# Navigating Eye to Blind People using Machine Learning

**Tejaswini B[1], Aishwarya S[2], Anushree R[3], Harshitha R[4], Inchara B R[5]**

[1]Assisstant Professor, Department of Information Science and Engineering

[2,3,4,5]Student, Department of Information Science and Engineering

East Point College of Engineering and Technology, Bengaluru, India[1,2,3,4,5,6]

Affiliated to Visvesvaraya Technological University, Belagavi,Karnataka, India

**Abstract***: In a world that heavily relies on visual information for navigation, visually impaired individuals face significant challenges in accessing and interacting with their environment. Everyday tasks, such as navigating unfamiliar places or recognizing objects, can become daunting and frustrating. However, advancements in artificial intelligence (AI) and computer vision have opened up new possibilities for improving the lives of visually impaired individuals. Navigating the eye to blind people using a machine learning system aims to bridge the gap between the visual world and the visually impaired people.*

**Keywords:** CNN, KNN, Machine learning, Disease Prediction

## I. INTRODUCTION

Visually impaired individuals encounter numerous challenges in navigating and interacting with the world around them due to their limited or non-existent visual capabilities. Everyday tasks that most people take for granted, such as independently moving through unfamiliar environments or identifying objects, become formidable obstacles. However, recent advancements in artificial intelligence (AI) and computer vision have paved the way for innovative solutions that can significantly improve the lives of visually impaired individuals. One such breakthrough is the development of a navigating the eye for blind people using a machine learning system tailored specifically to their needs.

## II. LITERATURE SURVEY

**[1].** Wearable Navigation Device for Virtual Blind Guidance.

Author: K. Dhivya, G. Premalatha, and S. Monica Year: 2019

Abstract: This device enables the blind to move from one place to another by wearing the navigation device. This system is based on the microcontroller with speech output. This may also give information about the urban walking routes. Blind could avoid the obstacle by using the ultrasonic sensor. They can easily detect the object in front, back, left, or right side of the blind. The Camera captures the image of the object and that will send the image to the raspberry pi board. The raspberry pi board is coded with the Python programming language. Then through the principle of optical character recognition, they will convert the image to speech or voice signal. Thus the blind can hear the object's name, the sign of direction symbol, etc. It makes the blind go to the outdoor environment and it makes them buy the product and see its quality through the camera and guide them with headphones. This can help the blind to move to various places of their own without depending on others. It may also be used for indoor navigation. The power supply is provided through the battery so it can be taken to anywhere.

This uses the TTS software because some adults and children face difficulties in reading the text. But it easily covers documents, e-pages, and web news in the speech signal. It makes the self-navigation. The proposed system has more features than other similar systems.

**[2].** A Recurrent Neural Network Approach to Image Captioning in Braille for Blind-Deaf People.

Author: S. Zaman, M. A. Abrar, M. M. Hassan, and A. N.

M. N. Islam Year: 2019

Abstract The Limitation of resources in braille is one of the biggest obstacles faced by the blind community trying to learn and integrate themselves better into our societies. The fact that not only text but also images are used for

DOI: 10.48175/IJARSCT-10645

communicating information does not help their cause. In this paper, we intend to help not only the visually impaired but also people with deaf-blindness, by converting images into captions which then can be read in braille. Being able to automatically translate the content of an image using properly formed English sentences is a challenging task in itself, but it could have a great impact by helping a blind person better understand his/her surroundings and even in mundane tasks like browsing the web for example. While text-to-speech methods can be then implemented using voice synthesizers for blind only people, the text needs to be converted into braille for people who have both hearing and sight loss. Our paper has been centered on the concept of transforming images into grade 1 braille. Here we present a deep recurrent neural network architecture that automatically generates brief descriptions of images directly in braille. Our model achieves a BLEU-4 score of 0.24 on the Flickr 8K Dataset, which is comparable to current text- based state-of-the-art deep learning models. Moreover, the generated captions are translated into haptic feedback readable for the blind by a microcontroller-based system on a 2 × 3 arrangement of push-pull solenoids.

**[3].** Automatic judgement of neural network-generated image captions.
Author: Biswas R, Mogadala A, Barz M, Sonntag D, Klakow D
Year: 2019
Abstract: Manual evaluation of individual results of natural language generation tasks is one of the bottlenecks. It is very time-consuming and expensive if it is, for example, crowdsourced. In this work, we address this problem for the specific task of automatic image captioning. We automatically generate human-like judgments on grammatical correctness, image relevance, and diversity of the captions obtained from a neural image caption generator. For this purpose, we use pool-based active learning with uncertainty sampling and represent the captions using fixed- size vectors from Google's Universal Sentence Encoder. In addition, we test common metrics, such as BLEU, ROUGE, METEOR, Levenshtein distance, and n-gram counts, and report the F1 score for the classifiers used under the active learning scheme for this task. To the best of our knowledge, our work is the first in this direction and promises to reduce time, cost, and human effort.

**[4].** How to get started with Google Text-to-Speech using Python
Author: Bharath K Year: 2019
Abstract: The Text-to-Speech API enables developers to generate human-like speech. The API converts text into audio formats such as WAV, MP3, or Ogg Opus. It also supports Speech Synthesis Markup Language (SSML) inputs to specify pauses, numbers, date and time formatting, and other pronunciation instructions.

**[5].** Towards Explanatory Interactive Image Captioning Using Top-Down and Bottom-Up Features, Beam Search and Re-ranking
Author:Biswas, R., Barz, M. Sonntag, D. Year: 2020
Abstract: Image captioning is a challenging multimodal task. Significant improvements could be obtained by deep learning. Yet, captions generated by humans are still considered better, which makes it an interesting application for interactive machine learning and explainable artificial intelligence methods. In this work, we aim at improving the performance and explain the ability of the state-of-the-art method Show, Attend, and Tell by augmenting their attention mechanism using additional bottom-up features. We compute visual attention on the joint embedding space formed by the union of high-level features and the low-level features obtained from the object-specific salient regions of the input image. We embed the content of bounding boxes from a pre-trained Mask R-CNN model. This delivers state- of-the-art performance, while it provides explanatory features. Further, we discuss how interactive model improvement can be realized through re-ranking caption candidates using beam search decoders and explanatory features. We show that interactive re-ranking of beam search candidates has the potential to outperform the state-of-the- art in image captioning.

DOI: 10.48175/IJARSCT-10645

## III. PROPOSED SYSTEM

This section presents a proposed system for visually impaired individuals that integrates image captioning technology with a smart navigation system. The system aims to provide real-time auditory descriptions of the surrounding environment to enhance accessibility and independence.

The proposed system follows the following workflow:

Image Acquisition: The camera captures real-time images of the environment.

Image Processing: Computer vision algorithms pre-process the images to enhance clarity and remove noise.

Image Captioning Model: Deep learning models analyze the processed images and generate descriptive captions.

Audio Interface: The generated captions are converted into auditory feedback using text-to-speech technology.

User Interaction: The user interacts with the system, providing input and receiving auditory descriptions of the environment.

```
                 +-----------------------------------+
              |        Image Acquisition        |
              |      +-------------------+    |
              |   Image Processing|    |
Camera <---->|            |                 |
            |       +---v----------------+    |
            |       |Image Captioning Model|  |
            |       +-------------------+    ||
            |       |   Audio Interface  |    |
            |       +-------------------+    |
            |       |  User Interaction  |    |
            |       +-------------------+    |
```

Fig.3.1 System Architecture

We are implementing this project using the following modules:

1. Data collection
2. Data Pre-processing and feature extraction
3. Building Training the Model
4. Caption Generation and voice output.

***Data collection:*** In this module, we are using images from the MS-COCO dataset. COCO is a large-scale object detection, segmentation, and captioning dataset.

COCO has several features:

- Object segmentation
- Recognition in context
- Superpixel stuff segmentation
- 330K images (>200K labeled)
- 1.5 million object instances
- 80 object categories
- 91 stuff categories
- 5 captions per image

***Data Pre-processing and feature extraction****: Image Pre-processing:*

- Resize and normalize images to a standard size to ensure consistency.
- Apply demonizing techniques to reduce noise and improve image quality.

- Adjust brightness, contrast, and color balance to enhance visual details.
- Convert images to grayscale if color information is not necessary for the captioning task.
- Remove irrelevant regions or backgrounds that may distract the captioning model.

**Object Detection and Recognition:**

- Utilize state-of-the-art computer vision algorithms (e.g., Faster R-CNN, YOLO, or SSD) to detect and localize objects within the pre-processed images.
- Extract bounding box coordinates for each detected object.
- Assign appropriate object labels to each bounding box (e.g., "person," "car," "chair," etc.).

**Feature Extraction:**

- Extract visual features: Utilize pre-trained CNN models, such as VGG16, ResNet, or Inception, to extract high-level visual features from the images.
- Global and local features: Consider both global features that capture the overall scene and local features that focus on specific objects or regions of interest.
- Dimensionality reduction: Apply techniques like Principal Component Analysis (PCA) or t-SNE to reduce the dimensionality of the extracted features while retaining their discriminative power.

**Caption Generation:**

- Tokenization and text pre-processing: Convert the captions or textual descriptions into a sequence of tokens or words for further processing.
- Vocabulary creation: Build a vocabulary or word dictionary that maps unique words in the captions to numerical indices for the captioning model to work with.
- Encode captions: Convert the tokenized captions into numerical representations, such as one-hot encoding or word embedding, to feed into the captioning model.

**Augmentation and Data Balancing:**

- Data augmentation: Apply techniques like rotation, scaling, or cropping to create additional variations of the image data, increasing the diversity of training samples.
- Address data imbalance: If there is a significant class imbalance in the object detection or captioning data, employ techniques like oversampling or class weighting to ensure a balanced representation.

These pre-processing and feature extraction steps help to extract relevant information from the image data and prepare it for subsequent training or inference in the Virtually Impaired Individuals with Navigating Eye to blind people using a machine learning system. The processed data can then be used to train and fine-tune deep learning models that enable accurate and descriptive caption generation for visually impaired individuals navigating their surroundings.

**Building Training the Model**

**Data Preparation:**

- Gather a well-annotated dataset that includes images, corresponding captions, and object labels if available.
- Split the dataset into training, validation, and testing sets. The typical split is around 70% for training, 15% for validation, and 15% for testing. Pre-process the images and captions as discussed earlier, including resizing images, tokenizing captions, and creating a vocabulary.

**Design the Model Architecture:**

- Choose a suitable architecture for image captioning, such as an encoder-decoder framework.
- The encoder processes the image and extracts visual features, while the decoder generates captions based on the encoded features.

DOI: 10.48175/IJARSCT-10645

- Popular architectures include CNNs for image feature extraction and recurrent neural networks (RNNs) or transformers for caption generation.
- Consider incorporating attention mechanisms to focus on relevant image regions while generating captions.

**Implement the Model:**

- Implement the chosen model architecture using a deep learning framework, such as TensorFlow or PyTorch.
- Set up the necessary layers, connections, and loss functions based on the model design.
- Define the training and evaluation procedures, including data loading, batch processing, and optimization.

**Training:**

- Initialize the model parameters with appropriate weights or pre-trained models if available.
- Train the model using the training dataset by feeding images and their corresponding captions.
- Optimize the model using backpropagation and gradient descent algorithms to minimize the defined loss function.
- Monitor the training process, track metrics such as loss and accuracy, and make adjustments if necessary.
- Experiment with hyperparameter tunings, such as learning rate, batch size, and regularization techniques, to improve performance.

**Validation and Evaluation:**

- Periodically evaluate the model's performance on the validation set to assess its generalization capabilities.
- Calculate metrics like BLEU (Bilingual Evaluation Understudy), METEOR, or CIDEr (Consensus- based Image Description Evaluation) to measure the quality of generated captions.
- Fine-tune the model based on the validation results and iterate on the training process if necessary.

**Testing and Deployment:**

- Evaluate the trained model on the testing set to assess its performance on unseen data.
- Deploy the trained model in a production environment, ensuring integration with the smart navigation system
- Monitor and evaluate the system's performance in real-world scenarios, collecting user feedback for further improvements.

### 4. Caption Generation and voice output

**Caption Generation:**

Utilize the trained image captioning model to generate descriptive captions for the input images.

Pre-process the input image using the same techniques employed during training, such as

Image Acquisition: Incorporate cameras or visual sensors to capture real-time images of the environment. Ensure the image acquisition component is lightweight, portable, and capable of capturing high-quality images in various lighting conditions.

Image Processing: Implement image processing techniques to enhance the captured images, including demonizing, resizing, and improving contrast. Apply object detection algorithms to identify objects, landmarks, and obstacles in the processed images.

Image Captioning: Integrate a deep learning-based image resizing, normalization, and feature extraction.

Feed the pre-processed image through the encoder part of the model to extract visual features.

Initialize the decoder with a start token and use the encoded features as the initial hidden state.

Employ a decoding algorithm, such as beam search or greedy search, to generate captions word by word.

At each time step, pass the previous word embedding and hidden state through the decoder to predict the next word in the caption. Repeat the process until an end token is generated or a maximum caption length is reached.

DOI: 10.48175/IJARSCT-10645

*Voice Output:*

Convert the generated captions, represented as textual data, into audible speech to provide auditory feedback to visually impaired individuals.

Utilize text-to-speech (TTS) synthesis techniques to transform the captions into spoken words.

Choose a suitable TTS system, such as Google Text- to-Speech, Microsoft Azure Cognitive Services, or open-source libraries like Festival or eSpeak.

Pass the generated captions through the TTS system

to generate corresponding speech audio.

Configure the TTS system to adjust voice parameters, such as pitch, speed, and language, based on user preferences or accessibility requirements.

Play the synthesized speech audio through headphones or speakers to provide real-time auditory descriptions to the visually impaired individual.

## IV. DESIGN

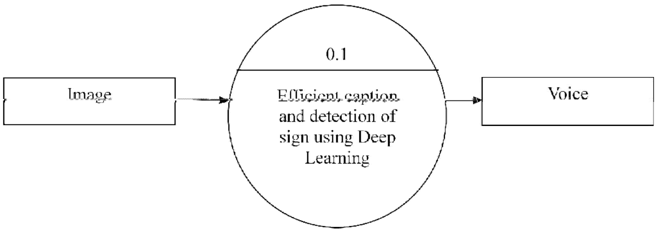

Fig.4.1 Data flow diagram of the overall design

- User Interface: Develop an accessible and user-friendly interface that accommodates different input modalities such as touch, voice commands, or physical buttons. Ensure the interface is designed with clear and intuitive navigation, allowing users to interact with the system easily
- Image Acquisition: Incorporate cameras or visual sensors to capture real-time images of the environment. Ensure the image acquisition component is lightweight, portable, and capable of capturing high-quality images in various lighting conditions.
- Image Processing: Implement image processing techniques to enhance the captured images, including demonizing, resizing, and improving contrast. Apply object detection algorithms to identify objects, landmarks, and obstacles in the processed images.
- Image Captioning: Integrate a deep learning-based image captioning algorithm that generates descriptive captions based on the detected objects and the scene. Utilize a pre- trained model or train a model specifically for the target domain to ensure accurate and contextually relevant captions.
- Text-to-Speech Conversion: Utilize a text-to-speech synthesis system to convert the generated captions into audible speech. Customize the voice output to be clear natural, and adaptable to different languages and dialects. Consider providing options for users to select their preferred voice characteristics and adjust speech parameters.
- Audio Interface: Design an audio interface that delivers the synthesized speech to the user effectively. Provide options for users to use headphones or external speakers based on their preferences. Ensure the audio output is loud and clear to overcome ambient noise and aid in effective communication.
- Navigation and Routing: Implement a navigation and routing module that utilizes the generated captions and object detection results to guide users in their navigation. Integrate with map services or databases to provide accurate location information, landmarks, and points of interest. Utilize audio cues, such as turn-by-turn directions or landmark descriptions, to assist visually impaired individuals in navigating their surroundings.
- Output and Feedback: Design a feedback mechanism that allows users to provide input or report issues encountered during navigation. Incorporate features to receive and process user feedback, helping improve the system's performance and addressing any limitations.

- System Integration and Accessibility: Ensure seamless integration of the designed system with other assistive technologies commonly used by visually impaired individuals, such as screen readers or Braille devices. Address accessibility standards, such as WCAG (Web Content Accessibility Guidelines), to make the system accessible to individuals with different disabilities.
- Continuous Improvement and Updates: Establish mechanisms to collect user feedback and usage data to identify areas for improvement and potential systemupdates. Regularly update the system to incorporate advancements in image processing, captioning algorithms, and accessibility features.

## V. CONCLUSION

In conclusion, Navigating the eye to blind people using a machine learning system is a valuable assistive technology that empowers visually impaired individuals to navigate their surroundings effectively. By leveraging image processing, object detection, caption generation, and text-to- speech synthesis, the system provides real-time auditory descriptions of the environment, aiding visually impaired individuals in understanding their surroundings and making informed navigation decisions.

The system's design encompasses several key components, including the user interface, image acquisition, image processing, image captioning, and text-to-speech conversion. The user interface serves as the primary interaction point, allowing users to input commands and receive auditory feedback. The image acquisition module captures real-time images of the environment, while the image processing module enhances and analyses these images, detecting objects and landmarks.

The image captioning module generates descriptive captions based on the detected objects and the scene, providing meaningful contextual information. These captions are then converted into audible speech through the text-to-speech conversion module. The synthesized speech, delivered through an audio interface, provides real-time auditory descriptions to guide visually impaired individuals in navigating their surroundings.

The system design ensures seamless integration with other assistive technologies and considers accessibility standards to cater to individuals with different disabilities. User feedback and continuous improvement play a crucial role in refining the system's performance and addressing any limitations.

Overall, Navigating the eye to blind people using a machine learning system offers an innovative solution to enhance independence and mobility for visually impaired individuals, providing them with valuable information about their environment and enabling them to navigate with confidence.

## REFERENCES

[1] Vision Loss Expert Group of the Global Burden of Disease Study. Trends in prevalence of blindness and distance and near vision impairment over 30 years: an analysis for the Global Burden of Disease Study. Lancet Global Health 2020. doi: 10.1016/S2214-109X(20)30425-3.

[2] S. R. A. W. Alwi and M. N. Ahmad, “Survey on outdoor navigation system needs for blind people,” 2013 IEEE Student Conference on Research and Development, Putrajaya, Malaysia, 2013, pp. 144-148, doi: 10.1109/SCOReD.2013.7002560.

[3] R. Ani, E. Maria, J. J. Joyce, V. Sakkaravarthy and M. A. Raja, “Smart Specs: Voice assisted text reading system for visually impaired persons using TTS method,” 2017 International Conference on Innovations in Green Energy and Healthcare Technologies (IGEHT), Coimbatore, India, 2017, pp. 1-6, doi: 10.1109/IGEHT.2017.8094103.

[4] Mande Shen and Hansheng Lei, “Improving OCR performance with background image elimination,” 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Zhangjiajie, China, 2015, pp. 1566-1570, doi: 10.1109/FSKD.2015.7382178.

[5] K. Dhivya, G. Premalatha, and S. Monica, “Wearable Navigation Device for Virtual Blind Guidance,” 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 2019, pp. 130-133, doi: 10.1109/ICCS45141.2019.9065816.

[6] S. Zaman, M. A. Abrar, M. M. Hassan, and A. N. M. N. Islam, “A Recurrent Neural Network Approach to Image Captioning in Braille for Blind-Deaf People,” 2019 IEEE International Conference on Signal Processing, Information,

Communication Systems (SPICSCON), Dhaka, Bangladesh, 2019, pp. 49-53,doi: 10.1109/SPICSCON48833.2019.9065144.

[7] Soh, Moses. "Learning CNN-LSTM architectures for image caption generation." Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep (2016).

[8] S. Sonth and J. S. Kallimani, "OCR-based facilitator for the visually challenged," 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), Mysuru, India, 2017, pp. 1-7, doi: 10.1109/ICEECCOT.2017.8284628.

[9] T. Chattopadhyay, P. Sinha, and P. Biswas, "Performance of Document Image OCR Systems for Recognizing Video Texts on Embedded Platform," 2011 International Conference on Computational Intelligence and Communication Networks, Gwalior, India, 2011, pp. 606- 610, doi: 10.1109/CICN.2011.131.

[10] Islam, N., Islam, Z. and Noor, N., 2016, 'A Survey on Optical Character Recognition System', Journal of Information Communication Technology-JICT Vol. 10 Issue. 2, December 2016