



Highly Commended

Programming, Apps & Robotics Year 3-4

Olivia Chong
Millicent Guerin

Vale Park Primary School



The Bad Bin Fixer

Vale Park Primary School

Code writers: Olivia Chong, Millie Guerin

Category: Programming, Apps & Robotics, Year Level 3-4

Aim

The aim of this Robotic Project is to build a robotic device that will make sure bins have the right things in them.

VIDEO DEMONSTRATION: <https://youtu.be/JZ7RGCWdFa0>

Scientific Purpose

This device is The Bad Bin Fixer and its purpose is to make sure the bins have got the right things in them because sometimes people are confused, or sometimes too lazy, to think about which bin they should use.

We know this is a problem because we have been seeing lots of rubbish in the recycling bin and we are trying to fix it.

How the Bad Bin Fixer works

We have used the Micro:bit program to do the coding.

Our coding for the Bad Bin Fixer works with a Huskylens AI camera to give feedback that sorts rubbish into the correct bin.

We have taught the program to learn different identities (IDs) that match different types of rubbish. ID1 is "bottles". When ID1 is scanned, the program will show a tick and make a tune telling you the item is a bottle and it can go in the 10c recycling bin. If anything is scanned and is *not* ID1, the Bad Bin Fixer will show a cross and make an error sound, telling you it is not a bottle and it needs to go into a different bin.

What this means

From now on we will be hoping to see recycling go into the right bin. The Bad Bin Fixer will make it easier for kids to know if an item can be recycled. Also, if the amount of 10c refundable items is increased, our school will raise more money. Overall we think that kids will also be really interested in using the Bad Bin Fixer because we have programmed a cute music sound to make it fun.

Potential applications and next steps

Our next step is to build something that can hit the item into the right bin. We have a motor and some parts and we are working on an extension so it does the job.

After that we want it to recognise more things, to also tell you if things should go into the other types of bin - rubbish for landfill and the food scraps in the green bin. The Micro:bit program and Huskylens knew automatically what a bottle looked like, and with more time we can train the program to recognise other types of rubbish. We would start with the most common green rubbish we see at school like apple cores and banana peels. Plastic wrappers are also a problem at school, and we'd definitely need to train the program for that.

Eventually we would like to bring this device into our school or into parks, and we expect the government to pay us a licence to use it in these public spaces. We understand this is how scientific inventions make an impact and profit so businesses are interested and will invest money.

We are wishing that this invention will make life easier for people who sort out the rubbish that is put in the recycling bin and not in the landfill or green bin. Now The Bad Bin Fixer is here, we are keeping the faith that our world is getting healthier because we have the correct things in the bins.

Project Materials

- Any normal laptop or chromebook
- Huskylens AI camera
- Micro:bit program
- A place to set up the camera and computer with a suitable background like a black cardboard sheet (see instruction below for more information)
- Examples of rubbish to train and test the program

Instructions for use

- Turn on the camera
- Turn on the computer and run the Micro:Bit program
- Set up a background so the camera can scan without other items being accidentally scanned or picked up by the camera. This could confuse the camera and program. We use a black cardboard sheet for this.
- Test the rubbish item by placing it between the background and the camera.
 - if we had a bottle and put it in front of the camera it would light up a tick and play a melody and it would hit it to the correct side
 - If we put something else in front of the camera with no bottle it will do the same thing but it will show a cross.

Project Method

- Draw a graph about what you are going to do
- Write the code
- Enter it into the program
- Get the correct Micro:bits
- Transfer the code to the Micro:bit
- Put the Micro:bits together
- Train the Huskylens AI camera to recognise the object you want to it to transfer.
- Make or buy a cover just incase it will rain (optional)

Bibliography

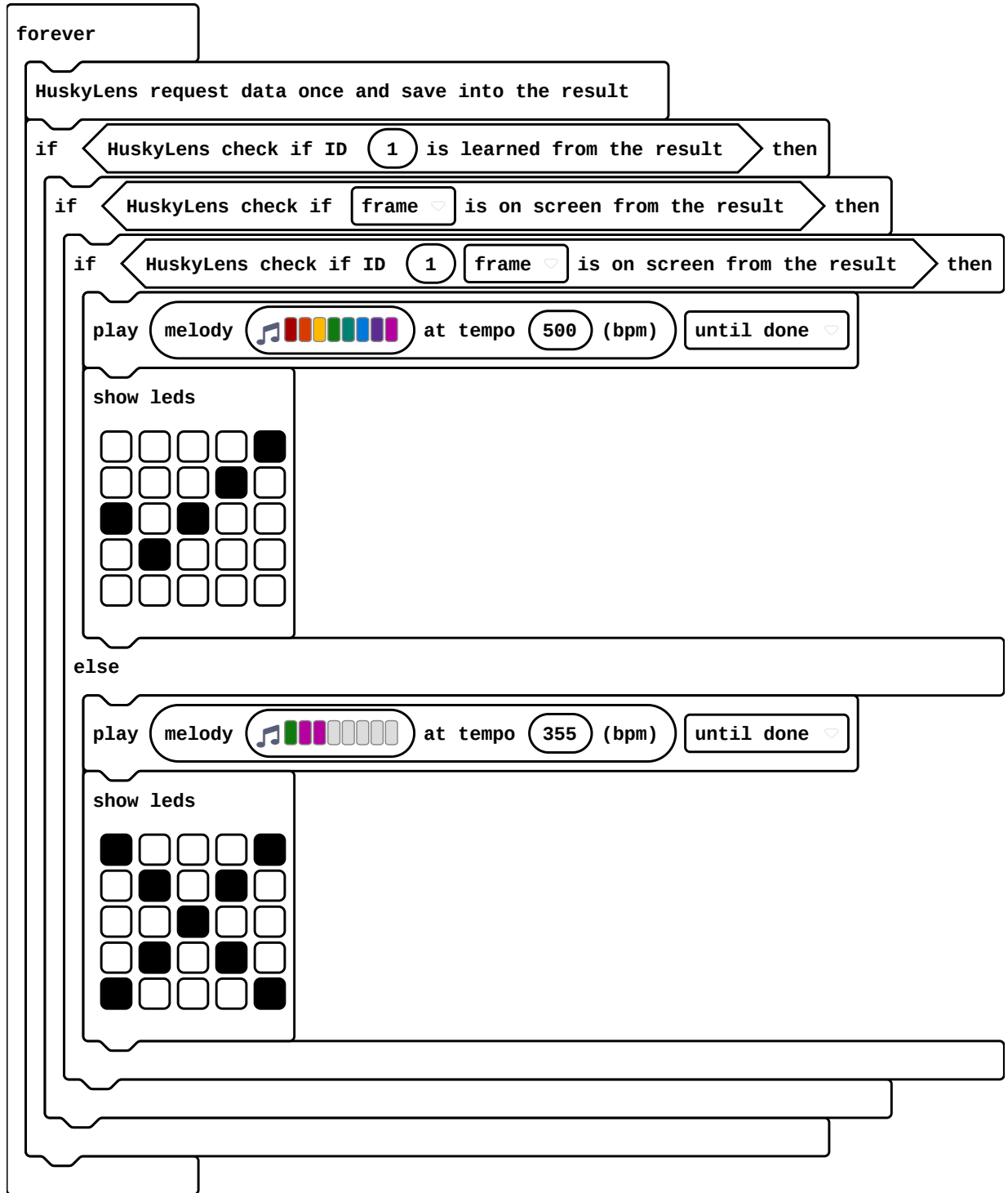
- Huskylens AI Instructions for use (part of the camera set)
- <https://makecode.microbit.org/#editor>

Acknowledgement of external support:

- Miss Hoskin
- Mr Hern
- Our parents

bad bin fixer

```
on start
  HuskyLens initialize I2C until success
  HuskyLens switch algorithm to Object Recognition
```



Extensions

- radio, *
- microphone, *
- Huskylens, [github:dfrobot/pxt-dfrobot_huskylens#v2.0.6](https://github.com/dfrobot/pxt-dfrobot_huskylens#v2.0.6)

```

huskylens.initI2c()
huskylens.initMode(protocolAlgorithm.ALGORITHM_OBJECT_RECOGNITION)
basic.forever(function () {
  huskylens.request()
  if (huskylens.isLearned(1)) {
    if (huskylens.isAppear_s(HUSKYLENSResultType_t.HUSKYLENSResultBlock)) {
      if (huskylens.isAppear(1, HUSKYLENSResultType_t.HUSKYLENSResultBlock)) {
        music.play(music.stringPlayable("C5 B A G F E D C ", 500), music.PlaybackMode.UntilDone)
        basic.showLeds(`
          . . . . #
          . . . # .
          # . # . .
          . # . . .
          . . . . .
          `)
      } else {
        music.play(music.stringPlayable("G C C - - - - - ", 355), music.PlaybackMode.UntilDone)
        basic.showLeds(`
          # . . . #
          . # . # .
          . . # . .
          . # . # .
          # . . . #
          `)
      }
    }
  }
})

```