What Sensors Are Best for Self-Driving Cars?

https://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p043/robotics/autonomous-car-sensors (https://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p043/robotics/autonomous-car-sensors)

Procedure PDF Date: 2023-10-25

Experimental Procedure

1. If you have never used an Arduino before, see the How to Use an Arduino (http://www.sciencebuddies.org/science-fair-projects/references/how-to-use-an-arduino) reference in the Bibliography.
2. Prepare two data tables like Table 1, one for the ultrasonic sensor and one for the infrared sensor. Plan out what objects you will test and how many trials you will conduct. Test each object at a few different distances from the sensor. Check that the distances you plan to test are within the sensor's operating range according to its datasheet. Make sure you measure the same objects/distances with each sensor.

		Sensor:		
Object	**Object description (size/shape/texture etc.)**	**Actual distance (measured with tape measure, cm)**	**Sensor distance (measured with Arduino, cm)**	**Error (cm)**

Table 1. Example data table. Make one table for each sensor.

3. Set up your experiment as shown in Figure 2. Tape the ultrasonic sensor to the top of a cup or other small object to prop it up off the floor or table. This will prevent interference from sound waves reflecting off the table. Aim the sensor along a tape measure so you can use it to measure the distance between the sensor and the target object. Make sure you measure the distance to the part of the object directly in front of the sensor, which is not necessarily the closest part. In Figure 2, that would be the rabbit's stomach, not its feet.

Figure 2. Example experimental setup.

4. To use the ultrasonic sensor:
    a. Connect the sensor to your Arduino using the male-female jumper wires. Connect the GND pin to GND, 5V pin to 5V, the Echo pin to Arduino pin 7, and the Trig pin to Arduino pin 8.
    b. To open code for the ultrasonic sensor, open the Arduino IDE and select File→Examples→06.Sensors→Ping.
    c. The example code is for the PING ultrasonic distance sensor, which uses a single pin (alternating between output and input) for the trigger and echo signals. If you purchased the Science Buddies kit and have an HC-SR04 ultrasonic sensor, you need to modify the code as shown in the following video.
    d. Upload the code to your Arduino and open the serial monitor (Tools→Serial Monitor). The monitor will show the distance measured by the sensor in both inches and centimeters.

    https://www.youtube.com/watch?v=n-gJ00GTsNg (https://www.youtube.com/watch?v=n-gJ00GTsNg)
5. Collect data for different objects using your ultrasonic sensor and fill out your data table. To take a reading, place an object in front of the sensor. Measure the actual distance between the sensor and the front of the object using the tape measure. Get the electronic sensor reading from the serial monitor. You should do multiple trials for each distance.
6. To use the infrared sensor:
    a. Connect the sensor to your Arduino using the JST connector. Connect the red wire to 5V, the black wire to GND, and the white wire to pin A5.
    b. Example code for the infrared sensor (https://github.com/sparkfun/simple_sketches/blob/master/sharp/sharp.ino) is available on Github. Download the code and save it locally on your computer.
    c. Download the code to your Arduino and open the serial monitor. The monitor will show the distance measured by the sensor in centimeters.
7. Collect data for different objects using your infrared sensor and fill out your data table. Use the same process that you used for the ultrasonic sensor.
8. Analyze your data.
    a. Calculate an error for each reading by subtracting the actual measurement (with the tape measure) from the electronic measurement.
    b. Calculate an average error for each distance you tested.
    c. Compare the errors for the two sensors when measuring different objects. Does one sensor work better for certain types of objects? What causes a sensor to give an inaccurate reading? Can you explain your observations based on your understanding of how the sensors work?
    d. Compare results between the sensors at different distances for the same objects. Is one sensor better at accurately detecting objects at a certain range?
    e. If you were building your own autonomous vehicle using an Arduino (for example, something that would drive around the floor of your house), which sensor would you choose and why? Is cost a factor in your decision (check the links in the materials section for the price of each sensor)? What other factors might influence your decision?