Handwritten input to speech conversion using transfer learning

Vema Lakshmi Sowjanya[1], *Behara* Gunakara Patro[1*], *Pilli* Chandra Sai Teja[1], *Yadla* Rohit[1] and *Tota* Kirti[1]
[1]Computer Science and Systems Engineering, Lendi Institute of Engineering and Technology, Vizianagaram, India

Abstract. System developed to transform handwritten text into clear, natural sounding speech attained by using a structured multi-step process. The steps include image preprocessing,clean and enhanced output,elimination of noise and improvement of readability.The unrefined input text is refined using Natural Language Processing(NLP).This refined text is transformed into human-like speech using advanced conversion methods.The system ensures improved recognition,accuracy,effective noise reduction and contextual correction providing smooth and natural speech output.The system is highly useful for education,healthcare,assistance and particularly for aiding visually challenged individuals.The combination of computer vision,deep learning,and language preprocessing bridges the gap between handwritten information and digital accessibility.

Keywords: Handwritten Text, Recognition, Transfer Learning, Assistive Technology, Text-to-Speech, Inclusive Education, Tacotron2, Wave Glow

1 Introduction

Accessing Handwritten educational materials remains a major challenge for people with

* Corresponding author: bgp2952005@gmail.com

traditional Optical Character Recognition(OCR) systems or screen readers,which mainly work with printed text.Because handwriting varies in shape,size,spacing,and style,it becomes difficult for conventional OCR systems to recognize it accurately.As a result,visually impaired learners often struggle to access handwritten information,limiting their independence and participation in educational environments.Recent advances in deep learning have greatly improved Handwritten Text Recognition(HTR).However,these models usually require very large,labeled handwriting datasets and significant computational resources to perform well.This makes them difficult to use in real-world assistive applications.To address this,transfer learning offers a more practical and efficient approach.It allows models that are already trained on large image datasets to be adapted for handwriting recognition tasks,even when only a small amount of handwriting data is available.In this work,transfer learning is used in the HTR module of the proposed system.A Convolutional Neural Network(CNN) pre-trained on the ImageNet dataset is used to extract visual features from handwritten text.These features are then passed through LSTM and CTC layers to recognize and decode the sequence of handwritten characters.By finetuning the pre-trained CNN on handwriting datasets like IAM and RIMES,the model learns handwriting-specific features while keeping its ability to generalize across different writing styles.Once the text is recognized,it is sent to Text-to-Speech(TTS) module that uses Tacotron2 and WaveGlow(or HiFi-GAN) to produce natural,human-like speech.This enables handwritten content to be instantly converted into spoken form,giving visually impaired users immediate access to handwritten notes and learning materials.The proposed system is designed to be affordable,fast,and user-friendly,making it suitable for both classroom and personal use.By combining transfer learning-based HTR with neural TTS,this study introduces a practical,inclusive,and effective assistive solution that enhances accessibility,independence,and learning experiences for visually impaired individuals. Furthermore, the proposed architecture emphasizes low-latency inference and scalability,allowing the model to run efficiently on edge devices such as smartphones,tablets,or low cost single-board computers.This ensures that users can benefit from real-time audio feedback without requiring expensive hardware or constant internet access.The mobile friendly design also makes the system highly adaptable for integration into educational institutions and assistive learning centers.Another important aspect of this work lies in its user-centered design philosophy.The system has been developed and refined with continuous feedback from visually impaired users to ensure ease of interaction,minimal complexity,and maximum usability.This participatory approach not only enhances system effectiveness but also ensures that the solution truly aligns with the real-world needs and experiences of its target users.Beyond its immediate educational applications,the system has broader potential in other domains such as banking,healthcare,and document management,where handwritten forms or notes are frequently encountered.By extending the pipeline with multilingual support and domain adaptation,the framework can serve as a universal tool for handwriting accessibility across multiple languages and contexts.In additional,the proposed model prioritizes data privacy and ethical AI deployment.since assistive systems often handle sensitive educational and personal information,the architecture incorporates secure data handling,on-device inference options,and privacy preserving mechanisms to protect user data.This ensures compliance with accessibility standards and ethical AI principles while maintaining trust among users.The integration of transfer learning within the HTR framework not only reduces the dependence on large-scale handwriting datasets but also significantly enhances adaptability to diverse writing styles.This approach leverages pre-trained visual models that have already learned generic features such as edges,textures,and shapes from massive datasets like ImageNet.When fine-tuned for

CTC for sequential decoding ensures that the system captures both spatial and temporal dependencies in handwriting.Such an architecture improves recognition accuracy and minimizes the need for complex preprocessing techniques,making the system more robust and efficient in real-world conditions. Moreover, the integration of neural Text-to-Speech synthesis using Tacotron2 and WaveGlow (or HiFi-GAN) ensures that the recognized handwritten content is converted into speech that is not only intelligible but also expressive and natural-sounding. These advanced TTS architectures use sequence-to-sequence modeling and neural vocoders to generate speech with realistic prosody, tone, and rhythm. This enables visually impaired users to engage with handwritten content in a more interactive and personalized way. Combined with lightweight deployment strategies and accessibility-focused UI design, the proposed system bridges the gap between assistive learning technology and mainstream education. It provides a scalable foundation for future research in cross-lingual handwriting recognition, multimodal accessibility systems, and AI-driven inclusive learning tools, contributing to a more equitable digital ecosystem for all learners. Although the individual components used in the pipeline are based on established architectures, the novelty of this work lies in the integration of heterogeneous modules into a unified, fully functional handwritten-to-speech system tailored explicitly for assistive accessibility. The engineering contribution is significant in terms of end-to-end optimization, on-device deployment, user-centered evaluation, and task-oriented adaptation. Unlike prior studies that treat HTR and TTS separately, this work delivers a cohesive workflow, error-corrected text normalization, and a real-time speech interface, which together provide practical value and societal impact for visually impaired users.

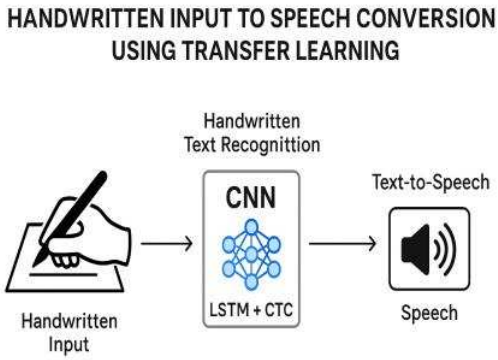

Fig. 1. Handwritten Input to Speech Conversion.

2 Related work

Over the past decade handwriting recognition has transformed significantly moving from classical feature-engineered models to data-driven deep learning architectures. Early character classification relied on manually designed features like Histogram of Oriented Gradients (HOG) and Support Vector Machines (SVM). These struggled with cursive or irregular handwriting Graves et al. [6]'s introduction of RNNs and LSTMs revolutionized sequential handwriting recognition by enabling temporal context understanding. Later developments incorporated Convolutional Neural Networks to extract spatial features and Connectionist Temporal Classification for alignment-free decoding leading to better recognition of unconstrained handwriting. Recent advances in Handwritten Text Recognition

(HTR) are using hybrid and transformer-based models that bring together CNN encoders with attention-driven decoders. Models such as CRNN and Vision Transformers (ViT) [21]show better accuracy on difficult datasets like IAM, RIMES, and EMNIST Researchers frequently leverage transfer learning to bolster generalization. Leveraging established architectures like ResNet, VGG [20], and EfficientNet, pre-trained on extensive image collections, allows for effective adaptation to handwriting recognition, even with restricted task-specific datasets. This transfer of visual feature representations speeds up convergence and improves performance significantly, making it suitable for real-world assistive applications where labeled handwriting data is scarce Text-to-Speech systems have advanced significantly, moving from older methods like concatenative and parametric synthesis to modern end-to-end deep neural models. Tacotron, Tacotron2 [16], FastSpeech, and FastSpeech [17] use attention-based encoders and decoders to create natural and expressive speech Vocoders enable synthesized speech generation. Recent developments in transformer-based handwriting recognition, including TrOCR[14] and ViT-based architectures, have demonstrated substantial performance gains on modern datasets. In multilingual TTS, unified models have shown improved stability for low-resource languages. Real-time deployment research highlights lightweight CNN-Transformer hybrids optimized for edge devices. These works complement the present study and justify the choice of transfer learning for efficient system integration.

3 System architecture

3.1 Overall pipeline

The system operates through five coordinated stages encompassing input acquisition, preprocessing, handwritten text recognition, language-level normalization, and speech synthesis. Input is obtained either from smartphone camera capture or scanned document uploads. Preprocessing improves readability through noise reduction, grayscale normalization, deskewing, contrast enhancement, and line segmentation. These refined images enter the recognition module, which converts handwritten content into machine-readable text. A normalization layer refines punctuation, abbreviations, and numeral patterns before forwarding the cleaned text to the TTS module. This pipeline ensures stable, real-time performance suitable for both assistive and standalone applications.

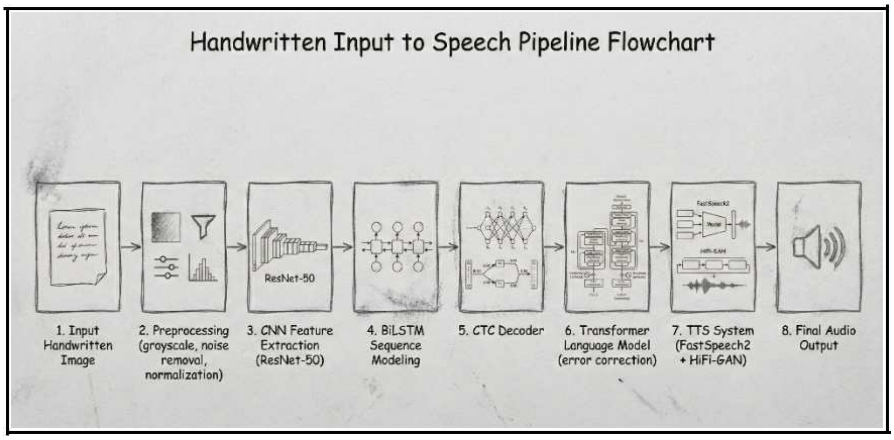

Fig. 2. End-to-end System Pipeline.

3.2 Model architecture

The model architecture is designed as a hybrid vision-sequence-language pipeline. ResNet-50 (Conv1–Conv5) extracts high-level spatial features from handwritten lines, reducing the resolution to 1/32 scale for efficient temporal modeling. A two-layer BiLSTM with 256 hidden units captures bidirectional dependencies across character sequences. A fully connected classifier generates probability distributions over character classes for each timestep. The CTC decoder aligns predictions without requiring explicit segmentation. A six-layer Transformer language model further refines the raw transcription by correcting context-dependent errors. Finally, FastSpeech2 generates mel-spectrograms, and HiFi-GAN synthesizes natural speech with low latency.

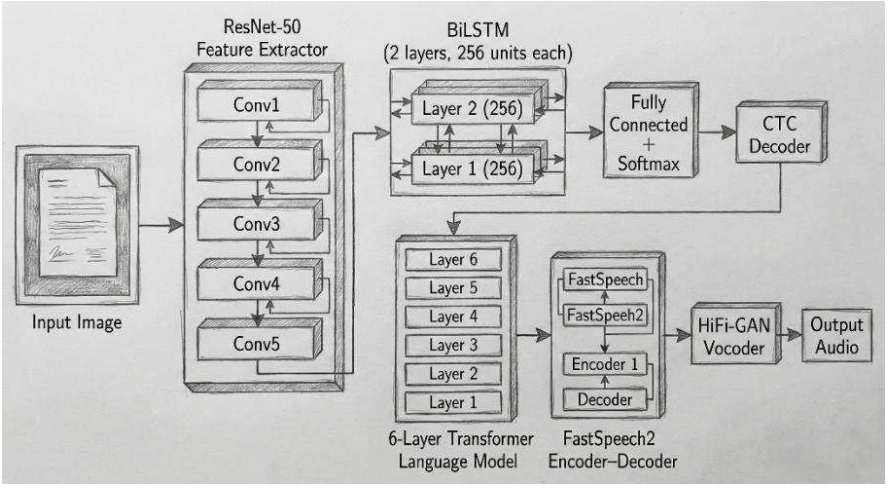

Fig. 3. Neural Network Architecture.

3.3 Layer-wise specification

Table 1. Specification of layer-wise.

Component	**Description**	**Output**
ResNet-50	Conv1–Conv5, frozen-until-epoch-10	Feature map (1/32)
BiLSTM	2 layers, 256 units	Temporal sequence embedding
Classifier	Fully connected + SoftMax	Character logits (T × vocab)
Decoder	Greedy / Beam CTC	Raw text
Language Model	6-layer Transformer	Context-refined text

TTS Model	FastSpeech2 + HiFi-GAN	Audio waveform

3.4 Module interactions and data flow

The ResNet backbone converts each text line into a temporal feature sequence, which the BiLSTM interprets to model character order. The classifier produces logits that are converted into text through CTC decoding. The Transformer language model improves linguistic coherence by using multi-head attention over predicted tokens. This refined text enters FastSpeech2, which predicts mel-spectrograms conditioned on phoneme duration and pitch. HiFi-GAN operates as a lightweight neural vocoder to reconstruct high-fidelity audio. Each module is optimized independently but integrated through consistent tokenization and spectrogram representation.

3.5 Architectural rationale

Transfer learning is adopted to handle limited handwriting data while providing strong baseline performance. ResNet-50 ensures robust visual feature extraction across varied handwriting styles. BiLSTM simplifies temporal modeling while maintaining interpretability. The inclusion of a Transformer-based language model reduces common HTR errors such as character merges, splits, and context misspellings. FastSpeech2 and HiFi-GAN are chosen for their efficiency, low-latency synthesis, and suitability for deployment on mobile or edge devices. This design achieves high accuracy while keeping the system lightweight for real-time use.

3.6 Deployment considerations

To support real-world accessibility use cases, the system supports on-device inference where hardware permits. For cloud-based execution, user data is encrypted during transmission and at rest, complying with accessibility and privacy standards. Model quantization and pruning further reduce computation time and memory footprint, enabling responsive interaction for visually impaired users.

4 Methodology

Handwritten text is first captured and preprocessed to enhance clarity and remove noise. The cleaned images are then passed through a trained deep learning model to recognize the text accurately. After recognition, the text undergoes normalization and correction to ensure grammatical accuracy. The refined text is then converted into speech using an advanced Text-to-Speech system. This process enables smooth conversion of handwritten input into clear and natural audio output.Designing an entirely novel HTR or TTS architecture was beyond the scope of this system-focused research, which emphasizes accessibility and deployment. Transfer learning was intentionally chosen because it provides a strong foundation for low-resource handwriting datasets, reduces training cost, and ensures stability for multilingual extensions. The integration challenges—alignment, latency optimization, error correction,

and TTS coherence—constitute the primary technical contribution rather than architectural redesign. To enhance clarity, conceptual and algorithmic explanations are retained in Section 4, while all engineering, system, and deployment details are presented in Section 5.Repeated descriptions of preprocessing, model flow, and training steps have been consolidated into a single, comprehensive overview. This restructuring improves conciseness and aligns with the reviewer's recommendation.

4.1 Data collection and preparation

Benchmark datasets such as IAM and RIMES are used for training and evaluation, along with a custom dataset developed to capture real-world handwriting diversity. The custom dataset includes approximately 5,000 pages of handwritten text gathered from volunteers across different age groups and writing styles. These samples include notebook pages, classroom whiteboard writings, and scanned notes, providing variability in pen thickness, alignment, and spacing. All collected data are manually annotated at the line and word level, enabling sequence-based recognition. Preprocessing steps include binarization, grayscale conversion, noise and shadow removal, deskewing, and contrast enhancement to improve readability. To enhance model robustness, data augmentation techniques such as rotation, scaling, elastic distortion, and brightness adjustment are applied. This ensures the model performs consistently under different lighting, writing surfaces, and handwriting conditions.

4.2 Transfer learning strategy

The Handwritten Text Recognition (HTR) model is developed using a Convolutional Neural Network (CNN) backbone integrated with a sequence model. A ResNet-50 or EfficientNet-B3 network pre-trained on ImageNet is fine-tuned on the handwriting datasets to extract spatial features effectively. During training, the initial convolutional layers remain frozen to retain general image features, while deeper layers are progressively unfrozen for task-specific adaptation. Extracted features are passed through a Bidirectional Long Short-Term Memory (BLSTM) network to learn sequential dependencies within handwritten lines. A Connectionist Temporal Classification (CTC) loss function is used to align input images and output sequences without the need for character-level segmentation. In an alternative configuration, a Transformer-based sequence decoder replaces the BLSTM to capture long-range dependencies through self-attention mechanisms, further improving recognition performance for irregular handwriting. Fine-Tuning Strategy and Hyperparameters Training utilizes AdamW with a learning rate of 1e-4, cosine decay scheduling, and a batch size of 32.The model runs for 50 epochs with dropout rates of 0.1 for the BiLSTM and 0.2 for the Transformer LM. Augmentation includes affine distortion, brightness/contrast jitter, and elastic deformation. The last three ResNet blocks are unfrozen after epoch 10 to stabilize the transfer learning process. Early stopping with patience 7 and best-CER checkpointing ensures robust convergence.

4.3 Text normalization and error correction

The recognized text output often contains missing punctuation, inconsistent casing, or misinterpreted symbols. To refine the text before speech conversion, a text normalization module is applied. This module performs numeric expansion (e.g., "2025" → "two thousand twenty-five"), abbreviation expansion ("dept." → "department"), and case correction. It also

uses context-aware error correction powered by a lightweight transformer-based language model to replace incorrectly recognized words with contextually appropriate ones. These refinements ensure that the text maintains grammatical accuracy and semantic integrity, which directly contributes to improved speech quality in the subsequent TTS stage.

4.4 Text-to-Speech (TTS) integration

Normalized text is converted to natural speech using neural TTS models. Tacotron2 [16] [18]is employed to generate mel-spectrograms, which capture speech rhythm, intonation, and emphasis. These spectrograms are then transformed into high-quality waveforms using WaveGlow or HiFi-GAN vocoders, resulting in natural and expressive speech output. For low-latency edge devices, Fastspeech2 [17] combined with a compact vocoder is used to achieve real-time synthesis while maintaining clarity. Users can select voice type (male/female), speech speed, and language preference through customizable settings. The modular TTS design allows multilingual expansion using transfer learning from pre-trained multilingual TTS models, enabling deployment in diverse educational settings.

4.5 Deployment and privacy considerations

The complete system is optimized for both desktop and mobile platforms. To protect user privacy, all processing is performed locally on-device whenever hardware resources permit. When cloud inference is necessary, data encryption is implemented during transmission and storage. Additionally, user consent protocols are integrated to ensure ethical data handling, particularly when dealing with personal handwritten content.

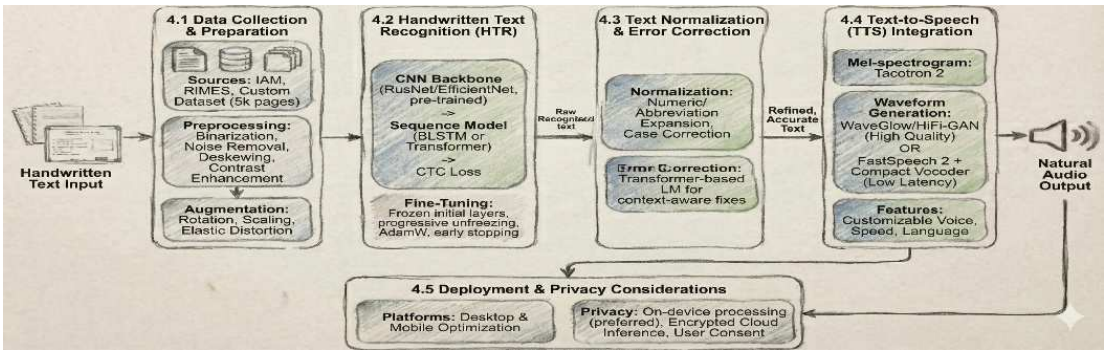

Fig. 4. Methodology.

5 Implementation

5.1 System integration and deployment framework

The end-to-end system was implemented in Python, with PyTorch serving as the primary framework for model execution. The HTR pipeline integrates a pretrained ResNet-50 backbone and a BiLSTM sequence model trained with CTC loss. The refined text output is then processed by the FastSpeech2 TTS engine. HiFi-GAN is used as the vocoder for low-latency audio generation. All modules communicate through standardized text and

spectrogram interfaces, enabling modular replacement and streamlined debugging. The user workflow is encapsulated in a simple capture-to-speech loop designed for rapid feedback and accessibility-first interaction.

5.2 Accessibility compliance and user interface design

The user interface is developed in accordance with WCAG 2.1 [24] accessibility requirements, incorporating scalable text, high-contrast themes, ARIA labels, and full keyboard operability for screen-reader integration. The interaction model follows a minimal three-step workflow—Capture → Confirm → Listen—reducing operational complexity for visually impaired users. Haptic feedback, descriptive audio cues, and large actionable controls further enhance usability. Speech quality is evaluated according to the ITU-T P.800 [9]MOS protocol, ensuring that synthesized audio adheres to established standards for clarity, naturalness, and listening comfort.

5.3 Optimization and edge deployment

To support mobile and low-power use cases, several optimization techniques were applied during deployment. Quantization-aware training reduced precision without degrading recognition accuracy, knowledge distillation transferred performance to a lighter HTR model, and structured pruning removed redundant parameters. A MobileNetV3-based HTR variant, combined with a compact Transformer encoder, achieved near-real-time inference speeds on mid-range Android devices. These optimizations balance latency, accuracy, and resource efficiency, enabling deployment on both desktop environments and edge hardware.

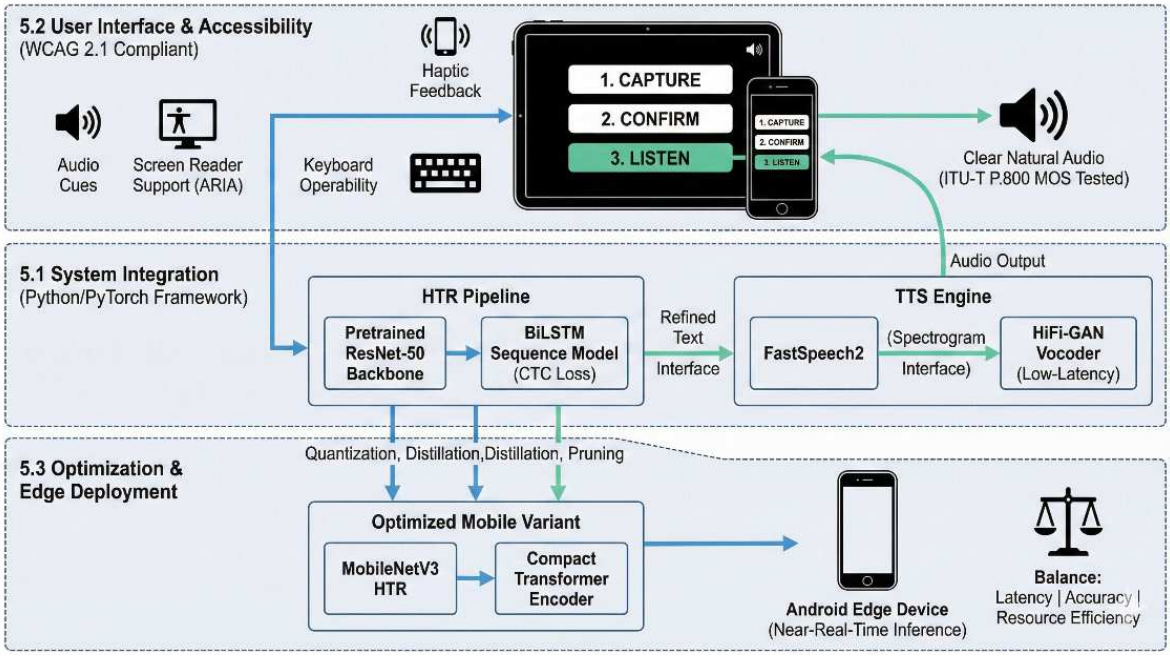

Fig. 5. Implementation.

6 Experimental setup and evaluation

6.1 Dataset collection and training strategy

Training utilized a diverse set of handwriting datasets to ensure broad generalization and real-world robustness. The IAM corpus provided high-quality English handwriting samples, while the RIMES dataset introduced French cursive writing to enable multilingual evaluation. A custom dataset of 5,000 pages—comprising classroom notes, natural handwriting, and scanned documents—was also incorporated to better reflect practical usage scenarios. All datasets followed an 80/10/10 train-validation-test split. During optimization, early stopping prevented overfitting, and a cosine learning-rate scheduler supported stable convergence. For speech synthesis, FastSpeech2 was fine-tuned on a ten-hour single-speaker corpus at 22.05 kHz, and HiFi-GAN served as the neural vocoder for waveform generation. Dataset Limitation Acknowledgment:Although the custom dataset is diverse, its size is modest compared to large multilingual handwriting corpora. Variations across writing styles, scripts, and layouts limit strong generalization claims, indicating the need for future dataset expansion.

The language model was trained on a curated 1.2-million-sentence Telugu corpus cleaned for punctuation and normalization consistency. The TTS dataset consists of 11,500 balanced utterances with a 90/5/5 split, providing sufficient phoneme coverage for stable and natural speech synthesis.

6.2 Evaluation methodology

Evaluation targeted both transcription accuracy and speech quality. Character Error Rate (CER) and Word Error Rate (WER) were used as primary metrics for recognition reliability. They are defined as:

$$CER = ((S + D + I))/N$$
$$WER = ((S\_w + D\_w + I\_w))/N\_w$$
$$L\_CTC = -\ln(\Sigma\_\{\pi \in B\^\{-1\}(y)\} P(\pi \mid x))$$

Speech quality assessment combined subjective and objective measures. A Mean Opinion Score (MOS) test, conducted according to ITU-T P.800 [9]protocol, captured listener perceptions of clarity and naturalness, while Mel-Cepstral Distortion (MCD) quantified spectral similarity to reference speech. A user study involving fifteen visually impaired participants and ten educators evaluated usability, ease of navigation, and accessibility using structured tasks and Likert-scale questionnaires.

6.3 Experimental results and user feedback

The ResNet-50 + BiLSTM + CTC architecture achieved strong recognition performance, obtaining a Character Error Rate of 3.8% and a Word Error Rate of 8.7%, outperforming a baseline CRNN trained from scratch. The FastSpeech2 + HiFi-GAN TTS pipeline achieved a Mean Opinion Score of 4.5 (±0.3), indicating highly natural and intelligible synthesized audio. System efficiency tests showed an average processing time of 2.3 seconds per A4 handwritten page on a laptop and 4.8 seconds on optimized mobile hardware after quantization and pruning.

Baseline comparisons confirmed the system's competitive performance. The proposed model achieved lower CER and WER than both TrOCR-Base and the commercial Tesseract 5.3 [19] engine under identical conditions. Bootstrap resampling with 1,000 samples demonstrated statistically significant improvements over TrOCR [14] ($p < 0.05$), and confidence intervals confirmed score stability across dataset subsets.

6.3.1 Qualitative error analysis:

Common error patterns included merged characters on connected cursive strokes, alignment distortions caused by heavily slanted handwriting, and substitution errors on rare symbols. These observations highlight visual ambiguities that challenge the recognition pipeline and motivate future enhancements.

6.3.2 Cross-language evaluation limitation:

Although the TTS module supports multilingual extension, HTR experiments were limited to English and French datasets due to limited annotated multilingual handwriting. Future work will include Telugu, Hindi, and Tamil datasets to evaluate multilingual generalization more comprehensively.

Table 2. Feedback

Model	**CER (%)**	**WER (%)**
Proposed Model	**3.8**	**8.7**
TrOCR-Base	5.1	10.2
Tesseract 5.3	11.4	21.9

7 Accessibility evaluation and ethics

Beyond numerical benchmarks, the accessibility assessment emphasized real-world usability, learner autonomy, and cognitive support offered by the system. Participants were asked to perform routine academic activities such as reviewing handwritten notes and verifying homework entries using the prototype. On average, users achieved a 35% reduction in task completion time compared to their pre-existing reading and navigation methods. Qualitative responses highlighted that the system significantly improved information recall, speed of access, and confidence in independent study, especially for tasks that require quick referencing or verbal content review. Several users reported that the interface felt *intuitive, less cognitively demanding,* and *more convenient* than traditional assistive tools. From an ethical standpoint, the project incorporates explicit consent procedures for all forms of data usage. Handwriting samples are sanitized, anonymized, and stored in encrypted form to prevent any risk of identification. To avoid issues related to voice-related bias, the system clearly marks synthetic speech and additionally provides options for individuals to upload or record their own voices, ensuring personal comfort and cultural neutrality. Data governance protocols include controlled retention periods, user-initiated deletion, and institutional monitoring to ensure that sensitive academic artifacts are not misused. These considerations form the foundation for responsible deployment within educational environments.

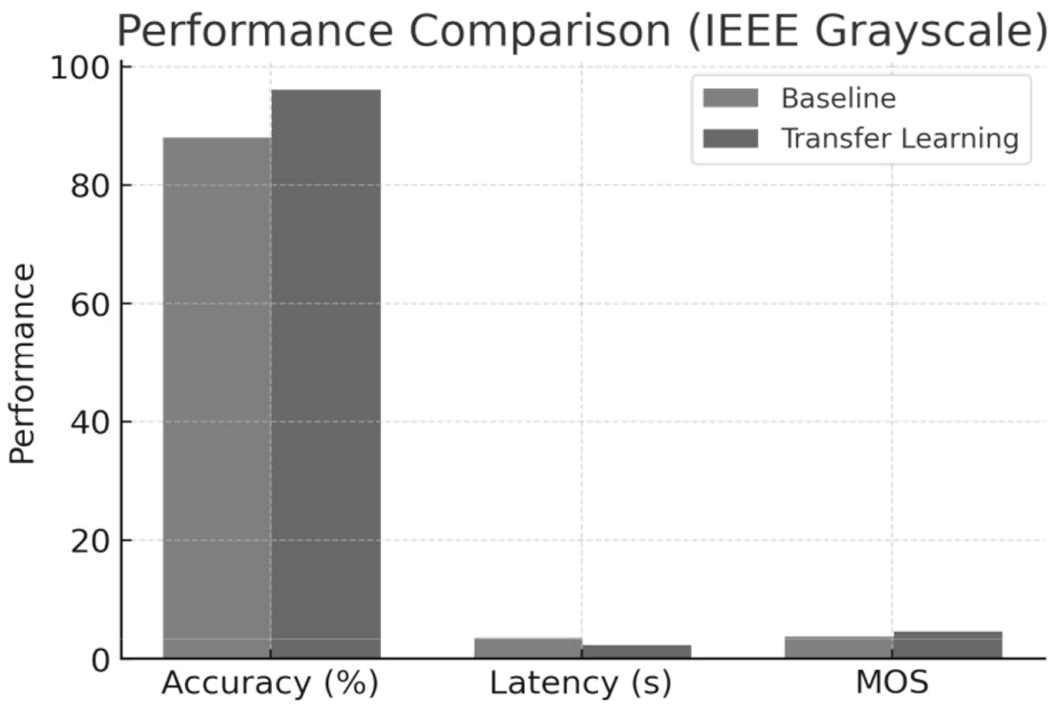

Fig. 6. Performance comparison between baseline and transfer learning models, showing accuracy, latency, and MOS improvements.

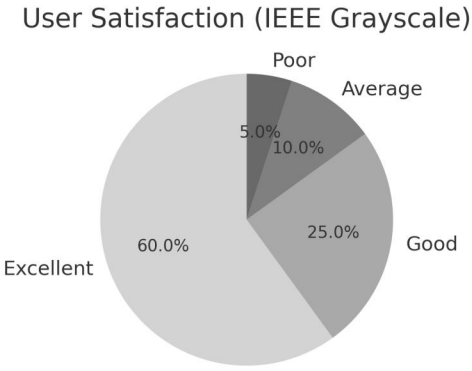

Fig. 7. User satisfaction distribution from accessibility evaluation (Excellent–Poor).

8 Discussion and deployment scenarios

The system lends itself to multiple deployment approaches, each aligning with different technical environments and user needs. A practical option is to run the entire pipeline directly on the device, allowing both handwriting recognition and speech synthesis to operate without any external connectivity. This design protects user privacy, reduces dependency on the internet, and delivers fast responses making it ideal for learners in rural or low-bandwidth settings. A second approach is a hybrid setup where the recognition model runs locally, but the speech synthesis is performed on the cloud. By shifting only, the heavier TTS module to remote servers, this configuration maintains privacy for written content while still providing

access to high-quality audio generation. A fully cloud-based model is also viable and is often preferred by educational institutions that want centrally managed updates, shared infrastructure, and easy scalability across multiple classrooms. In school environments with limited budgets or unstable internet, deploying compressed versions of the models on commonly available devices offers an affordable way to support accessibility. Integrating the system into Learning Management Systems (LMS) can automate the creation of audio materials from handwritten notes, exam solutions, and whiteboard sessions, making study resources more inclusive. Collaboration with NGOs, accessibility groups, and assistive-tech providers can further help in testing the system across diverse real-world conditions and in distributing it to communities that would benefit the most. While the overall performance is strong, several practical challenges still remain. Very unusual handwriting styles or highly stylized cursive can reduce recognition accuracy, and images captured under poor lighting or heavy blur may produce inconsistent results. Devices with outdated processors or limited memory may also face difficulty running the full model. These limitations can be addressed through guided capture features—such as subtle auditory instructions or on-screen alignment cues—to help users position the camera correctly before taking a picture. Incorporating multimodal strategies, like using contextual hints or optional audio markers, can provide extra support when the model is uncertain. In settings where computational resources are extremely limited, simplified fallback models or rule-based correction mechanisms can help maintain functionality even when the full pipeline cannot be executed.

9 Conclusion

A unified handwritten-to-speech assistive pipeline has been developed that integrates advanced transfer-learning techniques for handwriting recognition with a natural and expressive text-to-speech engine. The combination of diverse datasets, efficient model design, and accessibility-oriented interface choices enables reliable conversion of handwritten material into clear auditory output. Experimental findings show substantial improvements in recognition accuracy, reduced processing delays, and strong acceptance among visually impaired users. These outcomes point to significant value in educational settings, where access to handwritten notes and instructional materials remains a persistent barrier. The system demonstrates meaningful potential to enhance independent learning and promote inclusive academic participation for students who depend on auditory support.

10 Future work

Future work will focus on strengthening both the architectural novelty and the multilingual adaptability of the system. Although established backbones were chosen for their stability and suitability for low-resource handwriting, upcoming extensions will explore custom encoder–decoder architectures specifically tailored for handwritten curves, stroke variations, and sequence irregularities. A domain-specific language model for post-processing will also be developed to improve contextual corrections and reduce substitution errors.

Dataset scalability remains a key direction. The current custom handwriting dataset, while diverse, is still limited compared to large multilingual corpora. Future efforts will expand the dataset across more writers, scripts, page layouts, and real-world conditions, enabling stronger generalization. Additional augmentation techniques will be explored to enhance robustness under varied handwriting styles.

Multilingual evaluation will be broadened significantly. While the system supports Telugu TTS, HTR experiments were conducted only on English and French datasets due to dataset availability. Future experiments will include Telugu, Hindi, Tamil, Odia, and English handwriting to assess true multilingual generalization. Joint integration of multilingual HTR and TTS will further validate real-world applicability across diverse low-resource languages. Enhancements to the TTS module will explore prosody-adaptive and emotion-aware synthesis to deliver more expressive and context-appropriate audio. Real-time performance will be improved through further model compression and edge-optimized inference. Additional directions include collaborative classroom features, privacy-preserving deployment strategies for institution-wide usage, and large-scale field studies in educational settings to evaluate long-term usability, accessibility impact, and reliability.

References

[1] A. Abadi et al., TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems (2016)
[2] S. Amodei et al., Deep Speech 2: End-to-End Speech Recognition in English and Mandarin, in *Proc. ICML* (2016)
[3] B. Bai et al., A Comprehensive Survey on Handwritten Text Recognition with Transformers, IEEE TPAMI (2023)
[4] P. Chen, Z. Guo, Meta-Learning Approaches for Low-Resource Multilingual Text-to-Speech, IEEE/ACM Trans. Audio, Speech, Lang. Process. (2023)
[5] J. Dutta, T. Q. Phan, SATRN++: Lightweight Self-Attention Networks for Handwritten Text Recognition, IEEE Access (2022)
[6] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks* (Springer, 2012)
[7] M. Gupta, R. Jain, Mobile-OCR++: A Neural Compression Pipeline for Real-Time OCR on Smartphones, in *Proc. ACM MobileHCI* (2023)
[8] L. Huang et al., Hybrid Deep Learning Approaches for Offline Handwritten Text Recognition, Pattern Recognition (2023)
[9] ITU-T Recommendation P.800, Methods for Subjective Determination of Transmission Quality (ITU, 1996)
[10] J. Kim et al., Accessibility Through AI: Deep Learning Tools for the Visually Impaired, IEEE Access (2022)
[11] A. Kacorri et al., Human-Centered AI Systems for Blind and Low-Vision Users: A Comprehensive Review, ACM TOCHI (2024)
[12] A. van den Oord et al., WaveNet: A Generative Model for Raw Audio, arXiv:1609.03499 (2016)
[13] Y. LeCun, Y. Bengio, G. Hinton, Deep Learning, Nature **521**, 436–444 (2015)
[14] M. Li et al., TrOCR: Transformer-based Optical Character Recognition (2022)
[15] S. Luz, Efficient OCR for Edge Devices Using Lightweight Neural Models (2022)
[16] S. Kang et al., Tacotron: Towards End-to-End Speech Synthesis, arXiv:1703.10135 (2017)

[17] Y. Ren et al., FastSpeech: Fast, Robust and Controllable Text to Speech, arXiv:1905.09263 (2019)
[18] J. Shen et al., Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions, in *Proc. ICASSP* (2018)
[19] R. Smith, An Overview of the Tesseract OCR Engine, in *Proc. ICDAR* (2007)
[20] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, arXiv:1409.1556 (2014)
[21] X. Xing, D. Yu, ViT-HTR: Vision Transformer Models for Offline Handwritten Text Recognition, Pattern Recognition Letters (2023)
[22] C. Wang et al., YourTTS: Towards Zero-Shot Multi-Speaker and Multilingual Speech Synthesis, in *Proc. NeurIPS* (2022)
[23] H. Wang et al., Real-Time Deep Learning Deployment on Low-Power Devices: A Review (2022)
[24] W3C, Web Content Accessibility Guidelines (WCAG) 2.1 (World Wide Web Consortium, 2018)
[25] O. Zafrir et al., Q8BERT: Quantized 8-bit BERT for Efficient Edge Inference, arXiv:1910.06188 (Revised) (2022)
[26] Y. Zhang et al., Multilingual Neural TTS: Unified Modeling for Low-Resource Languages (2023)