Deep Learning for Computer Vision: CNN, R-CNN, Fast R-CNN, Faster R-CNN, YOLO, and Vision Transformer (ViT

Deep Learning Models for Computer Vision

1. Convolutional Neural Network (CNN)

Theoretical Background:

CNNs are inspired by the human visual cortex and are designed to automatically and adaptively learn spatial hierarchies of features using multiple layers, such as:

- Convolutional layers
- Pooling layers
- Fully connected layers

They excel in recognizing patterns like edges, shapes, and textures in grid-like data (e.g., images).

Architecture:

```
Input → Convolution → ReLU → Pooling → Fully Connected → Output (Softmax)
```

Applications:

- Image classification (ImageNet, CIFAR-10)
- Medical image analysis (e.g., tumor detection)
- Face recognition
- Handwritten digit recognition

2. R-CNN (Regions with CNN features)

Theoretical Background:

Introduced by Ross Girshick (2014), R-CNN uses selective search to generate ~2000 region proposals and classifies each using a CNN. Bounding boxes are refined using regression.

Architecture:

```
Input image → Selective Search → Crop each region → CNN → SVM + Regressor
```

Applications:

- Object detection in images
- Scene understanding
- Visual question answering

---

3. Faster R-CNN

Theoretical Background:

An improvement over R-CNN and Fast R-CNN. It introduces a **Region Proposal Network (RPN)**, making the detection pipeline fully end-to-end trainable.

Architecture:

```
Image → CNN → Feature Map → RPN → RoI Pooling → Classifier + Bounding Box
Regressor
```

Applications:

- Autonomous driving
- Real-time surveillance
- Face and pedestrian detection

---

4. YOLO (You Only Look Once)

Theoretical Background:

YOLO treats object detection as a single regression problem, directly predicting bounding boxes and class probabilities from full images in one evaluation.

Architecture:

```
Image → CNN → Grid cells → Bounding Boxes + Class Probabilities
```

Applications:

- Real-time object detection (drones, robots)
- Traffic monitoring

- Industrial automation

---

5. Vision Transformer (ViT)

Theoretical Background:

ViT applies the transformer architecture, originally from NLP, to image patches. It models global relationships using self-attention mechanisms instead of convolutions.

Architecture:

```
Image → Divide into patches → Patch embeddings + positional encodings →
Transformer encoder → MLP head
```

Applications:

- High-resolution image classification
- Medical imaging
- Fine-tuned for object detection and segmentation

Key Differences

Model	Type	Speed	Accuracy	Real-Time	Use Case
CNN	Classifier	Medium	High	❌	Classification, feature extraction
R-CNN	2-stage detector	Slow	High	❌	Detailed object detection
Faster R-CNN	2-stage detector w/ RPN	Fast	Very High	❌	Surveillance, autonomous driving
YOLO	1-stage detector	Very Fast	Moderate-High	✅	Real-time detection
ViT	Transformer	Medium	Very High	❌	Global pattern understanding

REAL LIFE ANALOGY

1. Convolutional Neural Network (CNN)

Real-Life Analogy (Full Architecture View):

Imagine a mail sorter at a post office. Letters (pixels) arrive at the input. First, the sorter uses magnifying glasses (convolution filters) to look for patterns like stamps and addresses. He marks important parts (activations) and then groups similar-looking letters together into boxes (pooling). Finally, he sends each box to a decision room (fully connected layers) to decide: is it a bill, a wedding card, or junk mail?

2. R-CNN (Regions with CNN features)

Real-Life Analogy (Full Architecture View):

Picture a detective examining a crime scene. First, he highlights 2000 suspicious spots using a magnifying glass (selective search). Then, he carefully zooms into each spot with a microscope (CNN), takes notes (features), and passes them to an expert (SVM) who determines if the object is a weapon, tool, or irrelevant.

3. Fast R-CNN

Real-Life Analogy (Full Architecture View):

Now imagine a drone scanning an entire area in one go (CNN on full image). It maps out all important zones on a screen (feature map). Then a zoom-in button (RoI pooling) lets the drone operator inspect each suspect region efficiently. Each region is classified by an on-board system (classifier + regressor).

4. Faster R-CNN

Real-Life Analogy (Full Architecture View):

Think of an automated airport security scanner. It scans every bag on a conveyor (CNN), and a built-in smart system (RPN) flags only the bags that seem unusual. These are then highlighted on the scanner screen (RoI), and passed to a customs officer (classifier) who checks the content and assigns a label (e.g., clothes, electronics, or contraband).

5. YOLO (You Only Look Once)

Real-Life Analogy (Full Architecture View):

Imagine a superhero with X-ray vision. She looks at the whole scene once and instantly identifies and points out every person, dog, or object in a single glance. She divides her field of vision into a grid (YOLO grid), and each grid cell predicts what's inside its area and draws a box.

6. Vision Transformer (ViT)

Real-Life Analogy (Full Architecture View):

Imagine reading a comic book page. You cut it into equal panels (patches), read each one carefully, and use your memory and context (transformer blocks + self-attention) to understand the whole story. Unlike CNNs that zoom into details, you think about how each part of the story relates to every other part — not just what's in the patch but how patches connect globally.