A Smart Navigation System for Blind People

Nikesh Ramnani[1], Mridul Jain[2], Atul Neware[3]

[1] Department of Computer Engineering,NBNSSOE, Ambegaon, Pune, Maharashtra, India
[2] Department of Computer Engineering,NBNSSOE, Ambegaon, Pune, Maharashtra, India
[3] Department of Computer Engineering,NBNSSOE, Ambegaon, Pune, Maharashtra, India

---

Abstract - Artificial intelligence (AI) refers to the simulation of human intelligence on computers designed to think like humans and imitate their actions. Science gives importance to the creation of smart systems or machines that operate and react like humans. For humans, it is straightforward to detect and recognize objects as objects. Humans have a remarkable ability to distinguish objects through their vision. However, for machine object detection and recognition, it is a big issue. To overcome the 'Neural Networks' problem, the concept of Artificial Neural Networks has been introduced in computer science. Neural networks are a form of artificial, non-symbolic intelligence. They are computational models of the human brain that help in the object.

Key Words: Artificial Intelligence (AI), 'Artificial Neural Networks (ANN), R-CNN (Regional) Advanced Classifier Convolution Neural Network)

I. INTRODUCTION

Computer vision is a segment in computer science that deals with how a system can be set up to gain a high-level comprehension from digital images. Researching strictly from an engineering perspective, it is trying to automate tasks that are typically manmade and controlled by human beings. Computer vision tasks consist of a variety of methods. This helps acquire, analyze, process, understand images, and extract high-dimensional information from the real world to provide numerical or symbolic information known as Outputs. Understanding in this context means that the transformation of visual images (input of the retina) can alter descriptions of the world and can interface with the rest of the world. Information from image data using models are built with support for geometry, physics, statistics, and learning theory. Computer vision consists of various sub-domains, such as learning, indexing, motion estimation, reconstruction of the scene, and the restoration of the image. Now, a sub-domain of interest, object detection, is relevant to us. It is effortless for humans to detect and recognize objects as humans have a remarkable ability to distinguish between their objects through vision. Nevertheless, for the machine, detection, and recognition is a significant issue. Overcoming this, the 'Neural Networks' issue has been introduced in the field of computer science. It is also called 'Artificial Neural' Networks.

II. Literature Survey

Visually impaired individuals live a lifestyle quite different from what we call "regular" and often face challenges in performing day-to-day tasks. A blind person's greatest challenge, particularly those with complete vision loss, is navigating around locations. Those with vision impairment can walk around their homes and most immediate surroundings since these spaces have been adjusted according to their needs. This creates a big problem for those visually impaired. Many visually impaired people need the help of adaptive technologies that take online information and read it, translate it into Braille, or magnify it. The exact manner in which a visually impaired person will communicate with information using adaptive technologies depends on the specific technology and the individual's particular disability. This literature review offers explanations for the already existing navigation technologies' ineffectiveness by examining various research papers.

In this literature review, we summarize the potential enhancement that technology can offer to already existing systems. For a decade, both Navigation and artificial intelligence have played a significant part in a user's life. Research has been conducted that looks at how effective these technologies are to the visually impaired. Kaiser et al. (2012) studied that there are available solutions that help visually impaired people read, write, and navigate but still have room for improvement. Kaiser et al. found that Navigation for visually impaired and blind individuals in an unfamiliar outdoor and indoor setting is a significant challenge to be solved. Navigation systems are available, but they cannot provide visually impaired and blind people's accuracy. While using the traditional navigation application, a blind pedestrian will explore an unknown environment while the system builds a map and simultaneously tracks its position. Once a map is constructed of the unknown environment, the system's primary goal is to choose safe and effective routes to guide the blind person from a starting point to destinations previously explored.

Both Lawo (2012) and Nagarhalli (2018) suggest that enhancement can be achieved using Machine Learning and Pinhole Camera approximation concepts. Intelligent parking assistance systems use advanced algorithms and the Pinhole Camera approximation principle to estimate the distance between the computer and the object. We can have a single camera that can detect the objects and their distance from the vehicle (its camera). The data so obtained can be used to avoid collisions and fatal accidents. The same concept can be applied to existing navigation applications. In this case, the phone camera will detect the objects and the distance and notify the user about them to avoid any obstacles in the path. Upon reviewing existing papers and doing individual research, we conclude that the current navigation system is not competent enough for the visually impaired users, and they need enhancements to provide a better experience to the users. Various changes can be made, and multiple concepts like machine learning and pinhole camera approximations can be further improved to betterment our topic mentioned above.

III.Methods

1. TensorFlow Object Detection Model

For data flow programming across various activities, TensorFlow is Google's Open Source Machine Learning Platform. In the network, mathematical operations are represented by nodes, while the graph edges represent the multidimensional data arrays (tensors) communicated between them. Tensors are just multidimensional arrays, an extension of a higher dimension of 2-dimensional tables to data. TensorFlow has many characteristics, making it ideal for deep learning. So, without wasting any time, let us see how we can use TensorFlow to implement Object Detection.

1.1 Advantages of TensorFlow over other frameworks:

1. Quick deployment
2. Better Support, compared to other versions, for GPUs.
3. For building models, it provides high-level APIs.
4. Unconventional and hard-core alterations are straightforward to do.

1.2 Object Detection and Recognition

Object detection is used to identify, locate, and track objects on the software system from a particular image or video. The unique object detection attribute identifies the Object detection is used to identify, locate, and track objects on the software system from a particular image or video. The unique object detection attribute identifies the object class (person, table, chair, etc.) in the provided image and its location-specific coordinates. The location is indicated by drawing a bounding box around the object. The bounding box will locate the location of the item correctly or not. The output of an algorithm used for detection is defined by the ability to locate the object within an image. Examples of object detection involve face detection.

Object detection and recognitions are relevant in the following topics:

1. Surveillance systems for monitoring behavior or changing information in order to detect unusual activities.
2. Video indexing, for the retrieval and recovery of videos in databases.
3. Traffic monitoring for simultaneous traffic inspection to direct traffic flow.
4. Vehicle navigation for real-time path planning and obstacle avoidance capabilities in robotics.

1.3 How does Object detection work?

In general there are three steps to perform the task of object detection:

1. Provided in the figure below, generate the small segments in the input. As you can see, the wide border boxes span the entire picture.

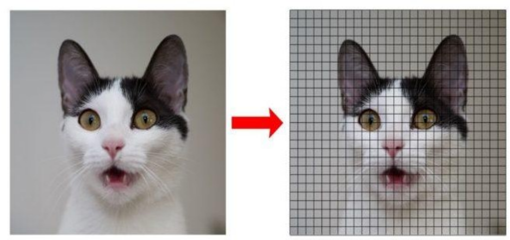

Fig. 1 Generation of segments in the image

2. The extraction function is done to determine if the rectangle contains a valid object for each segmented rectangular field.

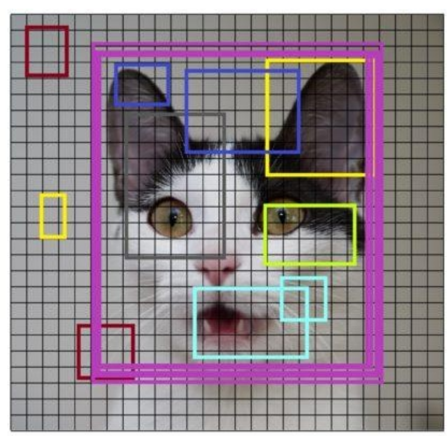

Fig. 2 Extraction of features from the image

3. Boxes that overlap are grouped into a singlerectangle (Non-Maximum Suppression)

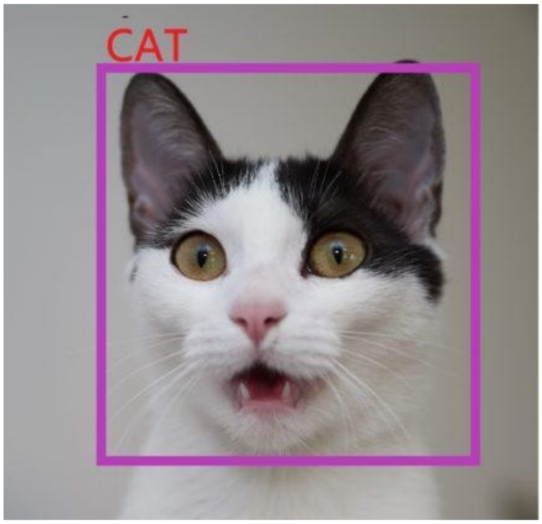

Fig. 3 Combination of bounding rectangles

1.4 Object Detection Problems

Detecting the desired object can be a tricky task since estimating a moving object's trajectory in the image plane has to be accurate. The main problems related to object detection are as follows:

1. Information loss when projecting from a 3D world to a 2D image.
2. Existence of noise in the images.
3. Complex motion of the objects.
4. Complex characteristics of the objects, such asnon-rigid/articulated objects.
5. Partial or full object occlusions;
6. Complex shapes of the objects;
7. Illumination variations in the scene; Real-time processing requirements.

2. Triangle Similarity

For this project, triangle similarity will be used in order to calculate the distance from the camera to a known object ormarker.

Something like this goes with the triangle similarity: assuming there is a marker or entity with a known W width.A marker is placed some distance away from the camera, D. Using the camera, an image of the object is taken and then its apparent width in pixels is measured in P. This enables the perceived focal length F of the camera to be derived:

$$F = (P \times D) / W$$

For instance, if there is a regular 8.5 x 11in piece of paper (horizontally; W = 11) D = 24 inches in front of a camera and a picture is taken. It is noted that the perceived width ofthe paper is P = 248 pixels when calculating the width of the piece of paper in the image. Then the focal length F is:

$$F = (248\text{px} \times 24\text{in}) / 11\text{in} = 543.45$$

As the camera continues to move both closer and further away from the object/marker to evaluate the distance of the object to the camera, the triangle similarity can be applied:

$$D' = (W \times F) / P$$

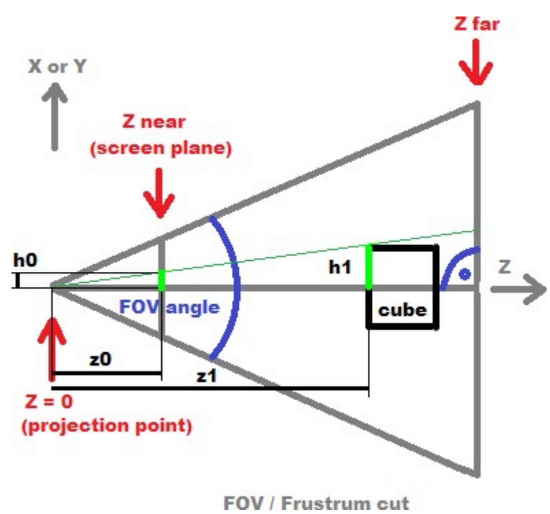

Fig. 4 Camera to object triangle

If the camera is moved 3 ft (or 36 inches) away from the marker and a snapshot of the same piece of paper is taken to make this more concrete, it can be determined that the perceived width of a piece of paper is now 170 pixels through automated image processing. Plugging it into the equation that is derived now:

$$D' = (11\text{in} \times 543.45) / 170 = 35\text{in}$$

Or 36 inches approximately, which is 3 feet.

IV.SYSTEM ARCHITECTURE

The TensorFlow runtime is a cross-platform library. This combination of size is flexible by the system architecture. TensorFlow's programming principles like the computer graph, operations and sessions are familiar to us.

Below are the terms and their functionality in the TensorFlow architecture.

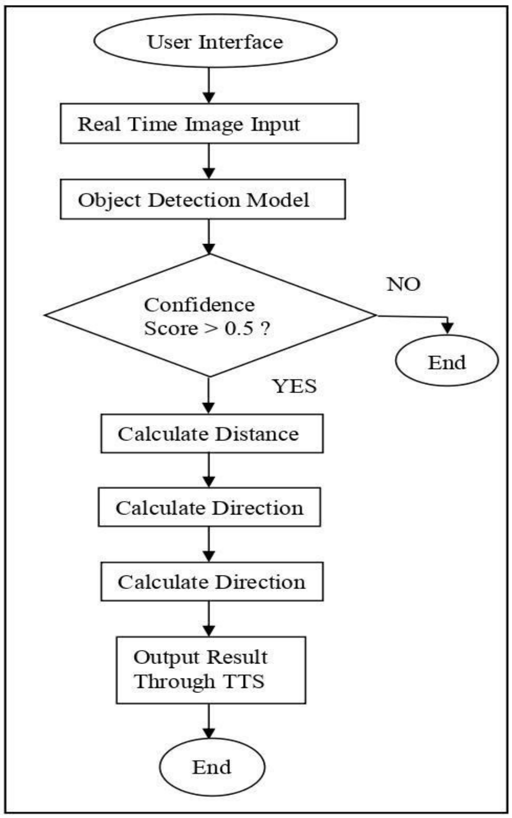

Fig. 5 Dataflow Diagram

4.1 TensorFlow Servable

These are the core unfinished units in the service of TensorFlow. Servables are the artefacts used to perform the computation by customers. A servable is flexible in its scale. A single serving machine can consist of many interface models from the lookup table to a single model. Any form and interface should be useful, allowing versatility and potential improvements such as:

- Streaming results
- Asynchronous modes of operation.
- Experimental APIs

4.2 Servable Versions

During the lifetime of a single server instance, TensorFlow server can handle one or more versions of the servers. It opens the door for new combinations of algorithms, weights, and it is possible to load other data over time.

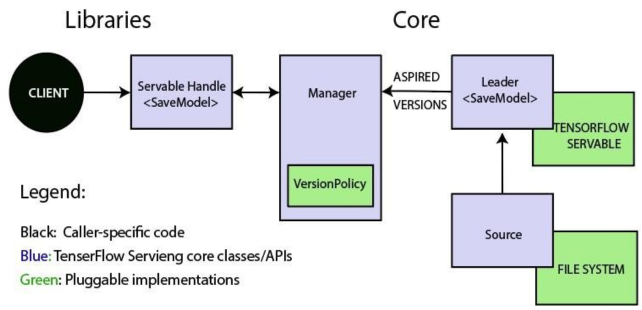

Fig. 6 TensorFlow Architecture

They can also allow for a charge of more than one edition of a servable at a time. They also allow for concurrent loading of more than one iteration of a servable, increasingly promoting roll-out and experimentation.

4.3 Servable Streams

Servable Streams are a sequence of versions of any servable sorted by increasing versions of numbers.

4.4 TensorFlow Models

A serving represents a model in one or more servables. A machine-learned model includes one or more algorithms and looks up the embedding tables. A servable can also serve as a fraction of a model; for example, a large lookup table can serve many instances.

4.5 TensorFlow Loaders

Loaders manage a servable's life cycle. The loader API enables common infrastructure independent of the specific learning algorithm, data, or product use-cases involved.

4.6 Sources in TensorFlow Architecture

In simple terms, sources are modules that find and provide servable. Each reference provides zero or more servable streams at a time. For each servable stream, a source supplies only one loader instance for every servable. Each source also provides zero or more servable streams. For each servable stream, a source supplies only one loader instance and makes available to be loaded.

4.7 TensorFlow Managers

TensorFlow managers are responsible for a Servables' entire life cycle, including:

- Loading Servables
- Serving Servables
- Unloading Servables

The manager tracks all versions of the sources. The manager wants to do something but can refuse to load a version of the Aspired. For example, a manager can wait until the loading of a newer version is complete, based on the policy of ensuring at least one version is loaded all the time. The manager can also delay unloading. For example, for customers to access the loaded servable instances, GetServableHandle( ).

4.8 TensorFlow Core

This manages the below aspects of servables:

- Lifecycle
- Metrics
- TensorFlow serves core satisfaction servables and loaders as the opaque objects.

TensorFlow is an open source library for mathematical calculation and enormous scope AI. This figure is a portrayal of how tensorflow workers are utilized for customer application. The accompanying chart shows the interaction that goes behind the work process of TensorFlow Server to Client Application.

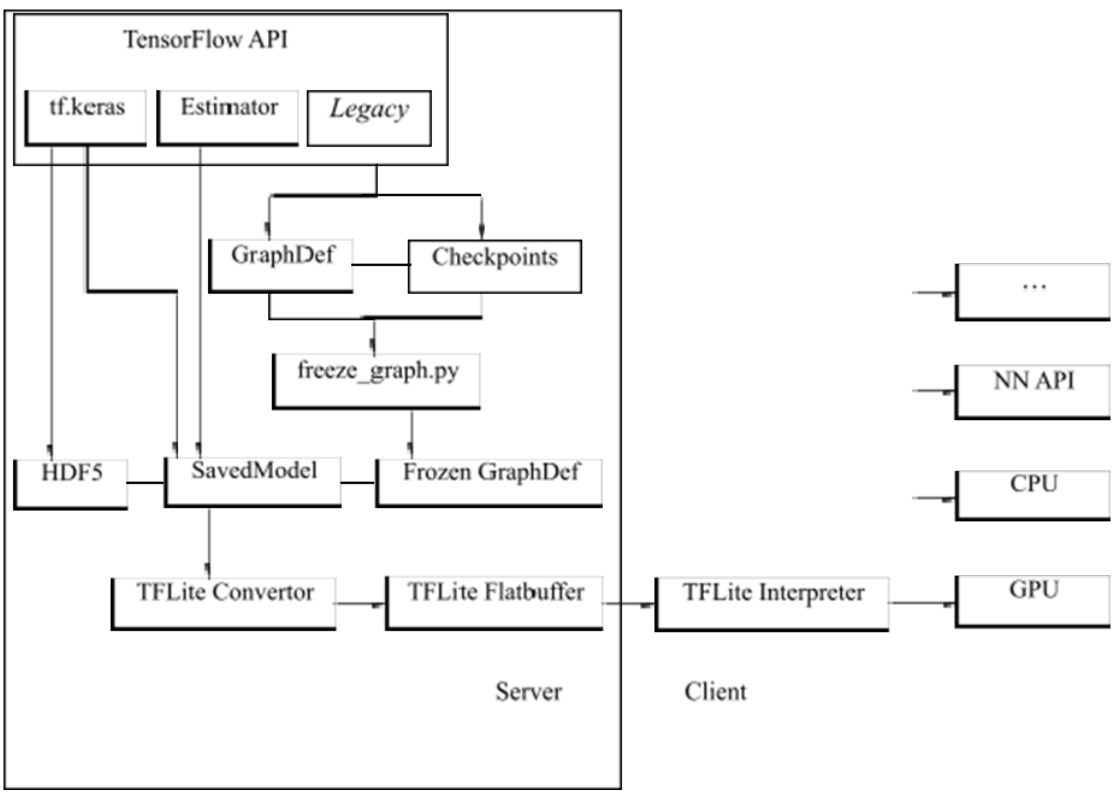

Fig. 7 Workflow of TensorFlow Server to Client Application

ssSr.no	***Author***	***Topic***	***Concept***	***Method***	***Limitations***	***Advantages***
1	*Reagan L. Galvez, Argel A. Bandala, et al;*	*Object Detection Using Convolutional Neural Networks*	*model is ideal for real-time application because of speed and the other can be used for more accurate object detection., To detect objects by shape and color pattern recognition.*	*Convolutional Neural Networks (CNN), Faster Region-based Convolutional Neural Network (Faster-RCNN), Single Shot Multibox Detector (SSD)*	*Less accuracy by SSD, less speed detection by Faster-RCNN*	*High Speed detection by SSD, More accuracy by Faster-RCNN*
2	*Rasika Phadnis, Jaya Mishra, et al;*	*Objects Talk - Object detection and Pattern Tracking using TensorFlow*	*To detect objects by shape and color pattern recognition.*	*TensorFlow, REMO, pattern recognition, SSD MobileNet*	*system goes into an unstable state and user gets an alert via mobile application*	*maximise the efficiency*
3	*Aishwarya Sarkale, Kaiwant shah, et al;*	*An Innovative Machine Learning Approach for Object Detection and Recognition*	*The system aims to deal with object detection and recognition of static objects*	*Faster R-CNN (Regional Convolution Neural Network), Artificial Neural Networks*	*the size of sample increases the accuracy decreases.*	*The performance in the terms of accuracy has greatly improved.*
4	*Adrian Rosebrock*	*Find distance from camera to object/marker using Python and OpenCV*	*To detect object by triangle similarity*	*Triangle Similarity*	*Not work in dark places*	*The distance and focal length of object is accurate*
5	*Rais Bastomi, Firza Putra Ariatama, et al;*	*Object Detection and Distance Estimation Tool for Blind People Using Convolutional Methods with Stereovision*	*to create a device that can detect an object and determine distance using the Convolutional Neural Network method.*	*Convolutional Neural Network (CNN)*	*Stereo Vision has limited distance in measurement*	*the success rate of detection is quite high*

Table 1 Literature Survey

Fig. 7 Detection of Objects

V. CONCLUSION

This model and the whole system surrounding it focuses on the need for smart Navigation to make movement easier for visually impaired people. It aimsto create an environment that is accommodating to a blind person's needs to ensure maximum comfort and efficiency for them in forthcoming times. This model aims to reform how things are done and work on making better outcomes for coming generations. This paper has looked at all the things that have been done so far, whether or not they have worked, and based on everything that has been analyzed, and there is hope for what can be done in the future. Research articles have played an immense role in supporting all arguments made in this research paper. Journals like "*Object Detection Using Convolutional Neural Networks"* talk about how this model is ideal for real-time application because of speed and the other can be used for more accurate object detection to detect objects by shape and color pattern recognition. Similarly, Journal paper "*An Innovative Machine Learning Approach for Object Detection and Recognition"* talks about how the system aims to deal with object detection and recognition of static objects.Compiling all this research and strategizing the best methods that can be used to create an efficient navigation has been the main goal of this research paper.

VI. REFERENCES

[1] Reagan L. Galvez, Argel A. Bandala, et al., "ObjectDetection Using Convolutional Neural Networks", *Proceedings of TENCON 2018 - 2018 IEEE Region 10 Conference (Jeju, Korea, 28-31 October 2018)*.

[2] Rasika Phadnis, Jaya Mishra, et al., "Objects Talk - Object detection and Pattern Tracking using TensorFlow",*Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018.*

[3] Aishwarya Sarkale, Kaiwant shah, et al., "An Innovative Machine Learning Approach for Object Detection and Recognition", *Proceedings of the 2nd International Conference on Inventive Communication andComputational Technologies (ICICCT 2018).*

[4] Adrian Rosebrock, "Find distance from camera to object/marker using Python and OpenCV",2015.

[5] Rais Bastomi, Firza Putra Ariatama, et al,"Object Detection and Distance Estimation Tool for Blind People Using Convolutional Methods with Stereovision", *International Symposium onElectronics and Smart Devices,2019*.