

SD card Preparation

1. Download [Jetpack 6.1](#) : SD Card Image Method -> Download Jetson Orin Super Developer kit
2. SD formatter: Format micro-SD card
3. Balena Etcher: Write the downloaded image file to the card
4. Refer to the [guide](#)

Initial Setup

1. Boot up Jetson with the SD card image
2. Connect computer with USB port on PC and USB-C port on jetson nano
3. Download PuTTY and connect to Jetson by selecting the connection as the following:
 - a. Serial -> Baud Rate 115200 -> Select correct COM
 - b. Type 'mode' into windows command prompt to find the correct COM
4. If this is the first time setting up connection to this PC follow the step in the guide provided in SD card Preparation (4th step)
5. Log in using your created user ID and password (write these down you will need these every time you use the jetson nano)

Connecting to WIFI

1. Command to connect to your Wi-Fi network (replace angle brackets with SSID and password):
 - a. `sudo nmcli device wifi connect <MY_WIFI_AP> password <MY_WIFI_PASSWORD>`
 - b. Further guide can be found [here](#)
2. A message should come up saying it is successfully activated, once this is active, find and record the IP address of the Jetson Nano using: `$ ifconfig` in the command line
3. If the PC and the Jetson nano are on the same Wi-Fi network, the serial cable can be disconnected and you can use PuTTY to connect to the jetson through wifi:
 - a. PuTTY -> enter IP of Jetson Nano -> Choose the SSH -> Use the same user ID and password that was used to set up serial connection

Acquiring Pictures from camera

1. From this [source](#) follow the following steps:
 - a. Start at 'USB Camera' section, paste the first 'nvgstcapture' code in the terminal (not in any folders) with N as 0
 - b. Scroll to the next 'USB camera' section to capture images, this gives 2 options to capture images, manually or automatically. Run these lines in the command window. If this process works as expected, video can be captured in part c.

- c. To capture video, run the lines in the next 'USB camera' section in the same fashion as previous. This also gives 2 options for video capture, manual and automatic.

Initializations for Coding

1. Install VS code:
 - a. Use chromium browser to download VS code
 - i. Choose ARM64 version
 - ii. Choose the '.deb' file
 - b. Install
 - c. Add the following extensions:
 - i. Code Runner
 - ii. Python Extension Pack
2. Install Python execute the following:
 - a. Install python, in command window:
 - i. `sudo apt install python3-pip`
3. Install OpenCV execute the following:
 - a. Install, in command window:
 - i. `pip install opencv-python`
 - b. Check to see if OpenCV is installed properly
 - i. `python3 -c "import cv2: print(cv2.__version__)"`
4. Reinstall different version of numpy, execute the following:
 - a. 'Reinstall' numpy, in command window
 - i. `pip install -force-reinstall numpy==1.23.5`
5. Obtain and test sample code from [here](#), execute the following:
 - a. From 'README' the code to inference with pytorch hub, was used to test the validity of the model. Use this code to ensure that the program goals have been met with respect to a still image.
 - b. This code will be put into a '.py' file and ran in VScode.
6. Modify '.py' code to take in USB camera input and display in real time
 - a. Delete the 'img' declaration and modify model parameters to take in USB camera source
 - i. Insert `source=0,stream=true` into the model parameters

Model selection and GPU allocation

1. Install jtop
 - a. Type the following into the command window or follow step from this [source](#)
 - i. `sudo pip3 install -U jetson-stats`
 - ii. `sudo reboot`
 - iii. `jtop`

- b. If jtop opens as expected, it is installed correctly
2. Install 'Ultralytics' from this [source](#) in the command window
 - a. Start with Native Installation -> Run on Jetpack 6.1
 - i. Run the second and third step in the "Install Ultralytics Package" section
 - ii. Run the 2 lines that will install torch 2.5.0 and torch vision 0.20
 - iii. Run the lines that will install 'cuSPARSElt' to resolve dependencies with 'torch'
 - iv. Install 'onnxruntime-gpu' with the specified line
 - v. Reinstall numpy to the proper version as it may have been changed
 1. `pip install numpy==1.23.5`
3. In the '.py' file used to test the camera operation, do the following
 - a. Under: Use TensorRT on Nvidia Jetson -> Convert Model to TensorRT and Run Inference example
 - i. Test this example as whole, delete previous code except
 1. `results.print()`
 2. `results.show()`
 - ii. If this takes too long (as it did with our experience)
 1. Remove lines 6-10 (leaving import, model and results declaration)
 2. Edit results declaration to use 'model' instead of 'trt_model'
 - b. Modify to take in the USB camera input
 - i. `results = model(source=0, stream=true)`
4. Finalize code in .py file, Refer to: Code and Output -> Code (repository) or Appendix A (this document)
 - a. Insure that lines 1-3 are present at the top of code
 - b. Load the pretrained AI model "YOLO11n" (line 6)
 - c. Bring the model to execute on the GPU for accelerated heterogenous computing using CUDA (line 7)
 - d. Use open cv to:
 - i. Access Video (line 9)
 - ii. Capture video frame by frame (lines 17-20)
 - e. Run the inference of each frame (line 23)
 - f. Annotate and display each frame using the AI model and open CV (lines 26 & 29)
 - g. Ensure the loop has a break key (lines 32 & 33)
 - h. Release the video and terminate the window using open CV when the video is over (lines 36 & 37)