



BrickPi Candy Sorter – Sort Halloween Candy with Google Cloud Vision and the Raspb

PROJECTS » BRICKPI CANDY SORTER – SORT HALLOWEEN CANDY WITH GOOGLE CLOUD VISION AND THE RASPBERRY PI



Candy Sorter
Using Google Vision and the Pi

Google Cloud Platform





BUY 2 GET 1 FREE
UNTIL FRIDAY DEC. 1
[BUY NOW](#)



Build & code your own robot car.


U.K./EU Local Shipping
[Click Here](#)

BUILD A ROBOT
OUR TOP PRODUCTS




TOP RATED PRODUCTS


Breadboard Adapter for Lego Mindstorms
~~\$45.00~~ **\$7.50**



Mars Rover Unit for GoPiGo (Single Classroom License)
\$500.00



Distance Sensor
\$19.99



PROJECTS BY ROBOT

- GoPiGo ☐
- GrovePi ☐

In this tutorial, we're going to build a machine to sort your Halloween candy, using the Raspberry Pi, the BrickPi, some LEGOs, and Google Cloud Vision.

Pile of candy too big this year?

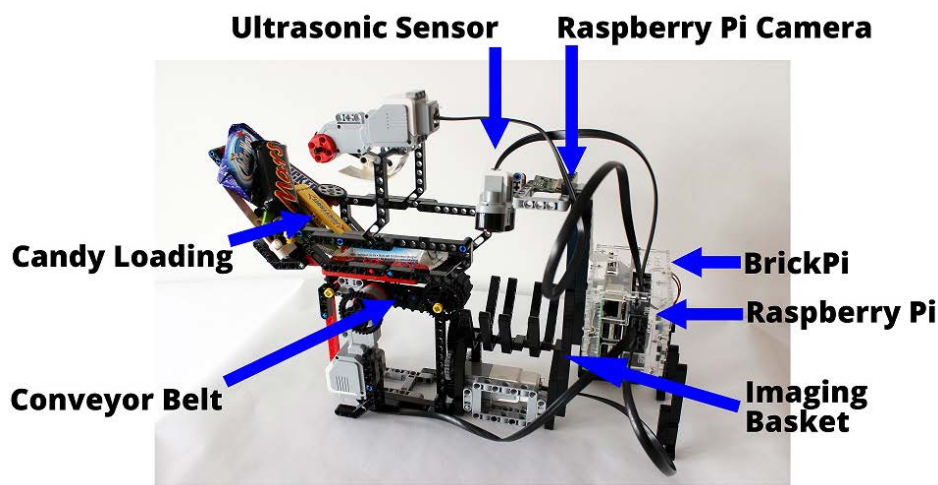
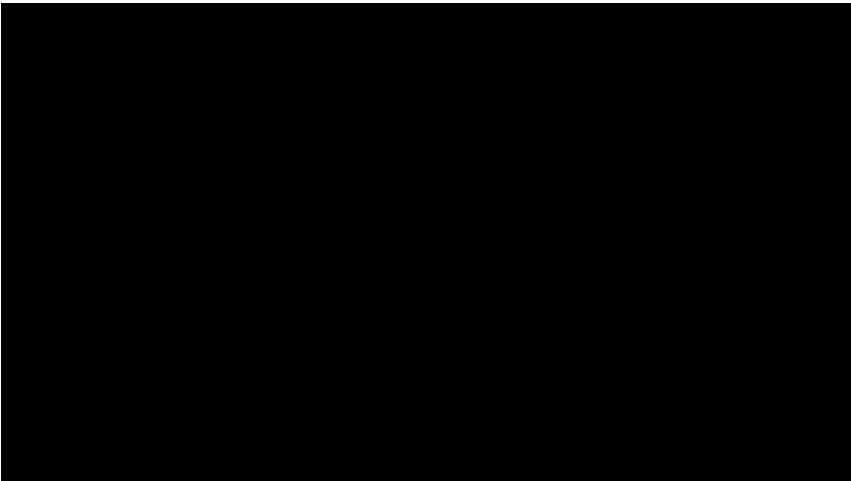
Need to sort the junk from the gold? Have a certain candy you hate? Fear not, the BrickPi Candy Sorter is here to help! In this tutorial we'll harness the powerful Google Cloud Vision, the BrickPi, the Raspberry Pi, the Pi Camera, and a pile of LEGO bricks to build a machine to find what you want (and don't want) automatically! After you run the BrickPi Candy Sorter, you'll have a pile of candy you want, and the junk you can give your little brother.

"Robots are coming for Dirty, Dangerous, and



Dull Jobs"; sorting candy isn't dangerous or dirty (hope not!), but it is kind of dull. The BrickPi Candy Sorter using Google Cloud Vision solves this problem!

A Quick Overview



BrickPi Candy Sorter

First, we'll make a python list of the candy we like and a list of the candy we don't like. After we fire up the python program, you're going to dump your unsorted candy into the machine. The individual candy bars will be sorted onto a conveyor belt. The conveyor belt dumps the piece of candy into the imaging basket.



Once they're dumped into the imaging basket, the ultrasonic sensor is tripped and the BrickPi turns off the conveyor belt. In the imaging basket, the Raspberry Pi camera will take a picture of the candy bar and upload it to the Google Cloud Vision, asking for a brand label.

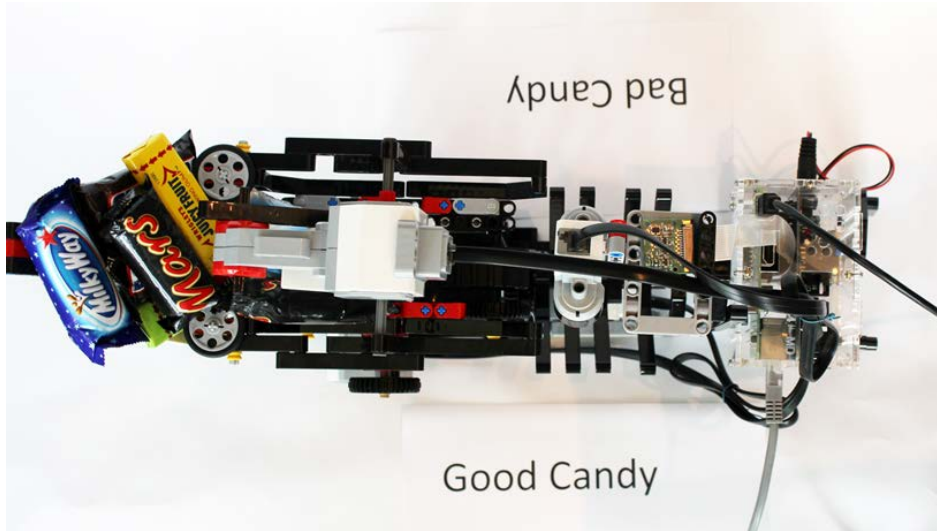
BrickPi ☐

CATEGORIES

- BrickPi
- C.H.I.P.
- GoPiGo
- GrovePi
- PivotPi
- Python
- Scratch



Google Cloud Vision will search the image and return its best guess for what the brand is of the object (candy) in the picture.



Finally we'll compare the value returned by Google Cloud Vision to a list of candy we want to keep, and a list we want to give to our brother. The candy sorter will dump in one direction if it's a keeper, and another if it's a reject! As a final touch, the speaker will yell out loud whether it was Good Candy or Bad Candy, and what brand candy it found.

Tools and Materials

- **BrickPi** – We'll use the BrickPi to control the LEGO motors and read the EV3 Ultrasonic Sensor.
- **Raspberry Pi** – The Pi will be doing the processing, taking pictures, and talking to Google Cloud Vision.
- **Raspberry Pi Camera** – The Pi Camera will take a picture of the candy.
- **Raspberry Pi Power Supply** – We'll power the Pi with this, and we'll also use the BrickPi Power Pack that comes with the BrickPi Base Kit.
- **Ethernet Cable** – You will need your machine to be connected to the internet. If you want to do this over wifi, that's fine too!
- **Raspbian for Robots SD Card** – The software that runs the Raspberry Pi. This comes with most of the software needed for this tutorial. You can also download the software for free.
- **LEGO Mindstorms EV3 Kit (31313)** – You'll need a pile of LEGO and two large motors, and one servo motor, and the Ultrasonic Sensor.
- **Raspberry Pi Speaker** – Can be anything that makes noise! This robot talks!

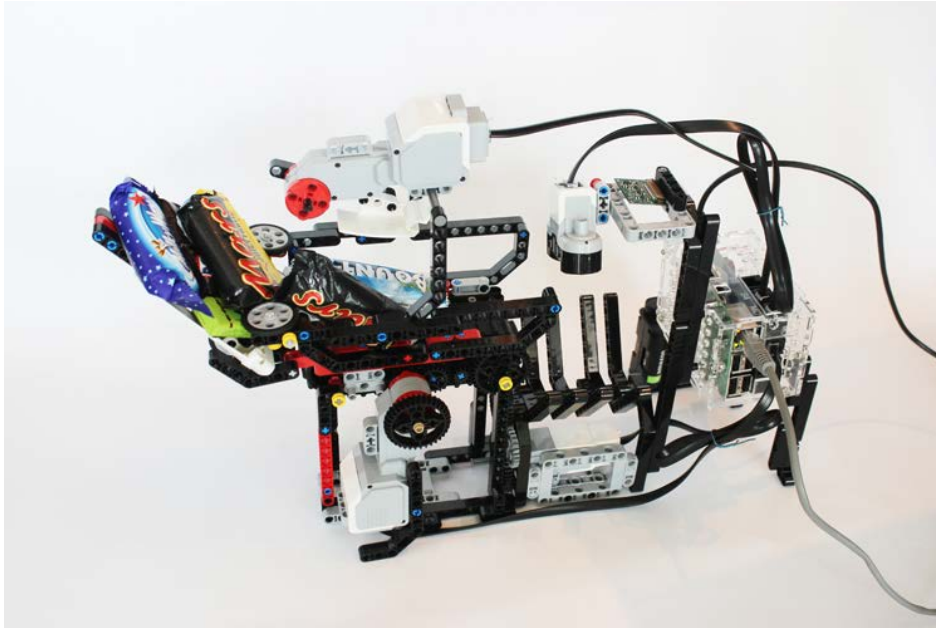


You will also need a Google Cloud account. Fear not, we have instructions here on how to set up a Google Cloud Account, and the first 60 days are free! You can use your existing Google or Gmail

account to set it up, and it takes about 10 minutes to set up.

Build the Google Cloud Vision and the Raspberry Pi Camera Sorting Machine

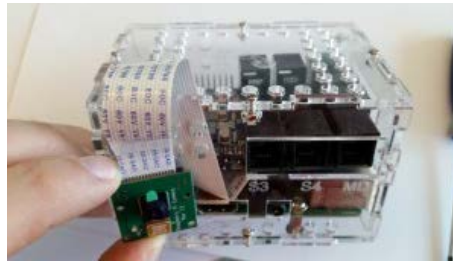
Coming soon: LDRAW Diagrams.



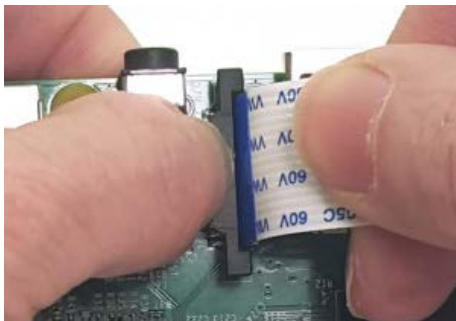
Wire Up the Raspberry Pi

To finish the project, you'll need to connect a few things:

- The Conveyor Belt Motor goes into Motor Port B.
- The Imaging Basket Servo connects to Motor Port C.
- The Rake (the motor on top) connects to Motor Port D.
- The Raspberry Pi Power Supply connects to the Micro USB Port of the Raspberry Pi.
- The BrickPi Power Pack connects to the power port of the BrickPi.



Remember to connect the Camera! We'll be using a Raspberry Pi Camera with the Raspberry Pi, here are some directions on setting up the camera.



These Raspberry Pi Camera set up directions assume you're using Raspbian for Robots on your SD Card. You can buy an SD Card with Raspbian for Robots on it, or you can download it for free and install it using our directions [here](#). If you're not using Raspbian for Robots, you will need to take additional steps to enable and set up your camera; here are some directions on setting up the camera.

Prepare the Raspberry Pi Software

The example python program is located on our Github Repo here. You should also check out our Github Repo on using the Raspberry Pi Camera with Google Cloud Vision here.

Since we're using Raspbian for Robots, there are only a few changes that need to be made to the software. All of them are specific to using Google Cloud Vision on the Raspberry Pi. We have a very thorough tutorial on how to get set up here. In short you'll need to run the following in the command line:

```
sudo pip install --upgrade pip
sudo apt-get install libjpeg8-dev
sudo pip install --upgrade google-api-python-client
sudo pip install --upgrade Pillow
sudo apt-get install python-picamera, espeak
```

Finally, you'll need to upload your Google Application Credentials in JSON to the Raspberry Pi.

Use an FTP program (such as FileZilla) or for Windows users Samba (see our tutorial here!) to move the JSON file over to your Raspberry Pi. Place the JSON file in the home directory ("home/pi").

Next, we'll turn on Super User and make our credentials available. In the command line, type the command "su":

```
su
```

You may need to run

```
sudo su
```

and you may need to enter in your password. Next, we'll make the JSON credentials available to Python. In the command line, run the following:

```
export GOOGLE_APPLICATION_CREDENTIALS=filename.json
```

Be sure to substitute your JSON filename in this command with the name of the file you have on your Raspberry Pi.

Finally, we'll download the code. Again, this should go into the home directory:

```
cd /home/pi
```

```
sudo
wget https://github.com/DexterInd/BrickPi_Python/raw/master/Project_Examples/candysorter/candy.py
```

List your Gold (or Trash)

Opening up the python program

```
sudo nano candy.py
```

we'll see up on lines 30 and 31, we have two lists: one of good candy ("good_candy") and one of bad candy going to our little brother Steve ("bad_candy").



```
good_candy = ['Mars', 'Milky Way', 'Bounty']
bad_candy = ['Juicy Fruit', 'Doublemint', 'Snickers']
```

Add all the candy names you like. Note that the code will default to "bad" so if it's not on the list, it's not going into the good pile! In my example, I am not really interested in gum, and I hate Snickers.

On the other hand, I love Bounty, Mars, and Milky Way, which are all in the Good Candy pile!

Run the Program

Make sure you're still in super user mode. If you see the word 'root' at the beginning of the prompt, then you're already in superuser mode. If you don't see 'root' run in the command line:

```
su

python candy.py
```

And the response should look something like this:

```
Last Candy Distance: 57
Milky Way
Candy Detected: Milky Way
Good Candy!
  espeak -ven+m7 -a 200 -s180 -k20 --stdout 'Good Candy!' | aplay
Playing WAVE 'stdin' : Signed 16 bit Little Endian, Rate 22050 Hz, Mono
  espeak -ven+m7 -a 200 -s180 -k20 --stdout 'Milky Way' | aplay
Playing WAVE 'stdin' : Signed 16 bit Little Endian, Rate 22050 Hz, Mono
Dump the good candy!
```

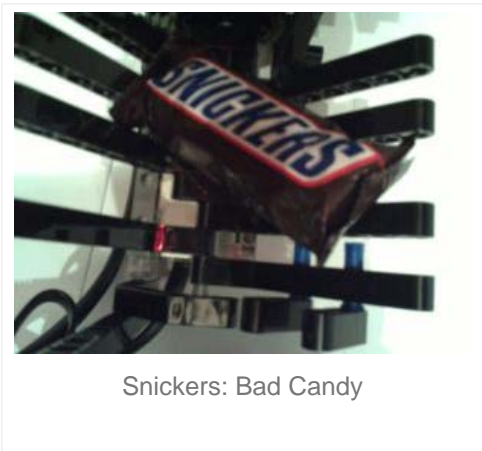
And if we examine the photo from the basket:



Indeed, it was a Milky Way, my favorite!

More Candy?

Aside from the cute tricks with LEGO, some awesome computing power is going on. In this project, we've harnessed some high end computing power to do a simple task: sort your candy. We're using a lot of computing power: a Raspberry Pi linked to the Cloud and accessing Google Vision – that's a whole lot of computers for sorting candy! Humans are still in charge of the robots and candy-sorting is serious business when you're two years old!



Snickers: Bad Candy



Mars – Good Candy!



Milky Way: Good Candy!

While the task may seem simple, that's kind of the point. This tutorial demonstrates a really simple but boring task that can be automated with Google Cloud Vision. Making this technology accessible is important, and combining Google Cloud Vision and the Raspberry Pi Camera does that!

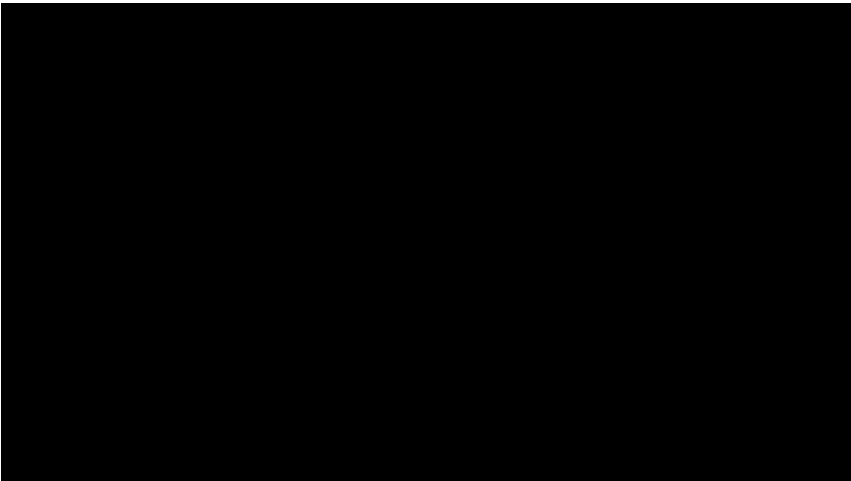
The robots are coming, and they'll probably be running Google Cloud Vision!

More Questions?

Ask on our forums [here](#)!

Bonus

Watch a quick video of my 2 year old son explaining the machine [here](#).



DEXTER INDUSTRIES

Home

About

Shop

Blog

Manuals

Forum

How To

Contact


Distributors

Shipping & Ordering

PRODUCTS


GoPiGo Robot Base Kit

\$99.99




GoPiGo Beginner Starter Kit

\$199.99




GoPiGo Beginner Classroom Kit

\$989.99



GoBox USA - Robot Subscription

\$15.99 on the 15th of each month for 11 months and a \$179.99 sign-up fee



EMAIL SIGNUP

Sign up to our amazing email newsletter, get all the latest Dexter offers and robot news, delivered straight to your inbox.

© 2017 Dexter Industries - Built by Nav.