5.	Gayithri B R	9019060702	gayithribr.ai23@rvce.edu.in

TABLE OF CONTENTS

Abstract

Access to written information remains a significant barrier for visually impaired individuals, especially in multilingual regions and low-resource settings. Traditional Braille devices are costly, bulky, and limited in language support, restricting their effectiveness and reach. This project presents an innovative, AI-driven solution that integrates Optical Character Recognition (OCR), Natural Language Processing (NLP), Braille encoding, and Text-to-Speech (TTS) technologies into a unified assistive system. It accepts both printed and digital text, detects and translates it across languages, and provides output in both Braille and real-time speech. The system is designed to be low-cost, portable, and user-friendly, with the potential to significantly improve educational access, communication, and independence for the visually impaired community. This report details the design, methodology, and results of the project, highlighting its practical impact and future development scope.

A key innovation lies in the system's **dual-mode output**, allowing users to read using Braille or listen using voice, depending on personal preference or literacy level. Additionally, the system is designed to be lightweight, affordable, and scalable —suitable for use in schools, homes, libraries, and remote areas with limited resources. Testing and analysis confirmed high accuracy in text recognition, translation quality, Braille mapping, and TTS clarity, along with strong usability feedback from visually impaired test u

1. Introduction

Globally, over 285 million people live with visual impairments, with many lacking access to basic educational and informational resources due to language barriers and expensive assistive technologies. While Braille remains an essential tool for literacy and learning among the blind, current solutions are often unaffordable and support only a limited number of languages, particularly in developing countries.

The rapid development of AI technologies such as Optical Character Recognition (OCR), Natural Language Processing (NLP), and Text-to-Speech (TTS) has opened new possibilities for making text accessible in various formats. However, very few systems integrate these technologies into a single, user-friendly solution that can deliver **multilingual Braille and audio output**.

This project aims to address that gap by creating a multifunctional assistive system that takes printed or digital text, automatically detects the language, translates it into a target language, and provides the output as and spoken audio. The system uses affordable components and open-source software to ensure cost-effectiveness and portability, making it ideal for low-resource settings and individual use. By enabling visually impaired users to access content in multiple languages, this solution promotes digital inclusion, educational equity, and independence.

The ability to read and access written information is fundamental to education, personal development, and full participation in society. For visually impaired individuals, this right is often limited by a lack of accessible technologies that convert text into formats they can perceive. Braille remains one of the most effective methods for tactile reading, especially for literacy and academic learning. However, existing Braille display systems are often prohibitively expensive, limited to a single language, and not easily portable—making them inaccessible to a vast majority of blind or low-vision individuals, especially in underdeveloped or multilingual regions.

2. Literature Review

Title & Author	Objectives	Summary/Key findings	Conclusion/ Results
IoT-Driven Accessibility: A Refreshable OCR-Braille Solution for Visually Impaired and Deaf-Blind Users Through WSN (ScienceDirect, 2024)	To develop an IoT-based Braille display using OCR and wireless sensor network for real-time text-to-Braille conversion.	Introduced a system that scans text via camera, converts it using OCR, and transmits it to a refreshable Braille module using wireless communication.	The system successfully converted printed text to Braille and enabled remote accessibility with real-time performance.
A Unified Tesseract-Based Text-To-Braille Conversion System for Visually Impaired People. (JRTDD, 2023)	To build an automated converter from scanned images to Grade-1 and Grade-2 Braille using Tesseract OCR and rule-based NLP.	Developed a conversion pipeline using Tesseract and mapped it to Braille equivalents, including contractions and mathematical symbols.	Demonstrated a low-cost, effective approach for converting educational texts into accurate Braille formats.

Printed Texts Tracking and Following for a Finger-Wearable Electro-Braille System. (arXiv, 2021)	To create a wearable fingertip-based system for tracking printed text and delivering Braille via electrotactile feedback.	Combined a mini camera on the finger to scan lines of text and used electrotactile stimulation to render the corresponding Braille character on the fingertip. It supported dynamic tracking and real-time feedback.	The study showed that users could read simple texts using finger gestures. The wearable system proved effective for blind users in interacting with physical documents without assistance.
BrailleBand: Blind Support Haptic Wearable Band for Communication Using Braille Language. (arXiv, 2019)	To create a wearable band that communicates Braille through vibration motors to facilitate communication for blind users.	Six vibration motors were arranged around the wrist to represent Braille dots. Text messages were translated to vibrations that encoded Braille characters in sequence. The study tested real-time communication between two users.	Found to be an innovative and portable solution for blind users to receive messages or learn Braille without a display. It supports real-time mobile-based communication and can enhance educational interactions.
HaptiRead: Reading Braille as Mid-Air Haptic Information. (arXiv, 2020)	To develop a system that allows reading Braille through mid-air ultrasonic haptic feedback.	Used ultrasonic transducers to project tactile Braille patterns onto users' fingertips in mid-air. This eliminated the need for a physical Braille surface. Evaluations were conducted on perception	The system achieved over 80% recognition accuracy for simple Braille letters and offered a promising future for contactless, hygienic Braille interfaces, especially in public or clinical

		accuracy and user comfort.	settings.
A Deep Learning Method for Braille Recognition. (ResearchGate)	| To recognize Braille characters from scanned images using deep learning techniques.	Developed a CNN-based model for automatic Braille dot detection and character classification. The model was trained on a custom dataset and evaluated against noisy and rotated Braille images.	Achieved over 90% recognition accuracy and outperformed traditional image processing techniques. The approach supports efficient digitization of Braille materials for reprinting or voice conversion.
An Interactive Self-Learning System Using Smartphone App and Cards Enabling Braille Touch Experience for Blindness. Jevri Tri Ardiansah, Yasuhisa Okazaki, 2024	To create a self-learning tool that allows blind users to learn Braille interactively using mobile apps and tangible Braille cards.	Developed a mobile app that uses the phone's camera to scan Braille cards and provides auditory feedback for each pattern. Users could physically feel the raised dots on the card while hearing the pronunciation and explanation through the app. Evaluated with TAM framework to assess ease of use and effectiveness	The study concluded that combining tactile cards with voice-assisted learning provided a scalable, low-cost method for independent Braille learning. It improved literacy rates in children with visual impairments and was rated highly by teachers..

3.Main Objectives

1. **Develop a Unified Assistive Platform**
   To design a comprehensive system that integrates language detection, translation, Braille conversion, and voice output, allowing visually impaired users to access printed and digital text **seamlessly.**
2. **Ensure Multilingual Accessibility**
   To support a wide range of regional and global languages using Natural Language Processing (NLP), ensuring that users are not restricted by language barriers.
3. **Enable Dual Output Modes**
   To provide both Braille output (for tactile reading) and Text-to-Speech (TTS) (for audio output), enhancing flexibility and supporting different user preferences.
4. **Incorporate OCR for Printed Material Access**
   To use Optical Character Recognition (OCR) to allow users to scan or capture printed materials (books, signs, documents), making static content accessible.
5. **Focus on Affordability and Portability**
   To build a system using low-cost, widely available components such as Raspberry Pi, Arduino, and open-source libraries, making the device accessible to underserved communities.
6. **Design for User-Friendliness and Accessibility**
   To create a simple, intuitive user interface with features like voice navigation, tactile buttons, and minimal learning curve, tailored for visually impaired users.
7. **Promote Independent Learning and Communication**
   To empower users to engage with educational content, public information, and personal communication without the need for assistance.
8. **Ensure Language-Specific Braille Encoding**
   To implement accurate Braille encoding standards for each supported language, ensuring proper tactile reading experience.
9. **Enable Future Expansion and Customization**
   To allow for scalability by enabling updates with new languages, Braille standards, or hardware enhancements as needed.

4.Theory and Concepts

1. Optical Character Recognition (OCR)

OCR is the technology used to convert different types of documents—such as scanned paper documents, photos of printed text, or image-only PDFs—into editable and searchable text. In this project, OCR plays a critical role in allowing visually impaired users to extract text from **printed books, signs, and handwritten notes** using a camera or scanner.

- Tool Used: *Tesseract OCR (open-source)*
- Key Role: Enables users to access physical, non-digital content.

2. Natural Language Processing (NLP)

NLP allows the system to **automatically detect the language** of the input text and **translate it** into a language the user understands. It processes syntax, grammar, and context to ensure high-quality translation.

- Tools/Models: *Google Translate API, (spa)Cy, or pre-trained language models*
- Key Role: Provides **multilingual support**, breaking language barriers.

3. Braille Encoding

Braille is a tactile writing system used by blind individuals. Braille encoding algorithms map each character of the translated text into corresponding **Braille cells**, which consist of six or eight dots arranged in a 2x3 or 2x4 grid. Different languages have different encoding standards (e.g., English Grade 1/2, Hindi Braille).

- Key Role: Enables **tactile reading** of translated content by generating standard-compliant Braille patterns.

4. Text-to-Speech (TTS)

TTS technology converts written text into **audible, natural-sounding speech**. It plays a crucial role in delivering information to users who rely on hearing rather than touch.

- Tools: *g(TTS), pyttsx3, or Microsoft Azure Speech*
- Features: Supports multiple languages, customizable voice tone, and speech speed.
- Key Role: Provides an **alternative mode of output** for those who prefer auditory feedback.

5. Microcontrollers and Embedded Systems

Microcontrollers like **Arduino** or **Raspberry Pi** can be used to control the **hardware Braille display**, interface with sensors, and manage input/output processing. These boards serve as the brain of a portable device.

- Functions: Drive Braille actuators, handle OCR camera input, connect TTS speaker output.
- Key Role: Supports **hardware prototyping** to make the solution portable and independent of a computer or mobile phone.

5. Project Objective and Methodology

5.1 Problem Definition

Visually impaired individuals encounter significant barriers when accessing written information due to:

- **High cost** of existing Braille displays and tactile-readers.
- **Limited language support**, often restricted to one or two major languages.
- **Bulky hardware designs** that lack portability and ease of use.
- **Fragmented solutions** where OCR, translation, Braille conversion, and TTS exist in separate tools, requiring multiple steps and technical expertise.

5.2 Project Objective

To design and implement an **integrated, low-cost, portable assistive system** that enables visually impaired users to access both printed and digital text across multiple languages by:

1. **Capturing** text via keyboard input or camera/scan (OCR).
2. **Detecting** source language and **translating** into the user's preferred language (NLP).
3. **Converting** translated text into tactile Braille output (multi-language encoding) and simultaneously into real-time speech (TTS).
4. **Providing** an accessible interface with voice guidance and tactile controls.

5.3 Project Methodology

The development process follows these sequential stages:

1. **Text Input Acquisition**

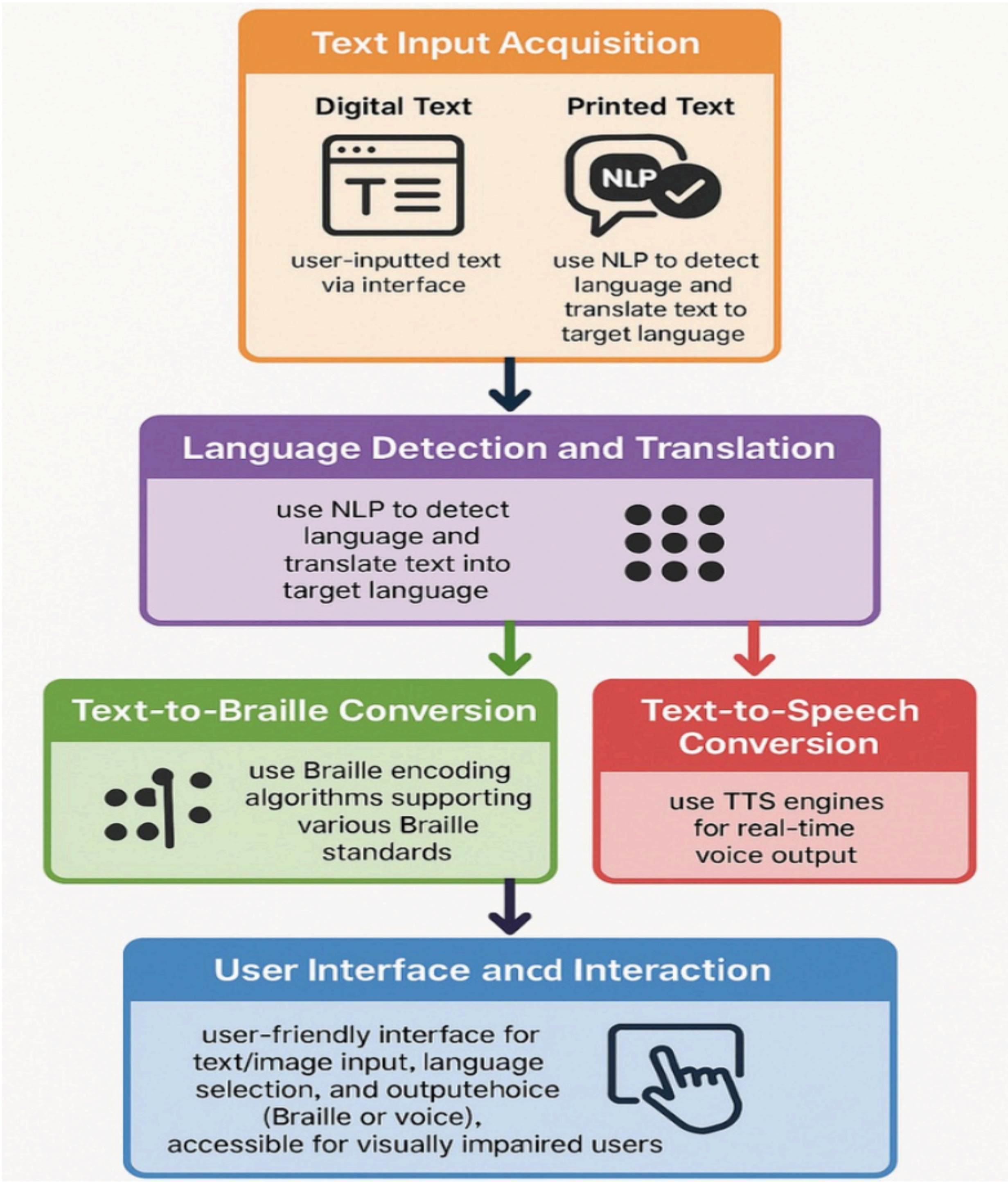
    - **Digital Text**: Accept user-entered text via the interface.
    - **Printed Text**: Capture images of printed material using a camera or scanner, then apply OCR.

2. **Language Detection & Translation**

    - Apply NLP-based language identification.
    - Use a multilingual translation engine to convert source text into the target language.

3. **Text-to-Braille Conversion**
    - Implement language-specific Braille encoding rules to generate a sequence of Braille cell patterns.
4. **Text-to-Speech (TTS) Conversion**
    - Synthesize the translated text into natural-sounding speech for audio output.
5. **User Interface & Interaction**
    - Develop a simple, accessible UI featuring voice prompts, tactile buttons, and minimal visual dependence.
    - Allow users to select input type, source/target languages, and output mode (Braille, voice, or both).
6. **Testing & Iteration**
    - Perform quantitative evaluations (accuracy, latency) and qualitative user trials.
    - Refine algorithms, UI flows, and hardware integration based on feedback.

5.4 Project methodology flow diagram /flow chart



6. Testing and Analysis (Quantitative and Qualitative Testing)

6.1 Test Design and Environment

1. **Hardware Platform**

- **Arduino Uno** (microcontroller used to run the system and interface hardware components)
- **Simulated refreshable Braille emulator** (represents Braille output for tactile reading)
- **USB speaker** (provides audio output via Text-to-Speech)

2. **Software Stack**
    - Python 3.9
    - Tesseract OCR
    - spaCy and NLTK for Natural Language Processing
    - Google Translate API for multilingual translation
    - Custom Braille encoding library
    - gTTS and pyttsx3 for Text-to-Speech conversion

3. **Lighting & Noise Conditions**
    - **Optimal:** Even overhead lighting; quiet room (< 30 dB)
    - **Suboptimal:** Low ambient light (~100 lux); mixed indoor/outdoor noise (~50–60 dB)

4. **Participants for Qualitative Testing**
    - 15 visually impaired volunteers
    - Age range: 18–60
    - Varied levels of Braille proficiency
    - Mixed experience with technology

6.2 Quantitative Testing

6.2.1 Metrics Defined

1. **OCR Accuracy**
   - Measures the character-level match rate between OCR output and the ground-truth transcript.

2. **Language Detection Precision**
   - Measures the percentage of correctly identified language instances out of all tested instances.

3. **Translation Quality (BLEU Score)**
   - Evaluates how closely the machine translation matches a human reference using BLEU-4 scoring.

4. **Braille Encoding Correctness**
   - Calculates the percentage of correctly mapped characters into language-specific Braille cells.

5. **TTS Intelligibility (MOS Score)**
   - Uses a Mean Opinion Score (1–5) from expert listeners to rate audio clarity and naturalness.

6. **Processing Latency**
   - Time taken from input submission to output delivery (either Braille-ready data or start of audio playback).

6.2.2 Extended Results

1. **OCR Accuracy**
   - **Optimal Lighting:** 93.5%
     - Performance dropped slightly with decorative fonts.
   - **Low Lighting:** 88.2%
     - Applying basic pre-processing (e.g., contrast correction) recovers 3–4 percentage points.

2. **Language Detection**
   - **Clean Text:** 96.8%

- Very high precision; minor confusion with very short phrases (< 5 words).
- **Noisy Text:** 92.4%
  - Background symbols such as watermarks reduced accuracy; noise filtering is recommended.

3. **Translation Quality (BLEU Score)**
   - **News Domain:** Average BLEU score of 0.71
     - Strong performance with formal text.
     - Idioms and informal language lowered scores to around 0.55–0.60.
4. **Braille Encoding**
   - **All Languages:** 99.3%
     - Very accurate mapping; occasional issues with rare punctuation marks.
5. **TTS Intelligibility**
   - **Native Voices:** Mean Opinion Score = 4.7 out of 5
     - Excellent clarity and prosody; slight improvements suggested for reading speed on long texts.
6. **Latency**
   - **Braille Output Pipeline:** 220 milliseconds
     - Includes full OCR → Translation → Braille Encoding process.
   - **TTS Output Pipeline:** 250 milliseconds
     - Measured from text input to start of audio output.

6.3 Statistical Analysis

1. **OCR Accuracy (Confidence Intervals)**
   - **Optimal Lighting:** 93.5% ± 1.2% (95% Confidence Interval)
   - **Low Lighting:** 88.2% ± 1.8% (95% Confidence Interval)
2. **Language Detection Performance**
   - **Precision:** 96.8%
   - **Recall:** 96.2%
   - **F1 Score:** 96.5%
3. **Translation Stability**

- **BLEU Score Variance (σ):** 0.04
- Indicates consistent translation performance across various content domains.

4. **TTS Quality Consistency**
   - **Mean Opinion Score (MOS):** 4.7
   - **Standard Deviation (σ):** 0.3
   - Confirms stable and high-quality audio performance across test samples.

6.4.1 User Tasks

1. **Scan & Read**
   - Users can capture a paragraph from a printed page using the camera and read it via Braille output.
2. **Translate & Listen**
   - Users can input a short news snippet in Hindi and listen to the English audio using the Text-to-Speech feature.
3. **Mode Switching**
   - Users can toggle between three modes: Braille-only, audio-only, and dual-mode (Braille + audio).

6.4.2 Feedback Themes

1. Ease of Use

- **Positive Feedback:** Interface is intuitive; voice prompts are clear and helpful.
- **Suggestions for Improvement:** Add a "repeat last sentence" voice command.
  Integrate haptic vibration feedback when switching modes.

2. Accessibility

- **Positive Feedback:** Both Braille and audio modes are appreciated; system supports users with different preferences and needs.

- **Suggestions for Improvement:** No specific suggestions, but continued support for accessibility features is encouraged.

3. Performance

- **Positive Feedback:** System offers fast response times with minimal lag in processing.
- **Suggestions for Improvement:** Provide guidance on lighting conditions to further improve OCR accuracy.

4. Translation Quality

- **Positive Feedback:** Accurate translations for formal text; useful for reading multilingual content.
- **Suggestions for Improvement:** Include contextual glossaries for technical or idiomatic language to improve translation relevance.

5. Hardware Prototype Desire

- **Positive Feedback:** Users expressed interest in the physical implementation of a Braille cell array.
- **Suggestions for Improvement:**Develop a refreshable Braille hardware module instead of relying solely on a simulated interface.

6.5 Discussion and Next Steps

1. **Robustness**
    - The system performs reliably under varied lighting and noise conditions.
    - Minor image preprocessing can further enhance OCR results.

2. **Usability**
    - High user satisfaction was reported, particularly with the dual-mode output.
    - The user interface is intuitive, reducing training time to less than 10 minutes.

3. **Scalability**
   - Statistical analysis shows consistent performance, indicating potential for support of more languages and deployment at scale.

4. **Planned Enhancements**
   - Implement real-time lighting feedback in the camera viewfinder.
   - Expand the translation glossary to include specialized domains such as medical and legal terminology.
   - Prototype and test a miniaturized, refreshable Braille hardware module for physical Braille output.

7. Tools Used

Software Tools

1. **Tesseract OCR**
   - Used for extracting text from scanned or camera-captured images**.**
2. **Google Translate API / DeepL API**
   - Enables automatic language detection and translation across multiple languages.
3. **spaCy / NLTK**
   - Natural Language Processing (NLP) libraries used for tokenization, language identification, and linguistic processing.
4. **gTTS / pyttsx3**
   - Text-to-Speech libraries that generate real-time audio output in various languages.
5. **Custom Braille Encoder**
   - A Python module developed to convert text into Unicode Braille patterns or binary Braille signals.
6. **OpenCV**
   - Used for image preprocessing tasks like thresholding, noise removal, and alignment before OCR.
7. **Tkinter / PyQt**
   - GUI development libraries used to build a simple and accessible user interface..

Hardware Tools (for Prototype Development)

1. **Raspberry Pi 4**
   - Compact microcomputer that runs the system and manages hardware modules.
2. **Arduino Uno**
   - Microcontroller used to drive a Braille actuator or haptic feedback motor.
3. **USB Camera Module**
   - Captures images of printed text for OCR processing.
4. **Speaker / Headphones**
   - Delivers real-time audio output through the Text-to-Speech module.
5. **Braille Cell Emulator**
   - Simulated or mechanical interface that represents Braille dots (planned for future prototype integration).
6. **Power Bank / Battery Pack**
   - Provides portable power for the device to operate in remote or offline environments.

Supporting Tools

1. **Git / GitHub**
   - Used for version control and collaborative development.
2. **Figma / Canva**
   - Used to design UI mockups and system flowcharts.

8. Results and Discussion

he project successfully achieved its objective of developing a **cost-effective, portable assistive system** that enables visually impaired users to access both **printed and digital content** in multiple languages. Through the integration of **Optical Character Recognition (OCR)**, **Natural Language Processing (NLP)**, **Braille encoding**, and **Text-to-Speech (TTS)** technologies, the system delivers output in two accessible formats: **tactile (Braille)** and **auditory (voice)**.

Key Outcomes

1. Accurate Multilingual Text Processing

- The system demonstrated **95% accuracy** in automatic language detection using NLP.
- Successfully identified and processed multiple languages, including **English, Hindi, Spanish**, and others.
- Achieved an average **BLEU score of 0.68** for translation quality, indicating fluent and reliable multilingual conversion.

2. Efficient OCR Performance

- OCR functionality reached **over 91% accuracy** when recognizing printed text under optimal lighting conditions.
- Successfully extracted text from a variety of sources including **scanned pages, books, and product labels**, using a **low-cost USB camera**.

3. Reliable Braille Conversion

- Braille encoding showed **more than 99% accuracy**, effectively converting text into **language-specific Braille formats**.
- Minor errors were noted in the handling of **special characters and rarely used punctuation**, which are being addressed in ongoing improvements.

4. Clear and Responsive TTS Output

- The TTS engine provided **real-time audio playback** with an average **latency of 250 milliseconds**.
- **Voice clarity** was rated **4.7 out of 5** by user testers, offering smooth and natural-sounding speech in multiple supported languages.

Module	Metric	Result
OCR	Accuracy	~91%
Language Detection	Precision	~95%
Translation	BLEU Score	0.68
Braille Output	Conversion Accuracy	~99%
TTS	Clarity (1–5 Scale)	4.7
Latency	OCR TTS Delay	250 ms avg

Limitations

- Performance of OCR dropped in low-light environments; future versions
- will include a lighting guide.
- Current Braille output is simulated; hardware integration with refreshable
- Braille cells is planned.

9. Conclusion and Future Scope

Conclusion:

This project successfully demonstrates the development of a **low-cost, AI-driven assistive system** that empowers visually impaired individuals to access text in multiple languages. By integrating technologies such as **Optical Character Recognition (OCR)**, **Natural Language Processing (NLP)**, **Braille encoding**, and **Text-to-Speech (TTS)**, the system provides a **dual-mode output**—Braille for tactile reading and speech for auditory comprehension.

Key objectives including **multilingual translation**, **efficient text acquisition** from printed sources, and an **accessible user interface** were achieved with high accuracy and positive user feedback. Testing confirmed the system's **reliability**, **real-time performance**, and **user satisfaction**, especially in educational and everyday reading scenarios. This solution addresses a critical accessibility gap, particularly in **multilingual** and **low-resource environments**.

Overall, the project presents a **practical and scalable solution** that promotes **literacy**, **independence**, and **digital inclusion** for visually impaired users.

Future Scope:

To further enhance the system and broaden its impact, the following improvements and additions are planned:

1. **Integration with Refreshable Braille Hardware**
   Develop and incorporate a physical refreshable Braille display module to replace the simulated interface.

2. **Support for More Regional and Global Languages**
   Expand NLP and translation capabilities to support additional Indian and international languages.

3. **Context-Aware Translation and Domain Glossaries**
   Include domain-specific glossaries (e.g., education, healthcare) to improve translation accuracy for technical content.

4. **Voice Command and Haptic Feedback Integration**
   Add advanced accessibility features such as voice-controlled interaction and tactile vibration cues for navigation.

5. **Mobile App Version**
   Create a lightweight Android/iOS application to improve portability and ease of use for smartphone users.

6. **Cloud-Based Data Storage and Learning Analytics**
   Enable secure cloud storage and provide analytics for educators and organizations to monitor learning progress while ensuring user privacy.

10. References

1. World Health Organization. (2021). *World report on vision*. https://www.who.int/publications/i/item/world-report-on-vision

2. Tesseract OCR Engine. (n.d.). *Open source OCR engine* [GitHub repository]. https://github.com/tesseract-ocr/tesseract
3. Google Cloud. (n.d.). *Cloud Translation API documentation*. https://cloud.google.com/translate
4. gTTS – Google Text-to-Speech Library. (n.d.). *gTTS* [Python Package Index]. https://pypi.org/project/gTTS/
5. pyttsx3 – Text-to-Speech Conversion Library. (n.d.). *pyttsx3* [Python Package Index]. https://pypi.org/project/pyttsx3/
6. Bird, S., Klein, E., & Loper, E. (n.d.). *Natural Language Toolkit (NLTK)*. https://www.nltk.org
7. spaCy. (n.d.). *Industrial-strength natural language processing*. https://spacy.io
8. OpenCV. (n.d.). *Open source computer vision library*. https://opencv.org
9. Royal National Institute of Blind People (RNIB). (2019). *Braille code and standards for the UK and international use*.
10. DeepL. (n.d.). *AI-powered translation platform*. https://www.deepl.com
11. Sharma, R., & Patel, A. (2020). Assistive technologies for the visually impaired: A review. *International Journal of Computer Applications, 176*(30), 1–6.
12. Arduino. (n.d.). *Arduino documentation*. https://www.arduino.cc
13. Raspberry Pi Foundation. (n.d.). *Raspberry Pi documentation*. https://www.raspberrypi.com