VISION

Re-imagining Mobility for the Visually Impaired

Centre For Innovation

SAHAAY Social Innovation Club

Team: Ira, Bhavnashri, Rishi, Jaayanth, Abesech

The Problem

36 million people are blind worldwide with 19 million (more than half) from India. Assistive technology is necessary not to make them feel dependent and to enable safe mobility & avoid injuries.

A white cane is a device used by visually impaired people and it allows its user to scan their surroundings by poking into objects. The white cane that is used for determining a safe path to walk by does not suffice to locate moving (like vehicles), protruding (like fences), over-hanging (like barriers) or hollow objects (like bicycles).

The picture below demonstrates how a white cane could fail in detecting objects. The cane does not indicate any obstacle since within the region in front of the user's lower half of body. In this case, a collision is bound to occur.

Existing solutions for this problem are listed below. These either fall short in a few aspects or incomplete in the set of choices they give the user:

- Ultrasonic Sensors to detect using reflection: These products in the market fail to get users. It is just as expensive as a Computer Vision based solution but inaccurate due to the indistinguishability of noise, limited field of vision, and high sensitivity to it's orientation.
- Using the Smartphone's camera and processing power: This solution is of good functionality but leaves out several aspects of a smartphone's daily usage. Calls might be an interruption, leaves little to no choice to the user to use their mobile for anything else, and it also puts on a constraint that mobility works only if the smartphone is with the users (and more importantly charged)
- Using auditory Beeps when an object is close: This does not give the user any indication of where the object is.
- Using audio descriptions: Usually Auditory senses of a visually impaired person are much sharper and they use this for several environmental stimulus. Using audio descriptions discounts on this sense and, although good in functionality, is better avoided.

Keywords

Assistive Technology, Social Innovation, Blindness, Visually-Impaired, Mobility, Stereo Cameras, Depth Map, Deep Learning, Computer Vision, White Cane, Cam-n-crank mechanism, Stepper Motors, Actuators, Microcontrollers, Bluetooth Communication, Compound rotating cylinder mechanism, Spring-based locking mechanism, Object localization, Somatosensory communication

Motivation & Solution

We attempt to mimic the human eyes using a stereo-camera. This let's us recieve depth information. We process this information to find out where the closest object is. This is indicated to the user through touch sense at the palms through a gripper attached to their existing cane.

Thus our solution can be divided inito two modules: The Vision Module and The Gripper Module.

The Vision Module is a belt worn that carries the stereo camera and a microcontroller for processing (Jetson Nano). This belt captures images, finds the closest object and sends the information to another microcontroller at the Gripper Module

The Gripper Module is an attachment to the white cane that carries a microcontroller (Mini Arduino). This alerts the user about where in the field of vision the closest object is located so that the user can deal with it but using a 3x3 matrix of pins that protrude out to let the user know that the object is there by when they hold the cane.

Brief of Previous Methods

Previously the Vision Module was based on the Deep Learning algorithm You Only Look Once (YOLO). Hence the microcontroller within the Vision Module was chosen to be a Jetson Nano to offer GPU Acceleration.

Results:

- YOLO v3 was found to be too slow with an FPS of 1.18 (Left Image Below)
- Tiny YOLO v3 was also found to be too slow for our application with an FPS of 1.38 and also comparatively much more inaccurate (Right Image Below)

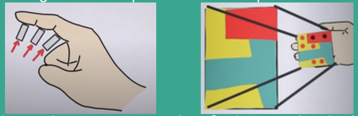
Previously the Gripper Module was based on a Cam-n-Crank mechanism operated by Servo Motors.

Results:

- Having 9 Servo-motors either made the Gripper much larger than what can be held in the arm or had no space left for the microcontroller to fit into

Mechanical Design

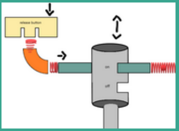
The Gripper Concept is the idea of communicating our results to the user using a matrix of pins at the user's palm.



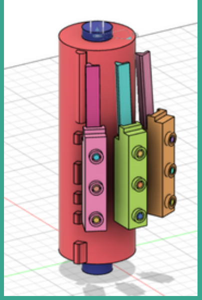
The gripper has a 3x3 matrix of pins. Each pin has two positions 'off' and 'on'. 'on' is when the pin is pushed further radially out. A pin is set to 'on' if the closest object within the field of vision covers that 1/9th of the field of vision.

Pin Locking Mechanism:

The pins are made with a spring-based locking mechanism shown. When the release button is not pressed, the pin retains its state.



Cylinder-Ramp Mechanism:



The pins are updated one column at a time using a rotating cylinder. This cylinder has column of features: 1. A long bulge that presses the release button 2. A sequences of ramps or nothing to push up or let the pin fall before locking again. Each vertical displacement of the cylinder corresponds to one combination for the column of pins. Thus the updation is done by choosing the vertical position first and then rotating it across the column of pins, repeating this process for all 3 columns of pins.

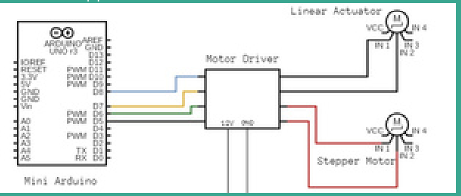
A Combinatorial Problem: The sequence need not have 3 x 8 elements corresponding to 3 pins each in all cases, but can reuse elements in different cases by overlapping them. The design for the sequence of ramps had several trade-offs: 1. No. of pins at a time vs No. of motor operations 2. Length of Cylinder vs Pin spacing. Taking appropriate constraints reduced the problem to this in the end from which one of the many solutions was used:

```
Find a binary sequence of length 14 such that each
substring str[n] + str[n+3] + str[n+6] is unique
```

Electronic Circuitry

The Vision Module's electronic circuit has a Jetson Nano, Power Source and a Stereo Camera and data will be sent to the Gripper through the wifi module (Capable of about 867Mbps).

The Gripper's circuit is shown below:

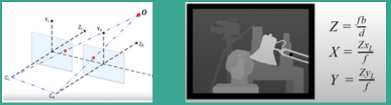


The Arduino translates the received 9-bit binary information into the 6 motor moves for pin updation.

Object Localization

Our Method starts with calibrating the stereo camera first. We remove distortions and rectify the images (using OpenCV).

Nextly, we find the Depth Map using Triangulation Method. Disparity between two similar points in the Right and Left channels are found. Then using some trigonometry we can find their distance from the camera.



The catch here in comparison with our previous model is that we do not need to detect objects but only localize them. Now using this depth map we can use a MaxPooling-inspired algorithm instead of going for a heavy-weight deep learning algorithm and increase our FPS.

1. We take the denoised depth map and we split it into several pools.
2. In each pool, we find the largest 'x' such that the fraction of entries within the range [x-a, x+a] is greater than b. 'a' and 'b' here are parameters for the algorithm
3. We repeat the process until we reach a 3x3

Reasoning behind the algorithmic choices:

1. Running a max pool directly will pick up all noises and in the end we will just end up with a white image, thus there should be some condition on the fraction of pool covered 'b'.
2. Hollow object such as a bicycle will get missed out if this condition with b is applied in a single iteration. Hence we run this on several fine levels iteratively
3. A range using 'a' is chosen because any object should be confined to some dimensions in reality.

Results:

Way Ahead

Possible developments from here:

1. Since the OpenCV based algorithm doesn't need a GPU. We can optimize cost of product by moving to a Raspberry Pi. Similarly we can go for a more elementary communication between the modules and maybe even an ARM or an AVR instead of Arduino
2. Redesign the belt so that the cane doesn't hide field of vision
3. Optimize Motor moves by running one motor at constant speed and analysing how the other should be moved
4. Expand resolution from 3x3 to higher
5. Optimize localization algorithm to handle edge cases more reliably

References

1.Gripper concept:https://youtu.be/8Au47gnXs0w
2.Triangulation:https://learnopencv.com/depth-perception-using-stereo-camera-python-c/
3.Wifi_Communication:https://learn.sparkfun.com/tutorials/adding-wifi-to-the-nvidia-jetson/all
4.Stereo_Calibration:https://temugeb.github.io/opencv/python/2021/02/02/stereo-camera-calibration-and-triangulation.html