Overfitting in CNN

Dr. Sagarika B

What is overfitting ?

High Training Accuracy, Low Validation Accuracy: The model performs significantly better on the training data compared to the validation or test data.

Increasing Training Accuracy, Decreasing Validation Accuracy: As training progresses, the training accuracy improves while the validation accuracy stagnates or deteriorates.

High Variance: The model shows large performance fluctuations on different subsets of the data.

Why CNNs might overfit?

- **Complex and deep Model Architecture**
- **Small Training Dataset**
- **Too Many Epochs**
- **Imbalanced Data**
- **Noise in the Data**

How can we resolve this issue?

- Add more patterns in data
  - Dropout
  - **Image augmentation**

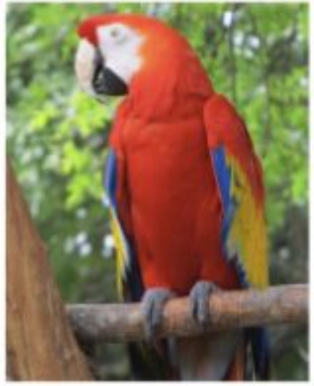
- Reduce the number of learnable parameters
  - Decrease the number of neurons and hidden layers
  - Batch normalization
  - Regularization

Image augmentation

Image augmentation is a technique used in computer vision to increase the diversity of images available for training models without actually collecting new images.

It involves applying various transformations to existing images to create new, modified versions.

Original image

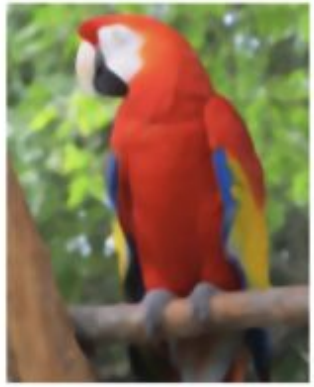


augmentation

Horizontal Flip

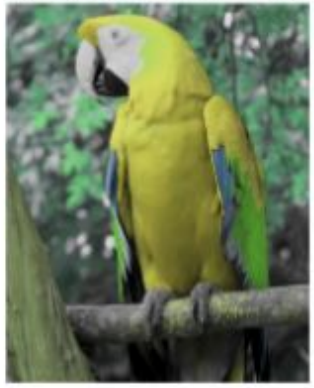
Crop

Median Blur



Contrast

Hue / Saturation / Value



Gamma

How do we perform this in code ?

Using Data generators

a `DataGenerator` is a powerful utility that allows for efficient data loading and real-time data augmentation during the training of machine learning models. It provides an easy way to preprocess, augment, and batch images dynamically, which is especially useful for large datasets where loading the entire dataset into memory is impractical.

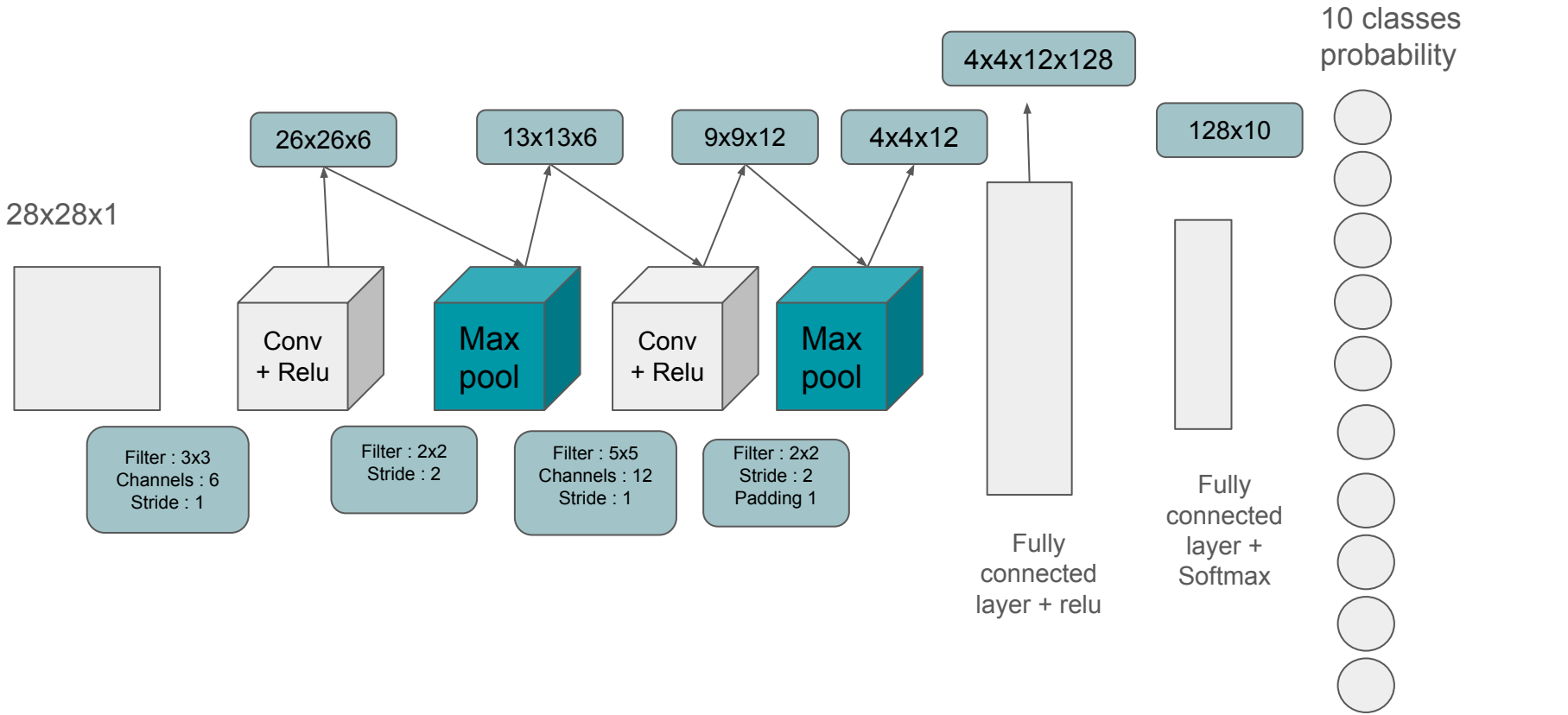
Image datagenerator

Key Features of `ImageDataGenerator`

1. **Data Augmentation**: Apply random transformations to the images to create variations that can improve the robustness of the model.
2. **Data Normalization**: Scale pixel values to a specified range (e.g., 0-1 or -1 to 1).
3. **Batch Processing**: Efficiently load and preprocess images in batches, reducing memory usage and speeding up training.
4. **Real-Time Processing**: Generate augmented images on the fly during model training, ensuring diversity in the training data.

Let’s do apply it in our Problem !!

https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html



Weights= conv1: 60 conv2:312 Dense1:24704 Dense 2:1290 = 26366 (total)