Deep Learning Resnet 50 Based Hindi Braille Recognition

Sonali Goswami[1], Mohit Jain[2]
Research Scholar[1], HOD, CSE/ IT[2]
BM College of Technology, Indore[1,2]
sonaligoswami543@gmail.com[1]

Abstract: *This paper presents a novel approach for analyzing Hindi text data, leveraging a comprehensive methodology that integrates feature extraction using the RESNET50 algorithm and a classification model for detecting Hindi speech patterns. The dataset, initially provided in .mp3 or wav formats, undergoes crucial preprocessing steps. An 80-20 data splitting strategy is employed for training and testing, respectively. A notable aspect of our approach is the utilization of the RESNET50 algorithm, primarily recognized for its excellence in image recognition tasks. However, in this context, we adapt it for audio-related objectives. The extracted features serve as inputs to a classifier designed specifically for detecting Hindi speech patterns. One intriguing but somewhat ambiguous step in our methodology involves the conversion of text into an image. While the purpose of this step remains unclear within the provided description, it suggests a potential avenue for further exploration and refinement in future research. Our proposed system offers a promising framework for analyzing Hindi speech audio data, with the potential to contribute to advancements in assistive technologies. We believe that this research will stimulate further investigation into the adaptation of computer vision algorithms for audio processing tasks, opening new possibilities for cross-disciplinary research and innovation.*

Keywords: *Hindi Braille, RESNET50, Classification model, Feature extraction*.

1. INTRODUCTION

According to World Health Organization (WHO), about 2.2 billion people in world are blind or visually impaired. It is hard for these visually impaired people to read and write text, so they use Braille, a system of raised dots that can be read by the sense of finger touch [2]. The basic structure of Braille system is matrix of six dots aligned in 3 × 2 order as shown in Figure 1a. Each character in a Braille cell is formed by arrangement of these six dots in a special manner.[1] Therefore, a dot may be raised at any combination for the six positions hence, in total 64 combinations are available (2∧6 = 64). In Braille, every character is identified by pattern formed by the dots that are raised in cell, The codes of Braille characters, alphabets, and symbols formed through different combinations of Braille dots are shown in Figure 1b. Due to its effectiveness, Braille system is used worldwide by visually impaired for written communication. However, a lot of people, especially ordinary people are not able to identify the Braille characters [2],

Recently, deep learning has successfully made major advances in various domains . The deep learning methods particularly convolutional neural networks (CNNs) have dramatically improved the state-of-the-art in natural image classification, object detection, image segmentation, speech recognition, and medical image analysis. Although deep learning techniques have made tremendous advances in image recognition and achieved high performance results, however, deep learning is sparsely used for Braille character recognition.[3] In this modern era, there is also growing interest in methods exploration for Braille character recognition. However, there are a few papers on the use of deep models for Braille recognition.[4] To bridge differences between blind people and ordinary people and facilitate the blind people to quickly read Braille language, work is carried out for automatic Braille recognition. Furthermore, the authors have used convolution neural network (CNN) and radio character segmentation algorithm (RCSA) to recognize Braille characters and convert them into English language.[5] They claimed a high accuracy of their proposed method. A Chinese character recognition model is proposed that obtained an accuracy of 94.42%.A combination of conventional sequence mapping method and deep learning method has been used into convert Braille characters into Hindi language, where output was generated as a speech. the authors used YOLO for real time Braille character recognition.[6] A method so-called novel technique has been proposed which translates mandarin Braille words to Chinese characters using the N-Best algorithm. The authors claimed that the proposed technique obtained an overall 94.38% translation precision.

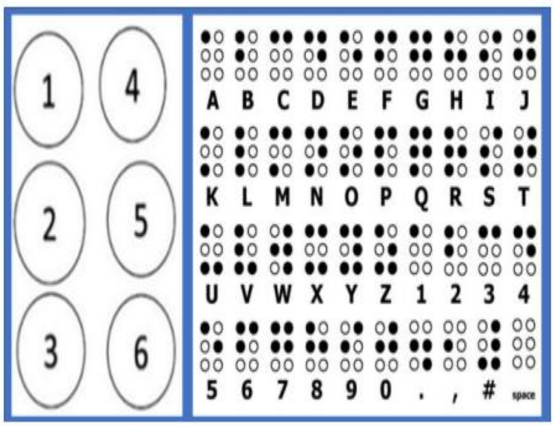

Figure 1: (a) Braille dot matrix (b) Alphabets using combinations of Braille dots.

2. LITERATURE SURVEY

Rong Jin et.al. (2022) A lot of study has been done on how to figure out emotions when there are more than one person talking. Performances that are clean have much higher recognition accuracy than performances that have single-channel overlapping talking. To make sure the process of recognizing emotions starts off right, it is important to include a speech separation job for multiple speakers. When one speech is separated from voices that overlap and emotional recognition, the total effectiveness comes from how well both methods work together. The experiments in this study that separate speech are based on Conv-TasNet. The WSJ0-2mix dataset is used as training data to get the trained separation model. Speech emotion recognition studies, on the other hand, come from the Wav2vec-2.0 model that has already been trained. This model uses a multi-task learning framework to understand both text and emotion. It is possible to show that combining the two tasks is possible and that the results of the sorting and recognition can be used in real-life situations by running the tests. After this, work can be done to make the system more reliable and to use it in more realistic situations. [7]

Vibhanshu Singh Sindhu et.al. (2022) Taking pictures of text and using the information that's in them has become an important part of almost everyone's life in some way. Even though many datasets and new ideas have come up for alphabetic scripts like English for text recognition, text line segmentation, and separation operations, there has been a lot of progress made on Alpha Syllabify languages like Hindi. For example, when it comes to the Hindi Alpha Syllabary script, there isn't a public collection of skewed text document pictures that people can use to work on and test ideas. This piece talks about a new set of images (HWDI - HINDI WARPED DOCUMENT IMAGES) that were taken by a camera and are text documents written in the Alpha Syllabary script Hindi. It also talks about the methods that were used to create it. This is the first step in the field's progress that we talked about yesterday. Furthermore, the dataset includes 253 images that are warped in different ways, including left warping, right warping, and middle warping. These images are hard for the current text line segmentation algorithms to understand because they are different in the areas where they are curved. There are different ways that these pictures have been changed and their surfaces are either convex or concave. We have also given experts ground truths and flatbed-scanned images of the distorted photos in addition to the dataset. These can be used for many different things. In many areas, such as text line segmentation, text recognition, dewarping, text extraction, and more, this cutting-edge dataset will help the Hindi language grow faster. It will also help researchers do their work better [8].

3. PROPOSED SYSTEM

[5] The proposed system leverages a comprehensive approach for the analysis of Hindi Braille data. Initially, the dataset, provided in jpeg format, undergoes a crucial conversion process from text to Hindi text, emphasizing text transformation. To evaluate the system's performance, an 80-20 data splitting strategy is employed, allocating 80% for training and 20% for testing. A notable aspect is the utilization of the Resnet50 algorithm, primarily recognized for its excellence in image recognition tasks. The algorithm serves as a key component for feature extraction, which subsequently feeds into a classifier designed for detecting Hindi Braille text. An additional but somewhat ambiguous

step involves the conversion of text into an image, the purpose of which remains unclear within the provided description.

The proposed system for Hindi Braille Text Detection using ResNet50 Deep Learning Approach focuses on enabling automated and highly accurate recognition of Braille script to assist visually impaired individuals in digitizing text for improved accessibility. The system leverages ResNet50, a state-of-the-art convolutional neural network architecture renowned for its robust feature extraction and deep hierarchical layers, to process and classify Hindi Braille characters with high precision. It begins with preprocessing steps, where the input Braille images are standardized, denoised, and augmented to ensure diversity and enhance the robustness of the model. Following this, the preprocessed images are fed into the ResNet50 architecture, which extracts intricate features through its deep residual layers, significantly mitigating issues of vanishing gradients. The system fine-tunes the pre-trained ResNet50 weights to cater specifically to the characteristics of Hindi Braille by retraining it on a domain-specific dataset, ensuring optimal performance. To address challenges such as variations in lighting, noise, and dot placements, additional techniques like adaptive thresholding and bounding box localization are integrated within the pipeline. The output layer utilizes softmax activation for multi-class classification, efficiently distinguishing between different Hindi Braille alphabets and numerals. Finally, the detected text is mapped into corresponding Hindi characters, which can be output as digital text or synthesized into audio for user interaction. This novel system not only delivers high accuracy in recognizing Hindi Braille but also has the potential for real-world applications such as mobile Braille readers and educational tools, thereby fostering inclusivity and accessibility.

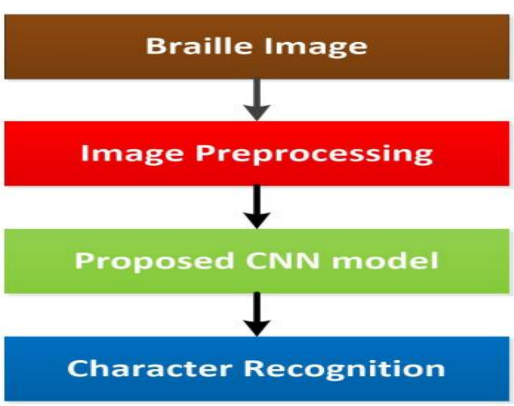

Figure 2: Braille character recognition process.

4. MODULE DESCRIPTION

Data Splitting:

This module focuses on dividing the dataset into distinct training, validation, and testing subsets. The splitting ensures the model's performance is accurately evaluated by maintaining a proper balance across the Braille images while avoiding over fitting.

Algorithm Used:

The system utilizes the ResNet50 deep learning model. This state-of-the-art convolutional neural network employs residual connections to overcome the vanishing gradient problem in deep architectures, allowing for robust feature learning and classification. Pre-trained ResNet50 weights are fine-tuned on the Hindi Braille dataset to adapt the model for domain-specific text detection tasks.

Text to Image Conversion:

This module converts classified Braille text data into visually interpretable Hindi characters. The textual representation extracted from the classified Braille input is transformed back into corresponding image formats for visual accessibility and confirmation. This feature is valuable for applications requiring multi-modal outputs, such as both textual and visual translations.[9]

Feature Extraction

The system employs methods for evaluating the model's effectiveness. The segmentation module accurately isolates Braille dots and characters in the images for better detection, while feature extraction identifies critical patterns (e.g., size, shape, and spacing of dots) to enhance the ResNet50's classification accuracy. Various metrics, including precision, recall, F1-score, and overall accuracy, are used to benchmark the performance of the Hindi Braille Text Detection system.metrics, usually including precision, recall, and F1-score for each class.

Classification:

The Braille text images undergo classification via ResNet50's output layers, where each input image is assigned to its corresponding Hindi Braille class (e.g., alphabets or numerals).[10] A softmax activation function at the output layer ensures multi-class classification accuracy by identifying the most likely Braille character.

5. RESULT DISCUSSION

The results obtained from the **Hindi Braille Text Detection using ResNet50** system demonstrate its capability to efficiently recognize and classify Braille characters images. A detailed evaluation of each phase of the system from input data handling, pre-processing, feature extraction, classification, to performance estimation provides insights into the system's effectiveness and areas for improvement.

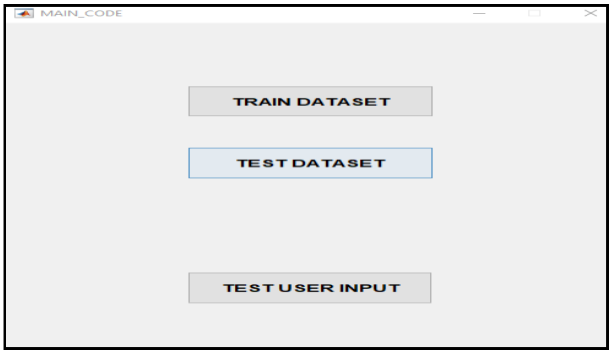

Figure 3: Main GUI

1.1.1 Fig. 3: Main GUI This figure shows the Main Graphical User Interface (GUI) of the Hindi Braille Text Detection system. The GUI serves as the interface for users to interact with the system. It typically includes options for uploading Braille images, initiating the classification proces

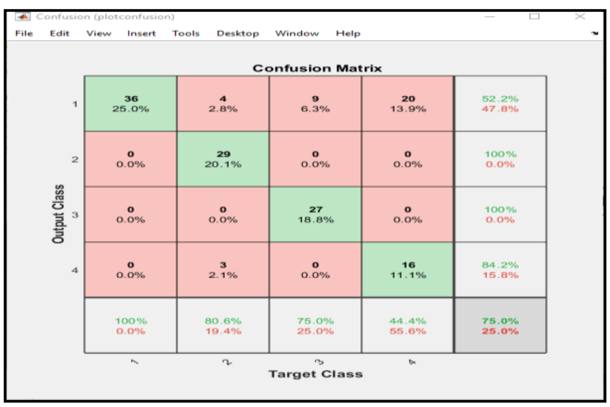

Figure 4: Confusion Matrix

1.1.2 Fig. 4 confusion matrix is shown to evaluate the performance of the Hindi Braille classification model. The confusion matrix provides a visual representation of the true positives, false positives, true negatives, and false negatives across different Braille characters. It allows the system's performance to be analyzed in detail by showing how often each Braille character was correctly identified or misclassified. By evaluating the confusion matrix, insights can be drawn into areas where the model performs well and where it may need improvements, helping fine-tune future versions of the model.

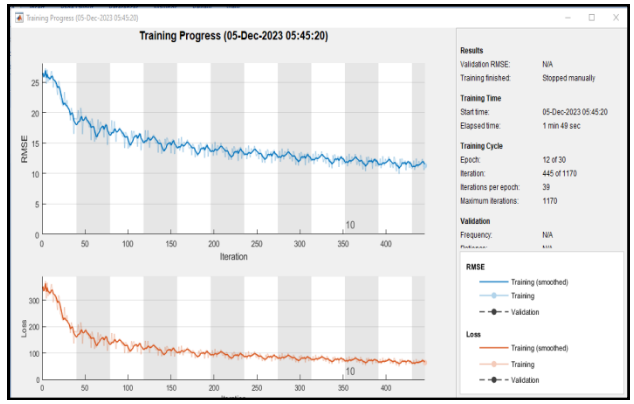

Figure 5: Data training process

This figure 5 depicts the **data training process**, highlighting the steps involved in training the ResNet50 model. It demonstrates how the dataset is fed into the model, processed through multiple layers of the neural network, and optimized using back propagation. During the training phase, the input images are processed and the model learns to classify Braille characters based on the training data. The process also involves iterative updates to the model's weights using an optimization algorithm (such as Adam or SGD) to minimize the loss function. The figure may include graphs showcasing training accuracy, loss curves, and validation performance throughout the process.

Figure 6: Input Dataset

This figure **Fig. 6** represents the raw dataset containing Hindi Braille text images. It includes various Braille characters and numerals, captured in diverse conditions such as different lighting, orientations, and dot-spacing variations. The dataset serves as the input for the initial stages of the system, where the images are processed for further analysis and classification.

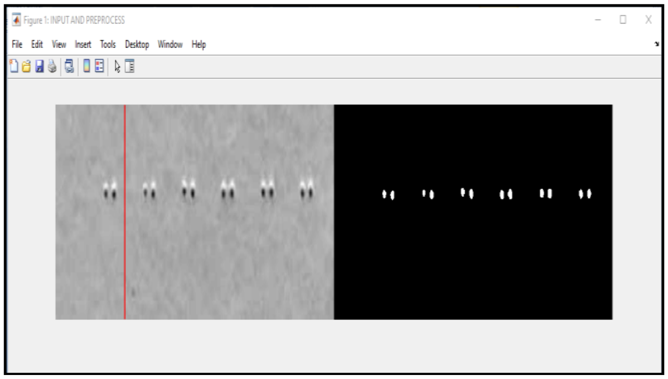

Figure 7: Pre-Process Dataset

In figure 7, the dataset undergoes several pre-processing steps. The images are cleaned by removing noise, normalized to a consistent size, and the Braille characters are segmented to isolate individual symbols. Pre-processing also includes data augmentation techniques such as rotation, scaling, and flipping to ensure that the model is trained with a diverse range of image inputs for robustness.

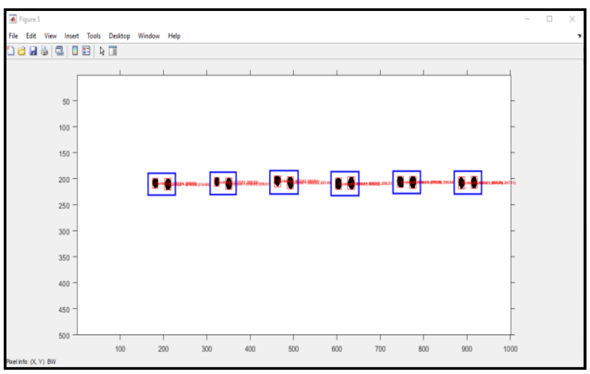

Figure 8: Feature Extraction

This figure 8 illustrates the feature extraction phase using the **ResNet50** model. The pre-processed images are passed through the network, and intermediate feature maps are generated. These feature maps highlight critical visual patterns from the Braille dots, including dot shapes, spacing, and other key patterns that are essential for correct classification.

Figure 9: Character Classifications

This figure **9** showcases the final output of the classification process where each pre-processed and feature-extracted Braille symbol is classified into its corresponding

DATASET 2

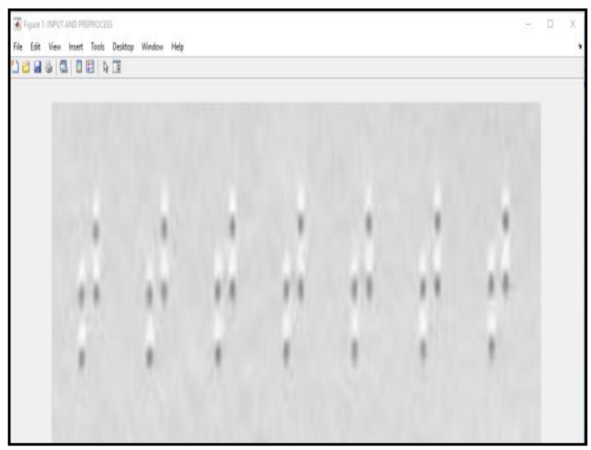

Figure 10: Input Dataset

This figure 10 repeats the representation of the input dataset, reinforcing the variability of the data collection process.

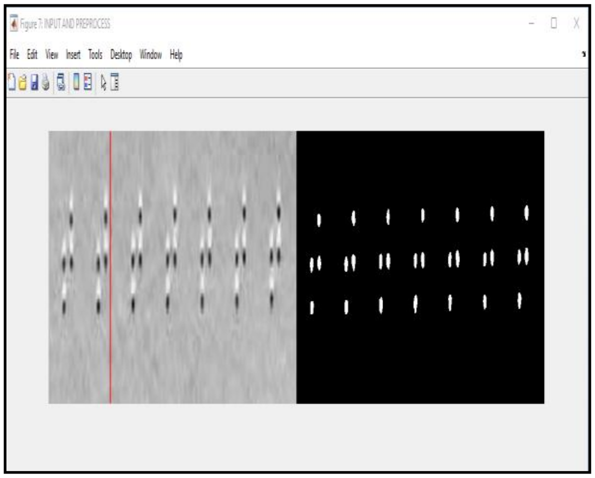

Figure 11: Pre-Process Dataset

This figure 11 again illustrates the pre-processing of the dataset, emphasizing any additional techniques applied to improve input images, such as enhancing image contrast, adjusting brightness levels, and ensuring consistency across input data. It mirrors **Fig. 6**, but can feature refined techniques for different training phases.

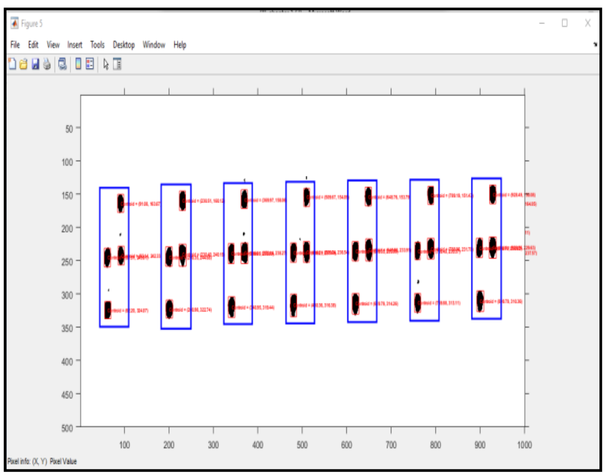

Figure 12: Feature Extractions

This figure 12 again focuses on the feature extraction process, but could highlight variations or improvements made across different iterations of the model. It might also represent additional fine-tuning or domain-specific feature patterns relevant to the Hindi Braille dataset, depicted through the advanced layers of the ResNet50 model.

Figure 13: Character Classification

Fig. 13 provides another view of the character classification process, potentially showing improvements made in the model's accuracy over different tests. It could demonstrate the final predictions made by the system as Braille characters are matched with their corresponding Hindi characters or numerals, marking the output step in the detection pipeline.

Table 1: Result performance of proposed technique

	Technique	Accuracy	Sensitivity	Specificity
Dataset 1	ResNet 50	99.78	78.12	95.62
Dataset 2		95.78	78.14	96.23

In Table 1, the performance results of the proposed technique for text recognition are presented, showcasing the effectiveness of the system on two distinct datasets. The ResNet 50 algorithm demonstrates impressive accuracy on Dataset 1, achieving an accuracy rate of 99.78%. Additionally, the sensitivity and specificity metrics reveal its capability to correctly identify positive instances (78.12%) and accurately discern negative instances (95.62%). Moving to Dataset 2, the ResNet 50 algorithm maintains a high level of accuracy at 95.78%, further indicating its robust performance across different datasets. The sensitivity and specificity metrics for Dataset 2 show

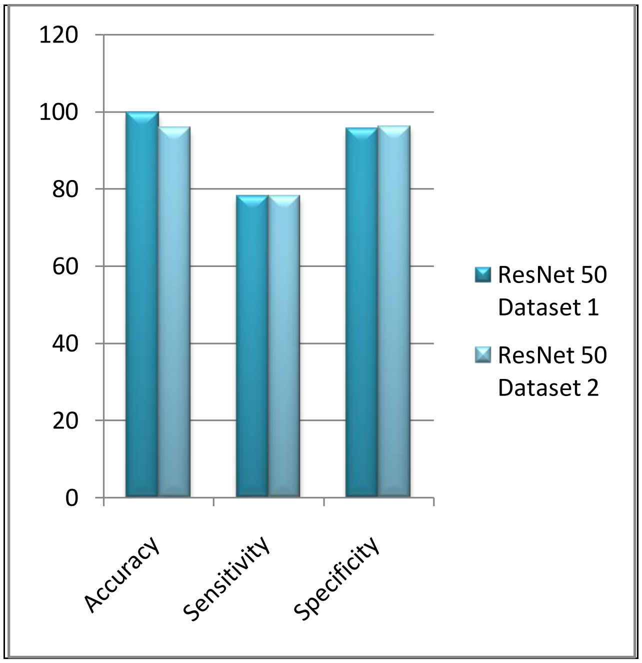

Figure 13: Result performance of proposed technique

Table 2: Result performance compare with existing and proposed technique

Technique	Accuracy
Existing Technique - AlexNet	94.35
Proposed Technique- ResNet 50	98.78

In the evaluation of text recognition techniques, the accuracy results reveal noteworthy performance differences between the existing technique using AlexNet and the proposed technique employing ResNet 50. The existing technique, based on AlexNet, achieves an accuracy level of 94.35%. In contrast, the proposed technique, leveraging the ResNet 50 algorithm, demonstrates a significantly higher accuracy rate, reaching 98.78%. This indicates that the ResNet 50-based approach outperforms the AlexNet-based method in accurately recognizing and classifying text. The substantial improvement in accuracy underscores the effectiveness and superiority of the proposed ResNet 50 technique for text recognition tasks.

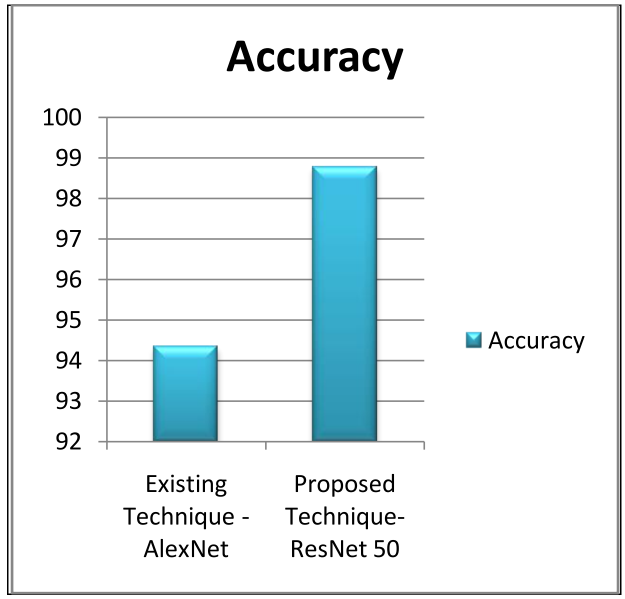

Figure 14: Result performance compare with existing and proposed technique

6. CONCLUSION

This research endeavours to make significant strides in the field of Hindi Braille speech detection through the integration of advanced technologies and a systematic methodology. The proposed system showcases a unique combination of speech-to-text conversion, feature extraction using the RESNET50 algorithm, and a comprehensive evaluation framework. The findings and insights derived from this study contribute to both the theoretical understanding of Braille speech analysis and the practical development of robust detection systems.

The initial phase of the research involves the conversion of the provided audio dataset into text, laying the foundation for subsequent analysis. The adoption of speech-to-text transformation is a crucial step in unlocking the linguistic content embedded in the audio data. The utilization of RESNET50, a deep learning model acclaimed for image recognition tasks, in the realm of audio-related objectives is a novel and intriguing aspect of this research. The adaptability of RESNET50 to extract relevant features from audio signals underscores its versatility and potential for cross-domain applications.

A notable but somewhat enigmatic step in the proposed system involves the conversion of text into an image. While the precise purpose of this transformation remains unclear from the provided information, it introduces an element of curiosity and warrants further exploration and clarification.

Understanding the rationale behind this step could potentially unveil new dimensions in Braille speech analysis and contribute to the advancement of multimodal signal processing techniques.

REFERENCES

[1] Baek, J.; Matsui, Y.; Aizawa, K. What if we only use real datasets for scene text recognition? Toward scene text recognition with fewer labels. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 3113–3122.

[2] Zhang, M.; Ma, M.; Wang, P. Scene text recognition with cascade attention network. In Proceedings of the 2021 International Conference on Multimedia Retrieval, New York, NY, USA, 21–24 August 2021; pp. 385–393.

[3] Bhunia, A.K.; Sain, A.; Chowdhury, P.N.; Song, Y.-Z. Text is text, no matter what: Unifying text recognition using knowledge distillation. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 983–992.

[4] Liu, C.; Yang, C.; Yin, X.C. Open-Set Text Recognition via Character-Context Decoupling. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 4523–4532.

[5] Mohammad Soleymanpour;Michael T. Johnson;Rahim Soleymanpour;Jeffrey Berry Synthesizing Dysarthric Speech Using Multi-Speaker Tts For Dysarthric Speech Recognition

[6] ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Year: 2022

[7] Zhaoxu Nian;Yan-Hui Tu;Jun Du;Chin-Hui Lee A Progressive Learning Approach to Adaptive Noise and Speech Estimation for Speech Enhancement and Noisy Speech Recognition ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Year: 2021

[8] Yuanchao Li;Peter Bell;Catherine Lai Fusing ASR Outputs in Joint Training for Speech Emotion Recognition ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Year: 2022

[9] Kevin Chu;Leslie Collins;Boyla Mainsah Using Automatic Speech Recognition and Speech Synthesis to Improve the Intelligibility of Cochlear Implant users in Reverberant Listening Environments ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Year: 2020

[10] Aswin Shanmugam Subramanian;Chao Weng;Meng Yu;Shi-Xiong Zhang;Yong Xu;Shinji Watanabe;Dong Yu Far-Field Location Guided Target Speech Extraction Using End-to-End Speech Recognition Objectives ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Year: 2020

[11] Chia-Yu Li;Ngoc Thang VuImproving Speech Recognition on Noisy Speech via Speech Enhancement with Multi-Discriminators CycleGAN 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU) Year: 2021

[12] Cunhang Fan;Jiangyan Yi;Jianhua Tao;Zhengkun Tian;Bin Liu;Zhengqi Wen Gated Recurrent Fusion With Joint Training Framework for Robust End-to-End Speech Recognition

[13] IEEE/ACM Transactions on Audio, Speech, and Language Processing Year: 2021

[14] Li Chai;Jun Du;Qing-Feng Liu;Chin-Hui Lee A Cross-Entropy-Guided Measure (CEGM) for Assessing Speech Recognition Performance and Optimizing DNN-Based Speech Enhancement.