NANDHA ENGINEERING COLLEGE, AUTONOMOUS, ERODE -52

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ASSIGNMENT -2

ACADEMIC YEAR: 2024-2025

REGISTER NUMBER   :   22CS065,22CS066

NAME   :   PRANESH KUMAR S , PRAVIN KUMAR S

COURSE CODE & NAME :   22CSX01 & DEEP LEARNING

CLASS /SEM   :   III- B.E(CSE) / V

TEAM – 4

TOPIC	MARKS
You are tasked with building a CNN for real-time object detection in autonomous vehicles. What components would be essential in your CNN architecture?	

**Student signature**   **Faculty Signature**

1 . You are tasked with building a CNN for real-time object detection in autonomous vehicles. What components would be essential in your CNN architecture?

CONVOLUTIONAL NEURAL NETWORK:

CNNs are a type of deep learning algorithm that uses linear algebra principles to identify patterns in images. They are often used for image recognition and object detection, but can also be used for other tasks like natural language processing and recommendation engines.
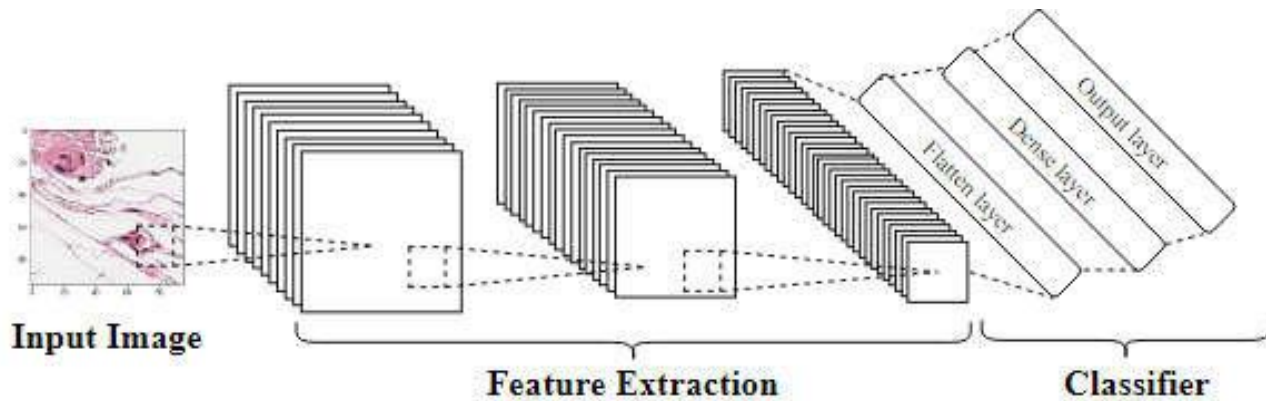
WORKING OF CNN:

- CNNs process input images through layers of filters that detect low-level features like edges, gradually building up to more complex shapes and textures as the network deepens.

- Convolutional layers apply these filters across the image, allowing the network to learn spatial hierarchies and essential visual patterns needed for object recognition.

- Activation functions like ReLU introduce non-linearity, enabling the CNN to capture intricate patterns beyond linear combinations of pixels.

- Pooling layers downsample feature maps, reducing computation while retaining important features and making the model invariant to small translations in the input.

- Fully connected layers towards the end combine the extracted features into a classification or detection output, assigning probabilities to potential classes or bounding boxes.

- During training, the CNN adjusts filter weights to minimize prediction errors, refining its feature extraction ability for tasks like object detection, segmentation, or recognition.

ARCHITECTURE OF CNN:

Convolutional layer

Max pooling layer

Dense layer

Input layer

IMAGE

Output layer
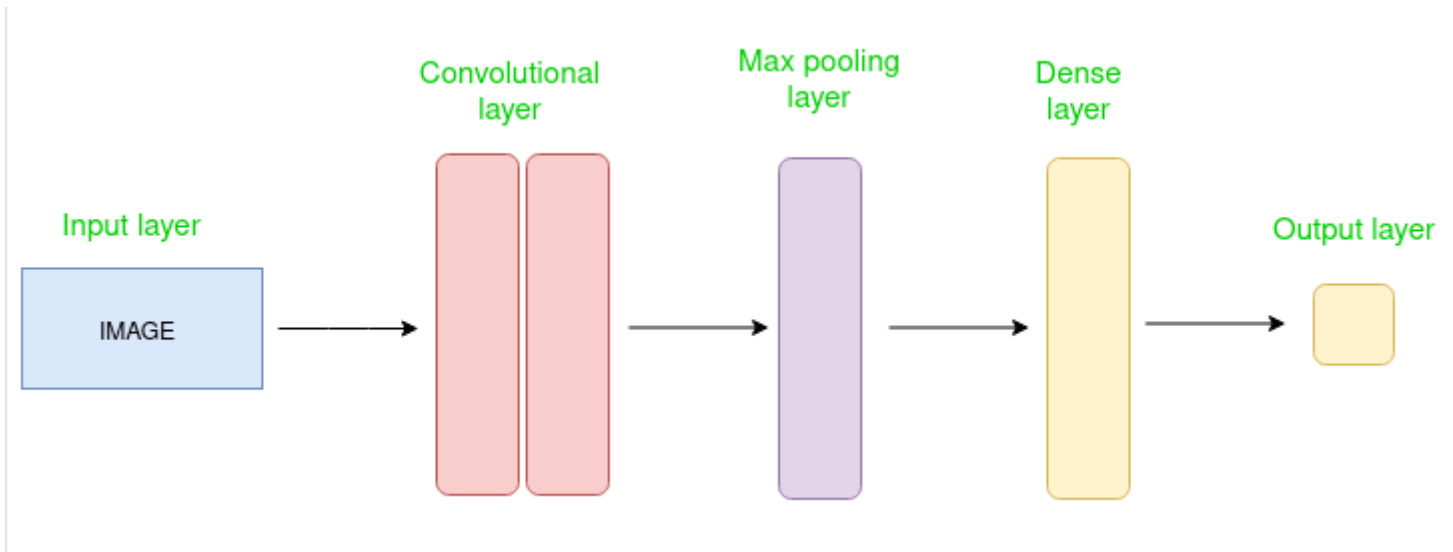
NEED OF OBJECT DETECTION IN AUTONOMOUS VEHICLES:

- Real-time object detection is essential for autonomous vehicles to identify and respond to dynamic obstacles like pedestrians, cyclists, and other vehicles, ensuring safe navigation.

- It enables quick recognition of traffic signs and signals, allowing the vehicle to adjust speed, stop, or follow traffic rules, which is crucial for compliance and safety.

- Real-time detection helps in accurately identifying lane markings, boundaries, and road edges, enabling the vehicle to stay within lanes and avoid off-road or incorrect path deviations.

- It provides instant environmental awareness, enabling the vehicle to make split-second decisions in complex scenarios, such as merging lanes, avoiding sudden obstructions, or stopping for emergencies.

- Real-time detection minimizes latency in decision-making, which is critical to maintaining a smooth and safe driving experience in ever-changing road conditions.

HOW CNN HELPS IN DETECTING OBJECTS:

- Convolutional layers apply learnable filters across the input image, detecting low-level features such as edges and textures by convolving small receptive fields over the image matrix.

- As layers deepen, CNNs capture more abstract features, with each layer learning spatial hierarchies of features that evolve from edges to shapes and finally complex objects.

- ReLU activation functions introduce non-linearity after each convolution, allowing the CNN to model intricate data patterns essential for distinguishing diverse objects in varying conditions.

- Pooling layers, like max pooling, downsample feature maps by retaining only the most significant values in each region, making the network invariant to minor spatial translations.

- The fully connected layers at the network's end consolidate the extracted features, mapping them to probability distributions over predefined classes for object classification or bounding box predictions.

- During backpropagation, CNNs optimize filter weights based on classification or localization error, improving their ability to accurately detect and locate objects across images.

COMPONENTS NECESSARY FOR BUILDING A CNN:

Input Layer

- Accepts raw image data from the vehicle's cameras or sensors and prepares it for processing by the network.
- It normalizes the pixel values, typically scaling them to a range between 0 and 1 to maintain consistency during training.
- The images are resized to a fixed size so that the network can handle varying image dimensions uniformly.
- Once processed, the image is passed onto the convolutional layer for feature extraction and learning.

Convolutional Layer

- Applies filters (kernels) to the image in order to detect basic features such as edges, textures, and patterns.
- These filters slide across the image, producing feature maps that highlight important patterns at different locations in the image.
- The network detects low-level features like lines, curves, and corners that are crucial for recognizing objects in the image.
- As the layers deepen, the network learns increasingly complex features, helping it recognize higher-level object shapes and parts.

Activation Layer (e.g., ReLU)

- Introduces non-linearity into the network by applying activation functions like ReLU (Rectified Linear Unit).
- ReLU sets all negative values in the feature maps to zero and keeps positive values, allowing the network to focus on significant features.
- By adding non-linearity, the network can learn more complex and diverse patterns needed to distinguish different objects.
- This process also helps the model converge faster during training and avoids problems like vanishing gradients in deep networks.

Pooling Layer

- Reduces the spatial dimensions (height and width) of feature maps, making the model more computationally efficient.
- Pooling operations, like max pooling, downsample the feature maps, retaining only the most important features for the detection task.
- This operation helps reduce the computational load and memory usage while preserving important spatial hierarchies.
- Pooling also adds translation invariance, meaning the detection is less sensitive to the location of the object in the image.

Dense (Fully Connected) Layer

- Flattens the multi-dimensional feature maps and connects each neuron to every neuron in the previous layer for higher-level processing.
- The dense layer integrates the features learned by the convolutional and pooling layers to make global predictions like object classification and bounding box coordinates.
- It helps the network decide what object is present in the image by analyzing all extracted features together.
- The output from this layer is used to predict the object class and its exact position in the image.

Output Layer

- Provides the final predictions, including the object class probabilities and bounding box coordinates for each detected object.
- The softmax activation function is used for classification, producing probabilities for each object class in the image.
- For object detection, this layer outputs bounding box coordinates (x, y, width, height) for each detected object.
- The final outputs are passed through post-processing techniques like Non-Maximum Suppression (NMS) to eliminate redundant or overlapping detections.