

ABSTRACT

My research is about learning how to build an Image Classification Artificial Intelligence model that recognized 60,000 images in the CIFAR-10 database accurately. Image Classification is being used more frequently in technology today with applications ranging from face ID on your phone to self driving cars. My progress in developing this program was primarily through trial and error, but continually speaking with my mentor for advise on how to optimize my program was a big advantage. As a result, my AI model filtered through all of the training and testing images in the database with a 98% accuracy level.

INTRODUCTION

Image Classification AI models are everywhere in todays technology sphere, with many different applications in fields like facial recognition, cybersecurity, traffic control systems, self driving car, and so much more. Without it, our lives would be drastically different and I am glad to have learned about it.

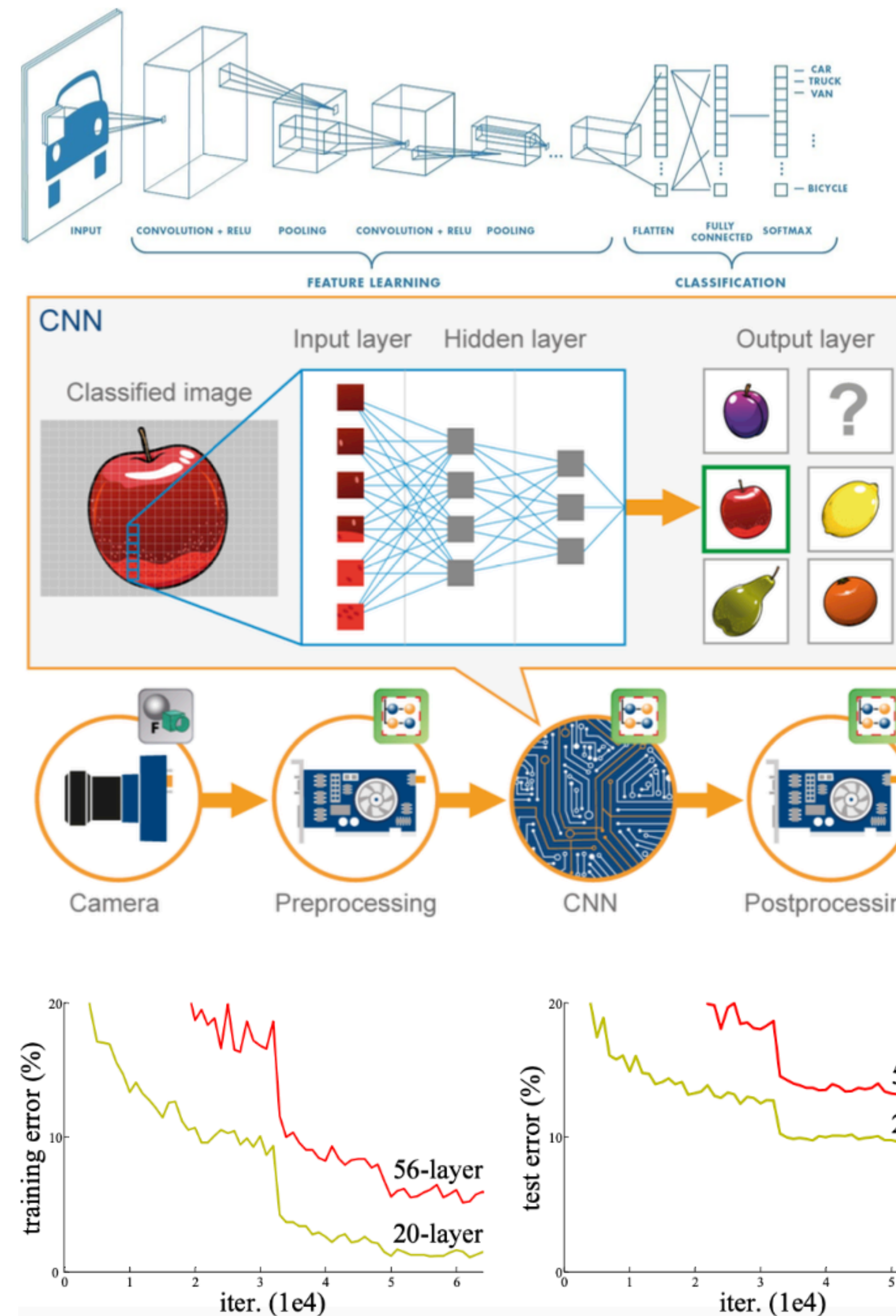
RESULTS

• The trained AI model is currently running to 99% accuracy

```
Train Epoch: 1 [48000/60000 ( 80%)] Loss: 0.105997
Train Epoch: 1 [48640/60000 ( 81%)] Loss: 0.025460
Train Epoch: 1 [49280/60000 ( 82%)] Loss: 0.045346
Train Epoch: 1 [49920/60000 ( 83%)] Loss: 0.058124
Train Epoch: 1 [50560/60000 ( 84%)] Loss: 0.124371
Train Epoch: 1 [51200/60000 ( 85%)] Loss: 0.361128
Train Epoch: 1 [51840/60000 ( 86%)] Loss: 0.031059
Train Epoch: 1 [52480/60000 ( 87%)] Loss: 0.011888
Train Epoch: 1 [53120/60000 ( 88%)] Loss: 0.142231
Train Epoch: 1 [53760/60000 ( 90%)] Loss: 0.156632
Train Epoch: 1 [54400/60000 ( 91%)] Loss: 0.027537
Train Epoch: 1 [55040/60000 ( 92%)] Loss: 0.040891
Train Epoch: 1 [55680/60000 ( 93%)] Loss: 0.048493
Train Epoch: 1 [56320/60000 ( 94%)] Loss: 0.095243
Train Epoch: 1 [56960/60000 ( 95%)] Loss: 0.063206
Train Epoch: 1 [57600/60000 ( 96%)] Loss: 0.124005
Train Epoch: 1 [58240/60000 ( 97%)] Loss: 0.059471
Train Epoch: 1 [58880/60000 ( 98%)] Loss: 0.033517
Train Epoch: 1 [59520/60000 ( 99%)] Loss: 0.013027
Test set: Average loss: 0.0039, Accuracy: 9857/10000 ( 99%)
```

Convolutional Neural Network

- Layers
 - Fully Connected
 - Dropout
 - Pooling
 - Convolution
 - Relu
- Used 2 convolution, 2 dropout, and 2 FC layers in my model with soft and max pooling and flattening



CONCLUSIONS

In this project, I learned about and trained Convolutional Neural Networks in order for the model to recognize data.

FUTURE DIRECTIONS

Based on the deep learning model I achieved, I can continue into efficiency field or cybersecurity.

REFERENCES

- “A Comprehensive Guide to Convolutional Neural Networks - the Eli5 Way.” *Saturn Cloud Blog*, 17 June 2023, saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way/.
- Deep Residual Learning for Image Recognition - Arxiv.Org, arxiv.org/pdf/1512.03385.pdf. Accessed 24 Aug. 2023.
- Kadlaskar, Amruta. “Image Classification Using Convolutional Neural Network with Python.” *Analytics Vidhya*, 14 June 2021, www.analyticsvidhya.com/blog/2021/06/image-classification-using-convolutional-neural-network-with-python/.
- *Packt Subscription*, subscription.packtpub.com/book/data/9781838823412/4/ch04lv11sec35/recognizing-cifar-10-images-with-deep-learning. Accessed 24 Aug. 2023.

