International Conference on Science, Technology, Engineering & Management
[ICON-STEM'15]
Journal of Chemical and Pharmaceutical Sciences ISSN: 0974-2115

Auditory guide for the visually challenged

Ambika.K*, Jayashree.L, Angel Diana.S
Department of Electronics and Instrumentation Engineering, Jeppiaar Engineering College, Chennai, Tamil Nadu
***Corresponding author: E.Mail: ambika811@gamil.com**

ABSTRACT

Visual impairment compels the need to depend on another person or a guide dog or instruments such as canes for guidance and navigation. This paper proposes an auditory guide that would use object tracking/obstacle detection and text/LED pattern recognition to generate navigational instructions. Its audio output would help the visually challenged user to traverse familiar or unfamiliar locales on their own without any external aid.

Keywords: Text Recognition, Structural analysis, Object tracking, Background subtraction, Histogram of Oriented Gradients, Blob detection, Open CV, Raspberry pi B+, OCR Tesseract, ESpeak

INTRODUCTION

In this paper, we have proposed an auditory guide for the visually challenged. The guide captures images of the environment and tracks objects, so as to help the user avoid such obstacles. Also, it reads LED patterns, such as in name boards and traffic signals to help the user discover his location and also to move amidst traffic. It uses a high speed camera to capture real time images. The image frames are extracted from the video feed and are analysed to detect objects and obstacles. Also, any pattern or text that is created using LEDs is analysed and read. In case of detection of an obstacle, a direction is generated to the user so as to avoid colliding into it. An obstacle includes both moving objects and non-moving objects. In case of an LED pattern, the alphabet or number alone is read and compared to the existing database to identify the text as entire words. The text or the direction of navigation is both fed to a software that converts text to speech. The speech is given to the user through an audio feed or using a Bluetooth device.The main objective is to make the device portable and compact so that it does not cause any inconvenience to the user or hinder their normal activities, while also making it cost effective and robust. We have implemented the proposed design on a Raspberry pi B+ module using an open source image processing software "Open CV". The software ESpeak was used in the conversion of the text instructions into audio output.

Text and LED pattern recognition using Structural Analysis: A combination of Background subtraction and structural analysis techniques were used in recognition of patterns of LED text.

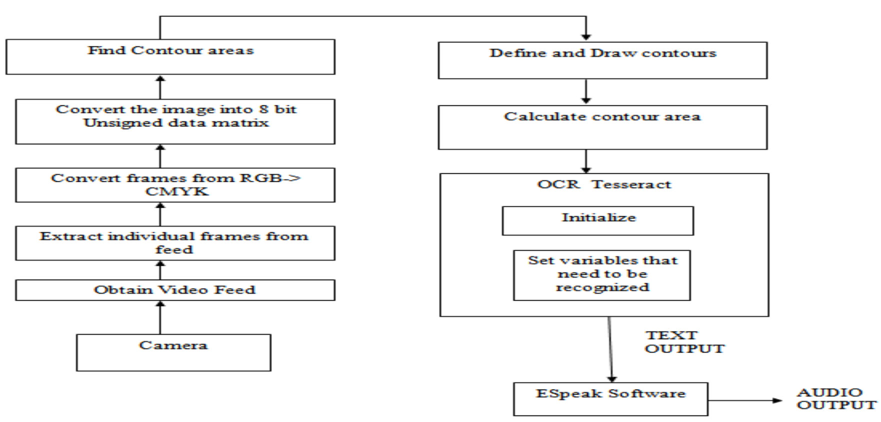

Fig.1.Flowchart for text detection

Initially, the video feed from the camera is extracted into individual image frames. The number of frames extracted depends on the speed of the camera. The frames are then converted to CMYK color mode for better resolution and subsequently into 8 bit unsigned data for faster processing. Structural analysis engages contours for detection. Contours are just a curve joining all the continuous points (along the boundary), having identical color or intensity.[1] The contours are a helpful tool for shape analysis in Text detection.

Canny edge detection to find contours: Contours are characterized by canny edge detectors. The Canny edge detector is operator for detecting edges where a multi-stage algorithm is used to detect a wide range of edges in images.[2][3]The progression of Canny edge detection algorithm can be identified as 5 different steps:

1. Implementation of the Gaussian filter to remove the noise by smoothening the image
2. Identification of the intensity gradients of the image
3. Application of non-maximum suppression to prevent spurious response to edge detection
4. Application double threshold to determine potential edges
5. Tracking edge using hysteresis: Finalization of the recognition of edges by suppression of all the other edges that are feeble and not linked to strong edges.

OCR Tesseract: Tesseract is an optical character recognition engine for diverse operating systems. Tesseract supports formatting of output text, positional information on hOCR and analysis of page layout and a number of new image formats added using the Leptonica library. Tesseract can also perceive if the text is mono-spaced or proportional. Tesseract's response will be of poor quality if pre-processing of the input images is not done: Input must be rescaled so that the height of the text is at least 20 pixels, any rotation or distortion must be corrected, low-frequency changes in brightness must be high-pass filtered. Also dark borders must be manually removed to avoid their misinterpretation as characters.

Optical Character Recognition: Optical character recognition (OCR) is the mechanical or electronic conversion of images of typewritten or printed text into machine-encoded text. It is extensively used as a type of data entry from printed paper data records. It is a widespread method of converting printed texts to digital format so that it can be easily edited and searched electronically, stored more efficiently, presented on-line, and implemented in processes such as automatic machine translation, conversion of text-to-speech, text and data mining. Advanced systems now have recognition accuracy that is of a high degree for most fonts and are even capable of replicating output that strongly resembles the original page including columns, non-textual images etc.

A) Pre-processing: OCR software does "pre-processing" of images to increase the odds of successful detection using techniques such as:

- De-skewing
- De-speckling
- Binarization
- Line removal
- Layout analysis or "zoning"
- Line and word detection
- Script recognition
- Character isolation or "segmentation"
- Normalize aspect ratio and scale

B) Character Recognition: There are basically two kinds of OCR algorithm that can bring about the recognition of characters: Matrix matching or pattern matching or image correlation entails the comparison of the image to a stored glyph pixel-by-pixel. Correct isolation of the input glyph from the remaining image and the similarity and proper scaling of input glyph to the stored glyph is essential in the process. Feature extraction causes decomposition of the glyphs into "features" such as lines, loops, line direction, and line intersections. Then comparison of these with an abstract vector-like representation of a character is done and reduction to one or more glyph prototypes takes place. This method is termed as "intelligent" handwriting recognition. For choosing the nearest match by comparing the features of images with stored glyphs, classifiers such as the k-nearest neighbors algorithm are implemented. Tesseract uses a two-pass approach in recognition of characters. The second pass is termed as "adaptive recognition". The letter contours identified with more confidence on the first pass are used to detect the remaining letters on the second pass. This is beneficial for atypical fonts or poor-quality scans with font distortion such as blurs.

C) Post-processing: The accuracy of OCR can be improved if the output is controlled by a lexicon which restricts the words that are allowed in a certain type of document. Tesseract employs its dictionary to manipulate the character isolation step, for better precision. "Near-neighbor analysis" includes the rate of co-occurrence to rectify errors. This is done by noting that certain words are often bound to occur together. Knowledge of the grammar can also help determine if a word is likely to be a verb or a noun, for example, allowing greater accuracy.

III. Object tracking and Obstacle avoidance using HOG descriptors: The flow of the process of object tracking is defined by the techniques such as background subtraction and blob detection for nonmoving objects and Histogram of Oriented Gradients descriptor function for moving objects.

Background Subtraction: Background subtraction (BS) is an extensively implemented technique to generate a foreground mask a foreground mask contains all the pixels of the moving object in the image captured by the static camera. The background is taken as the static part of the observed scene. A foreground mask is created by subtraction of all the objects in the current frame from the static background model. The two main steps of Background modeling include Background initialization where the first background model is created and Background update where variations are added to the model so that adaptation to changes in the scene can be made.

Blob Detection: Blob detection includes methods used to detect areas in an image that vary in properties like intensity or color as compared to surrounding regions. However, all the points inside the blob are considered to be similar to each other. The two main classes of blob detectors are: (i) *differential methods* and (ii) *methods based on local extrema*. Blob detection offers complementary data about regions, whichwere not given by edge or corner detectors. Blob descriptors are also used in histogram analysis to detect peaks and in segmentation.

Histogram of Oriented Gradients (HOG) Descriptor: Histogram of oriented gradients (HOG) is a feature descriptor used in detection of objects. The HOG descriptor technique counts occurrences of gradient orientation in localized portions of an image - detection window, or region of interest (ROI).

HOG Descriptor Algorithm: The HOG Descriptor algorithm is implemented using the following steps:

1. Division of the image into small associated regions termed cells and computation of a histogram based on gradient directions or edge orientations for the pixels within each cell.[11]
2. Discretization of every cell into angular bins depending on their gradient orientation.
3. Each cell's pixel contributes weighted gradient to its corresponding angular bin.
4. Adjacent cells are grouped together to form regions called blocks. Then, normalization of the histograms is done based of the data from the blocks.
5. The descriptor is derived from the normalized histograms.

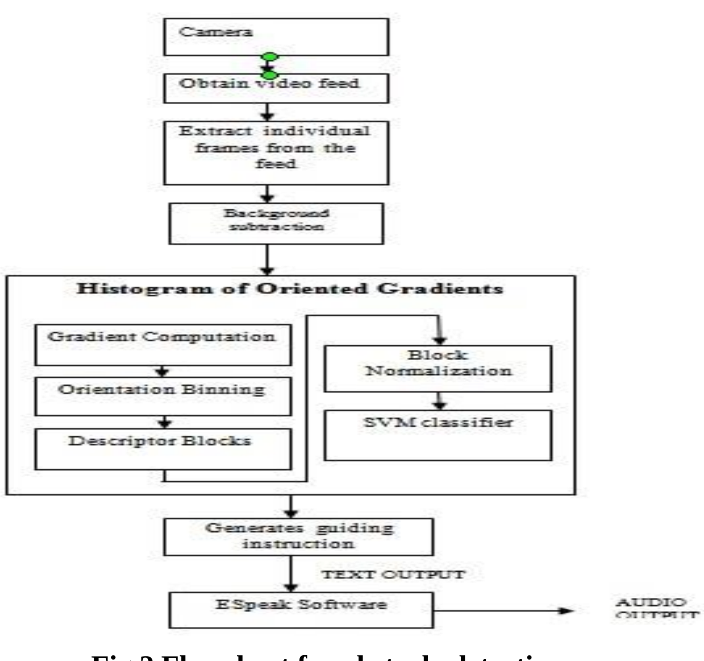

Fig.2.Flowchart for obstacle detection

Simulated output: Executing the Open CV program using Linux commands in the Raspian operating system, we get the output from ESpeak software. For Obstacle tracking, the following output is acquired. As the video feed is obtained, three display windows are shown. The first window displays the constant background model. The second window shows the foreground mask. The foreground mask is in monochrome since that makes identification of movement easier. The third window searches for obstacles, i.e., any object that is a deviation in the detected foreground. This is shown in Fig (3) and Fig (5).When the obstacle is detected, an instruction is relayed to the user through audio. For demonstration purposes, here it is represented by a text instruction and then given as an audio output. Since the instruction was pre-defined as "obstacle detected", it is displayed on the top left corner of the window in Fig (4) and Fig (6).Fig (3) and Fig (4) represent stationary objects while Fig (5) and Fig (6) represent stationary objects.

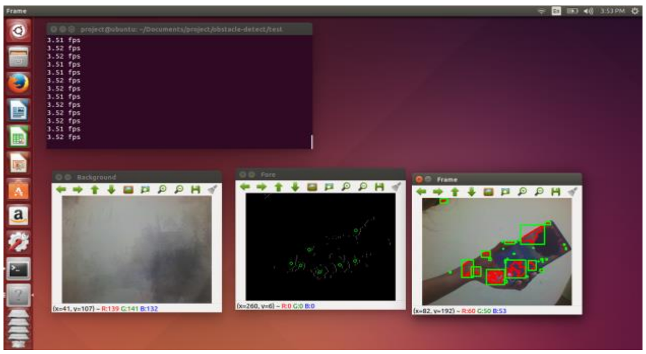

Fig.3.Background subtraction for stationery object 1

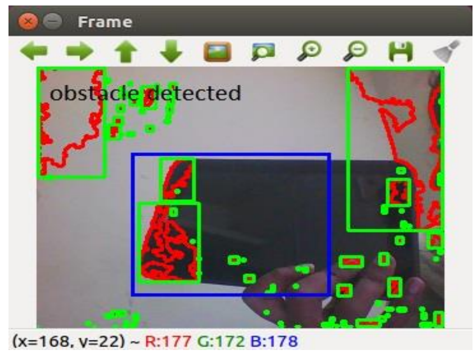

Fig.4.Object 1 detected as obstacle

For Text/LED pattern recognition, the following output is acquired. Fig (7) shows the performance of background subtraction process to detect the region where the character content is available. Then structural analysis of the region takes place. A contour is detected, drawn and its area is calculated to isolate the characters from the background. The background is given a threshold value. Any pixel whose value is greater than the threshold value (light) is recognized in the contour.

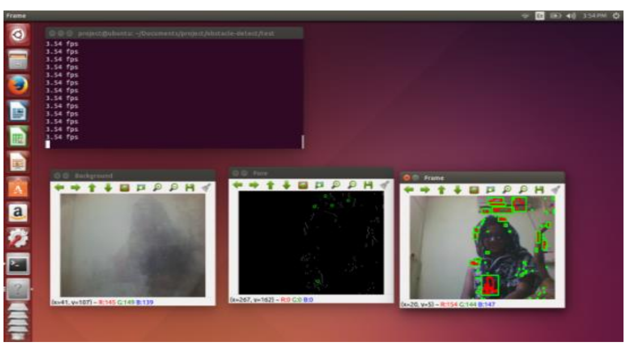

Fig.5.Background subtraction for non-stationery object (a person)

Fig.6.Non-stationery object detected as obstacle

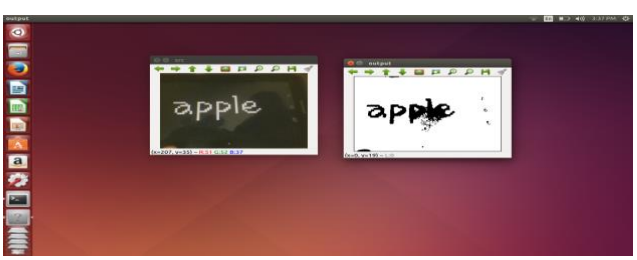

Fig.7.Recognition of the word as a whole

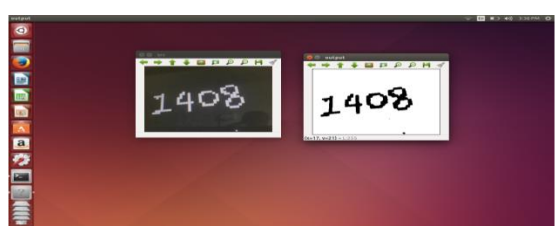

Fig.8.Recognition of numerical characters

Journal of Chemical and Pharmaceutical Sciences ISSN: 0974-2115

By recognizing the string of pixels, the character is given form. Then the characters are recognized and the output is given not as individual letters but as entire words based on the lexicon. Fig (7) shows detection of the word "apple" and Fig (8) shows the identification of numerical characters.

CONCLUSION

The algorithms used were found to be effective in identifying their respective objectives despite variations in lighting, orientation and environments. The output response was found to be quick and the noise immunity was significant. Owing to compact Hardware, the device is portable. Since all the softwares used are open-source softwares, the implementation was cost effective. The heat produced by the processor was expelled rapidly by the usage of a heat sink. This ensured efficient working of the processor without the requirement to be turned off. The scope for implementation of this idea is extensive. This idea can be used not only for the visually challenged but also by etc.The only drawback found was the need of a high resolution camera with rapid response speed. In future, this drawback can be overcome by increasing the efficiency of the algorithms to compensate for the video feed of low end camera with even lesser resolution.

REFERENCE

A.K. Jain and B. Yu, Automatic Text Location in Images and Video Frames, Pattern Recognition, 31(12), 1998, 2055-2076.

Bretzner and Lindeberg, The scale-invariant feature transformas well as other image descriptors for image matching and object recognition, 1998.

D. Chen, J.M. Odobez, H. Bourlard, Text Detection and Recognition in Images and Video Frames, Pattern Recognition, 37(3), 2004, 596-608.

D. G. Lowe. Distinctive, Image features from scale-invariant keypoints, IJCV, 60(2), 2004, 91–110.

H. Li, D. Doermann, and O. Kia, Automatic Text Detection and Tracking in Digital Video, IEEE Trans. Image Processing, 9, 2000, 147-156.

J. Canny, A computational approach to edge detection, PAMI, 1986

Moeslund, T, Canny Edge Detection. Retrieved December 3, 2014

Suzuki, S. and Abe, K, Topological Structural Analysis of Digitized Binary Images by Border Following, CVGIP, 30(1), 1985, 32-46.

T. Lindeberg, Detecting Salient Blob-Like Image Structures and Their Scales with a Scale-Space Primal Sketch: A Method for Focus-of-Attention,(abstract page). International Journal of Computer Vision, 11 (3), 1993, 283–318.

X. Chen, A.L. Yuille, Detecting and Reading Text in Natural Scenes,In Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition, 2004, 366-373.

Z. Liu, S. Sarkar, Robust Outdoor Text Detection Using Text Intensity and Shape Features, in Proc. IEEE Int'l Conf. Pattern Recognition, pp. 1-4, 2008.