# **Image Classification Game**

This Snap! game uses Nvidia Jetson capability to classify images.

# **Offline Snap! downloading**

Please download and open Offline version of Snap! for our project. Go to https://snap.berkeley.edu/offline and follow the steps.

# Snap! files' downloading

Please open the link Classification Game to download our project on your computer. Probably you would see the xml in raw format. Click the right button of your mouse and save it on the disk.



# Web camera Image in Snap!

You can get picture from your web camera in Snap!.

- video capture block to enable video capturing.
- Change value of set video transparency block to 0 for clear image.
  set video transparency to 0
  video snap on stage block reports picture from stage
- video snap on stage block reports picture from stage.

#### video snap v on Stage v

# **Connection to Jetson from Snap!**

If you have not imported it yet, please download jetson blocks and import it to your Snap! project.



Enable Javascript Extensions for



• Use connect to Jetson url block to connect Jetson.

connect to jetson url ws://ip\_address:4040



You need **ip address** of Nvidia Jetson. You can use *ifconfig* command in a terminal to get **ip address**.

• Store the value of jetson\_name in a variable.



### **Response from classification**

Here we will send **video snap on stage** to Jetson for processing. Jetson will respond back class name, confidence value and class ID.



- Use get response from Jetson block to send image , and get class name and confidence value.
  - $\circ\,$  First input slot is for jetson variable that stores websocket data.
  - $\circ\,$  Second input slot is for  ${\color{black} costume}$  you want to be classified by Nvidia Jetson.



#### **Class name and confidence value**

This section will demonstrate how to handle **response** variable to access **class name** and **confidence value.** 

• class name is the 2nd item of 1st item of response block.



• confidence value is 2nd item of 2nd item of response block.

round 100 × item 2 → of item 2 → of response



Multiply **confidence** value by 100 to get percentage of **confidence**.

You can create custom blocks, to get **class name** get class name from response and to get **confidence** get confidence from response .

#### **Speech functionality**

**Speech functionality** is available as a library in Snap!. Select *export libraries* from settings then choose *speech module* .

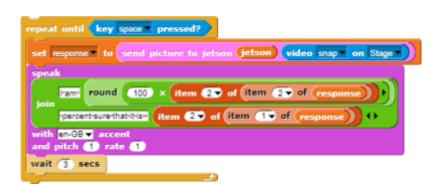
• Use join block to create text like I am confidence percent sure, it is class name .



#### **Repeat block for game**

Last step is adding loop for the game.

• Use repeat block and put script inside of it.



Last update: 2022/08/06 snapcon2022:image-classification-game https://wiki.eolab.de/doku.php?id=snapcon2022:image-classification-game&rev=1659787650 14:07



This example used **repeat until** block to break loop when *space* key pressed. You can download full game from Github page of EOLab-HSRW.

From: https://wiki.eolab.de/ - HSRW EOLab Wiki

Permanent link: https://wiki.eolab.de/doku.php?id=snapcon2022:image-classification-game&rev=1659787650

Last update: 2022/08/06 14:07

