Third Assignment

Shahid Beheshti University - Bachelor’s Program

Artificial Neural Networks - Winter 2024

Theoretical Exercises

Exercise 1

What is gradient accumulation? When should we use this technique? How to perform this in PyTorch?

Exercise 2

Describe the backpropagation details in the convolutional layers. (For a better understanding, check out this link)

Exercise 3

What are the benefits of the pooling layers? What are the drawbacks of the pooling layers? How do max pooling and average pooling differ in their operations? When might one be preferred over the other? Can you use these layers frequently?

Exercise 4

Discuss the concept of transfer learning in the context of CNNs. How does it allow leveraging pre-trained models for tasks with limited training data? Provide examples of popular pre-trained CNN models and the tasks they are commonly used for.

Exercise 5

True/False, Explain the reason:

- Face verification requires comparing a new picture against one person's face, whereas face recognition requires comparing a new picture against K person's faces.

- In order to train the parameters of a face recognition system, it would be reasonable to use a training set comprising 100,000 pictures of 100,000 different persons.

❖ You train a CNN on a dataset with 100 different classes. You wonder if you can find a hidden unit that responds strongly to pictures of cats. (I.e., a neuron so that, of all the input/training images that strongly activate that neuron, the majority are cat pictures.) You are more likely to find this unit in layer 4 of the network than in layer 1.

Practical Exercises

Architectural Heritage Elements Image64 Dataset

The Architectural Heritage Elements Dataset (AHE) is an image dataset for developing deep learning algorithms and specific techniques in the classification of architectural heritage images. This dataset consists of 10235 images classified into ten categories:

- Altar: 829 images
- Apse: 514 images
- Bell tower: 1059 images
- Column: 1919 images
- Dome (inner): 616 images
- Dome (outer): 1177 images
- Flying buttress: 407 images
- Gargoyle (and Chimera): 1571 images
- Stained glass: 1033 images
- Vault: 1110 images

Objective

The objective of this project is multi-fold:

- **CNN Model Training**: The first objective is to build and train a Convolutional Neural Network (CNN) that can classify images from the Architectural Heritage Elements Image64 Dataset. This involves preprocessing the data, designing the CNN architecture, and training the model using a suitable optimization algorithm.

- **Deconvolution Visualization**: The second objective is to use deconvolution techniques to visualize what the trained CNN model sees when it processes an input image. This involves implementing deconvolution operations in the CNN and visualizing the feature maps produced by the model. This will help us understand how the model interprets the input images and which features it finds crucial for classification.

- **Image Generation for Specific Classes**: The third objective is to use the trained model model in previous sections and manipulate it so that, given a specific class, it generates an image corresponding to that class.

Dataset

The dataset used in this project is the Architectural Heritage Elements Image64 Dataset. It consists of 10235 images classified into ten categories. The categories represent different architectural elements such as Altar, Apse, Bell tower, etc. You can download this dataset from this link.

Methodology

- **Data Preprocessing**: The images will be preprocessed for the model. This may include resizing the images, normalizing the pixel values, etc.
- **Model Building**: A CNN model will be built using libraries such as TensorFlow or PyTorch. The model should include several convolutional and pooling layers, followed by dense layers.
- **Training**: The model will be trained on a portion of the dataset. Various strategies like data augmentation, dropout, etc., can be used to improve the model's performance and prevent overfitting.
- **Evaluation**: The model's performance will be evaluated on a separate test set. Metrics such as accuracy, precision, recall, and F1-score can be used for this purpose.
- **Optimization**: Based on the evaluation results, the model can be further optimized by tuning hyperparameters, modifying the architecture, etc.

Expected Outcome

At the end of this project, students should have a working CNN model that can classify images of architectural elements with reasonable accuracy. They should also gain practical experience in handling image data, building and training CNN models, and optimizing these models for better performance.

References

- Architectural Heritage Elements Image64 Dataset