# Image Classification Game: Part 1

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.

		🕒 https://www.githubusencontent.com/EOL	ab H	RMSnapcon workshop/main/snap/	inepton_mit_finel_base.x	nd - [hiPrivate]						
େ ଲ	🗇 topo/raw.gittub	usercontent.com/tOLim-HSRW/Snapton-wor								InPrivate		
cthumbna[]> steps:lklCom w00101/NIAt huffyd:Sturf s i rom 180v/m wktobiadki gyggladdiadk /7vtfspcTBuA uvditv7dcy2E	data (ange/jng)boséd, 20 edengta/fg0hoti 316 e 2003 a Societenty (Metsburg) (Mr. 2) KRTed (2019/Mittae 2) (SML 4) 40 Pří (2019/Mittae 2) (SML 4) 40 Pří (2019/Mittae 2) (Mittae 2) Mathia Mathi	apt 7, https://sep.berkeley.nds*versi Wohengedaanstokeugaanstaanstokeugaanstaanstokeugaanstaanstokeugaanstaanstaanstaanstaanstaanstaanstaans	Wied MR21 M2P1 M2P1 M2P1 M2P1 M2P1 M2P1 M2P1 M2	AAAAANSERICAY sac GQAAE 2V ) BEFU a 3MaaAQF / KYTENg2gC cpr (xilo BDAA LaggBu TDT 3   Stag2 gG / QopABAV3 j W 1AE TOQ: 4155081148 AAAANSEC109 I cos 4 Lage sachad (? 3150877 4400477 K 9057   1644 y 4987 gSOb Iane SEAC a 2	GANGENICayla+YNGgCCS CaP/AcGyk1r14f7a40c Esvleffedhg14a18014f1 Er63MLookso1/2140vf E16915304waa7E5sAol	PG1+G+bG071wA2mlm OBpsahe LoRG (HoA (S ST LoQtb1Ap467mg27 N970545g4 (pL402wA V/NS3+mp31Kywsaff	nyqilli di Yvan Si Dolla di Yvan di Dolla di Dolla di Ulpita di di Dolla	EVSADNe cyline3L NOSD-Phy AdPig1N4 LCS42TQN LCS42TQN LCS42TQN LCS42TQN LCS42TQN LCS42TQN LCS42TQN LCS42TQN	ks2t1Mu rungerr Lex0M80 1141xW Ap61/41X Sekas110 y100400 01xc000	LittfTCFhty 2928h/YRM g7buYt/cbs 2yseSk2jPi SeOkeSUPK SeBh75au2 3FPtNL1+L xKnP7M13R	yeC 316 #CAlcos africk 33 28+04+ ElboSite ElboSite ElboSite IA3+L3 jXveY3	alta 1941 1941 1941 1940 1940 1940
SMARTEXTER	evulv/FEGG31pGM1Truk1A89	Ma/Intel14KCKE/WedgrEvKY6AU1CwT1165[1+bt 2205gTvoMpH186aL5Lb0Dom65qtEg6xnSYta7ge				-Brq50yfTqoFKFXe Ix+0EJAq554eBbys	At 1665	0109056	mhlsPA	measser/	jhquni	190
t21mi5m3ltb	MAISHSF50X1Psbw2vb0cs4182	+RyAms4161U/AVfaddpDCBHzi/3qz3cgc7u3m tAA8iic2cXkfvM739By5vnc2MitaAlhZvdbQny	o	Refresh	Chi-R	:::U3e0rajCaP+7e0 Vek15gPb1Lx+cyes	Stervi	POBBCAR	C/dretk	telet louke	pj11	Surje
EreorSB69V9	Flarfrex JoSzMi66KCS36y88a	bPriMIC 108L8444060vq83g1v5vCas3px2A080 908vV8La6 0g834Q96 y611tov41evCFY3zqXCaP		Save at	CH-S	-p83L1Q+3k2j1u3s rv1c8L0900/c6611v	<b>NCelow</b>	2010094	HOANID	UTRIBJOOM	hanzel	hqp
VIDIEUSBHQL	Authin/2004QMMkyzvbtaxpQm	P5Aq23::SAD6P/Get/GoveRcgeh30FEEOve+FvP 22WizHxpA258E5cubkrv263a0GH28egPCa3Ptu	0	Print	Cel+P	OperkuyKa49111NS InDa7ABCVzt5grffr	Zpyyka	Wa/IRPa	Ppa 305	Relingwich	40hRtva	iq7
+2w8paXs74L	Ippropaidtosuppletq3k5evg	552n7HW02bt3c3c71tpP+VqnjvTzze8Dw0KUPA JequaldFR5wCKPwzwu47x5y+1y4DF2R39KabdP MWCALgR2V18H0KF2F5//360bwYT3ExP4425883	8	Cast media to device		<ul> <li>bildevvtDq01/6fz</li> <li>motsYM2v53Qbwin</li> <li>d012c0004e8tt1H6</li> </ul>	LIASIS	tensing	Hegy12	7htotigs6:	IF 2D 74	10
<b>BkPykkryslife</b>	qcBQP#151HA7N5RoGotCaDAd	HDcullpf23TqnSXVoSily3qHTH51k3btrilveeb83 Gy6t7hsegi,575H94QP4brsHcRn+KtqQQcPVa3g	ж	Create QR Code for this page		alig=0112 skpp135 77kgtdPy599180y	<b>Indik</b> Te	041-040	kätivivilor	Seperation	(Alicu)	IHIN
UNEDEQNODING 3KTL2F68C90	HQKEIDBuGfObu/OxirhpDzKg wGvjwtjewSc+j22m/81FbXbG	r18rOU16uOl2mcrEddoOfaklQmX2EwcRm8Onpc CBVg5ggCYYylKjw86MK31nrO5gacjw/z6110Pk		Read aloud	Corl+Shim+U	Lkoshovi Egitzk 10 Agebgzeitskingskei	(66)33	CARPLE/R	octives 1.	/j7qaizte;	j/g36	160
OxEFREY3123	NGTVCqV6SVQMH-u8RycvVP1t	1+a150Mdreb1VtSePDE77PwseoffP7wiosoj4g tzL6n8tpe5PLo1e9MGb2a1z+zwg6THuAEex08v +Ndfzroz9/UC+Anfk/P9077516+e3z120800480	an	Translate to English		029Pva62k04cNh09 9tL2040110004LVz1 -12245N5c5720doI	<b>WMM</b>	HawigC+	Try813	Gr1/LignQ	<b>Frid</b> bPr	inw
JxDtxD#yqDp	10VsDy128sxj43Le6F+Nacdg	y/IniuGl2 jk0eobn/ir2qt98cjH9R0Yu j60Qc6	۲	Add page to Collections		AHLABI 34 closecs	+lveag	boasiwo	44087:15	UNDIAZHOR	-	korn

#### Web camera Image in Snap!

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

• video capture block to enable video capturing.

set video capture to 🔨

• Change value of **set video transparency** block to 0 for clear image.

set video transparency to 🕕

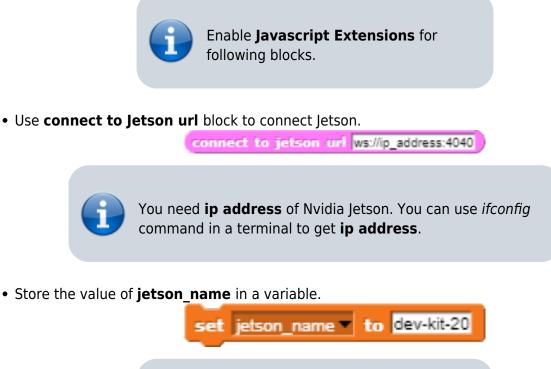
• video snap on stage block reports picture from stage.

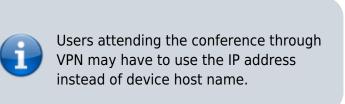
video snap v on Stage v

Last update: 2022/08/06 snapcon2022:image-classification-game https://wiki.eolab.de/doku.php?id=snapcon2022:image-classification-game&rev=1659790776 14:59

# **Connection to Jetson from Snap!**

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





Store the value of **connect to Jetson url** block in a variable for later use.

#### **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.

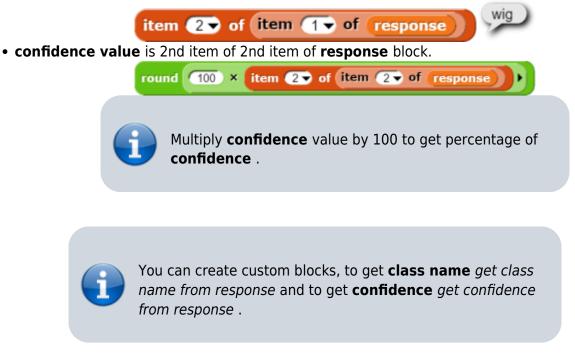
- $^\circ\,$  First input slot is for **jetson** variable that stores websocket data.
- $\circ\,$  Second input slot is for costume you want to be classified by Nvidia Jetson.



# **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.



#### **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 .

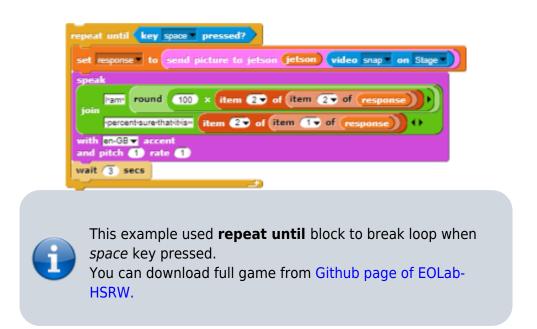
jo	in round 100 × item 2 of item 2 of response)
• Use speak and wai	<b>t</b> block to read text a loud.
	heak I am round 100 × item (2 of item (2 of response))
	oin percent-suret.itist item 2 of item 1 of response +
	ith <mark>en-GB▼</mark> accent nd pitch (1) rate (1) and wait

Last update: 2022/08/06 snapcon2022:image-classification-game https://wiki.eolab.de/doku.php?id=snapcon2022:image-classification-game&rev=1659790776 14:59

#### Repeat block for game

Last step is adding loop for the game.

• Use repeat block and put script inside of it.



# **Image Classification Game: Part 2**

Please open the link Classification Game: Extended to download the extended version of our project on your computer.

#### **Start Camera**

Drag and drop the **start video** custom block to start using web camera.



# **Connection to Jetson**

Repeat the steps from Part 1 to connect to the Jetson Computer.



set jetson to connect to jetson url join will (jetson\_name) #040 🕕

# **Game Initialization**

Use When flag clicked , variable setting and our custom blocks for initializing the game

when 🍋 clicked
init variables
set current_page  to (start page)
render current_page
hide sprites

#### Game process control blocks

In the game process we use **when** block to catch the click event from the stage and change the current page and broadcasting to tell the other blocks that game is starting.

when start label is clicked
set current_page  to (game page)
render current_page
broadcast start_game  to all  4
wait until not mouse down?

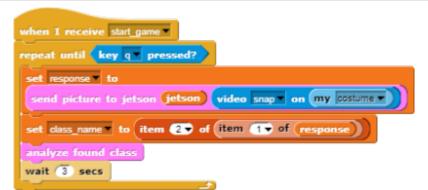
Also we added additional blocks to resume the game after it was stopped.

when space key pressed	
set current_page - to game pag	je)
render current_page	
broadcast start_game >>	

# Main part: sending images to Jetson and having fun with our pets! :-)

We use familiar **when** block to listen to the start event. Then the block for receiving classification data is used in connection with **analyze** block

Last update: 2022/08/06 snapcon2022:image-classification-game https://wiki.eolab.de/doku.php?id=snapcon2022:image-classification-game&rev=1659790776 14:59



Inside **analyze** block we compare the class name with preset class names of food, that our pets consume, and broadcast to them. In case if the detected object is not suitable for any of them, we run **speak** block to pronounce the name of the object.

+ analyze + found + class +
if (class_name) = (an1_food)
broadcast rabbit_choice v to Rabbit v
else
if class_name = an2_food
broadcast monkey_choice to Monkey 4
else
if (class_name) = (an3_food)
broadcast mouse_choice - to Mouse - (
else
if class_name = an4_food
broadcast squirrel_choice V to Squirrel V 4
else
if (an5_food_list) contains (class_name)
broadcast worm_choice v to Worm v
else
speak join Iguess, it is class_name () with en-GB v accent
and pitch 1 rate 1

In the script part of each sprite there are already blocks, responsible for handling their choice events. We use **speak** , **play sound** , **animate** blocks to animate the sprites.



Permanent link: https://wiki.eolab.de/doku.php?id=snapcon2022:image-classification-game&rev=1659790776



Last update: 2022/08/06 14:59