Caleb Martinez

Robotic RFID scanner

Low Costing Computer Vision

Introduction

Robotic navigation is the way the robot will change its position without hitting obstacles. Automated ground vehicles navigation takes advantage of the visual information provided by pre-existing landmarks [2]. Many Robotic navigation systems utilize vision sensor data to navigate and to map out data entry points. A type of navigational system collects environmental data when navigating and builds a local database to support navigation [3]. This paper will review the aspects of using low-cost, real-time vision systems with a microcontroller to be used with a navigational system.

Commercial Pixy 2 and Google Vision Kit

A real-time navigation sensor embedded with a microcontroller are used in various navigational robots. Pixy 2 ($59.99-$69.99) is a computer vision color recognition system which pairs a processor and image sensor to output data information about learned-detected pixel blocks. The information is available through UART serial, SPI, I2C, and analog out [1]. With the rate of 50 FPS (frames per second) and an open source software integration with Raspberry Pi, Arduino, LEGO Minstorms, and roboRIO, the Pixy2 camera has been implemented with a variety of projects [1]. Some of the Pixy2 projects include “Arbotix Arm Sorter With Conveyor”, “Nellie, the Weed Picking Robot”, “Air Hockey Robot”, and “Automated Rubik’s Cube solving robot” [7].

Another low powered, mobile computer vision is the Google Vision Kit ($89.99–$99.99). The Vision Kit includes a VisionBonnet that attaches to the Raspberry Pi GPIO pins, and the VisionBonnet includes an Intel Movidius MA2450 VPU (vision processing unit) that can process up to 30 FPS [4]. The Vision Kit uses “Edge AI” that runs the vision models completely on the device [5]. The kit processes an image and returns objects that were identified and a confidence score to go along with the object [6].

Technology Low-Cost Computer Vision

The simpler real-time image processor uses a hue color filtering algorithm to detected colors set by user input. The camera image gets processed by each pixel's color dimension and then gets compared to color signature desired with levels of tolerance [1]. With filtering and growing signature algorithm, the false detects are minimized in real-time. The Pixy also offers a line tracking program that predicts the best vector candidate to follow [1]. At rate of 60 FPS, the Pixy tracks the line frame to frame while also detecting command barcodes to the left or right of the line being tracked [1].

The more advanced computer vision system uses a model trained from a dataset to process the cameras incoming data [5]. A model is trained from a complex program with a huge dataset of images. The dataset of images entails one object at a variety of angles and lighting.

Both processes use software to provide detailed information on the current image being processed. Also, both processes use a machine learning algorithm that continues to learn the object desired to detect

Implementing the Pixy and Google Vision

The Pixy 2 has a variety of API calls by sending packets through serial protocol [1]. The Pixy 2 consist of two programs, color connected components and line tracking [1]. The Pixy requires 140mA with 5V through the USB, I/O connector, or through the unregulated power connector [1]. Furthermore, the Pixy offers tilt control with LED lights place near the camera [1]. By using the Pixy 2 data output, the information can be used to calculate the distance of detected objects using the Pythagoras equation [2].

The Google Vision Kit includes Raspberry Pi camera, Vision Bonnet, and Raspberry pi Zero WH. Including a one and half hour's time required to build [6], the kit entails a setup guide that will image your Raspberry Pi [6]. To power the kit, you power the Raspberry Pi at 2.1A with 5V.

Both the Pixy 2 and the Google Vision Kit have processor chips to process the current image of the camera. Both use a serial port connection to be embedded with a microcontroller. Pixy and Google offer open source software to be easily deployed on a Raspberry Pi. Pixy also offers Arduino, LEGO Minstorms, and roboRIO software to be deployed and used with the Pixy hardware. The open source code can be implemented in programs outside of the given code [6] & [1].

References

[1] Charmed Labs, “PIXY Documentation,” Charmed Labs, Pixycam, 2018. [Online]. Available: https://docs.pixycam.com/wiki/doku.php?id=start [Accessed: June 14, 2019].

[2] B. Rahmani, A. Harjoko, T. K. Priyambodo, and H. Aprilianto “Research of smart real-time robot navigation system,” AIP Conference Proceedings 1707, 2016. [Abstract]. Available: Scitation, https://doi.org/10.1063/1.4940854 [Accessed June 12, 2019].

[3] A. Aziza and O. Arafa, “Microcontroller-based mobile robot positioning and obstacle avoidance,” Journal of Electrical Systems and Information Technology, vol. 1, no. 1, pp 58-71, May, 2014. [Online serial]. Available: https://www.sciencedirect.com/science/article/pii/S2314717214000105 [Accessed Jun. 11, 2019].

[4] B. Rutledge, “Introducing the AIY Vision Kit: Add computer vision to your maker projects,” Google Developers, Nov. 30, 2017. [Online]. Available: Google Developers, https://developers.googleblog.com/2017/11/introducing-aiy-vision-kit-add-computer.html [Accessed June 14, 2019].

[5] C. Hart, “Computer Vision Training, the AIY Vision Kit, and Cats,” cogint.ai, June. 05, 2018. [Online]. Available: https://cogint.ai/custom-vision-training-on-the-aiy-vision-kit/. [Accessed June 15, 2019].

[6] Google, “Vision Kit,” Google, Aiy Projects, 2018. [Online]. Available: https://aiyprojects.withgoogle.com/vision#project-overview [Accessed: June 14, 2019].

[7] Charmed Labs, “Pixy Project Gallery,” Charmed Labs, Pixycam, 2018 [Online]. Available: https://pixycam.com/projects/ [Accessed June 16, 2019]