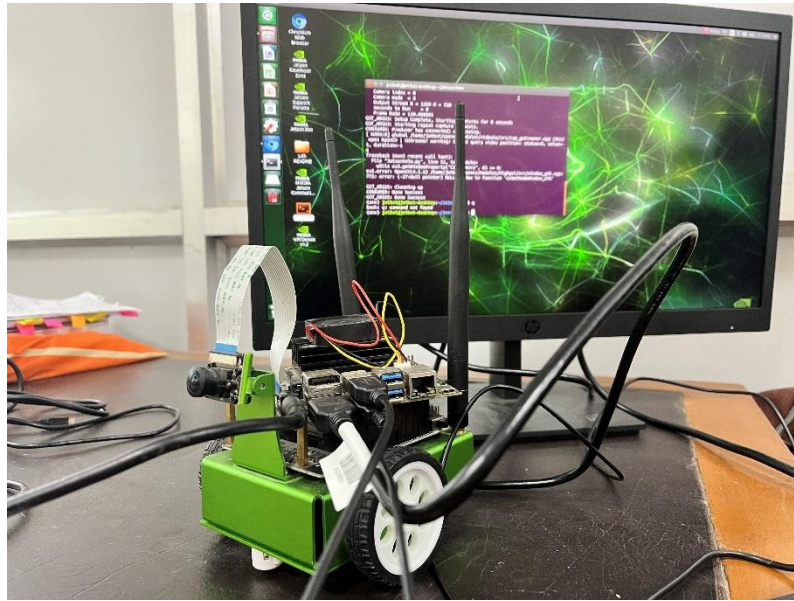


Object Detection on Jetson Nano using YOLOv5

Hardware Requirements: Jetson Nano, CSI camera, power adapter, mouse, keyboard, monitor.

Software Requirements: SD card with 32 or 64 GB. In this we have used 64GB



Setup Image a

Download the Jetson Nano 2GB image file from Nvidia jetson official website link is given below and follow the step as given in the website: <https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-2gb-devkit#prepare>

1. Then download **balenaetcher** software to flash the Jetson image in your SD card through this balenaetcher software using card reader.
2. Then insert the SD card in the SD card slot on Jetson Nano device.
3. Then a screen will be displayed once image was installed properly as shown above, initially it will ask for user name, password, wifi connection so do all the setting one by one.
4. Now once the setup is over next step is to install all dependencies and software required for object detection. will install one by one.

Software setup: install the following

1. `sudo apt-get install python3-pip` // for pip install
2. `pip3 install virtualenv` // creating virtual environment
3. `python3 -m virtualenv -p python3 env --system-site-packages` // installing all default package which comes with jetson nano in env folder
4. `source env/bin/activate` // for activating the env folder once you will execute this command you will get something like in my case **(env) jetbot@jetbot-desktop:**
5. `python -c 'import cv2; print(cv2.__version__)'` // this will give you default opencv version. And for deep learning application to run, we need to install higher version with CUDA support.

Create a swap file: this will make space for when out of memory storage occurs, to overcome this issue we need to create swap memory

1. `sudo fallocate -l 4G /var/swapfile`
2. `sudo chmod 600 /var/swapfile`
3. `sudo mkswap /var/swapfile`
4. `sudo swapon /var/swapfile`
5. `sudo bash -c 'echo "/var/swapfile swap swap defaults 0 0" >> /etc/fstab'`

Reboot the system by: `sudo reboot`

Swap space by using this command: `free -h`

Now install these Dependencies before installing OpenCV: these are essential to install the opencv

1. `sudo sh -c "echo '/usr/local/cuda/lib64' >> /etc/ld.so.conf.d/nvidia-tesla.conf"`
2. `sudo ldconfig`
3. `sudo apt-get install build-essential cmake git unzip pkg-config`
4. `sudo apt-get install libjpeg-dev libpng-dev libtiff-dev`
5. `sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev`
6. `sudo apt-get install libgtk2.0-dev libcanberra-gtk*`
7. `sudo apt-get install python3-dev python3-numpy python3-pip`
8. `sudo apt-get install libxvidcore-dev libx264-dev libgtk-3-dev`
9. `sudo apt-get install libtbb2 libtbb-dev libdc1394-22-dev`
10. `sudo apt-get install libv4l-dev v4l-utils`
11. `sudo apt-get install libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev`
12. `sudo apt-get install libavresample-dev libvorbis-dev libxine2-dev`
13. `sudo apt-get install libfaac-dev libmp3lame-dev libtheora-dev`
14. `sudo apt-get install libopencore-amrnb-dev libopencore-amrwb-dev`
15. `sudo apt-get install libopenblas-dev libatlas-base-dev libblas-dev`
16. `sudo apt-get install liblapack-dev libeigen3-dev gfortran`
17. `sudo apt-get install libhdf5-dev protobuf-compiler`
18. `sudo apt-get install libprotobuf-dev libgoogle-glog-dev libgflags-dev`

Download OpenCV:

1. `cd ~`
2. `wget -O opencv.zip https://github.com/opencv/opencv/archive/4.5.1.zip`
3. `wget -O opencv_contrib.zip https://github.com/opencv/opencv_contrib/archive/4.5.1.zip`
4. `unzip opencv.zip`
5. `unzip opencv_contrib.zip`

Now rename the directories. Type each command below, one after the other.

1. `mv opencv-4.5.1 opencv`
2. `mv opencv_contrib-4.5.1 opencv_contrib`
3. `rm opencv.zip`
4. `rm opencv_contrib.zip`

Lets build OpenCV now:

1. `cd ~/opencv`
2. `mkdir build`
3. `cd build`

Copy and paste this entire block of commands below into your terminal. Paste it on note pad or chrome make it single line then paste

```
cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr -D
OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib/modules -D EIGEN_INCLUDE_PATH=/usr/include/eigen3 -D
WITH_OPENCCL=OFF -D WITH_CUDA=ON -D CUDA_ARCH_BIN=5.3 -D CUDA_ARCH_PTX="" -D WITH_CUDNN=ON -D
WITH_CUBLAS=ON -D ENABLE_FAST_MATH=ON -D CUDA_FAST_MATH=ON -D OPENCV_DNN_CUDA=ON -D ENABLE_NEON=ON
-D WITH_QT=OFF -D WITH_OPENMP=ON -D WITH_OPENGL=ON -D BUILD_TIFF=ON -D WITH_FFMPEG=ON -D
WITH_GSTREAMER=ON -D WITH_TBB=ON -D BUILD_TBB=ON -D BUILD_TESTS=OFF -D WITH_EIGEN=ON -D WITH_V4L=ON -D
WITH_LIBV4L=ON -D OPENCV_ENABLE_NONFREE=ON -D INSTALL_C_EXAMPLES=OFF -D INSTALL_PYTHON_EXAMPLES=OFF -
D BUILD_NEW_PYTHON_SUPPORT=ON -D BUILD_opencv_python3=TRUE -D OPENCV_GENERATE_PKGCONFIG=ON -D
BUILD_EXAMPLES=OFF ..
```

make -j4 // this will take around 3hr time/ according to the processor speed

Finish the install of OpenCV

1. cd ~
2. sudo rm -r /usr/include/opencv4/opencv2
3. cd ~/opencv/build
4. sudo make install
5. sudo ldconfig
6. make clean
7. sudo apt-get update

Verify whether OpenCV installed properly or not

open the python shell by

python3

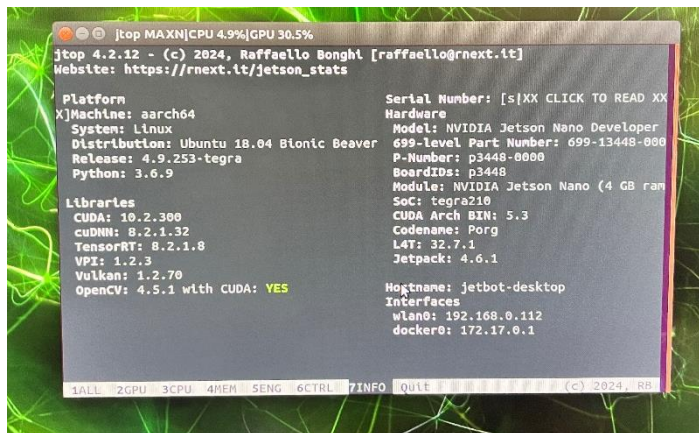
```
>>> import cv2
```

```
>>> cv2.__version__ //shows the updated opencv version
```

Next install the jtop which will show the cpu, version, memory usage everything

Press ctr+d for returning back to the folder or default bash than copy and paste command on by one

1. cd ~
2. sudo -H pip3 install -U jetson-stats
3. sudo reboot
4. jtop



jtop image b

Here you can see the version of jetpack and updated OpenCV, CUDA, python version. And in below option you also see memory usage, cpu and gpu. [2]

Next is to test the camera

open new terminal the type

1. ls /dev/video0
2. nvgstcapture-1.0 // this will open the csi camera and to exit from camera type q this will exit

Installing Torch and Torchvision

The Torch and Torchvision cannot be installed using pip, so go to link given below. than find your Jetson Nano version using **jtop** command as shown above image [b]

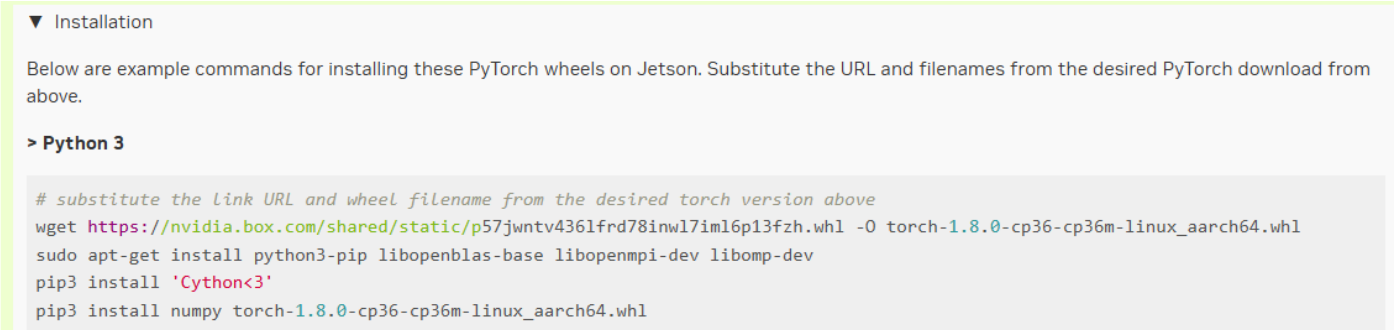
1. jtop //version of Jetson Nano in my case mine is 4.6.1 once you know you jetson you have to install
2. <https://forums.developer.nvidia.com/t/pytorch-for-jetson/72048> //link for install torch and torchvision
3. Scroll on to your jetpack in my case its jetpack 4

- I have installed PyTorchv1.8.0 it is compatible with jetpack version 4.6.1 click on the PyTorchv1.8.0 next click on the Installation (blue tip mark) in **instruction** menu



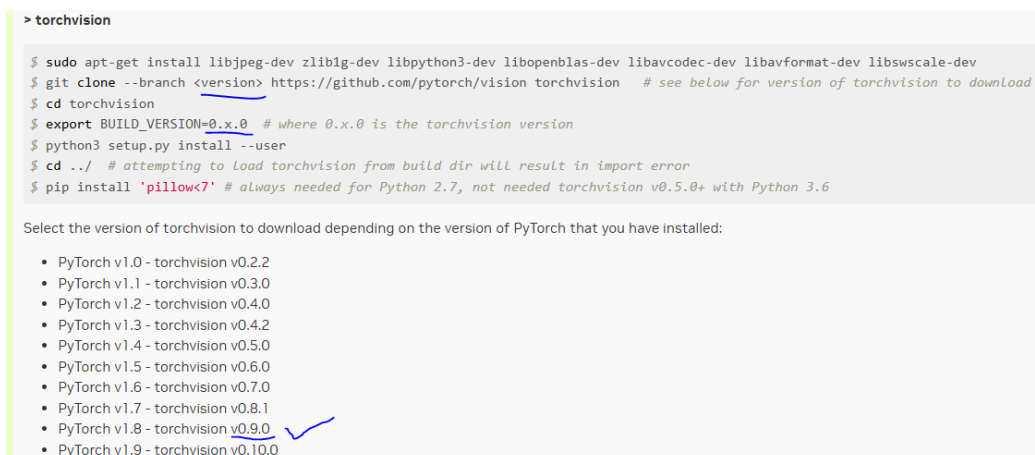
Under lined text has to be copied and pasted on the **highlighted** command below

Install the below command one by one for Torch



- `wget https://nvidia.box.com/shared/static/p57jwntv436lfrd78inwl7iml6p13fzh.whl -O torch-1.8.0-cp36-cp36m-linux_aarch64.whl`
- `sudo apt-get install python3-pip libopenblas-base libopenmpi-dev libomp-dev`
- `pip3 install 'Cython<3'`
- `pip3 install numpy torch-1.8.0-cp36-cp36m-linux_aarch64.whl`

Next is to install Torchvision



See the version of your torch according to jetpack version the paste the version of torchvision on blue line marked command Ex: I have installed pyTorch v1.8 – torchvision v0.9.0. (blue tick shown)

Installing Torchvision

1. `sudo apt-get install libjpeg-dev zlib1g-dev libpython3-dev libopenblas-dev libavcodec-dev libavformat-dev libswscale-dev`
2. `git clone --branch <version> https://github.com/pytorch/vision torchvision` // write your version of blue tick shown above Ex: v0.9.0 for jetpack 4.6.1
3. `cd torchvision`
4. `export BUILD_VERSION=0.x.0` // replace 0.x.0 to your torchvision 0.9.0 is the torchvision version , for torch 1.8.0 version for jetpack 4.6.1
5. `python3 setup.py install --user`
6. `cd ../`
7. `pip install 'pillow<7'` // always needed for Python 2.7, not needed torchvision v0.5.0+ with Python 3.6

once the Torch and Torchvision installation of is over next is to test the version so that to know whether installed properly or not using below command

go to python shell

`python3`

`>>> import torch`

`>>> print(torch.__version__)` // it should give you which Torch version you have installed in my case its 1.8.0

`>>> import torchvision`

`>>> print(torchvision.__version__)` //it should give you which Torch version you have installed in my case its 0.9.0

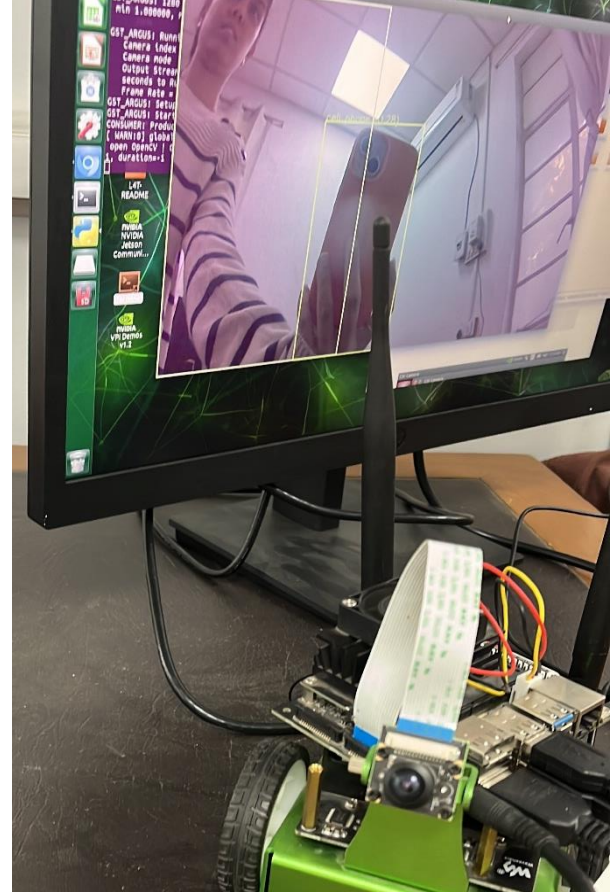
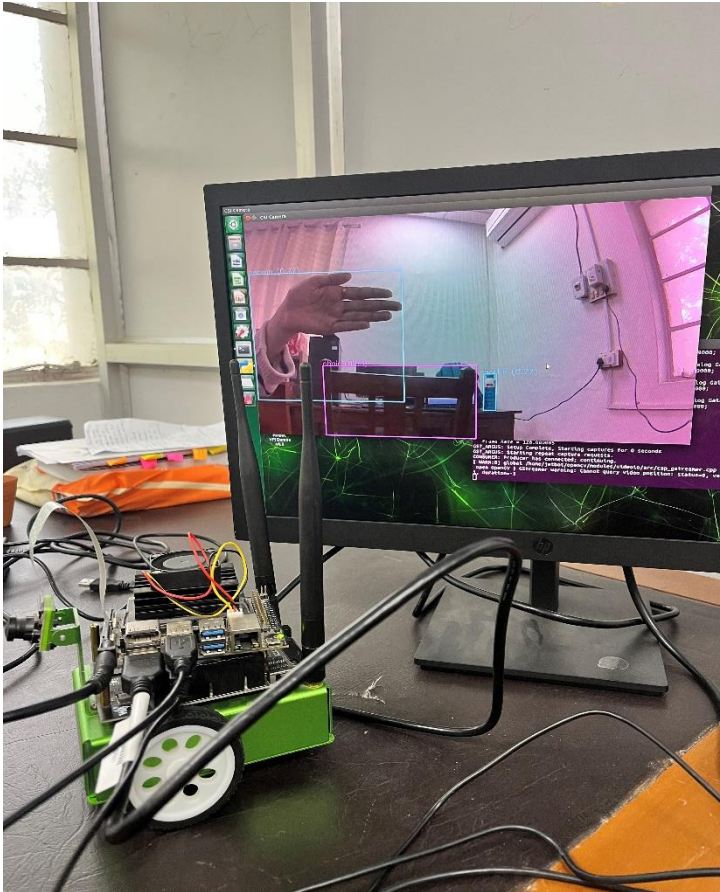
ctr+d for returning back to command bash

Running an object detection on Jetson Nano:

Clone the repository

1. `git clone https://github.com/amirhosseinh77/JetsonYolo.git` //once you clone the yolov5 there is folder which will have data, models, python scrips
2. enter into to cd weights folder: `cd weights`
3. `wget https://github.com/ultralytics/yolov5/releases/download/v5.0/yolov5s.pt` //this will create yolov5
4. `python3 JetsonYolo.py` //finally run this in the cd JetsonYolo folder
this will take 3 to 4 minute to run the command later camera will be ON, you can show the objects to the camera to detect.
after this if you get error with tqdm run: `pip install tqdm`

Final output: the python script is a JetsonYolo.py in that if you open, total 80 class are mentioned where it can detect 80 different object. As some of them shown below.



Reference:

1. <https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-2gb-devkit#prepare> for jetson image download
2. <https://www.youtube.com/watch?v=P-EZr0zy53g&list=PLv8Cp2NvcY8AkXRldCAYCvFxRUs0h5JJF&index=3> opencv & cuda [YouTube video]
3. <https://forums.developer.nvidia.com/t/pytorch-for-jetson/72048> for pythorch and torchvision
4. <https://towardsdatascience.com/yolov5-object-detection-on-nvidia-jetson-nano-148cfa21a024> object detection [article]