CNN Fundamentals Assignment Questions

1. Explain the basic components of a digital image and how it is represented in a computer. State the differences between grayscale and color images.

A digital image is composed of a grid of tiny units called pixels. Each pixel contains information about intensity or color. In computers, these pixels are stored as numerical values in a 2D or 3D matrix format.

In a **grayscale image**, each pixel holds a single value typically ranging from 0 to 255, where 0 represents black, 255 represents white, and values in between represent varying shades of gray. Grayscale images are stored as 2D arrays.

In contrast, a **color image** is usually represented in the **RGB format**—Red, Green, and Blue channels. Each pixel in a color image contains three values (one for each channel), also ranging from 0 to 255. These three values are combined to form a specific color. Thus, color images are stored as 3D arrays with dimensions: height × width × 3.

The key differences are: grayscale images are simpler (single channel), consume less memory, and are easier to process. Color images, though more complex, contain richer visual information and are necessary for tasks involving color-based object detection or classification.

2. Define Convolutional Neural Networks (CNNs) and discuss their role in image processing. Describe the key advantages of using CNNs over traditional neural networks for image-related tasks.

Convolutional Neural Networks (CNNs) are a class of deep learning models designed specifically for working with grid-like data such as images. CNNs extract and learn patterns from images by applying filters that highlight important features like edges, textures, shapes, and objects.

In image processing, CNNs play a critical role by automatically identifying spatial hierarchies of features—starting from simple edges in early layers to complex shapes in

deeper layers. This enables models to understand visual content in a scalable and efficient manner.

Compared to traditional fully connected neural networks, CNNs offer several advantages. They require fewer parameters due to weight sharing in convolutional layers, which makes them more efficient and less prone to overfitting. CNNs also maintain the spatial structure of the image, making them better suited for visual recognition tasks. Additionally, they can generalize well to images with varying positions and orientations of objects.

3. Define convolutional layers and their purpose in a CNN. Discuss the concept of filters and how they are applied during the convolution operation. Explain the use of padding and strides in convolutional layers and their impact on the output size.

A convolutional layer is the fundamental layer in a CNN that performs the convolution operation. Its purpose is to scan the input image with small learnable filters (also called kernels) to extract local features.

Each **filter** is a small matrix (like 3×3 or 5×5) that slides over the image. At each step, it performs an element-wise multiplication with the underlying pixel values and sums the result to produce a single value. This process creates a new feature map highlighting specific patterns such as edges or textures.

Padding refers to adding extra pixels (usually zeros) around the image border. This is used to control the output size of the convolution. Without padding, the output shrinks in size with each convolution. Padding allows the filter to cover the edge pixels and helps preserve the original spatial dimensions.

Strides determine how far the filter moves with each step. A stride of 1 means the filter moves one pixel at a time, while a higher stride skips pixels and results in a smaller output.

Padding helps retain image size, and stride controls how much the image is downsampled. Together, they influence the balance between computational cost and feature resolution.

4. Describe the purpose of pooling layers in CNNs. Compare max pooling and average pooling operations.

Pooling layers are used in CNNs to reduce the dimensions of feature maps, which helps in decreasing computational load and controlling overfitting. Pooling also helps the model become more robust to variations in the input image, such as slight shifts or distortions.

In a pooling operation, a small window (like 2×2) slides over the feature map and applies a summary operation within that window.

Max pooling selects the highest value in each window. This captures the most prominent feature in that region and helps preserve strong activations. It's widely used due to its ability to retain key features effectively.

Average pooling, on the other hand, computes the average of all values in the window. This results in a smoother, more generalized feature map but may lose finer details.

The key idea is that pooling helps to downsample feature maps while keeping the most important information, aiding in faster and more effective learning in deep CNN architectures.