BBM 433 - Introduction to Computer Vision Laboratory

HACETTEPE UNIVERSITY

Department of Artificial Intelligence Engineering

Programming Assignment 4
Spring 2021-2022
Assoc. Prof. Dr. Ali Seydi Keçeli

Due Date: 23:59 pm on Sunday, May 15th, 2022

Image Classification with Convolutional Neural Networks

In this assignment, you will get familiar with image classification by training Convolutional Neural Networks (CNN).The goals of this assignment are as follows;

1. Understand CNN architectures and build a model from scratch and train on data.
2. Understand and implement transfer learning from a pre-trained CNN.
3. Analyze your results.
4. Experience with a deep learning frame, TensorFlow.

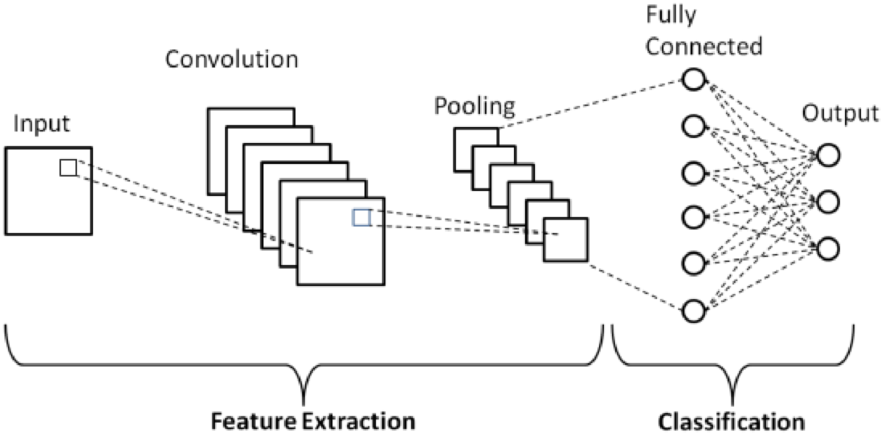

Figure 1: A simple CNN architecture [1].

Dataset

You will download MIT Indoor Scenes dataset [2]. Dataset consists of sample of scene images, see Figure 2. There are 67 indoor categories categories. In this assignment, you will train classifiers to classify images to one of the 67 categories. If the computation time is a problem, you can use a subset of this dataset. However, make sure that your training set is at least 3000 images (200 images per class). The validation and test sets should include at least 50 images per class, a total of 750 images each. Note that the more data you use, the better the learning will be.

PART 1 - Modeling and Training a CNN classifier from Scratch [60 pts]

In this part, you are expected to model CNN classifier and train it. You should first define the components of your model.

- Give parametric details with relevant TensorFlow code snippets; number of in channels, out channels, stride, etc. Specify your architecture in detail. Write your choice of activation functions, loss functions and optimization algorithms with relevant code snippets. [15 pts]

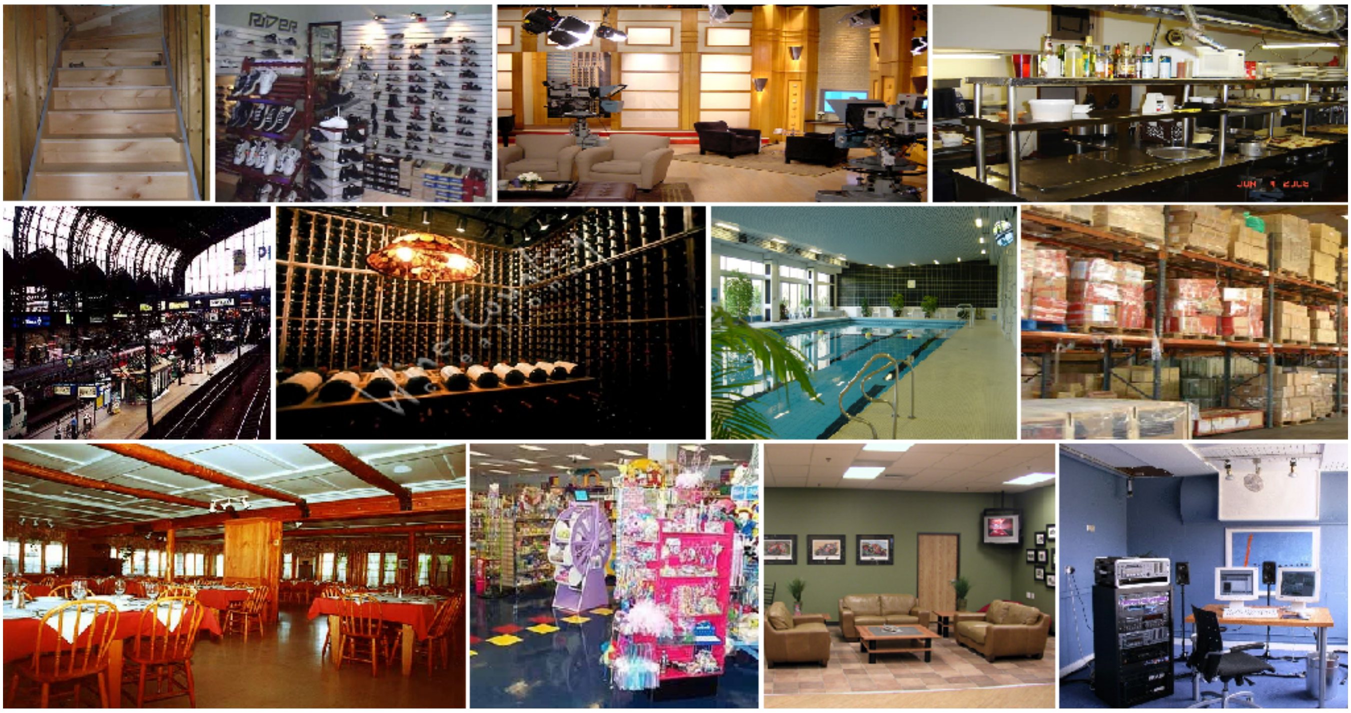

Figure 2: Dataset class samples.

Training and Evaluating your model

For both of your models;

- you will set epoch size to 100 and evaluate your two model with three different learning rates and two different batch sizes. Explain how you calculate accuracy with relevant code snippet. Moreover for each of the points below add relevant code snippet to your document.

1. Draw graphs of loss and accuracy change for three different learning rates and two batch sizes. [15 pts]
2. Integrate dropout to your your model. In which part of the network you add dropout and why? Explore four different dropout values and give new validation and test accuracies. [10 pts]
3. Plot a confusion matrix for your model's predictions. [10 pts]
4. Explain and analyze your findings and results. [10 pts]

PART 2 - Transfer Learning with CNNs [40 pts]

Now, you will fine-tune the pre-trained VGG-16 network which is available at TensorFlow. This network is trained on ImageNet dataset so you are not initializing the weights randomly, instead, you are using the pre-trained weights from this network. Freeze all the layers before training the network, except the FC layers. Consequently, the gradients will not be calculated for the layers except the ones that you are going to update (FC layers) over the pre-trained weights. However, since the number of classes is different for our dataset, you should modify the last layer of the network, which the probabilities will be calculated on. Therefore, the weights will be randomly initialized for this layer.

1. What is fine-tuning? Why should we do this? Why do we freeze the rest and train only FC layers? Give your explanation in detail with relevant code snippet. [5 pts]
2. Explore training with two different cases; train only FC layers and freeze rest, train last two convolutional layers and FC layers and freeze rest. Tune your parameters accordingly and give accuracy on validation set and test set. Compare and analyze your results. Give relevent code snippet. [20 pts]
3. Plot confusion matrix for fine-tuned model and analyze results. [5 pts]
4. Compare and analyze your results in Part-1 and Part-2. Please state your observations clearly and precisely. [10 pts]

What to Hand In

Your submission format will be:

- README.txt *(give a text file containing the details about your implementation, how to run your code, the organization of your code, functions etc.)*
- code/ *(directory containing all your code)*
- report.pdf

Archieve this folder as **b<studentNumber>.zip** and submit to `https://submit.cs.hacettepe.edu.tr`.

Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.

References

[1] `https://medium.com/techiepedia/binary-image-classifier-cnn-using-tensorflow-a3f5d6746697`
[2] `https://www.kaggle.com/datasets/itsahmad/indoor-scenes-cvpr-2019`