UCF REU 2022 Week 2

Connor Cabrera
Background

- School
  - University of Central Florida
  - Senior
  - Major in Computer Science
  - Minor in Intelligent Robotic Systems

- Courses
  - Artificial Intelligence
  - Robot Vision
  - Robot Systems
  - AI for Game Design

- Progress
  - Setting up PyCharm and libraries to support a GPU
  - Exposure to multiple ideas and aspects regarding computer vision
MNIST Part One: Simple NN

Testing Loss: 0.0681
Testing Accuracy: 0.9786
MNIST Part Two: Simple CNN

Testing Loss: 0.0310
Testing Accuracy: 0.9895
CIFAR-100 Key Notes

● Originally had an issue where loss and accuracy were not improving
  ○ Implemented ResNet to confirm it wasn’t a data input issue
  ○ Solution: Change optimizer from “Adam” to “SGD”

● To improve Testing Accuracy (decrease overfitting)
  ○ L2 Regularization
  ○ Early Stopping, Monitoring Loss, Patience = 3
  ○ Dropout Layers
CIFAR-100 Part One: Differing Number of Layers

- **8 Layers**
  - 29.7M Parameters
  - Training Time: 32 min
  - Testing Loss: 2.3112
  - Testing Acc: 0.4704

- **11 Layers**
  - 90.3M Parameters
  - Training Time: 50 min
  - Testing Loss: 2.1243
  - Testing Acc: 0.5235

- **16 Layers**
  - 20M Parameters
  - Training Time: 100 min
  - Testing Loss: 1.7328
  - Testing Acc: 0.6002

- **18 Layers**
  - 21.1M Parameters
  - Training Time: 123 min
  - Testing Loss: 1.8587
  - Testing Acc: 0.58
CIFAR-100 Part Two: Differing Training Sizes on the Best Model

50 Images Per Class
- Training Time: 11 min
- Testing Loss: 3.2832
- Testing Acc: 0.2629

125 Images Per Class
- Training Time: 31 min
- Testing Loss: 2.6522
- Testing Acc: 0.3948

250 Images Per Class
- Training Time: 42 min
- Testing Loss: 2.1776
- Testing Acc: 0.4954

400 Images Per Class
- Training Time: 82 min
- Testing Loss: 1.8157
- Testing Acc: 0.5830
Conclusions

- Larger models make for better results, but a limit exists where too large a model will start to deteriorate performance.
- Larger training datasets make for better results, but with diminishing gains.
- Multiple techniques for reducing overfitting are necessary for achieving the best results.
Progress on Research

- Topic: Building an Attention-Based Spike Neural Network
- Began reading the first reference provided by Dr. Vaca-Castano on Spiking-YOLO for SNN
  - This required a refresher on the traditional YOLO methodology
    - Useful for object detection with multiple objects in an image
    - Divide training images into cells (ex. 4x4, 16x16) and assign a vector to each cell
    - Apply non-max suppression on output using Intersection Over Union to eliminate multiple bounding boxes for the same object
    - If multiple objects still reside in the same cell either consider increasing the size of the vector or increasing the number of cells
  - Somewhat understand the calculations behind the “spike” but don’t fully realize what happens once a spike is generated