Assignment 2

Task 1: (marks=15)

Implement a basic CNN model from scratch using TensorFlow for image classification. In our dataset, we have 3 classes of digits which are 0, 1 and 2.

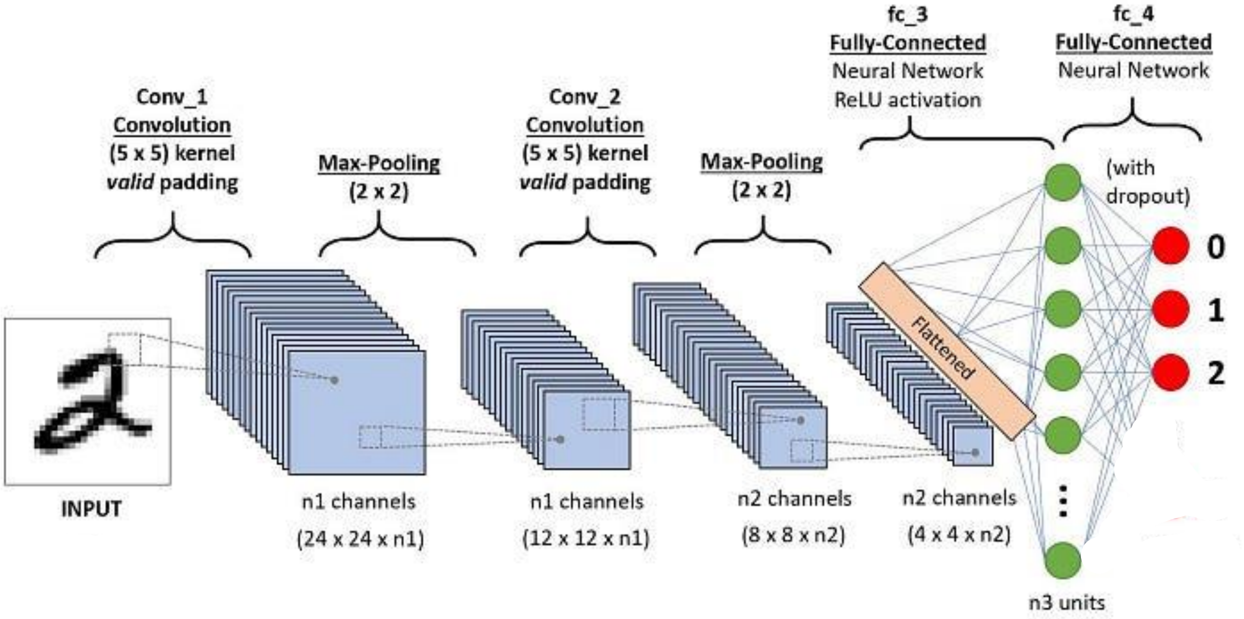
What you need to do is:

1. Split the dataset into training and validation set. 80% for training and the remaining 20% for validation. You should do this split in code. You are only provided with a single folder which is having all the 3 classes. Each sub-folder in the Hand_written_digits folder is considered a separate class/category by the TensorFlow. TensorFlow allows you to do the split. Check TensorFlow official documentation to see how it is done. Images in the dataset are gray scale of size (100x100) pixel. The dataset provided to you is in the following format:
   - ⇨ Hand_written_digits
     - o 0_digits
     - o 1_digits
     - o 2_digits
2. Train model on the provided dataset. Do the data augmentation suitable for the dataset provided. Be careful while doing the augmentation as all the augmentations are not suitable for our dataset like doing the vertical flip or horizontal flip on digit 2 changes it to something else.

Train the model on the provided hand-written digits dataset and report the accuracy achieved after 10 epochs of training. You can increase the number of epochs as per your choice but be careful of model overfitting. This task will be marked based on validation accuracy so do play around with the model architecture to achieve the best possible accuracy.

Following is the model architecture. You can change the model architecture as per your creativity.

Assignment 2



Task 2: (marks=15)

Now this time do Task 1 again by fine-tuning the pre-trained model of your choice. Contents of week 6 and week 7 are focused on the concept of transfer learning so the main objective of this task is to solidify and reinforce your core concepts of transfer learning and how to fine-tune a pre-trained model trained on different dataset for your own custom dataset. Observe the accuracy of the model and compare it with the accuracy of the model in task 1.

Task 3: (marks=20)

Building an Image Retrieval System using Convolutional Neural Networks

Importance of CNN in Image Searching and Retrieval

Assignment 2

Convolutional Neural Networks (CNNs) are playing a pivotal role in revolutionizing image searching and retrieval systems. CNNs have become a fundamental technology in computer vision and image processing due to their ability to learn and extract complex visual features from images. Here's why CNNs are crucial in image searching and retrieval systems:

1. **Feature Extraction:** CNNs excel at automatically extracting hierarchical features from images. They can identify edges, shapes, textures, and even high-level concepts in an image. These learned features are essential for comparing and retrieving similar images efficiently.

2. **Spatial Hierarchies**: CNNs capture spatial hierarchies of features, which means they understand the relationship between different parts of an image. This knowledge is vital for recognizing similarities in images even when they have different scales, rotations, or perspectives.

3. **Semantic Understanding**: Advanced CNN architectures can even encode semantic understanding of images. They can recognize objects, scenes, and context, enabling more meaningful image retrieval.

Google Lens and Advanced CNNs for Image Searching

Google Lens is a real-world example of how Advanced CNNs are utilized for image searching and retrieval. Google Lens is an image recognition technology that allows users to search for information by taking pictures of objects or scenes. It uses Advanced CNNs to perform the following tasks:

- **Image Recognition:** Google Lens can recognize various objects, animals, plants, and landmarks by analyzing the images captured through a smartphone camera. It does this by using CNN-based models trained on extensive datasets.

- **Text Extraction:** It can extract and understand text from images, making it easy to translate, copy, or search for information from printed or handwritten text.

- **Shopping and Information Retrieval**: Users can point their camera at products or objects, and Google Lens provides information about those items, including shopping links, reviews, and related content.

Assignment 2

Task 3 Coding part for you:

Now that you understand the importance of CNNs in image searching and retrieval, it's time to apply your knowledge and create your first image retrieval system. Follow the instructions below to complete this assignment:

1. Write a Python program that allows users to select a query image from folder "***query_images***" and retrieve the top N similar images from a local folder named "***images_database***". Put N=4 for this task so the system must spit out 4 similar images from the given folder closely similar to your query image. Your program should use a pre-trained CNN model (e.g., VGG16 or ResNet) for feature extraction and a similarity metric (e.g., Euclidean distance) for retrieval.

2. Ensure that your program can handle images in various formats (e.g., JPG, PNG, JPEG).

3. Provide clear instructions on how to run your program and demonstrate its functionality using sample query images.

This coding task allows you to apply your understanding of CNN-based feature extraction to build a practical image retrieval system, like the technologies used by Google Lens.

Important Instructions:

Please note that you must solve all the tasks as all tasks are mandatory. You will be required to use Google Colab for solving this assignment. To speed up the training of your models, Google Colab provides limited access to GPU so you can use that for quick training. Once all the tasks are solved, you must make a public repository on Git Hub and push your assignment notebook to it. You must write a readme file for your repository explaining all your tasks and the approach you adopted to solve the problems. You will be delivering/uploading to LMS a word file having a link to your Git Hub repository containing all the tasks. The word file should be named as "***assignment_2_your_name".*** You must follow the best practices of how to write an attractive readme file for your code. If you are stuck and need help, you can contact me ***(your Teaching Assistant)*** on WhatsApp, I will try to resolve your issues.

Good luck with your assignment!