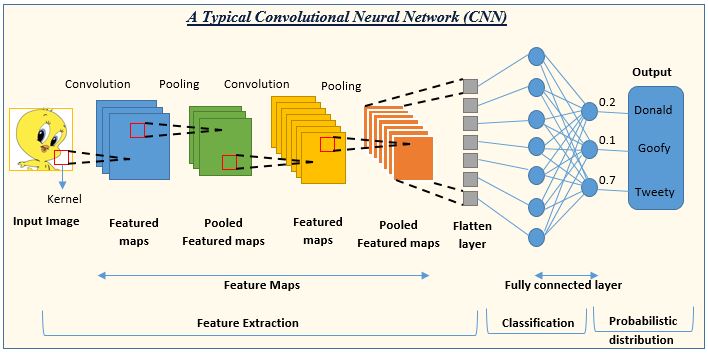
Deep Learning Classifications

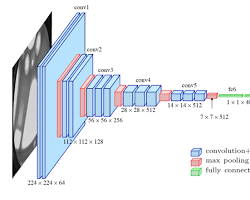
Computer Vision

Image Classification

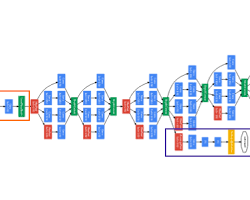
**Convolutional Neural Networks (CNNs)**:



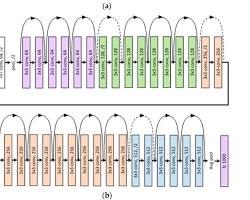
* **LeNet**: One of the earliest CNNs, primarily used for digit recognition.
* **AlexNet**: Popularized deep learning for image classification by winning the ImageNet competition in 2012.
* **VGGNet (Visual Geometry Group Network)**: Known for its simplicity and depth, uses very small (3x3) convolution filters.



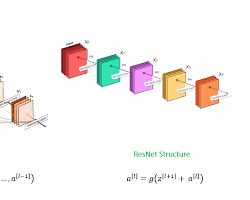
* **GoogLeNet (Inception)**: Introduced the inception module which dramatically reduced the number of parameters.



* **ResNet (Residual Networks)**: Introduced residual blocks, allowing the training of very deep networks by mitigating the vanishing gradient problem.



* **DenseNet (Densely Connected Convolutional Networks)**: Each layer is connected to every other layer in a feed-forward fashion, improving information flow and gradient propagation.



Object detection

Object detection is a more complex task than image classification as it involves not only identifying objects in an image but also localizing them with bounding boxes.

**Region-Based Convolutional Neural Networks (R-CNN)**:

* **R-CNN (Regions with CNN features)**: Proposes regions using selective search, then applies CNNs to each region.
* **Fast R-CNN**: Improves R-CNN by sharing computation and using a RoI (Region of Interest) pooling layer.
* **Faster R-CNN**: Introduces the Region Proposal Network (RPN) to replace selective search, making the region proposal process nearly cost-free.
* **Mask R-CNN**: Extends Faster R-CNN to also predict segmentation masks for each region of interest.

**You Only Look Once (YOLO)**:

* **YOLO**: Treats object detection as a single regression problem, directly predicting bounding boxes and class probabilities from full images in one evaluation.
* **YOLOv2 (YOLO9000)**: Improves YOLO with better performance and multi-scale training.
* **YOLOv3**: Enhances the architecture with feature pyramid networks and multi-scale predictions.
* **YOLOv4** and **YOLOv5**: Further improvements in accuracy and speed, with enhanced architectures and training techniques.

Image segmentation

Image segmentation involves partitioning an image into multiple segments or regions to simplify its representation and make it more meaningful.

**Fully Convolutional Networks (FCNs)**:

* **FCN**: One of the earliest deep learning approaches to segmentation, which replaces fully connected layers with convolutional layers, allowing the network to produce spatially dense outputs.

**U-Net**:

* **U-Net**: Originally developed for biomedical image segmentation, it has a U-shaped architecture with symmetric encoder-decoder paths, featuring skip connections between corresponding layers of the encoder and decoder.

**Attention U-Net**:

* **Attention U-Net**: Enhances the U-Net architecture by incorporating attention mechanisms to focus on relevant parts of the image, improving segmentation accuracy.

**Mask R-CNN**:

* **Mask R-CNN**: Extends Faster R-CNN to perform pixel-level segmentation tasks by adding a branch for predicting segmentation masks on each Region of Interest (RoI).

**GCN (Global Convolutional Network)**:

* **GCN**: Utilizes large convolution kernels to capture global context, enhancing segmentation performance, especially in boundary prediction.

**SegNet**:

* **SegNet**: Similar to U-Net, it has an encoder-decoder architecture but focuses on memory efficiency by storing only the max-pooling indices in the encoder and using them in the decoder to perform non-linear up-sampling.

Face recognition

Face recognition is a specific application of computer vision that involves identifying or verifying a person from a digital image or a video frame.

**DeepFace**:

* **DeepFace**: Developed by Facebook, it uses deep learning techniques to achieve human-level performance in face verification. It employs a deep neural network to learn a robust representation of the face.

**FaceNet**:

* **FaceNet**: Developed by Google, it uses a deep convolutional network to map faces into a compact Euclidean space where the distance directly corresponds to face similarity. FaceNet uses triplet loss to train the network.

**VGGFace**:

* **VGGFace**: Based on the VGGNet architecture, it is trained on a large dataset of faces and is known for its high accuracy in face recognition tasks.

**MobileFaceNets**:

* **MobileFaceNets**: Designed for efficient face recognition on mobile and embedded devices, combining high accuracy with low computational cost.

**LightCNN**:

* **LightCNN**: A lightweight convolutional neural network specifically designed for face recognition, balancing accuracy and efficiency.

pip install --upgrade charset-normalizer