Week 4 Presentation

Akhil Arularasu
UCF CRCV REU in Computer Vision
Project: Robust Image GeoLocalization
Mentors: Parth Kulkarni and Dr. Gaurav Nayak
06/16/2023
What happens if the input street-view images from the previous slide have noise?

- In the real world, images are not perfect (blurry, low resolution, rain, snow, etc.)
- Obviously, accuracy will decrease because model is not currently trained to handle imperfections
Working with: CVUSA (Cross-View USA) Dataset
From Last Week: Training Results on Clean CVUSA Data

- Accuracies (Percentages) after 59 Epochs (Stage 1)
  - top 1: 91.49031967582171
  - top 5: 97.5574065736155
  - top 1%: 99.74110760918505
Adding Noise

- This week:
  - I was able to accomplish my goal from last week of understanding the different types of noise and how I can apply it to CVUSA testing images.
  - Added 18 different types of noise to the CVUSA dataset (each variation of the testing images can be found within the folders depending on what noise is being added)
Adding Noise

- More info:
  - added noise only with severity level 1, will be changing this to levels 2-5 in the future, only need to modify a couple lines of code
  - each folder contains 8884 images (which is also the size of the testing dataset so there were no issues)
Strategy + Changes

❖ Original Strategy I took to accomplish task: modify Dataloader (CVUSA.py file) to apply noise to an image as soon as it is opened, then save each image.

❖ New Strategy (simpler): instead of modifying the Dataloader directly, I created an instance of the CVUSA object in a different file, moved all the testing .jpg images from existing csv file into a list and applied various noises through the list of images before saving.
Contrast (severity level 1)

Brightness (severity level 1)
Elastic (severity level 1)

Fog (severity level 1)

Gaussian Blur (severity level 1)
Gaussian Noise (severity level 1)

Glass Blur (severity level 1)

Impulse Noise (severity level 1)
Saturate (severity level 1)

Shot Noise (severity level 1)

Snow (severity level 1)
Saturate (severity level 1)

Shot Noise (severity level 1)

Spatter (severity level 1)
Evaluating TransGeo Model on Clean Test Data

- Validation Accuracies (Percentages):
  - top1: 0.03376857271499324
  - top 5: 0.27014858171994593
  - top 10: 0.49527239981990095
  - top1%: 3.388134624043225
  - time: 4.0766072273254395

- I wasn’t able to reproduce existing test results through the pretrained model, currently looking into code to find potential bug
Next Steps

- Increase severity levels
- Fix previous bug and then evaluate the model with corrupted data
- Train the Transgeo model on the clean training data + corrupted training data
- Look into noise detection/removal of noise to improve the model