THE UNIVERSITY OF TENNESSEE KNOXVILLE

National Science Foundation

Training and Navigating with the Jetson Nano

Subir Warren
Katie Cheung

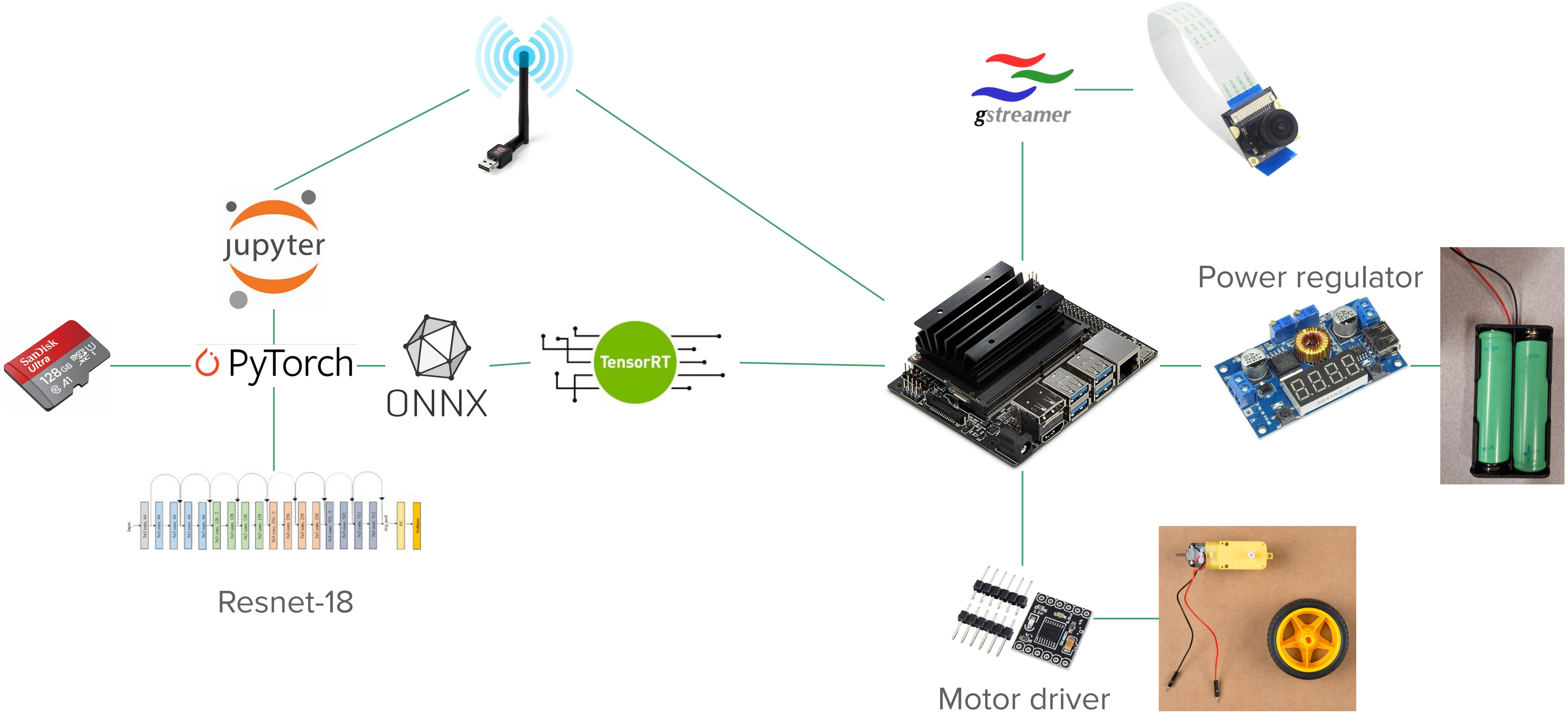
Baruch COLLEGE

Stony Brook University

Acknowledgements

This project was sponsored by the National Science Foundation (NSF) through the Research Experiences for Undergraduates (REU) award and hosted by the University of Tennessee Knoxville. We also would like to thank the NSF for sponsoring this project as well as Dr. Kwai Wong as a mentor for this project.

(Jetbot) Jetson Nano Schematic

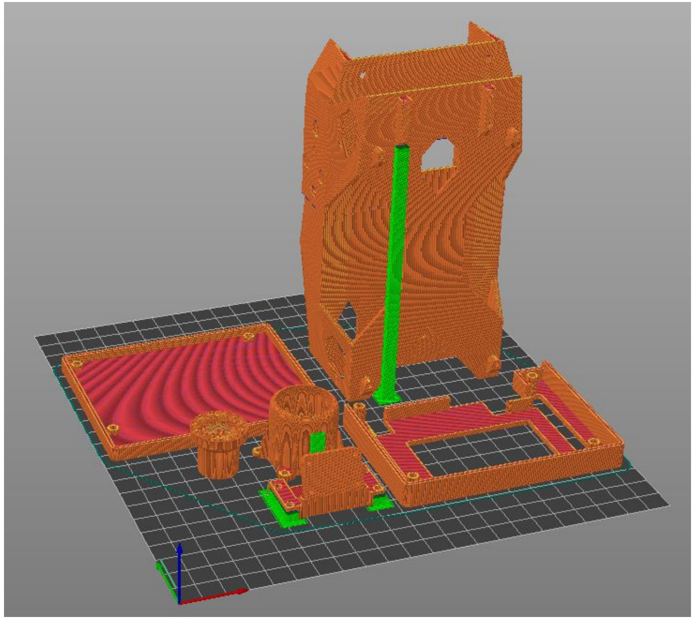


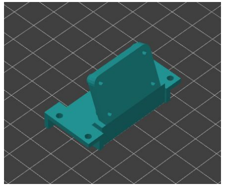
Power regulator

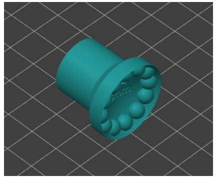
Resnet-18

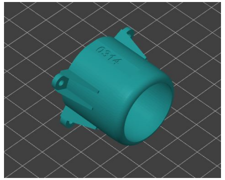
Motor driver

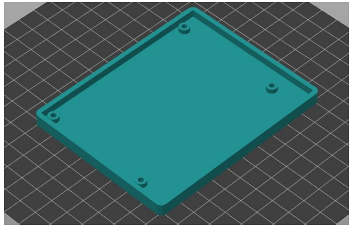
3D printed parts (Version 3 Jetbot)

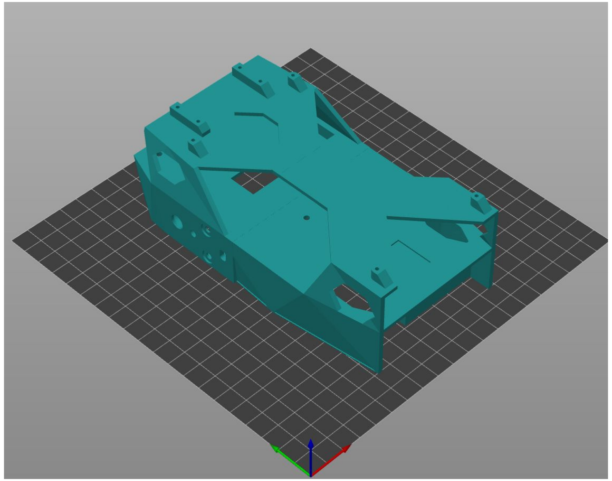


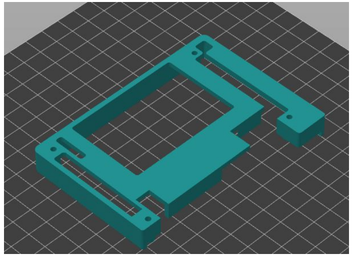


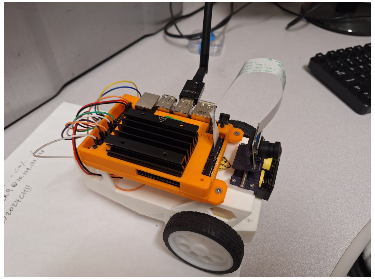










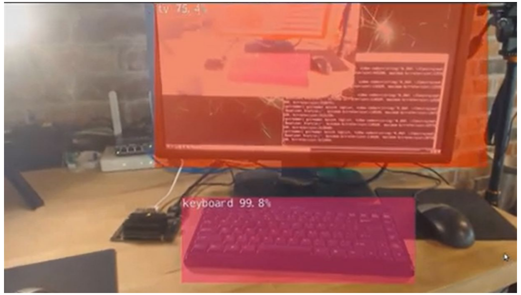


Software components

- PyTorch is a deep learning framework that provides tools and libraries for building and training neural networks.
- ResNet-18 is one of the pre-trained model architectures based off ImageNet, specifically designed for image classification tasks.
- ONNX allows an exported model to be used on different hardware platforms or different software systems that use different deep learning frameworks.
- TensorRT can optimize the inference performance model, taking advantage of NVIDIA GPU-specific optimizations.
- Using Pytorch and TensorRT, we can take advantage of torch2trt, which allows jetbot to move while detecting objects.

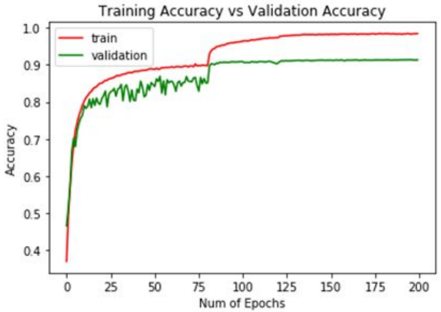
Here is detectnet, which is

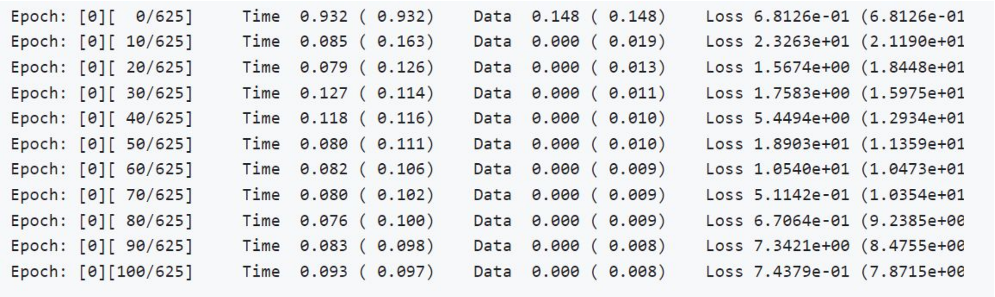
Object detection for multiple objects



Training Models

- The group was successfully able to train a Jetbot to detect tools, such as a wire cutter, screw driver, and scissors. The batch size for the model was 10, the amount of workers were set to 1 and we did only one epoch. Accuracy was approx. 43%.





```
Epoch: [0][  0/625]     Time  0.932 ( 0.932)    Data  0.148 ( 0.148)    Loss 6.8126e-01 (6.8126e-01
Epoch: [0][ 10/625]     Time  0.085 ( 0.163)    Data  0.000 ( 0.019)    Loss 2.3263e+01 (2.1190e+01
Epoch: [0][ 20/625]     Time  0.079 ( 0.126)    Data  0.000 ( 0.013)    Loss 1.5674e+00 (1.8448e+01
Epoch: [0][ 30/625]     Time  0.127 ( 0.114)    Data  0.000 ( 0.011)    Loss 1.7583e+00 (1.5975e+01
Epoch: [0][ 40/625]     Time  0.118 ( 0.116)    Data  0.000 ( 0.010)    Loss 5.4494e+00 (1.2934e+01
Epoch: [0][ 50/625]     Time  0.080 ( 0.111)    Data  0.000 ( 0.010)    Loss 1.8903e+01 (1.1359e+01
Epoch: [0][ 60/625]     Time  0.082 ( 0.106)    Data  0.000 ( 0.009)    Loss 1.0540e+01 (1.0473e+01
Epoch: [0][ 70/625]     Time  0.080 ( 0.102)    Data  0.000 ( 0.009)    Loss 5.1142e-01 (1.0354e+01
Epoch: [0][ 80/625]     Time  0.076 ( 0.100)    Data  0.000 ( 0.009)    Loss 6.7064e-01 (9.2385e+00
Epoch: [0][ 90/625]     Time  0.083 ( 0.098)    Data  0.000 ( 0.008)    Loss 7.3421e+00 (8.4755e+00
Epoch: [0][100/625]     Time  0.093 ( 0.097)    Data  0.000 ( 0.008)    Loss 7.4379e-01 (7.8715e+00
```

By using the github repository provided by Patrick Lau and Franklin Zhang, we were also able to run the model where we were also able to replicate the jetbot to move along a track once with some errors.

Problems Encountered

1. When assembling the Jetbot, the camera module would frequently show connection to the Nano, while other times it would not.
2. Drivers needed to be updated and removed as they were either updated or discontinued. (Ex. wifi, cameras, python, etc.)
3. Balena Etcher doesn’t work with flashing the images to the micro SD cards on Ubuntu and terminal download failed
4. Running libraries in Jupyter Notebooks often lead to errors due to space.
5. Often power supply would stop and Jetbot would shut down
6. Motors could move in every direction but not forward or backwards.
7. Jetson Nano has a bootloader issue, despite updating software and other troubleshoots.

Solutions and Adjustments

1. We switched camera modules and reinstall github software from 2023.
2. We updated and downloaded drivers that worked, such as for the wifi dongle.
3. Since Etcher didn’t work we’ve switched to an alternative: Startup Disk Creator for flashing the SD cards.
4. Gparted must be used to resize partitions on SD Card to have more space.
5. Wiring was changed so that power was supplied directly from the power regulator to the Jetson Nano instead of using a DC jack
6. *Possible solution to bootloader problem: flashing SD card through SDK Manager*

Documentation will need to be updated to help others build using the Jetson Nano if and when bootloader & motors has been troubleshooted.

Improving with the Jetson Orin Model (Future Plans)

Last year’s Jetbot team believed that due to the Jetson Nano crashing, the Jetson Orin would be better for image processing and visualization. We will be moving towards creating and replicating the processes with it. Orin is 80x more powerful than the Jetson Nano and thus will also be able to run ResNet 50, which will allow for higher accuracy and the ability to run more datasets to train models.

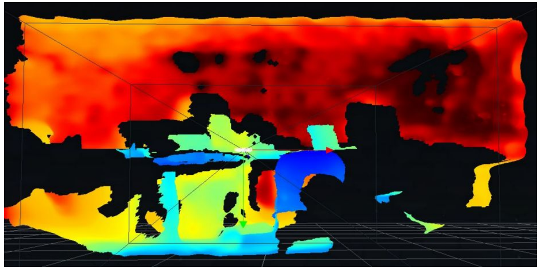
3D models will accommodate the Orin and will have 2 cameras, which will be able to follow other jetbot cars (after troubleshoot). A larger chassis could accommodate more expansion boards and wheels.

Future Prospects with the Orin

The Orin has many features which can significantly improve the Jetbot, such as...

- Improved object detection and recognition for better obstacle avoidance and navigation
- Increased autonomous capabilities, including path planning, mapping, and self-localization
- Integration of advanced sensors like depth cameras for more accurate 3D perception
- Deployment of complex deep learning models for tasks like semantic segmentation, object tracking, and action recognition
- Enhanced real-time performance for low-latency control and decision-making

Research being done by Realsense Team via the Orin can help to add a stereo camera and allow SLAM features to be added.



Thank You!

Any Questions?

Reference

1) Dusty-Nv. (2019, June 18). Dusty-NV/Jetson-inference: Hello AI World Guide to deploying deep-learning inference networks and deep vision primitives with TensorRT and Nvidia Jetson. https://github.com/dusty-nv/jetson-inference
2) Zhang, Franklin. “Jetbot” *GitHub*, github.com/franklinzhang12/jetbot.
3) Barlow, Nicholas. “Jetbot Documentation”, https://bitbucket.org/EDKLW/jetbot-car-2023/src/master/documentation/JetbotDocumentation_Nicholas.pdf
4) Lau, Patrick, “Jetbot Assembly Guide”, https://docs.google.com/presentation/d/1MOtegvjN1XGcSuTHRs4KnJr0uxTNdZtAVcILk1ymLaA/edit#slide=id.g2bc93d7f113_0_4
5) “Jetbot.” *JetBot*, jetbot.org/master/.
6) “Jetson AI fundamentals”, https://www.youtube.com/watch?v=VWdJ4BCtam8&list=PL5B692fm6--uQRRDTPsJDp4o0xbzkoyf8
7) Chart 1 “Epoch vs Accuracy”- https://datascience.stackexchange.com/questions/47209/what-happened-to-the-accuracy-before-and-after-75th-epoch-why-its-unstable-at-f