Flattening to a 1D Array

Pre-work: Computer Vision

Agenda

- What is Flattening?
- Why is it required?

What is Flattening?

Flattening

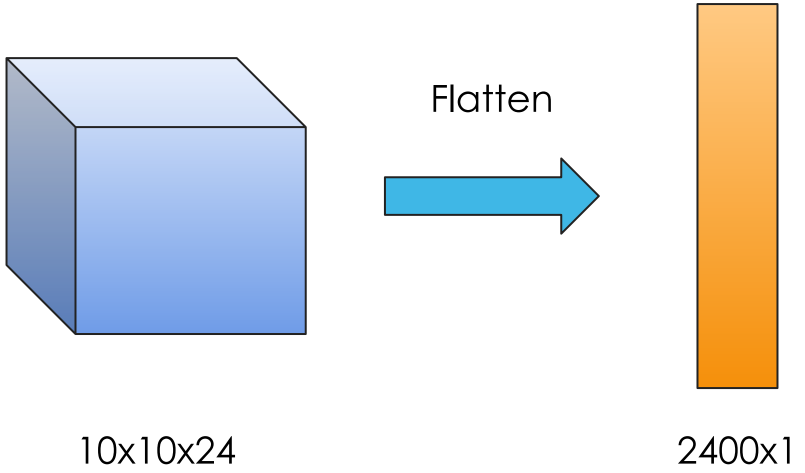
- **After completing multiple iterations of the convolution and pooling** operations on the image, we get feature maps of reduced dimensionality, also having multiple channels.
- This is a **3D Array** of size (Width, Height, Channels).
- The number of units or values in this 3D array will be Units (U) = **W x H x C**
- Our task here is to **flatten this 3D array into a 1D array of size (WxHxC, 1)** = (Units, 1)

Procedure for Flattening a 3D array:

1. **We read row-by-row** for each channel starting from the first channel.
2. **Each row is appended** to the 1D flattened array.

Flattening

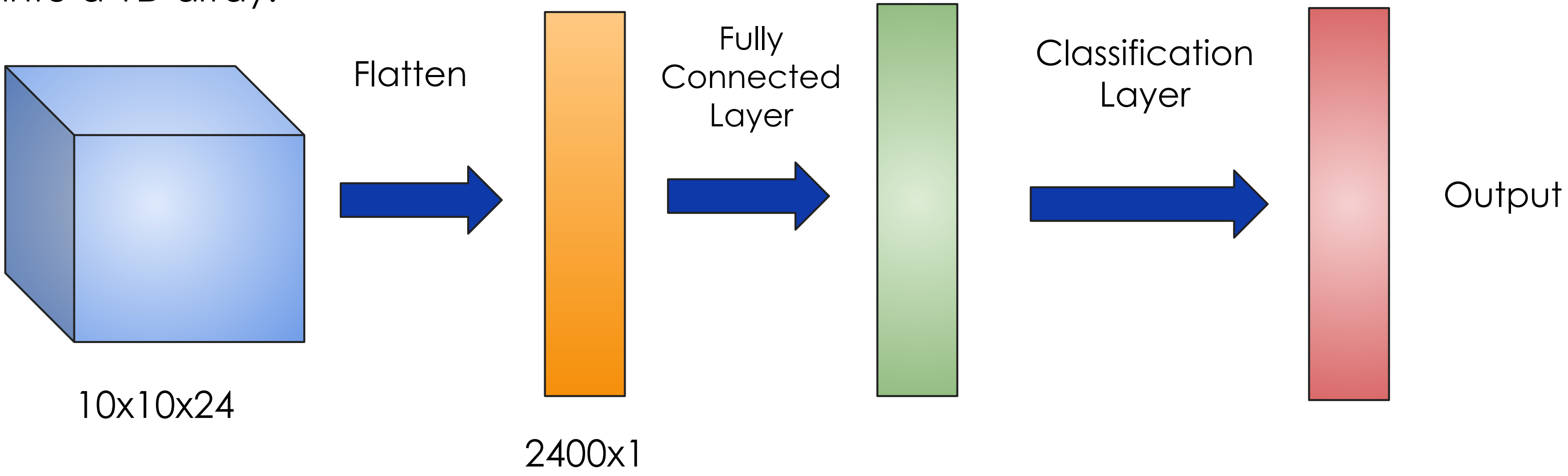
- **For example:**
  - The initial size after the Convolution and Pooling Operations: **10x10x24**
  - Here we have height and width as 10, and 24 channels.
  - Size after Flattening this 3D Matrix: (2400, 1)



Why is it required?

Requirement

- Since the output from the Convolution and Pooling operations is a 3D array, we can't directly use them in the fully connected layers of Neural Networks, which only accept 1D arrays of values.
- **This is the main reason** we have to **flatten** the output from Convolutions and Pooling into a 1D array.



Thank You