

Typical CNN Architecture Flow:

1. **Input Layer:** Accepts raw input data (e.g., images).
2. **Convolutional + Activation Layers:** Extract features.
3. **Pooling Layer:** Downsample feature maps.
4. (Repeat Convolution + Pooling layers as needed.)
5. **Flatten Layer:** Prepares data for fully connected layers.
6. **Fully Connected Layer:** Maps features to predictions.
7. **Output Layer:** Produces the final result (e.g., class probabilities).

Convolutional Layer:

- **Purpose:** Extracts features from the input using convolutional filters.
- **Key Attributes:**
 - **Filters (Kernels):** Small matrices that slide over the input to detect patterns.
 - **Stride:** Step size of the filter movement.
 - **Padding:** Determines how the edges of the input are handled (e.g., "same" or "valid").
- **Output:** Feature maps.

Activation Layer

- **Purpose:** Introduces non-linearity into the network.
- **Common Activation Functions:**
 - ReLU (Rectified Linear Unit)
 - Sigmoid
 - Tanh
- **Location:** Typically applied after each convolutional layer.

Pooling Layer

- **Purpose:** Reduces the spatial dimensions of the feature maps while preserving important information.
- **Types:**
 - **Max Pooling:** Retains the maximum value from a region.
 - **Average Pooling:** Retains the average value from a region.
- **Key Attributes:**
 - **Pool size:** Size of the pooling filter.
 - **Stride:** Step size of the pooling filter

Fully Connected (Dense) Layer

- **Purpose:** Maps the learned features to the output classes or regression values.
- **Structure:** Each neuron in this layer is connected to every neuron in the previous layer.
- **Use Case:** Typically placed at the end of the CNN for classification or prediction

Dropout Layer

- **Purpose:** Prevents overfitting by randomly setting a fraction of input units to 0 during training.
- **Dropout Rate:** The proportion of neurons to drop (e.g., 0.5 for 50%).

Flatten Layer

- **Purpose:** Converts 2D feature maps into a 1D vector, making them compatible with fully connected layers.
- **Location:** Between the convolutional/pooling layers and fully connected layers.