Sentri:

Motion and Sound Detection Using OpenCV and Arduino

Authors:

* Vinay Chilakamarri, Student, Manipal Institute of Technology
* Ajitha Chalasani, Student, Manipal Institute of Technology

Abstract:

Sentri aims to develop a system that combines motion detection and sound detection capabilities using OpenCV and Arduino. The system utilizes a webcam or camera for motion sensing and a microphone for sound detection. When motion is detected, an LED light will illuminate, and when sound is detected, a separate LED light will turn on. These two functionalities operate simultaneously and independently.

1. Introduction:

Sentri aims to develop a system that combines motion detection and sound detection capabilities using OpenCV and Arduino. The system utilizes a webcam or camera for motion sensing and a microphone for sound detection. When motion is detected, an LED light will illuminate, and when sound is detected, a separate LED light will turn on. These two functionalities operate simultaneously and independently.

II. Objectives:

1. Implement motion detection using OpenCV and a webcam or camera.

2. Implement sound detection using a microphone and audio processing techniques.

3. Integrate the motion detection and sound detection components with an Arduino board.

4. Control separate LED lights to indicate motion and sound detection events.

III. Technology Stack:

A. Hardware:

- Arduino board

- Webcam or camera for motion detection

- Microphone for sound detection

- Two LED lights for indicating motion and sound detection

B. Software:

- OpenCV for motion detection using the camera

- PyAudio library for audio processing and sound detection

- Python for scripting the logic

- GPIO (General Purpose Input/Output) library to control the Arduino's GPIO pins

C. Development Environment:

- Visual Studio Code for coding

IV. Implementation:

1. Motion Detection:

Motion detection is implemented using OpenCV with the following steps:

1. Pre-processing: The video frames are preprocessed using Gaussian blur to reduce noise.

2. Background Subtraction: To detect motion, the background frame in the first few seconds is captured and compared to the consequent frames to identify regions of motion.

3. Thresholding: Thresholding is applied to the frames captured after the initial seconds to identify regions of motion. Pixels that differ from the threshold value of the background image are marked as objects in motion.

4. Contour Detection: After thresholding, contour detection is used to identify individual moving regions or objects.

5. LED Control: When motion is detected, a designated LED is turned on using the Arduino's GPIO pins.

1. Sound Detection:

The PyAudio library is used to capture audio from the microphone. The implementation involves the following steps:

1. PyAudio Setup: Initialize PyAudio and open a stream to capture audio data from the microphone. Define parameters such as audio format, number of channels, sample rate, and frames per buffer.

2. Sound Detection Algorithm: Detect sound events from the audio data by calculating average amplitude of the audio signal.

3. Thresholding: Define a threshold value based on the ambient noise level. When the average amplitude exceeds this value, it is considered a sound event.

4. LED Control: Upon detecting a sound event, a separate designated LED is turned on using the Arduino's GPIO pins.

V. Integration:

The motion detection and sound detection components are integrated to work simultaneously. The Arduino board receives input from both the webcam (for motion detection) and the microphone (for sound detection). When motion is detected, the corresponding LED is turned on, and when sound is detected, the other LED is turned on. These two functionalities operate independently, allowing both LEDs to be illuminated simultaneously if motion and sound events occur simultaneously.

VI. Conclusion:

Sentri demonstrates the integration of computer vision techniques for motion detection and audio processing for sound detection using OpenCV and Arduino. The system responds to both motion and sound events by illuminating separate LED lights, providing a visual indication of these events. The simultaneous operation of motion detection and sound detection enables versatile applications in various domains, such as home automation, security systems, and interactive installations.

References:

1. OpenCV documentation: https://docs.opencv.org/

2. PyAudio documentation: https://people.csail.mit.edu/hubert/pyaudio/

3. Arduino GPIO library: https://www.arduino.cc/reference/en/language/functions/digital-io/