ENPM673 – Perception for Autonomous Robots – Spring 2021

Project 4

Srikumar Muralidharan, Samer Charifa

Due date: 17th May 2021, 11:59PM

Submission guidelines:

- This project is to be done and submitted individually.
- Your submission on ELMS/Canvas must be a zip file, following the naming convention **YourDirectoryID_proj4.zip**. If you email ID is abc@umd.edu or abc@terpmail.umd.edu, then your Directory ID is **abc**. Remember, this is your directory ID and NOT your UID.
- Please submit only the python script(s) you used to compute the results, the PDF report you generate for the project and a detailed README.md file.
- For each section of the project, explain briefly what you did, and describe any interesting problems you encountered and/or solutions you implemented.
- Include intermittent steps outputs in your report.
- Please do not include the dataset provided to you and your video/images results in your submission, rather provide the link to your video/images output in the report.
- For this project, you are free to use any in-built and/or third-party functions, unless instructed otherwise.
- For Problem 2, please remember to cite the dataset (explained under problem description).

Project Description

In this project, your aim is to understand how Optical Flow can be used to estimate the velocity and directions in which cars move on a highway. Secondly, you will use the concept of Classification to classify 9 different seafood type (Fish) collected from a Supermarket in Turkey. This is easy for humans, but your computer will find it a lot more difficult.

Problem 1:

Dataset for Problem 1

Here, you are given a video of cars being driven on a highway. Your aim is to use the concept of Optical Flow (discussed in lectures) to track the motion of vehicles on the highway. This link gives you a brief explanation on Optical Flow.

Deliverables:

1. In one video, plot vector field of the optical flow for every frame. This can be done by comparing subsequent frames and checking how a pixel has progressed. We need the vectors to be spaced 25 pixels apart from each other (both vertical and horizontal).
2. In a separate video, remove the background and show only the movement of cars on the highway. You **can** use any inbuilt function for both these deliverables.

Problem 2:

Dataset for Problem 2:

The given training archive contains 9 different seafood types. For each class, there are 1000 augmented images and their pair-wise augmented ground truths. Each class can be found in the "Fish_Dataset" file with their ground truth labels. All images for each class are ordered from "00000.png" to "01000.png".

The dataset includes the following species: gilt head bream, red sea bream, sea brass, red mullet, horse mackerel, black sea sprat, striped red mullet, trout, shrimp. Train your network on these images to predict the labels for the testing dataset. Use a training to testing split of 80:20. Split the 9000 original images into your training and testing dataset. We need to shuffle the training data and then train our network.

Resources

In case you do not have access to a machine with a GPU for training the neural network, you can follow this link to use Google Colab and use GPUs for this project.

Also, here you can find a turtorial for using Pytorch with Google Colab.

Suggested Pipeline

You are required to implement a Convolution Neural Network (CNN) to perform classification on the given dataset. For such classification task, generally VGG-16 architecture is deployed. You are expected to implement the same using Tensorflow, Pytorch or Keras.
A good starting point for Tensorflow2
tutorials is the official Tensorflow tutorial and this great tutorial by Hvass Labs.

Improving Accuracy of your neural network

Now that we have a baseline neural network working, let's try to improve the accuracy by doing simple tricks.

1. Standardize your data input if you haven't already. There are a lot of ways to do this. Feel free to search for different methods. A simple way is to scale data from [0,255] to [-1,1].
2. Decay your learning rate as you train or increase your batch size as you train.
3. Augment your data to artificially make your dataset larger.
4. Add Batch Normalization between layers.
5. Change the hyperparameters in your architecture such as number of layers, number of neurons.

Now, feel free to implement as many of these as possible and present a detailed analysis of your findings as before. Present the same details as before, train and test accuracy over epochs, number of parameters in tour model, loss value over epochs, your architecture, and details of other tricks you employed.

Results

You need to provide two plots for both training and testing between:

- Accuracy and epoch
- Loss and epoch

Citation

Make sure you include this citation in your report.

- O.Ulucan, D.Karakaya, and M.Turkan.(2020) A large-scale dataset for fish segmentation and classification. In Conf. Innovations Intell. Syst. Appli. (ASYU)