Convolutional Neural Networks (C

Machine Vision

Transforming visual recognition through deep learning.

Instructor: Stephin Rachel Thomas

Feb 05, 2026

Today’s Topics

- Artificial Neural Networks
- Disadvantages of simple ANN for Image classification
- Introduction to CNN
- CNN architecture
- Deep dive into CNN layers
- Application of CNN
- Performance Evaluation Metrics

What are Artificial Neural Networks?

1 **Biological Inspiration**

ANNs are inspired by the structure and function of the human brain, composed of interconnected nodes called neurons.

2 **Learning Through Data**

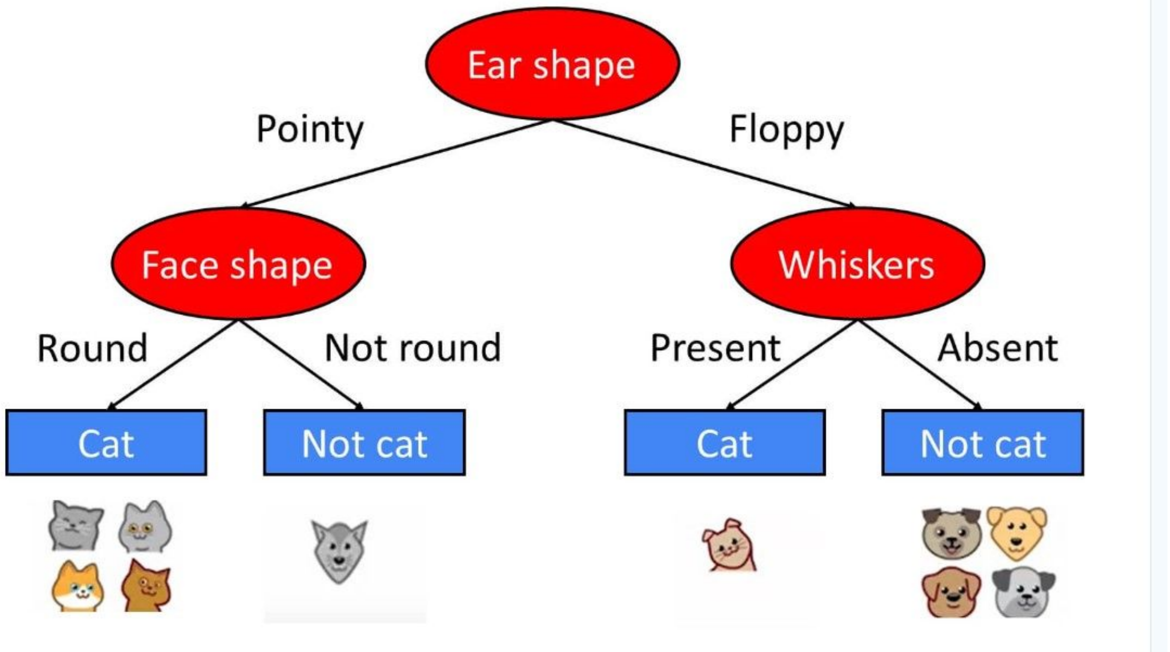
These networks learn by analyzing large datasets, adjusting the connections between neurons to improve their performance.

3 **Pattern Recognition**

ANNs are particularly effective at recognizing complex patterns in data, making them ideal for image classification.

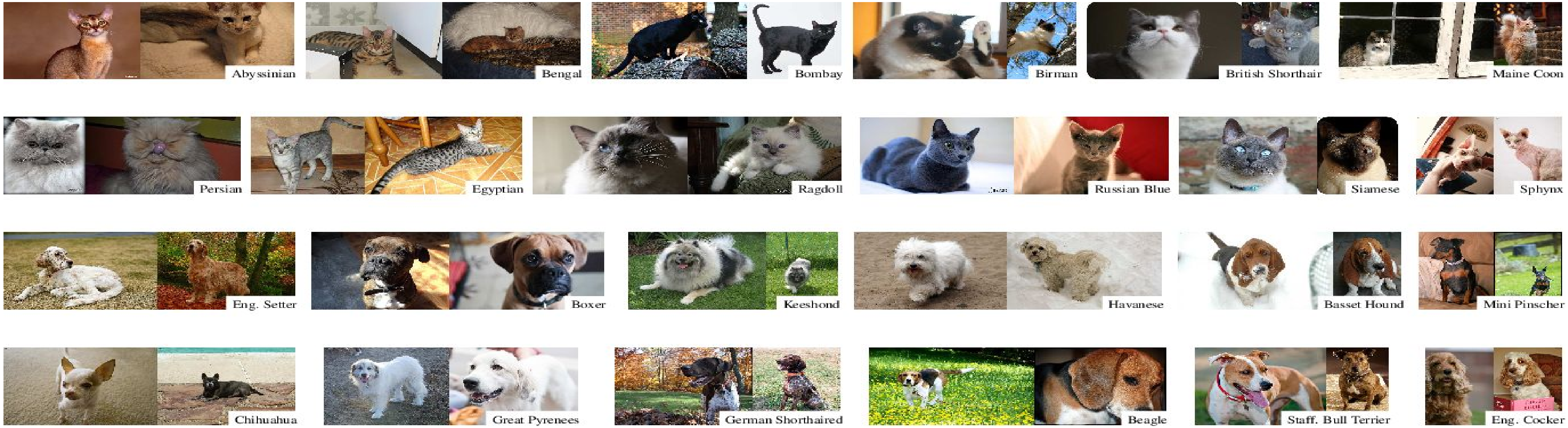
Classification using Traditional Methods

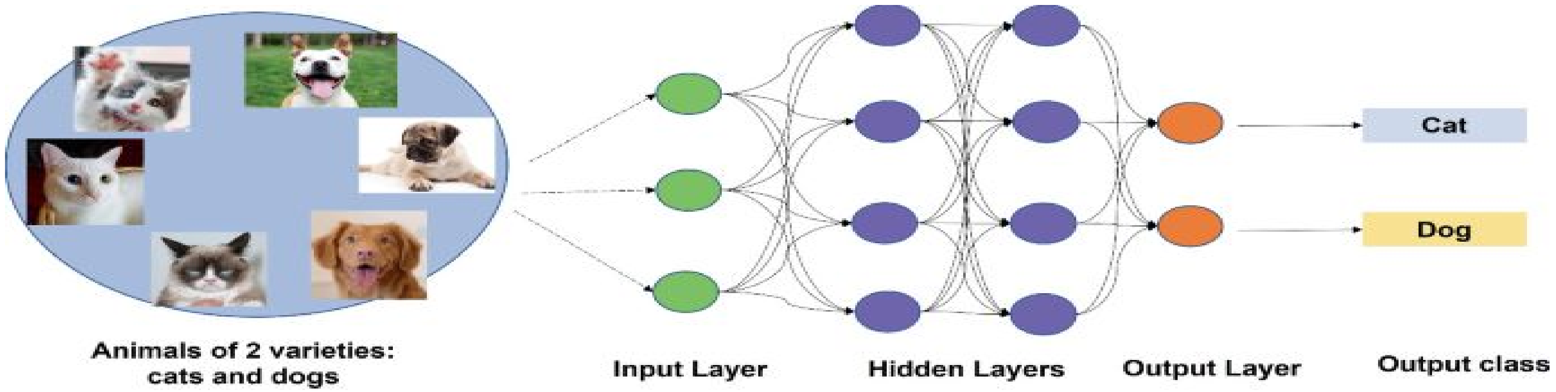
Ear shape	Face shape	Whiskers	Cat/No cat
Pointy	Round	Present	1
Floppy	Nor Round	Present	1
Floppy	Round	Absent	0
Pointy	Nor Round	Present	0
Pointy	Round	Present	1
Pointy	Round	Absent	1
Floppy	Nor Round	Absent	0
Pointy	Round	Absent	1
Floppy	Round	Absent	0
Floppy	Round	Absent	0



Decision-tree method

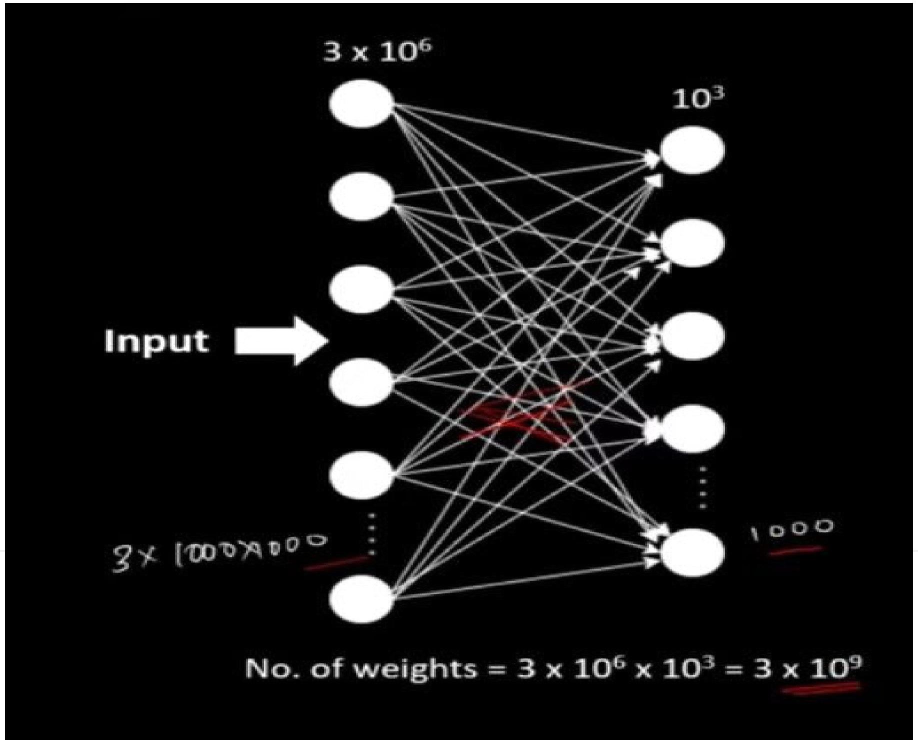
ANN for Image Classification



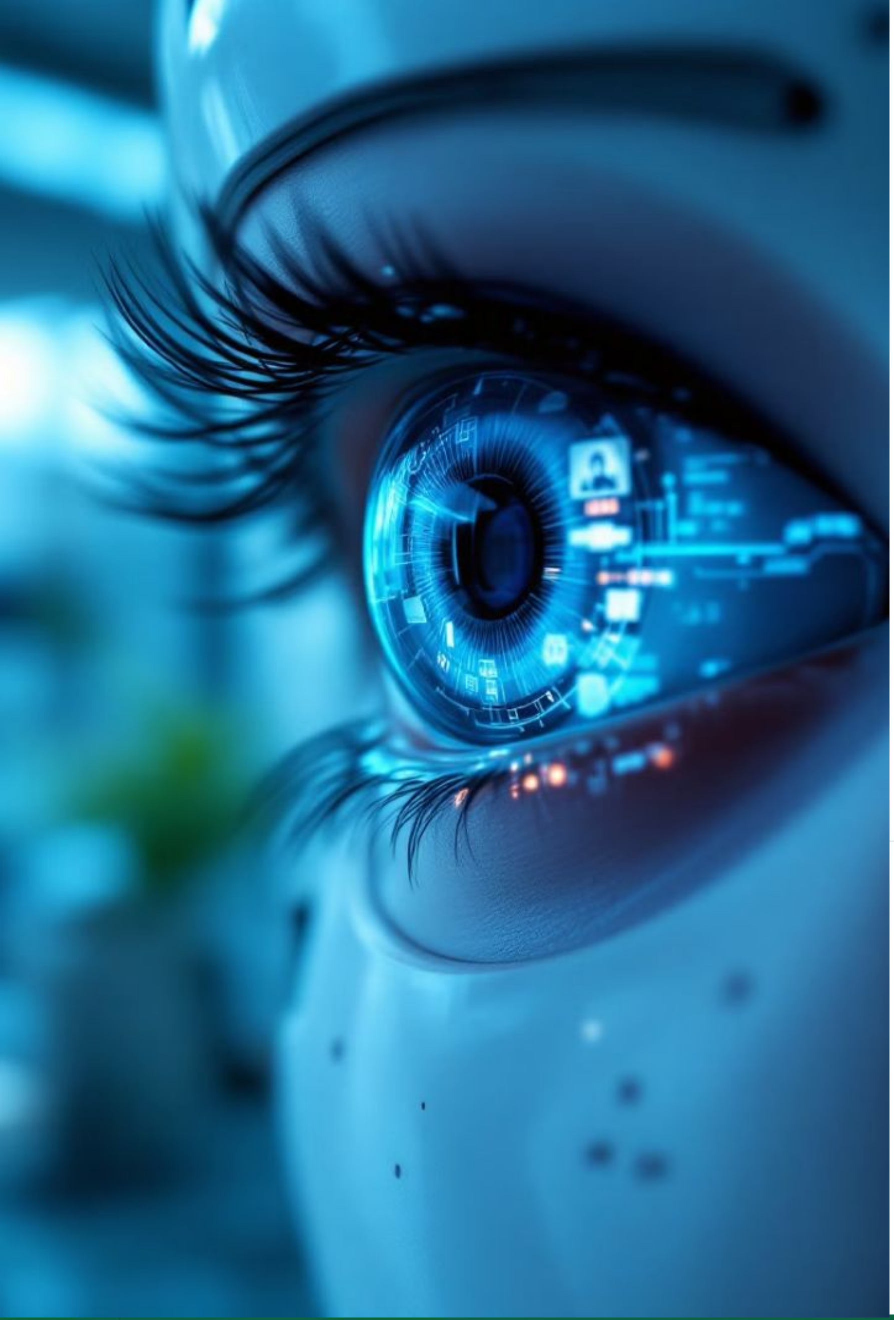


Limitation of ANN for Image Classification

1000 * 1000px



- High compuational cost
- Over-fitting problem
- Longer training time



Convolutional Neural Network (CNN)

1 Definition

A deep learning model designed for processing images to identify patterns and make decisions.

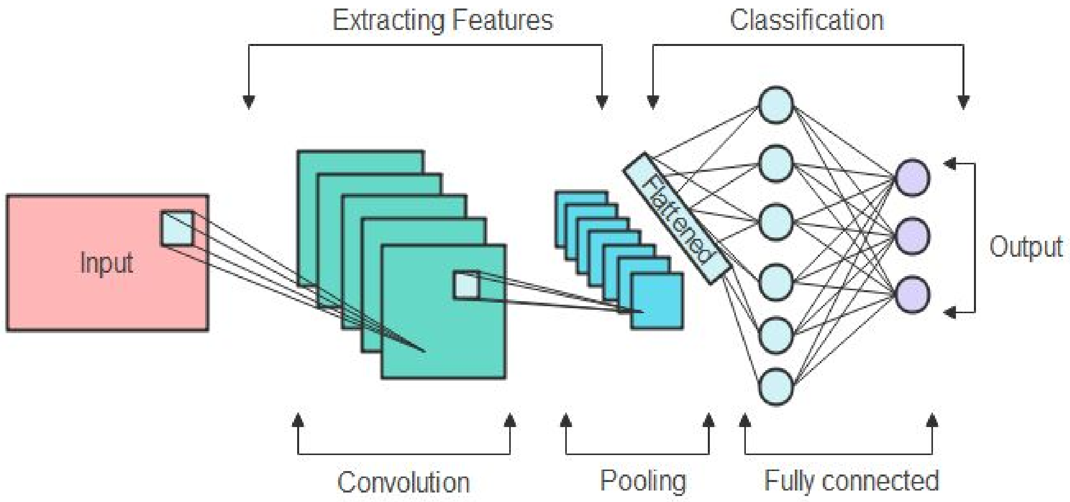
2 Objective

Solve complex visual tasks with deep learning.

3 Benefits

- Handles high-dimensional, structured data like images, videos and audio.
- Hierarchical feature learning.
- Robust to translation of object.

CNN Architecture



CNNs typically consist of an input layer, multiple hidden layers, and an output layer.

The hidden layers include a series of convolutional layers, pooling layers and fully connected layers.

Each layer performs distinct operations: Convolutional layers apply a convolution operation, Pooling layers perform down-sampling, Fully connected layers compute the class scores.

Key Components of CNN

1 Convolutional Layers

Extract spatial features from input images.

2 Pooling Layers

Reduce spatial dimensions, simplify computation.

3 Fully Connected Layers

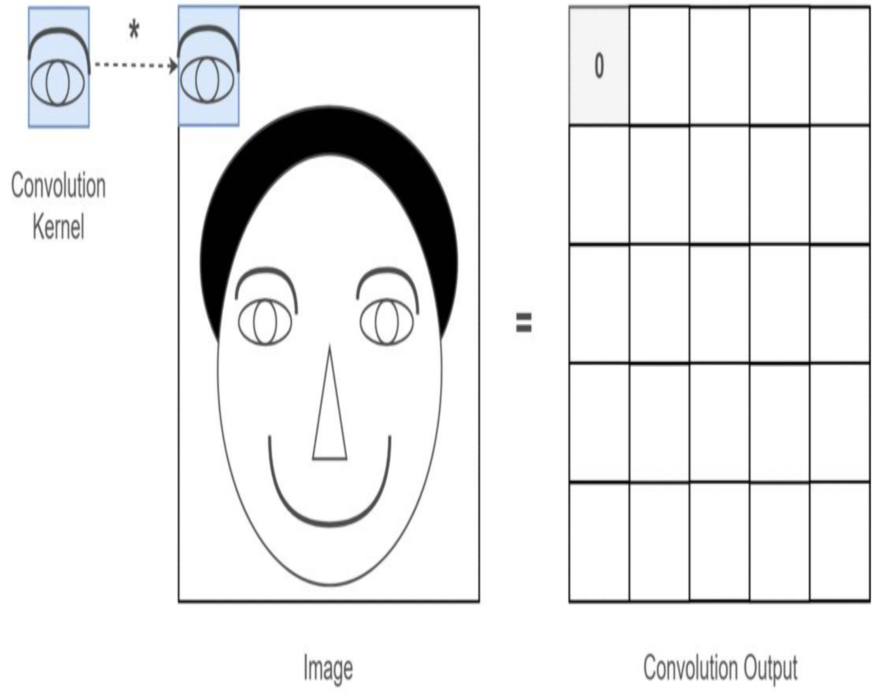
Integrate features for final classification.

Deep Dive into Convolutional Layers

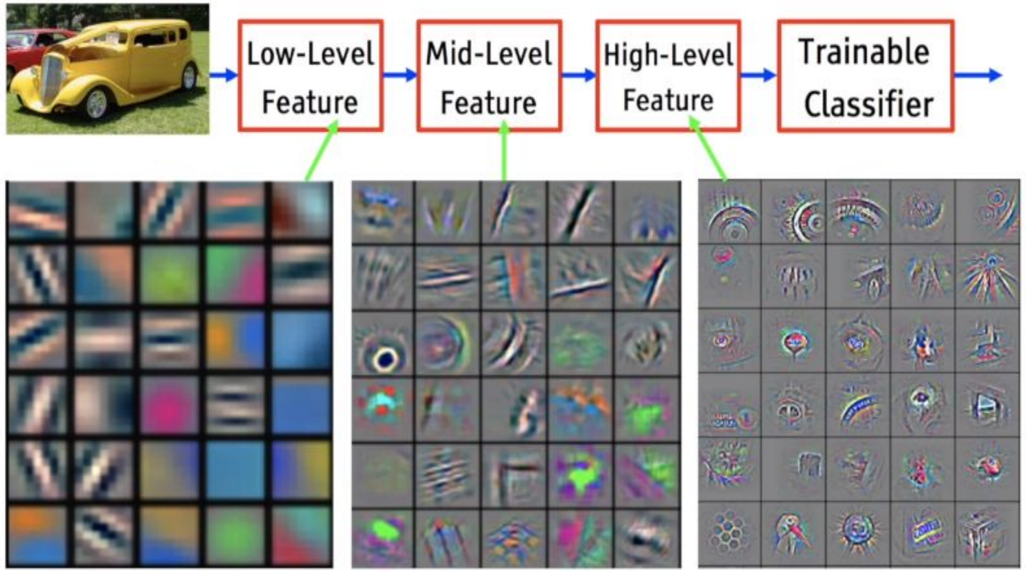
In these layers, small, learnable filters slide over the input data (like images) to extract features such as edges, textures, and shapes.

Each filter in a convolutional layer detects different features, and multiple layers work together to capture increasingly complex aspects of the data.

The convolutional layers thus play a crucial role in feature detection and representation, enabling CNNs to effectively perform tasks like image recognition and classification.



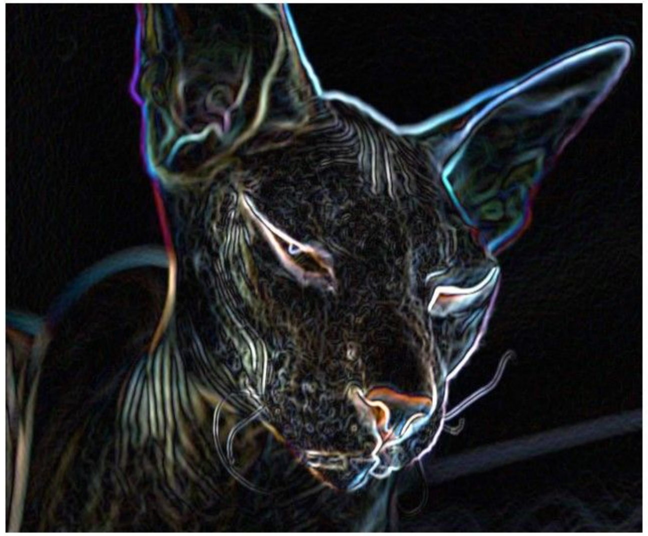
CNN Fundamentals



The basic principle of a Convolutional Neural Network (CNN) is to automatically learn and extract hierarchical features from input data, typically images, through the use of convolutional layers.

Convolutional Layers

Picture of my cat Mew!

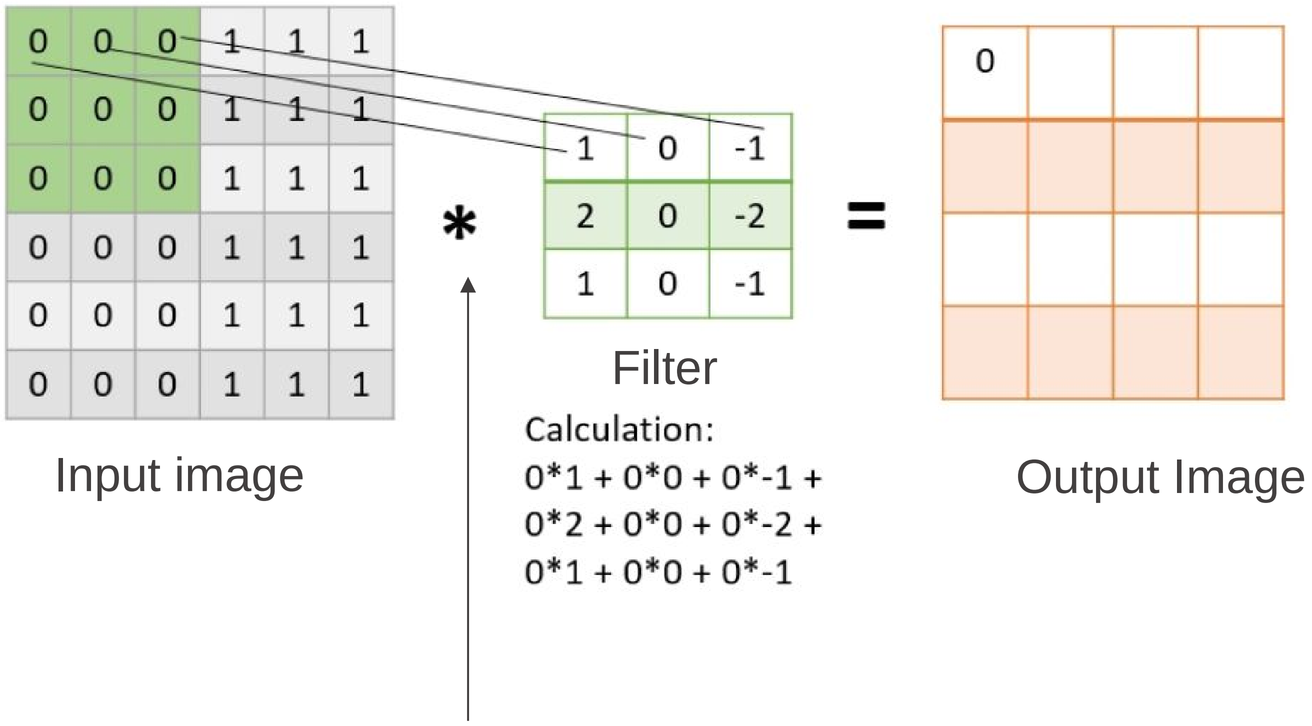


Basic edge filter applied to a RGB image of my cat

Feature Maps

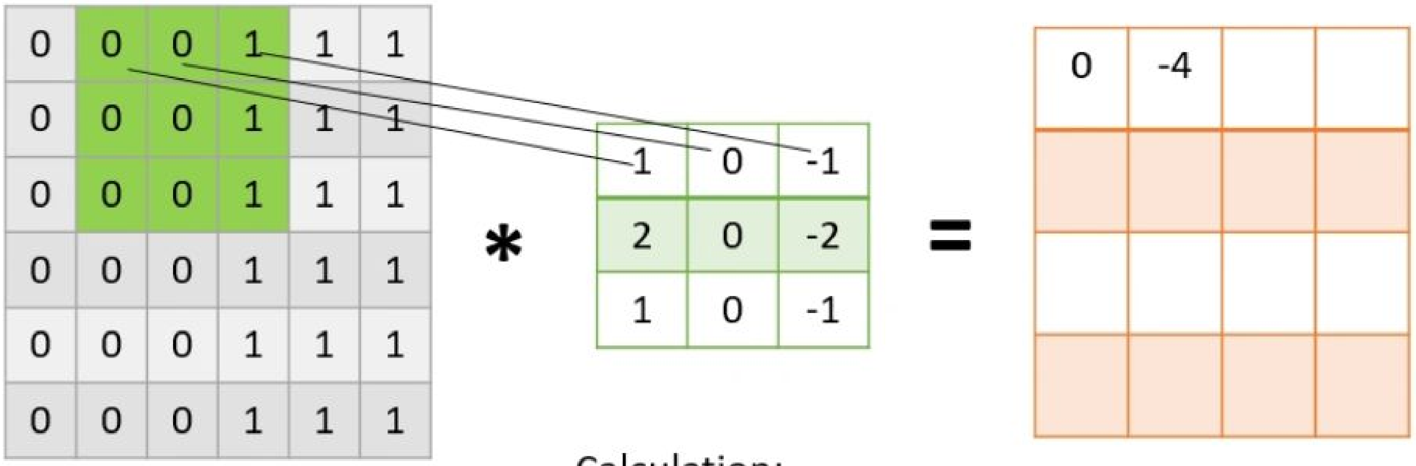
- Convolutional layers help the network focus on only the most important features
- Not all the pixel information in the image is relevant for training the model
- Improves performance and accuracy

Convolution Operation



Convolution
operator

Convolution Operation



Calculation:
0*1 + 0*0 + 1*-1 +
0*2 + 0*0 + 1*-2 +
0*1 + 0*0 + 1*-1

Convolution Operation

$$
\underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}}_{6*6}
*
\underbrace{\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}}_{3*3}
=
\underbrace{\begin{bmatrix} 0 & -4 & -4 & 0 \\ 0 & -4 & -4 & 0 \\ 0 & -4 & -4 & 0 \\ 0 & -4 & -4 & 0 \end{bmatrix}}_{4*4}
$$

Convolutional Layers

The filter size determines the extent of the input data that each filter covers, affecting the granularity of the features detected; smaller filters capture fine details, while larger filters identify broader patterns.

Stride, the step size with which filters move across the input, influences the overlap of receptive fields and the size of the output feature map; larger strides result in smaller, more abstract feature maps.

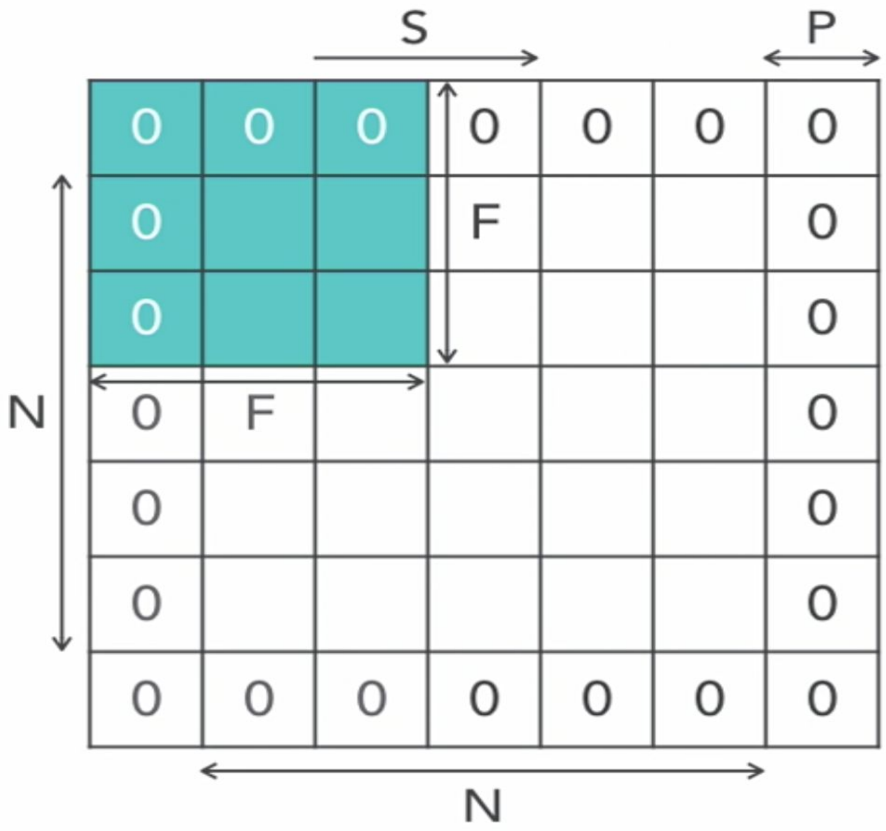
Padding, the addition of zeroes around the input border, allows control over the spatial dimensions of the output, preserving edge information and enabling deeper layers to build a spatial hierarchy of increasingly complex and abstract features.

Convolutional Layer – Output Image Size

The image output size is given by the following

$$\frac{N - F + 2P}{S} + 1$$

F: size of filter
S: stride
N: size of image
P: amount of padding



Pooling Layers

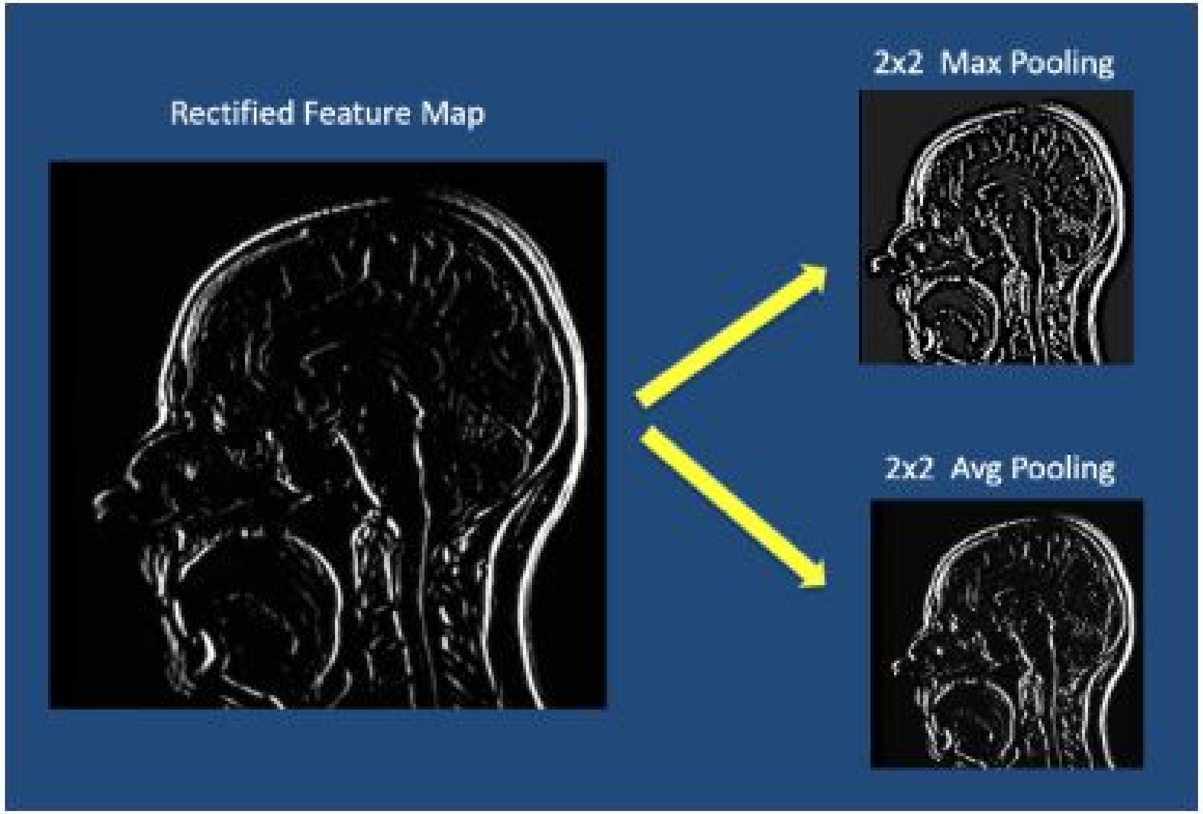
Responsible for reducing the spatial size of the feature maps generated by convolutional layers

By performing operations such as max or average pooling, they down sample the input features, which helps to decrease the computational load and the number of parameters in the network

This reduction also contributes to making the network more tolerant to variations and distortions in the input data, enhancing its ability to generalize.

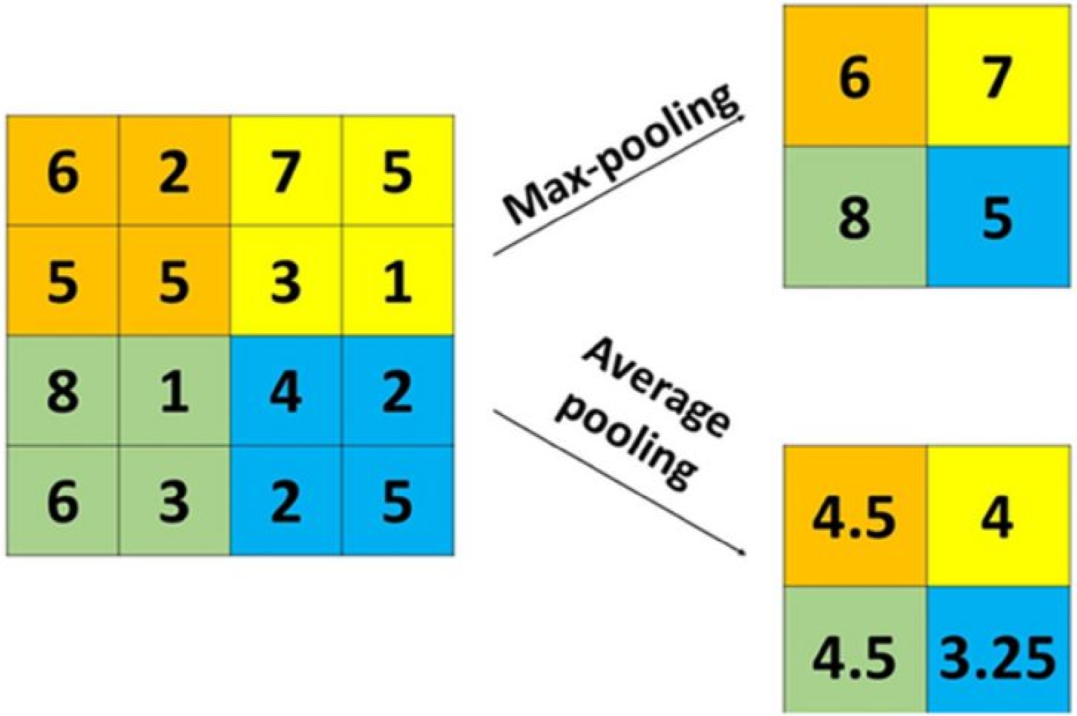
Pooling Layers

The pooling layer reduces the spatial dimensionality of the input feature map.

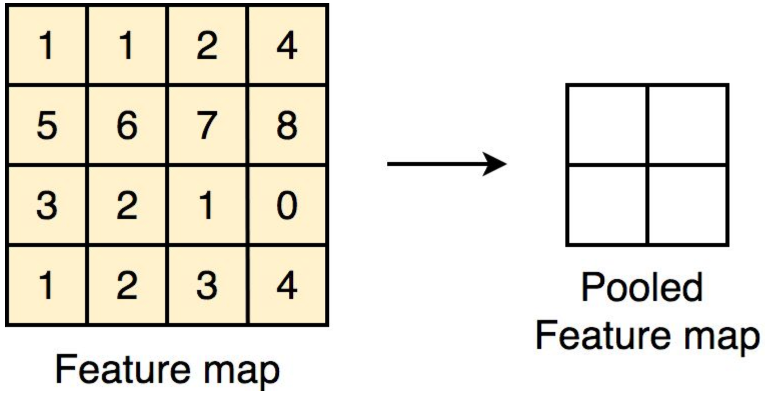


Max and average pooling of feature map using a 2x2 matrix

Pooling Operation



Expand

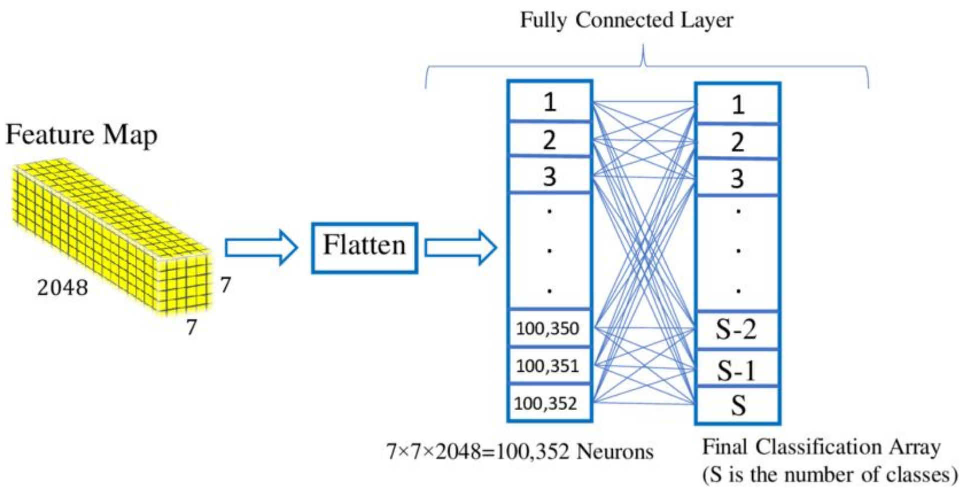


Fully Connected Layers

Where the high-level reasoning based on extracted features occurs. Transform high-dimensional feature maps into a probability distribution

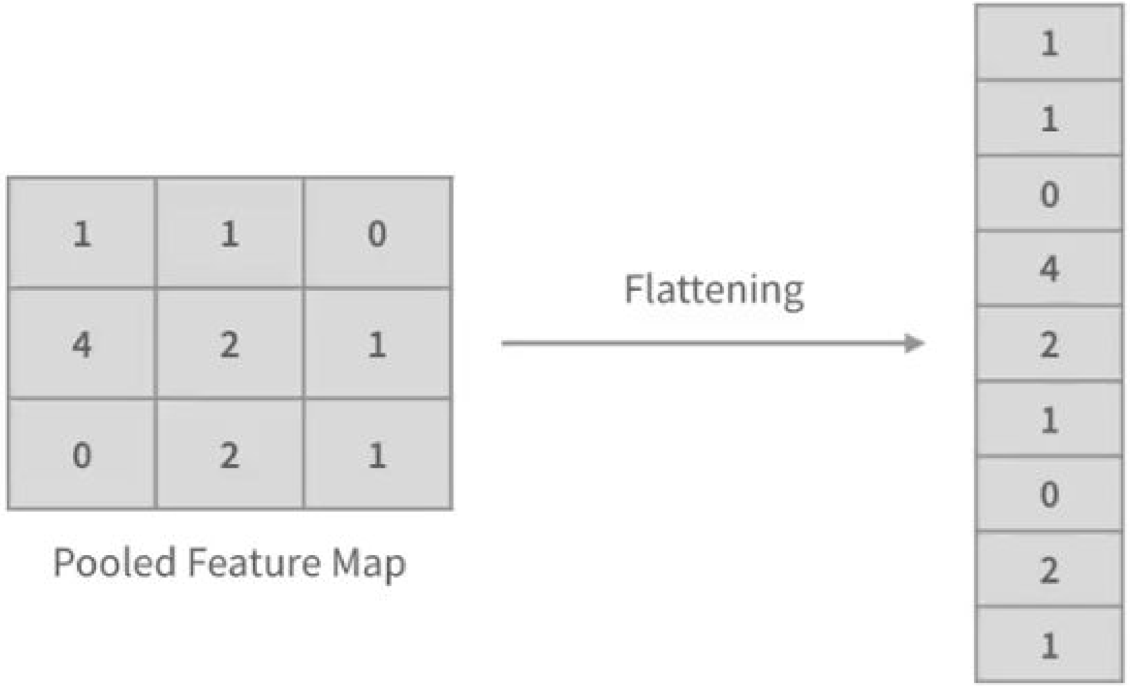
After convolutional and pooling layers extract and down sample features, fully connected layers integrate these features to make predictions or classifications.

Each neuron in these layers is connected to all activations in the previous layer, allowing the network to consider the entire representation of the input data.



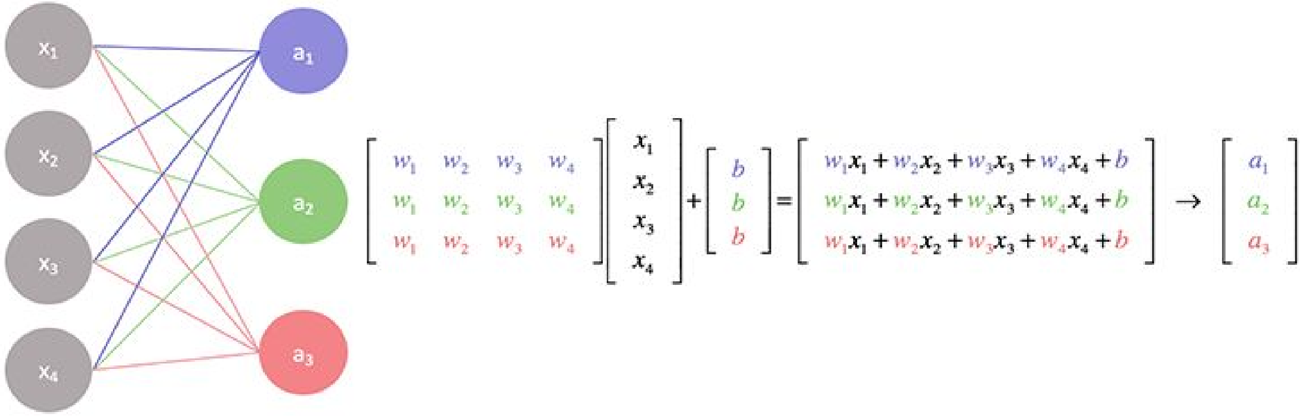
Flattening

- Convolutional and pooling layers produce feature maps
- Feature maps are multi-dimensional arrays
- Flattening converts feature maps to one-dimensional vector
- Concatenates elements along depth dimension
- Enables feeding into fully connected layers



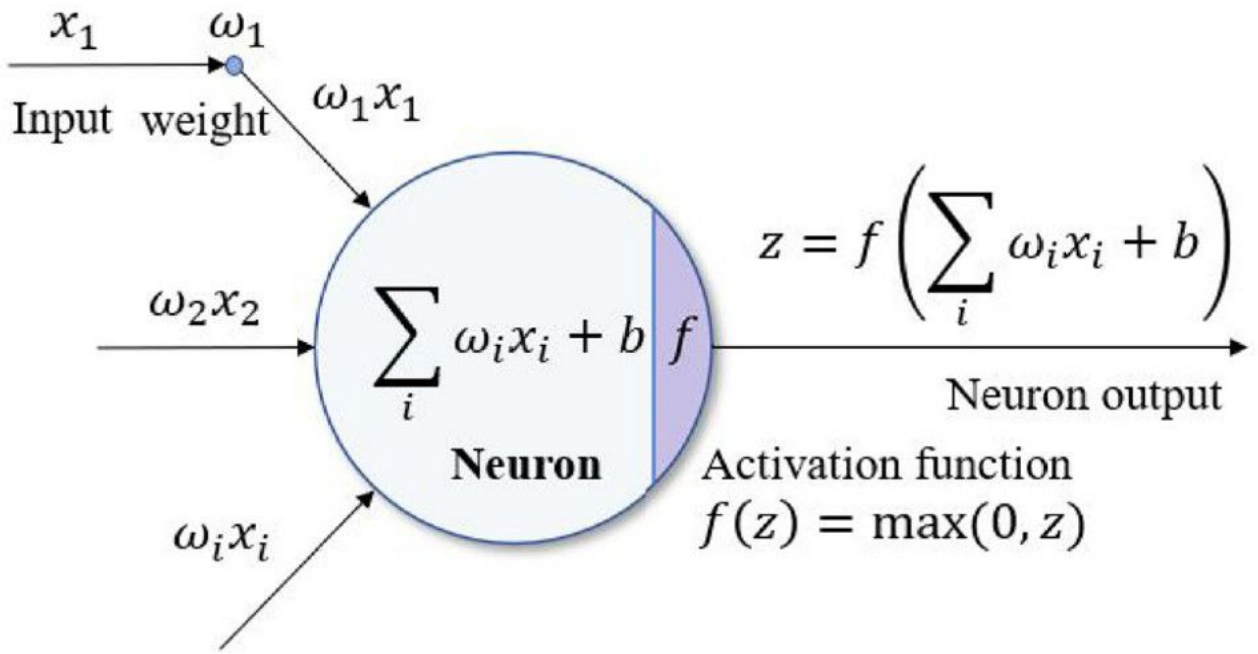
Weight Matrix and Bias Vector

- Foundation for deep learning algorithms.
- Fully connected layer have weight matrix (W) and bias vecor (b)
- Weight matrix: (n x m), n = neurons, m = flattened vector length
- Bias vector length: number of neurons in the current layer
- Learnable parameters of the fully connected layer
- Enable transformation and introduce nonlinearity
- Input vector is multiplied by weight matrix and bias vector is added
- Operation: W * input + b
- Output represents weighted sum of input from previous layer



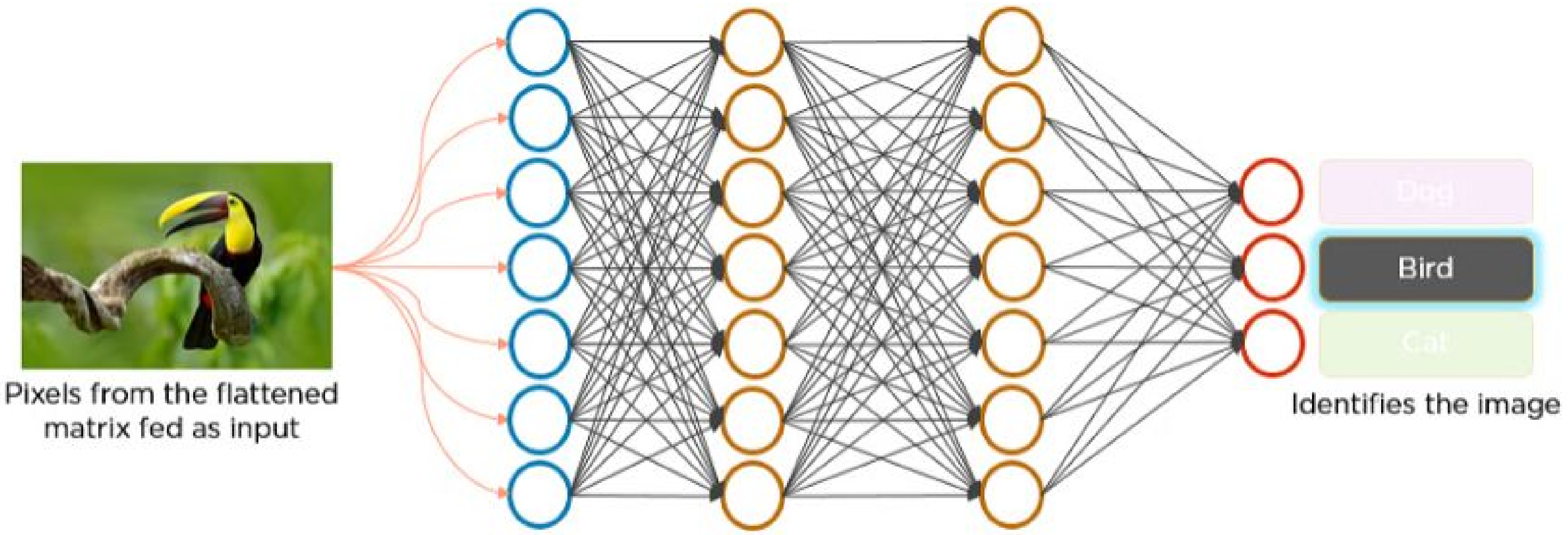
Activation Functions

- Activation function determines if a neuron fires
- Introduces nonlinearity to the network
- Applied after convolution layer, after each fully conncted later and output layer allowing the network to learn and represent complex patterns in the data\
- Most commonly used actiavtion function is ReLU



Output Layer

- The final layer generates predictions
- Neurons in the last layer match number of classes
- Activation function differs in final layer
- Softmax commonly used for multi-class classification
- Highest probability neuron represents prediction



Back Propagation

- A supervised learning algorithm used for training neural networks.
- It happens only during training
- Optimizes the parameters (weights and biases) of a neural network by minimizing the error between the predicted output and the actual target value.
- Basic Steps are;
    1. Feed a sample to the network
    2. Calculate the mean squared error
    3. Calculate the error term of each output neuron
    4. Iteratively calculate the error terms In the hidden layers
    5. Apply the delta rule
    6. Adjust the weights

Image Processing in CNNs

1

Input

Raw image data enters the network.

2

Feature Extraction

Convolutional layers detect edges, shapes, textures.

3

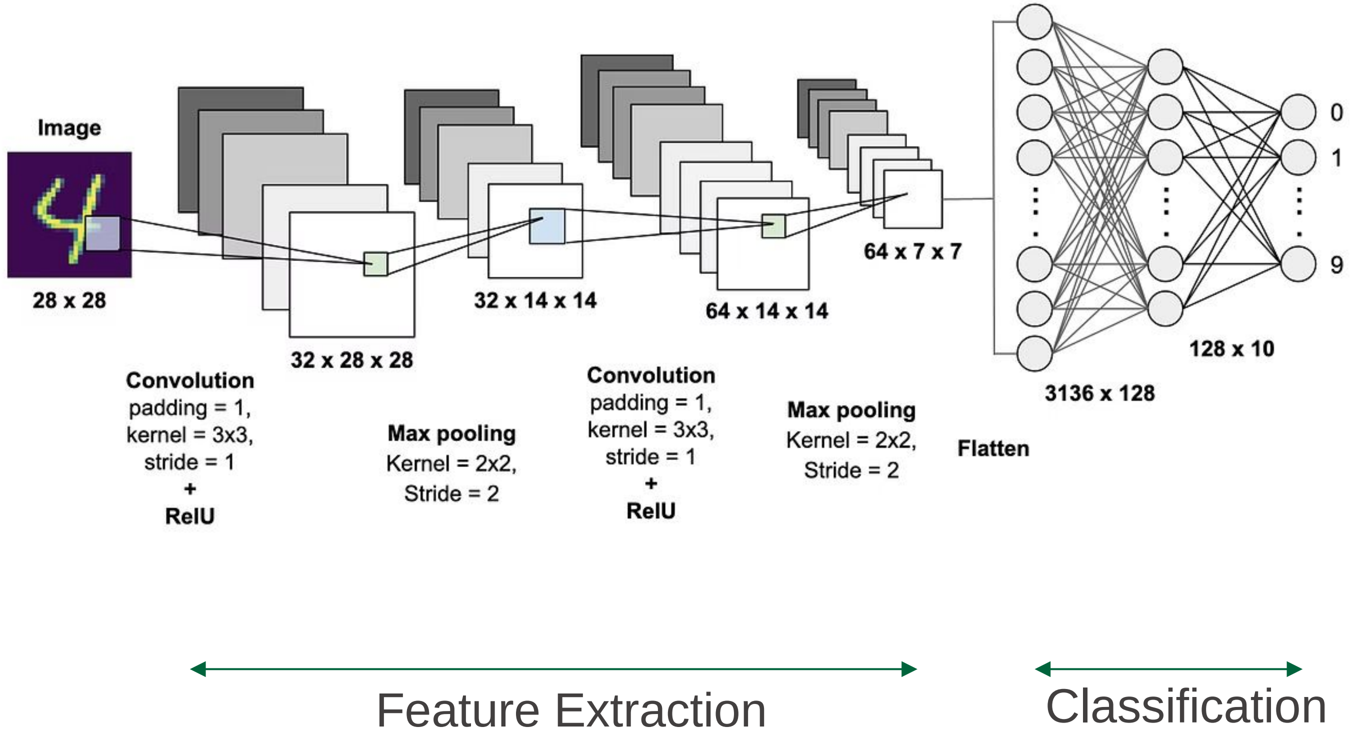
Down-sampling

Pooling layers reduce data complexity.

4

Classification

Fully connected layers determine image content.



Applications of CNNs

CNNs have revolutionized the field of computer vision. Applications include image and video recognition, image segmentation, object detection, face recognition, and automated medical diagnosis. They are also used in self-driving cars for detecting objects and pedestrians.

Can be used for tasks like:

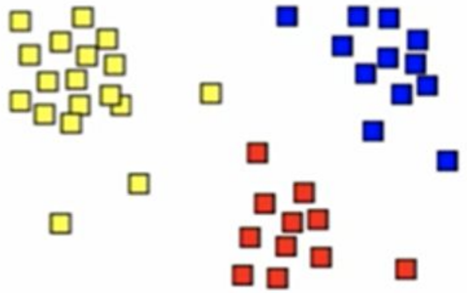
- Image classification
- Object detection
- Semantic and instance segmentation
- Multiple object tracking
- Re-identification
- Any vision task

Real-World CNN Impact

Medical Imaging	Anomaly detection in scans
Autonomous Vehicles	Real-time environment perception
Facial Recognition	Security and user authentication
Quality Control	Defect detection in manufacturing

Performance Evaluation Metrics

Classification, Regression or Clustering?

Clustering	Regression	Classification
		
Average **distance** to other center Average **distance** to cluster center Number of points Maximal **distance** to cluster center Average **distance** to cluster center Combined evaluation	**Mean absolute** error (MAE) **Root mean squared** error (RMSE) **Relative absolute** error (RAE) **Relative squared** error (RSE) **Coefficient** of determination	**Accuracy, precision**, recall, F1 score, AUC

1) https://learn.microsoft.com/en-us/azure/machine-learning/component-reference/evaluate-model?view=azureml-api-2

Performance Evaluation Metrics

- **Accuracy** measures the proportion of total predictions (both positive and negative) that the model got correct, offering a general sense of its performance across all classes.
- **Precision** assesses the accuracy of the positive predictions made by a CNN, specifically calculating the proportion of true positive predictions out of all positive predictions made (true and false positives), which is crucial in scenarios where false positives have significant consequences.

Performance Evaluation Metrics

- **Recall** (or sensitivity) evaluates a CNN's ability to correctly identify all actual positive cases, measuring the proportion of true positives out of the sum of true positives and false negatives, and is important in contexts where missing positive cases is costly.
- **F1 score** provides a balance between precision and recall by calculating their harmonic mean, offering a single metric for situations where it's crucial to maintain a balance between minimizing false positives and false negatives.
- **Receiver Operating Characteristic (ROC) curve** plots the true positive rate against the false positive rate at various threshold settings, and the Area Under the Curve (AUC) provides a single value summarizing the overall performance of a CNN across all possible classification thresholds.

Confusion Matrix

A confusion matrix is a tool used in machine learning and statistical classification to evaluate the performance of a classification model. It provides a summary of the prediction results on a classification problem. The matrix itself is a table that compares the actual target values with the predicted values.

True Positives (TP): The number of correct positive predictions.
True Negatives (TN): The number of correct negative predictions.
False Positives (FP): The number of incorrect positive predictions.
False Negatives (FN): The number of incorrect negative predictions.

BMI	BP	Glucose	Age	Gender	Heart Issue
25	140x90	90	78	M	No
23	120x80	87	48	F	Yes
21	110x70	116	37	F	Yes
28	135x90	129	23	M	No
25	150x110	144	89	M	Yes
...	...	...	---	...	...

		Prediction	
		No (29000)	Yes (1000)
Actual	No	28000 True Negative	100 False Positive
	Yes	1000 False Negative	900 True Positive

Performance Evaluation Metrics

Accuracy:

$$\text{Accuracy} = \frac{\text{True Positives (TP)} + \text{True Negatives (TN)}}{\text{Total Number of Samples}}$$

Precision:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

Recall (also known as Sensitivity):

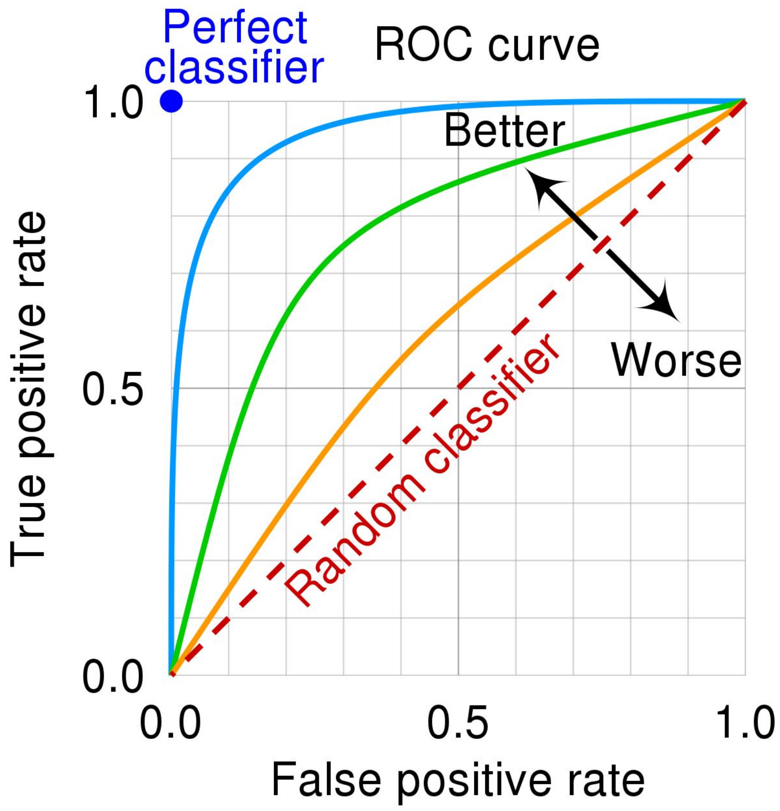
$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$$

F1 Score:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$\text{True Positive Rate (TPR)} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{False Positive Rate (FPR)} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$



Ethical Considerations and Bias in CNNs

Privacy: Privacy concerns arise when CNN models process sensitive personal data, such as facial images or medical records, potentially leading to unauthorized access or misuse of personal information if data security is not adequately maintained. This also causes issues when collecting and annotating data.

Surveillance: The use of CNNs in surveillance systems can enhance public safety and security by identifying threats more efficiently; however, it also raises ethical issues related to mass surveillance and the potential infringement on individuals' rights to privacy and freedom.

Bias in AI: particularly in CNNs, occurs when the data used to train these models contain inherent prejudices, leading to skewed or unfair outcomes in decision-making processes, often reinforcing existing societal stereotypes and discriminations.

References

1) https://austingwalters.com/edge-detection-in-computer-vision
2) https://www.kaggle.com/datasets/tongpython/cat-and-dog
3) Google search
4) https://gamma.app/#images
5) https://www.semanticscholar.org/paper/Cats-and-dogs-Parkhi-Vedaldi/84b50ebe85f7a1721800125e7882fce8c45b5c5a
6) https://www.simplilearn.com/tutorials/deep-learning-tutorial/convolutional-neural-network
7) https://www.analyticsvidhya.com/blog/2021/08/beginners-guide-to-convolutional-neural-network-with-implementation-in-python/
8) https://learn.microsoft.com/en-us/azure/machine-learning/component-reference/evaluate-model?view=azureml-api-2

Next Week Topics

- CNN Training Process
- Loss Function
- Different types of Activation Functions
- Back propagation Algorithm
- Common Problems in Machine Vision
- CNN Solutions