Real-time Gesture Controlled Robot using Virtual Reality

Rohan Sarkar
Ambujan Sothinathan
Dwarakanath Reddy
Vishveswaran Jothi

I. Summary

Video Link: https://youtu.be/GxsXzXl-2BA

- **Streaming Video from Raspberry PI** The images from the raspberry PI cam are captured and hosted as a webpage in raspberry pi. The images captured through the raspberry pi cam are also rendered as left and right views to allow them to be viewed as virtual reality through the google cardboard. The webpage is updated when a new image is captured; hence simplifying viewing the webpage allowing us to stream the video with minimal delay.
- **VR –Head Orientation tracking Android Application** A custom android application was written which allowed us to view the raspberry pi image stream and sending the head rotation data. The head rotation of the user is calculated by using the rotation vector virtual sensor of the android phone. Rotation vector combines data from the accelerometer and gyroscope to calculate device position. Since the phone has been tilted in landscape mode with the camera facing forward, the co-ordinate system is remapped for the tilted orientation. The Euler angles are calculated with respect to the north pole and these have to be re-oriented for current user orientation. After the user wears the google cardboard headgear, on press of the cardboard trigger, the on touch event handler is called and the application reads the current yaw angle. The head orientation of the user is then calibrated using this captured yaw angle and sent to the robot using TCP/IP socket asynchronously.
- **Leap Motion-Hand Palm Gesture Recognition to control the Speed and Direction of Robot** The Leap Motion 3D Gesture Sensor allows us to use our hands as an universal human interface to interact with the robot in real time. We acquire the following information from the Leap Motion Sensor:

  1. 3D position and normal vector for palms of both hands.
  2. 3D position and direction vectors of all 10 fingertips.
  3. 3D direction vector for both hands.

  Using the above information we wrote a ROS Node that can independently detect each finger of the hand by checking the position of each joint with respect to the other joints of the same finger. Also using the direction vectors and orientation of the hand, we can recognise gestures that can be used to move the car in reverse direction or turn in any direction.
  Using this gesture recognition technique,we have developed the following functionalities to control the speed and direction of the robot:

  1. The linear velocity of the robot is proportional to the finger count.
  2. The angular velocity of the robot is proportional to the tilt angle (orientation of the hand about the Z axis of the sensor).
  3. The robot stops if we close our hand.
  4. The robot moves in the reverse direction if I orient my palm in the reverse direction.

II. Achievements

- **Video Streaming from camera with minimal delay:** Real time streaming of video to the android phone had been one of the bigger challenges we had faced in this project. To use the google cardboard sdk(software development kit) the input to the android application has to be a video of mp4 format. But using any available video compression techniques for a stream of raspberry pi images caused substantial delay. For example using raspivid coupled vlc player for mp4 encoding created a delay of 8 seconds. Hence to achieve real time streaming of video, the images are hosted as webpage that updates at the rate of image capture (25 frames per second). However google cardboard sdk cannot be used with this webpage and new android application had to be written with webview class to view this webpage. This method avoids the delays caused by the compression and decompression allowing us to stream the raspberry pi camera feed in real-time.
- **New and accurate method for Hand Gesture recognition:** Several approaches were tried to find the best and most accurate way to detect hand palm gestures, described as follows:
  1. **Human Skeleton Tracking:**The human skeleton was tracked using the Kinect sensor and the motion of the arms was used to control the robot.Limitations include inaccuracy and data loss due to occlusion,non-linear output, dependency on the camera's position.
  2. **Computer Vision using Convex Hull approach:** This method is much more accurate than the skeletal tracking method but some major limitations are that using this method we cannot accurately determine the number of fingers, if there is only one finger or if all the fingers are joined together. Also it depends on the background and so proper thresholding to eliminate irrelevant background information is required.
  3. **Leap Motion:** We can get the accurate joint and palm positions and orientations and takes care of the previous limitations as it can can detect each finger seperately in any configuration, even if the fingers are joined. We can detect each part of the hand independently with great accuracy, thereby resolving the limitations of other methods.
- **Controlling a Robot in Real Time using Multi-Sensor Data Fusion:** The data from different sensors like the gyroscope and accelerometer from the mobile device that gives information regarding the head orientation and the data from the leap motion that gives us the linear and angular velocity data are being used to generate the control inputs for the robot. The position of the robot at any instant is given by the odometer of the robot which is used as a feedback to generate the error signal and design the control parameters for driving the robot.We interfaced with various devices/sensors asynchronously to avoid delays in communication ,thereby making the system real-time.This is explained in the diagram below.