Stesias – Stereo Eyes for the Visually Impaired

Mark Andrei Traya
University of San Augustin
mtraya@usa.edu.ph

Jason C. D'Souza*
University of San Augustin
jasondsouza2003@gmail.com

Jose Ryandale D. Gonzaga
University of San Augustin
jrgonzaga@usa.edu.ph

Eury T. Azucena
University of San Augustin
eazucena@usa.edu.ph

Joost Besonia
University of San Augustin
eazucena@usa.edu.ph

Leonard Nathaniel Majaducon
University of San Augustin
eazucena@usa.edu.ph

Abstract

Most technological assistance, such as walking sticks, guide dogs, surgery, or bionic implants, available for a partially or fully blind person cannot completely serve the needs of the visually impaired since they typically focus on one type of visual impairment: mobility or color perception; most are far too expensive and unattainable for many. Therefore, we designed the "Stesias" device, wherein vision-based information is translated into sound clues. The solution can help totally blind individuals in scanning the space around them with echolocation-like properties or partially blind individuals who want to read a text or identify distant objects. Stesias applies several AI techniques adapted for various purposes, combining depth mapping, text recognition, environmental descriptions, and language models.

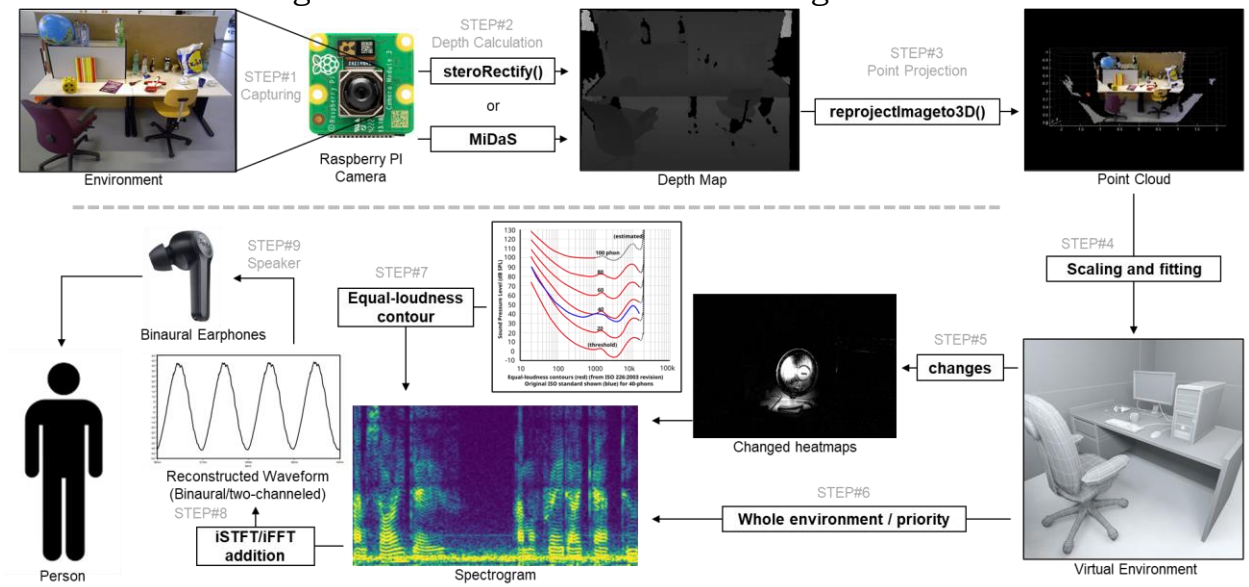
1. Introduction

The Stesias device is based on a microcontroller of the Raspberry Pi type and consists of various input and output devices for processing and forwarding of visual and auditory information. Thus, it enables users to interact with their environment in ways they could not otherwise. The components of the Stesias system are:

No.	Component	Purpose
1	Raspberry PI	Main Microcontroller
1	Microphone	Listens to user request *(optional)*
1	Binaural Earphones	Outputs audio information
1	Lithium Battery	Powers the device
1	Eyeglass foundation	Holds the components
1	Printed PCB Board	Assembles the glasses
1	ESP32 Dev Board	Manages RTSP connection *(for stereo)*
1	Stereo Camera	Captures video streams *(for stereo)*
1	Raspberry PI Camera	Captures video stream *(for one camera)*
1	Oscilloscope	Detects head movements of the user

2. Methodology

The following diagram illustrates the program's data flow, with a primary focus on converting visual information into audio signals:



2.1. Video Capturing

The system uses either:

- The following diagram illustrates the program's data flow, with a primary focus on converting visual information into audio signals:
- A Stereo camera that connects via an ESP32 microcontroller, transmitting video as an RTSP stream to the Raspberry Pi's Access Point

2.2. Depth Calculation

For the program we used C++ with OpenCV has any alternative like python is too slow or doesn't have enough online resources like documentations. If the video is through a Raspberry pi camera then the MiDaS model is downloaded and loaded upon program startup which it then processes each unbuffered(so its realtime) individual frame converting it into a depth map. If its through a stereo camera then each pair of unbuffered frame is compared with the "stereoRectify" method to get the depth map, it is a Horizontal stereo setup therefore the equation for Q(depth) based on C1(Camera1) and C2(Camera2):

$$C_1 = \begin{bmatrix} f & 0 & cx_1 & 0 \\ 0 & f & cy & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \text{ and } C_2 = \begin{bmatrix} f & 0 & cx_2 & T_x f \\ 0 & f & cy & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \text{ thus}$$

$$Q = \begin{bmatrix} 1 & 0 & 0 & -cx_1 \\ 0 & 1 & 0 & -cy \\ 0 & 0 & 0 & f \\ 0 & 0 & -T_x^{-1} & T_x^{-1}(cx_1 - cx_2) \end{bmatrix}$$

2.3. Point Projection

The depth map is then converted to a Point Cloud through using OpenCV's built-in "reprojectimageto3D" method which transform a single-channel disparity map(depth map) to a 3-channel image representing a 3d surface where pixels (x,y) is to (X,Y,Z):

$$\begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix} = Q \begin{bmatrix} x \\ y \\ disparity(x,y) \\ 1 \end{bmatrix}$$

2.4. Scaling and Fitting

Using the previous Point cloud instance and the change in the Oscilloscope's measurement, the Point cloud for that instance can be properly mapped into the virtual environment. One problem is that the depth map especially if it's calculated by an AI algorithm like MiDaS should not be Normalized but instead relative to the captured image. To fix this we can assume that the distance represented in the depth map has a coefficient times the actual distance, to find this coefficient is through using a Root Algorithm, starting from 1 and then increasing or decreasing based on the error margin.

2.5. Changes in Environment

Our senses are more sensitive to the logarithmic changes in our environment that linear or static information this is called the "Web-Fechner law". By applying this logic then the device shouldn't keep on converting the same environment into audio information, but it should convert the changes in the environment or if the user moves their head a certain way. It also This way the user won't be overwhelmed with excess information, and since the volume is proportional to the change relative to distance to the user, it allows the user to deal with focusing on near objects even when in a chaotic environment like a party.

2.6. Environment Priority

The result will be a spectrogram, the frequency, volume, and binaural mix can be configured to whatever the user prefers or most prioritize. Like the frequency represents the color of the area, the volume represents the movements of the area, and the binaural mix will be based on the direction of the area. Another coefficient represents the whole static environment, since moving objects relative to the observer have a higher priority, the rest of environment have their own coefficient

of volume which overtime goes to 0 has the user doesn't need to know the static environment all the time, but if the user prefers to see the environment again the user can rotate their head in a preferred axis of rotation like the X axis for the coefficient to average to 1 so that the user can hear the whole environment. But this time it has a different configuration has the original configuration would be confusing has normally the environment has a lot of colors and the would-be converted audio with be just a pile of noise, to fix this the frequency now represents the Y distance from the user, volume represents the distance from the user, and the binaural mix represents the same thing, the direction of the area.

2.7. Equal-Loudness Contour

Even with the same volume, two different frequencies will sound like they have different volumes. This is because of the "Equal-Loudness Contour", so when constructing the final spectrogram, we would have to consider this effect so that if there are two static objects in front of the user, one up and one down it wouldn't sound like the upper one is nearer.

2.8. Reconstructed Waveform

Now that we have two spectrograms representing the left and right channels of the earphones, we then convert this spectrogram into a live waveform through the Inverse-Short-Time Fourier Transform (ISTFT) using the "fftw3.h" library, and the "portaudio.h" audio library.

2.9. Speaker Output

The earphone or small speaker that is either connected to the raspberry pi through the audio jack or the GPIO pins receive this audio information and then outputs them to the user

3. Future/Possible implementation

3.1. Portable Phone

The raspberry pi is a single board computer, it is therefor also capable of navigating the web and apps. Just with a microphone to receive commands and a speaker to output the results, the user would be then able to navigate the web without touching the device at all.

3.2. Smart assistant

With some custom commands through a microphone, different AI models or APIs can be triggered like Object detection so that the user can also know what is approaching them, Text Recognition so the user can also read what is in front of them, Facial Recognition so the user can identify different people, Google Maps so the user can navigate to their destination, and even a Language Model so that the user can make more specific commands or discuss with the AI.

3.3. Extended eye

Since the visual recognition is captured by a camera, the camera can be switched depending on the desired measurement like temperature through a Thermal camera, or just the environment using a night goggle. This way the user can also have an extended range of vision if desired

3.4. Argumented Reality

This can also act has a smart glass for those that are not visually impaired, and by using like an OLED glass piece or a reflected mirror then images can be projected upon the environment allowing the user to have additional information and just visual like with Google maps or object detection or zooming, etc.

4. Implementation

The Stesias device consists of a modified eyeglass frame, incorporating a printed circuit board (PCB) that holds the Raspberry Pi, battery, and other components. The stereo or Raspberry Pi camera is mounted at the top front of the glasses, while the earphones or speaker/microphone are positioned on each side. Below is the previous concept design of the device:

5. Conclusion

The Stesias device is one of the great wearable technologies by designing it and was meant for helping visually challenged people. It is purely scientific and innovative technology which can convert visual information into audio cues just like a system of echo location. Interaction through environment with this system is dynamic and adaptive, which produces a more inclusive perception of surroundings than traditional ones such as using canes or guide dogs. The system will support various degrees of visual impairment by using artificial intelligence models and a Raspberry Pi, and it enhances the view of the user with depth mapping, prioritized environment, and sound-based feedback.

A multi-purpose facility, Stesias is more than just a simple navigator. Its potential can be extended to being a smart assistant, an extended vision tool, or even an augmented reality device. This flexibility also shows high potential of such technology in the long-term concerning growth and solution of many challenges imposed on visually handicapped individuals. The future improvements may be even more and would be available, friendly, and adaptive to integrate other technologies for the betterment of the lives for those who possess such a disability. Ultimately, Stesias is a promising advancement in assistive technology that aims to better equip users in navigating their environment independently and with confidence.

6. References

6.1. Hartley, R., & Zisserman, A. (2019). *Pinhole camera model*. HediVision. https://hedivision.github.io/Pinhole.html

6.2. *Camera calibration and 3D reconstruction*. OpenCV. (2024). https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html

6.3. Mike. (2004, October 1). *Weber's law and Fechner's law introduction*. Center for Neural Science. https://www.cns.nyu.edu/~msl/courses/0044/handouts/Weber.pdf

6.4. Selesnick, I. (2009, April 14). *Short-time fourier transform and its inverse*. NYU Tandon School of Engineering. https://eeweb.engineering.nyu.edu/iselesni/EL713/STFT/stft_inverse.pdf