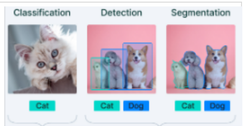
SCALER

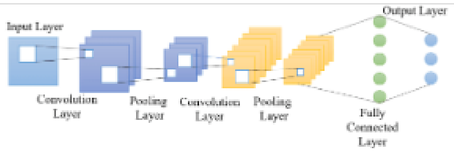
COMPUTER VISION CHEATSHEET

L1 : Introduction to Computer Vision (CNN) Cheat Sheet

- Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs.



A. What is a Convolutional Neural Network ?

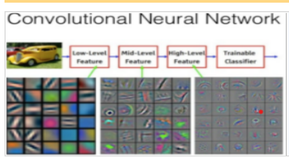


- A Convolutional Neural Network (CNN) is a **deep learning model** used for image and signal processing, where convolutional and pooling operations are applied to input data to extract features and make predictions.

B. Why use CNN over MLP?

- **Feature extraction**: CNNs are particularly well-suited for computer vision tasks because they can automatically learn and extract meaningful features from images and videos, while traditional machine learning algorithms like MPLs require manual feature engineering.
- **Handling large datasets**: CNNs can handle large datasets and can be trained on a large set of image data, while MPLs are not that efficient in handling large datasets.
- **Handling image variations**: CNNs can handle variations such as different lighting conditions, angles, and rotations, while MPLs can struggle with these variations.

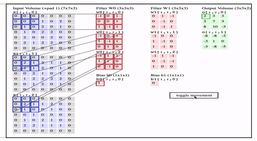
D. CNN Feature levels



- In a Convolutional Neural Network (CNN), features can be classified into different levels based on their complexity and abstraction, such as low-level features (e.g. edges, textures), mid-level features (e.g. parts of objects), and high-level features. (e.g. whole objects, scenes).

E. What is a Convolutional and Convolution operations?

- A kernel in a CNN is a small matrix used for performing a convolution operation which involves element-wise multiplication followed by addition and activation (such as ReLU)



D. What is Padding?

- Valid and same padding in CNNs are methods of handling the spatial dimensions of input and output matrices during the convolution operation, with "valid" reducing the output size and "same" preserving the input size in the output.

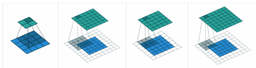
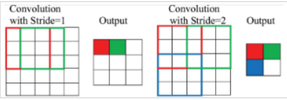


Fig 1 shows 'Valid' and other 3 shows 'Same' Padding

E. What is Stride ?



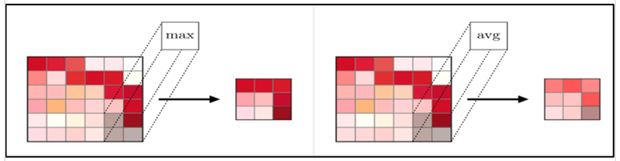
- In Convolutional Neural Networks (CNNs), strides are the number of pixels the convolution filter moves when scanning over the input.

H. Shape of output layer

$$output = \frac{input - kernel\_size + 2 * padding}{stride} + 1$$

H. What is pooling ?

Pooling layers provide an approach to down-sampling feature maps by summarizing the presence of features in patches of the feature map.



- **Max- Pooling -** Each pooling operation selects the maximum value of the current view.
- **Average-pooling** - Each pooling operation averages the values of the current view

D. What is Padding?

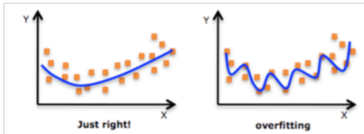
- For Convolution layer with k filters of size f x f and the input channel of image c:
- Total Parameter = k x (f x f x c_in) + k

- For a Fully Connected layer with n neurons and number of input I:
- Total Paramenter = n x (I + 1)

- By calculating the number of parameters in each layer and summing them up, you can obtain the total number of parameters in the entire CNN.

L2: Revisiting CNN Deal with Overfitting Cheat Sheet

A. What is Overfitting ?



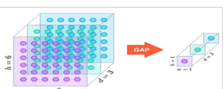
Overfitting occurs when a model is too complex and learns irrelevant details from the training data, causing poor performance on new data.

B. Methods to avoid Overfitting

1. Regularization: L1/L2, Dropout reduce model complexity, prevent overfitting.
2. Early Stopping: Stop training when validation performance declines.
3. Data Augmentation: Apply random image transformations to increase dataset.
4. Transfer Learning: Use pre-trained models and fine-tune.
5. Model Ensemble: Combine multiple models for robust results.
6. Reduce Network Complexity: Use smaller filters, fewer layers.

C. Global Average Pooling

- Global Average Pooling 2D is a type of pooling operation used in Convolutional Neural Networks (CNNs).



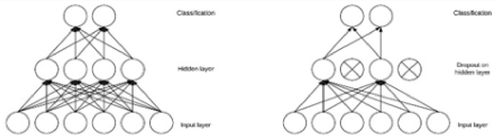
D. Batch Normalization

- Batch normalization normalizes neuron activations within batches, improving training stability and speed. It transforms each batch's activations to zero mean and unit standard deviation before passing to the next layer.

$$\hat{x}^{(k)} \leftarrow \frac{x_i^{(k)} - \mu^{(k)}}{\sqrt{\sigma^{2(k)} + \epsilon}}$$

D. What is Padding?

- Dropout is a deep learning regularization technique that randomly drops out some neurons during training to prevent overfitting, encouraging the network to learn multiple representations of data.



A. When to use Batch Normalization and Dropout

- Empirically, dropout is most effective placed between Dense layers with 0.5 probability. Lower probability (0.1-0.25) dropout after MaxPooling improves performance.
- Normalizing feature distribution after Conv Layer and passing it to ReLU block clamps negative values to 0, defeating normalization's purpose.

A. When to use Batch Normalization and Dropout

Lasso Regularization(L1)

$$loss = \sum_{i=0}^{n}(y_i - X_i\beta)^2 + \sum_{j=0}^{m} |\beta_j|$$

Ridge Regularization(L2)

$$loss = \sum_{i=0}^{n}(y_i - X_i\beta)^2 + \sum_{i=0}^{m} \beta_j^2$$

D. Example of Data Augumentation

A. When to use Batch Normalization and Dropout

- Data augmentation artificially increases training data in CNNs by applying various transformations to input images, preventing overfitting.
- Data augmentation is particularly useful when the size of the training dataset is limited.
- Techniques include rotation, scaling, translation, and flipping.

A. When to use Batch Normalization and Dropout

- There are many ways to apply augmentation in Tensorflow/Keras, few of them are discussed here:

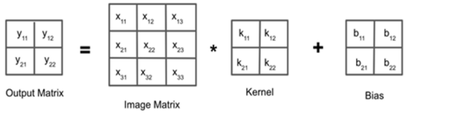
-- To apply augmentation in TensorFlow/Keras, use Keras Preprocessing Layers such as RandomFlip, RandomRotation, or tf.image methods like stateless_random_flip_up_down and stateless_random_brightness.

-- Use Keras ImageDataGenerator API for a suite of out-of-the-box augmentation

L2: Revisiting CNN Deal with Overfitting Cheat Sheet

A. Forward Propagation in CNN

The forward function iterates through every filter in every image channel and
- it pass over a kernel of specified size over the image and
- perform the convolution operation between the kernels and the input image.



$$Y_i = X_j * K_{ij} + B_i$$

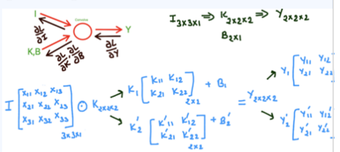
Forward propagation mathematically for the Convolutional Layer is as follows:

$$Y_i = \sum_{j=1}^{n} X_i * K_{ij} + B_i, i = 1 \ldots depth/n\_filters$$

B. Backward Propagation in CNN

Backward Propagation refers to updating the weights and biases of our model with respect to the loss of our model.

The derivative of error with respect to :
- Kernel
- Bias
- Input is calculated during backpropagation



C. Loss Gradients

Derivative of the Loss with respect to the Kernels

$$\frac{\partial L}{\partial K} = Conv\left(X, \frac{\partial L}{\partial Y_i}\right)$$

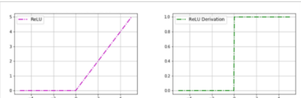
Derivative of the Loss with respect to the input of the layer

$$\frac{\partial L}{\partial X} = Conv\left(Padded\left(\frac{\partial L}{\partial Y_i}\right), 180°RotatedKernel_{K_i}\right)$$

Derivative of the Loss (L) with respect to the bias of the layer

$$\frac{\partial L}{\partial B_i} = sum\left(\frac{\partial L}{\partial Y_i}\right)$$

C. Loss Gradients



Forward Propagation

$RELU(x) = \begin{cases} 0 \text{ if } x < 0 \end{cases}$

Backward Propagation

$RELU'(x) = \begin{cases} 0 \text{ if } x < 0 \end{cases}$

C. Loss Gradients

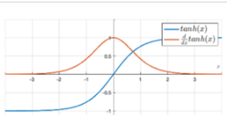
Forward Propagation

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Backward Propagation

$$\frac{\partial(Softmax)}{\partial x_1} = \frac{e^{x_1}.(e^{x_2} + e^{x_3})}{(e^{x_1} + e^{x_2} + e^{x_3})^2}$$

C. Loss Gradients



Forward Propagation

$$f(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$$

Backward Propagation

$$f'(x) = 1 - \tanh^2(x)$$

C. Loss Gradients

Forward Propagation

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(p(y_i)) + (1 - y_i) \cdot log(1 - p(y_i))$$

Backward Propagation

$$\frac{1}{N} \cdot \left(\frac{1 - y_{true}}{1 - y_{pred}} - \frac{y_{true}}{y_{pred}}\right)$$

SCALER

COMPUTER VISION CHEATSHEET

L4 : Introduction to Transfer Learning Cheat Sheet

A. What is ImageNet competition?

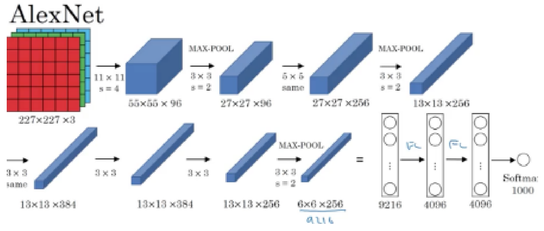
- ImageNet Competition: The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is an annual computer vision competition held to evaluate the performance of computer vision models on a large scale image classification task.
- Significance: The ImageNet competition has been instrumental in advancing the state-of-the-art in computer vision, with deep learning models winning the competition since 2012.

B. What is Top-K accuracy?

- Top-k accuracy refers to the accuracy of a model's top k predictions compared to the actual target classes.
- For example, if we have a multi-class classification problem with 5 classes and a model makes the following predictions:
- Prediction: [0.1, 0.3, 0.5, 0.8, 0.2] | True class: 3 If we set k=3, the model's top 3 predictions are classes 2, 3, and 4.
- Since class 3 is the true class, the top-k accuracy would be 1.0, indicating a correct prediction.

C. Why use CNN over MLP ?

- AlexNet is the first CNN to win ImageNet Competition in 2012.
- It was primarily designed by Alex Krizhevsky.
- They achieved a top 5 accuracy of 84.7%, while runner-up was at 74!



D. What was so new about AlexNet?

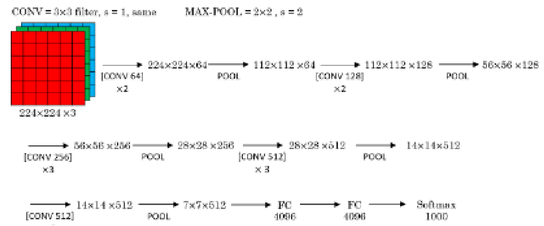
1. Graphics processing units (GPUs)
2. Rectified linear unit (ReLU) activation
3. Regularization (Dropout and Data Augmentation)
4. Pooling layers
5. Increasing the Depth

E. Improvements of VGG-16 over AlexNet

1. Deeper Network
2. Smaller Filters
3. Pooling Layers: VGG-16 employs more pooling layers
4. Uniform Filter Size - VGG-16 uses the same filter size (3x3) throughout the network

C. Why use CNN over MLP ?

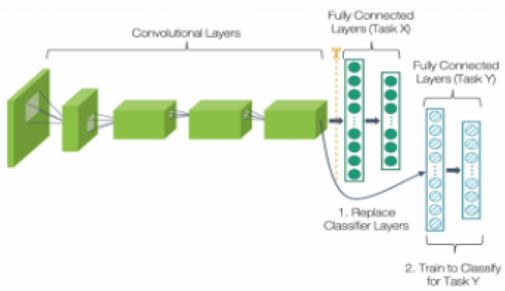
- VGG19 won second place in 2014 It was designed by the Visual Geometry Group (VGG) at the University of Oxford, UK.
- VGG16 has top 5 acc of 91.9 and top 1 acc of 74.4 on ImageNet dataset.



What is Transfer Learning ?

- Definition: Transfer learning is a method in machine learning that uses a pre-trained model as a foundation to solve a related but distinct problem.
-
- Advantages:
  - Leveraging learned knowledge from large data sets
  - Improved performance and faster training times
  - Reduced data and computational requirements
  - Useful when limited annotated data is available

What is Transfer Learning ?



What is Transfer Learning ?

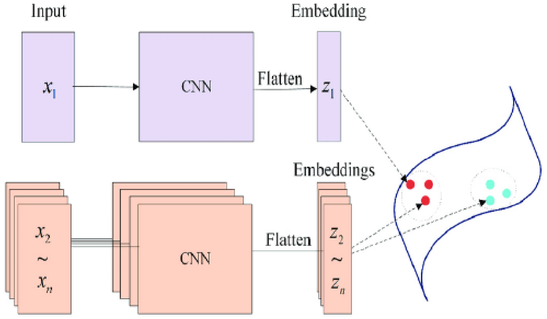
1. VGG-16 is computationally expensive to train and not suitable for low-power devices due to its high parameter count.
2. It's prone to overfitting and requires significant memory for parameter storage, limiting its use on smaller devices.
3. Training VGG - 16 can be challenging and time - consuming due to its deep architecture and many parameters, especially with limited annotated data.
4. Additionally, its use of max pooling can reduce spatial resolution, causing loss of information and reduced ability to capture fine details in images.

L4 : Introduction to Transfer Learning Cheat Sheet

What is Image Similarity?

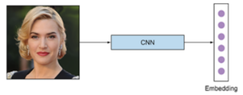
- Image similarity is the measure of how similar two images are.
- In other words, it quantifies the degree of similarity between intensity patterns in two images.
- Finding Image similarity also called reverse image search

Architecture of Image Similarity



D. What is Padding?

- An image embedding is a lower-dimensional representation of the image.
- In other words, it is a dense vector representation of the image which can be used for many tasks such as classification.
- A convolutional neural network (CNN) can be used to create the image embedding.

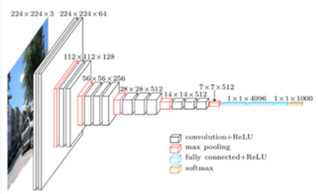


D. What is Padding?

- **VGG:**
  - **Many parameters, prone to overfitting and memory-intensive.**
  - **Small 3x3 filters make it computationally expensive.**
- **AlexNet:**
  - **Shallow architecture limits its ability to learn complex features.**
  - **Over-reliance on time-consuming data augmentation techniques.**

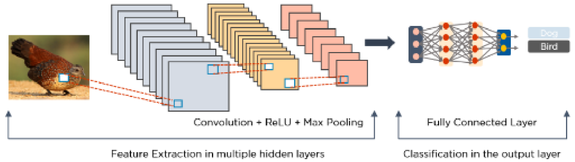
E. What is a Convolutional and Convolution operations?

- Deeper architectures in Convolutional Neural Networks (CNNs) are needed to improve the performance of the network by allowing it to learn more complex representations of the data
- A deeper network has more layers, and each layer can learn to extract a different and more abstract representation of the data.



C. Why use CNN over MLP ?

- We use multiple levels of Feature Extraction in CNN because At lower levels, the network extracts simple features such as edges and corners, which are combined to form higher-level features such as textures and shapes.
- These higher-level features are then combined to form even higher-level representations such as object parts and eventually complete objects.

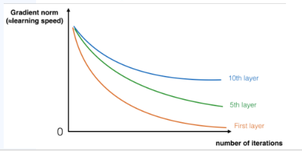


How ResNet and Inception better than AlexNet and VGG?

- ResNet:
  - Uses residual connections to learn residual representations and prevent vanishing gradients
  - Enables training deeper models without performance degradation
  - Has a deeper architecture than VGG and AlexNet, allowing it to learn more complex representations
- Inception:
  - Uses multiple parallel branches with different filter sizes and concatenates results to learn multiple scales of features efficiently
  - Reduces number of parameters by using multiple smaller filters instead of a few large filters
  - Reduces risk of overfitting

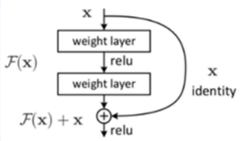
D. What is Padding?

- The speed of learning decreases very rapidly for the early layers as the network trains

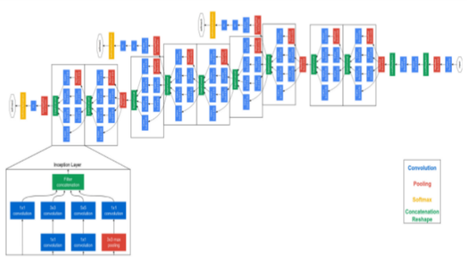


D. What is Padding?

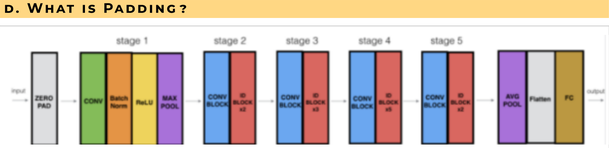
- Skip connections in CNNs bypass one or more layers in the network, aiding gradients flow during training.
- Skip connections mitigate the vanishing gradient problem in deep networks, which can make training difficult.
- Activations from earlier layers are combined with those from later layers via skip connections, maintaining a stronger gradient signal throughout the network.



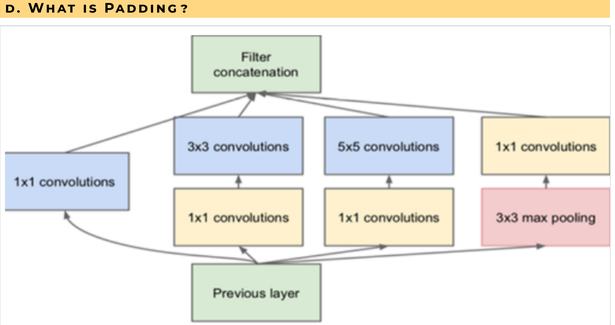
D. What is Padding?



D. What is Padding?



D. What is Padding?



SCALER

COMPUTER VISION CHEATSHEET

L9 : Object Segmentation CheatSheet

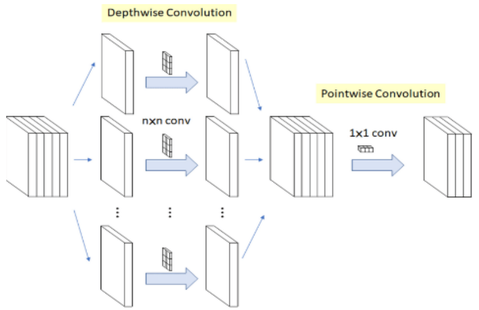
A. What is Object Segmentation ?

- MobileNet is a compact and efficient CNN architecture ideal for mobile and embedded devices.
- It uses Depthwise-Separable convolution and pointwise convolutions, yielding good results while being computationally inexpensive.
- Despite having 10x fewer parameters than ResNet50, it still offers great results.

A. What is Object Segmentation ?

- Depthwise Separable Convolution splits the computation into two steps:

1. Depthwise convolution : It applies a single convolutional filter per each input channel.
2. Pointwise convolution: It is used to create a linear combination of the output of the depthwise convolution.

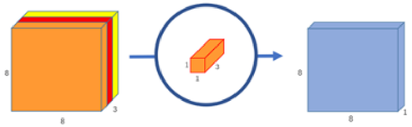


A. What is Object Segmentation ?

- In Depthwise Convolution, we apply a single convolutional filter for each input channel

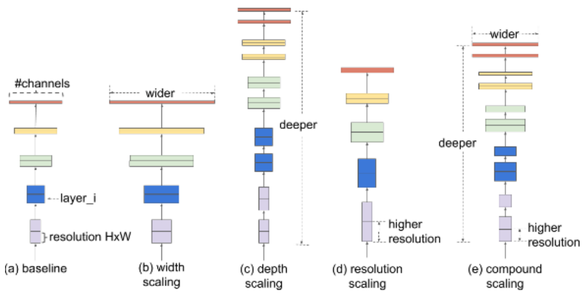
A. What is Object Segmentation ?

- In Pointwise Convolution, we use a 1x1 kernel, which iterates across each and every point. This kernel has a depth equal to the number of channels in the input picture.

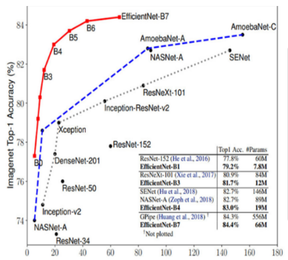


A. What is Object Segmentation ?

- EfficientNet is a CNN architecture that beats all previous ones with fewer parameters.
- It's a family of neural networks with scaling emphasis using the compound coefficient technique.
- It offers eight models of increasing size, from B0 to B7.



A. What is Object Segmentation ?



Model Name	Number of params	Top 1 Acc	Top 5 Acc
EfficientnetB0	5.3M	77.3	93.5
MobileNet	2.3M	71.0	90.5
ResNet50	25.6M	83.2	96.5
Inception	22.9M	79.0	94.5
VGG16	138M	74.4	91.9
AlexNet	62M	63.3	84.6

A. What is Object Segmentation ?

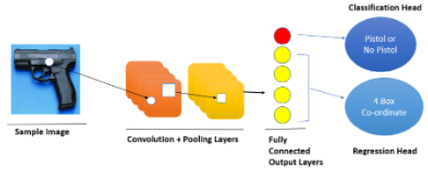
- Size of the new dataset
- Similarity of new dataset with the original dataset
- New Dataset is small and similar : Fine-tuning might lead to overfitting.
- New Dataset is large and similar : Fine-tune the pre-trained network
- New Dataset is large and different : Train a Conv. Network from scratch

L7 : Object Detection with Two Stage Methods Cheat Sheet

A. What is Object Segmentation ?

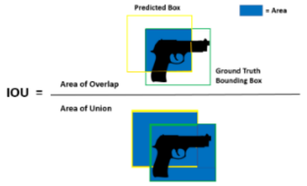
- Object detection is a computer vision task to locate and identify objects in an image or video.
- It answers the following question:
- Where is the object of interest present in the Image?
- How many objects are present in Image?

A. What is Object Segmentation ?



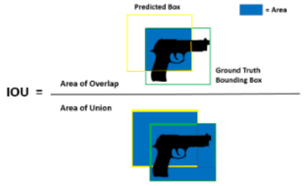
A. What is Object Segmentation ?

- IoU stands for Intersection over Union, which is a commonly used evaluation metric for object detection tasks.



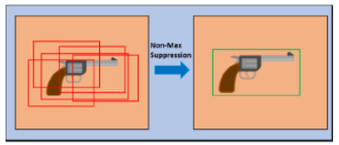
A. What is Object Segmentation ?

- IoU stands for Intersection over Union, which is a commonly used evaluation metric for object detection tasks.



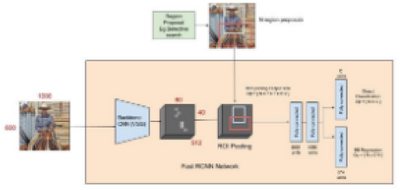
A. What is Object Segmentation ?

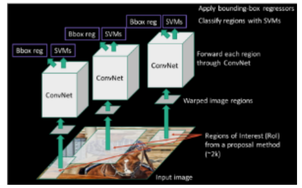
- NMS (Non-Maxima Suppression) is a technique used to suppress redundant bounding boxes (BBOXES) in object detection algorithms in computer vision.
- It works by keeping only the bounding box with the highest confidence score and removing other overlapping boxes, reducing the number of false positive detections.



A. What is Object Segmentation ?

- Instead of extracting features from each region proposal independently, Fast RCNN uses a shared convolutional layer to extract features from the entire image, reducing the number of computations needed.





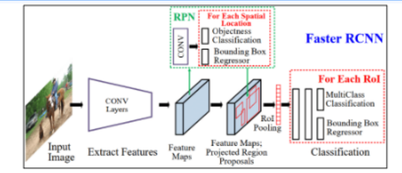
A. What is Object Segmentation ?

- RPN in R-CNN generates object region proposals using sliding windows and anchor boxes with objectness scores.
- RPN reduces the number of region proposals processed by the classifier.
- RPN provides high-quality proposals for objects of various shapes.
- RPN speeds up object detection.

A. What is Object Segmentation ?

- ROI (Region of Interest) pooling is a technique used in computer vision and deep learning for object detection and classification.
- It works by dividing the input image into regions of interest and then resizing each region to a fixed size, allowing the network to process each region independently and efficiently.

A. What is Object Segmentation ?



❑ Comparison of Speed and mAP metric

External region proposals method: (Selective Search Algorithm)	✓	✓	✗
	R-CNN	Fast R-CNN	Faster R-CNN
Test time per image	50 seconds	2 seconds	0.2 seconds
Speed-up	1x	25x	250x
mAP (VOC 2007)	66.0%	66.9%	66.9%

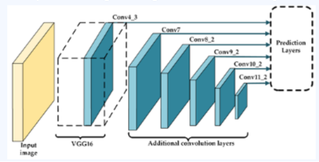
L9 : Object Segmentation CheatSheet

A. What is Object Segmentation ?

- Computational cost: Slow inference due to two-stage architecture.
- Poor recall: Can miss small or occluded objects.
- Inefficient proposal generation: Selective search is slow and doesn't scale well.
- Limited versatility: Designed for limited object categories.
- Difficult to fine-tune: Challenging for specific tasks with limited training data.
- Limited scalability: Scaling to larger object categories is challenging, leading to slower inference and more memory usage.

A. What is Object Segmentation ?

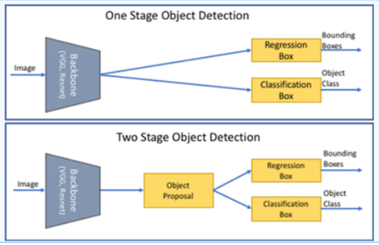
- Object detection method that predicts class labels and bounding boxes in a single stage.
- Offers an efficient alternative to two-stage detectors like R-CNN.
- Designed for real-time processing of large numbers of object categories.



SCALER

COMPUTER VISION CHEATSHEET

A. What is Object Segmentation ?

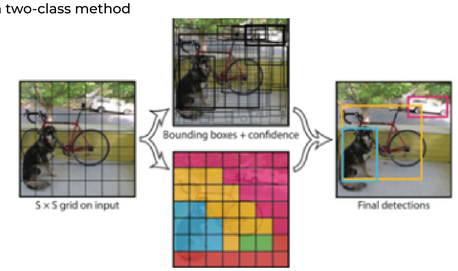


A. What is Object Segmentation ?

- You Only Look Once (YOLO) is a one-stage object detection method that predicts class labels and bounding boxes for objects within an image.
- It's designed to handle a large number of object categories in real-time and is known for its fast inference time and efficient use of computational resources.

A. What is Object Segmentation ?

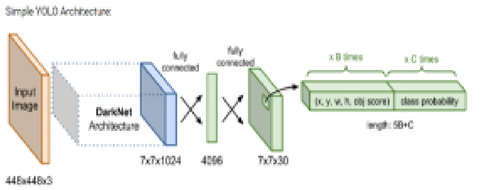
- The bounding box contains four values: x, y, w, h, (x, y) represents the center of the box. (W, h) defines the width and height of the box.
- Confidence indicates the probability of containing objects in this prediction box, which is the IoU value between the prediction box and the actual box.
- The class probability indicates the class probability of the object, and the YOLOv3 uses a two-class method



A. What is Object Segmentation ?

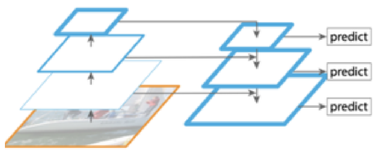
- Both one stage detection methods, like SSD and YOLO evalute almost 10^4 to 10^5 candidate locations per image.
- But only a few locations contain objects (i.e Foreground) and rest are just background objects.
- This leads to class imbalance problem.
- Small objects and close - by objects may be missed by YOLO like algorithms
- YOLO can detect a lmitied number of objects per image, making it less uiatable for applications where a large number of objects need to be detected.

A. What is Object Segmentation ?



A. What is Object Segmentation ?

- RetinaNet is superior to single-stage object detection methods because it addresses the class imbalance problem, resulting in better accuracy.
- Its two-stage approach utilizes class-specific confidence scores to eliminate false positive detections.
- Additionally, RetinaNet only generates one detection per object, which leads to faster inference times than YOLO and SSD, which use anchor boxes to detect objects.



L9 : Object Segmentation CheatSheet

A. What is Object Segmentation ?

Image segmentation is the process of classifying each pixel in the image as belonging to a specific category.

- Semantic segmentation : We treats multiple objects within a single category as one entity
- Instance segmentation : We identify individual objects within these categories

B. Transposed Convolution

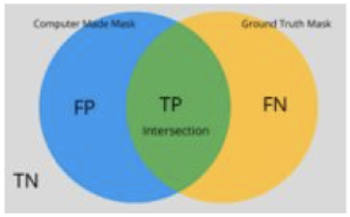
Transposed Convolutions is a method to up-sample the output. It can be considered as an opposite process to any simple CNN.

Output Shape :

output size = (input - 1)*stride - 2*padding + (kernel size - 1) + 1

C. Intersection over Union

$$\frac{\text{Intersection}}{\text{Union}} = \frac{TP}{TP + FN + FP}$$

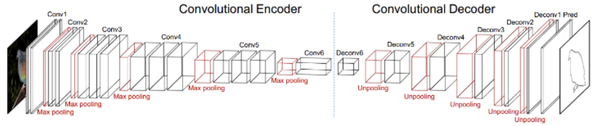


D. Dice Coefficient

$$\text{Dice Coefficient} = \frac{2\times\text{Intersection}}{\text{Union} + \text{Intersection}} = \frac{2TP}{2TP + FN + FP}$$

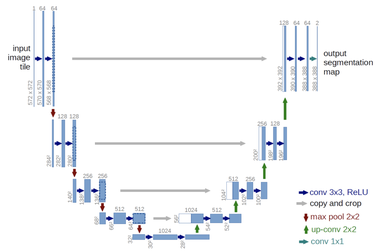
E. Encoder-Decoder Network

Encoder-Decoder Network is an architecture consisting of two parts, an encoder that compresses input into a lower dimensional representation, and a decoder that reconstructs the output from the encoded



F. U-Net Architecture

- U-net Model has a “U” shape architecture with a symmetric Encoder and Decoder.
- It uses Skip Connections between layers of Encoder and Decoder are used to make the information loss as minimal as possible.
- The final output layer produces a per-pixel prediction of the target mask or segmentation.



G. What is Object UpSampling2D ?

UpSampling2D is a simple scaling up of images by using nearest neighbour or bilinear upsampling.

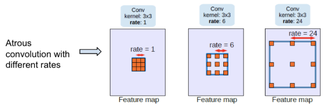
H. Mask R-CNN Architecture

- Mask R-CNN is an extension of the Faster R-CNN model for object detection and instance segmentation.
- The RPN generates object proposals by predicting object scores and bounding box coordinates.
- The RoIAlign layer resamples object proposal features to a fixed size to ensure they can be processed by the fully connected layers.

I.

J. What is Atrous Convolution ?

- For each location i on the output y and a filter w, atrous convolution is applied over the input feature map x where the atrous rate r corresponds to the stride with which we sample the input signal.
- It is also called dilated convolution.



Atrous convolution formula ⟹ $$y[i] = \sum_{k} x[i + r \cdot k] w[k]$$

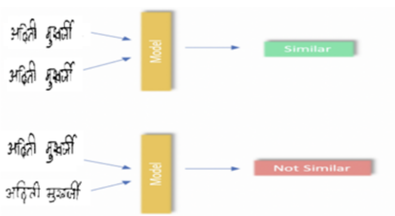
L10: Signature Verification Using Siamese Network Cheat Sheet

A. What is a Siamese Network?

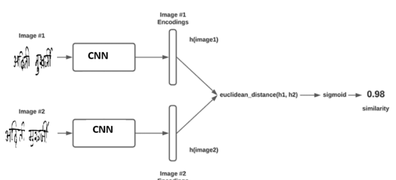
The Siamese Network architecture uses identical sub-networks to extract features from input data for similarity measurement or verification tasks, such as face recognition and signature verification.

B. Contrastive Loss

- The contrastive loss function uses two feature representations and a binary label indicating similarity to compute Euclidean distance and apply a penalty.
- Similar pairs are encouraged to be close and dissimilar pairs far. It trains a Siamese Network to learn a discriminative embedding space for accurate similarity measurement, one-shot learning, and verification.



C. Architecture of Siamese Network



C. Contrastive Loss Eqution

$$Y * D^2 + (1 - Y) * max(margin - D, 0)^2$$

- The label Y is 1 for the same-class image pairs and 0 for different-class pairs.
- The Euclidean distance between sister network embeddings is variable D.
- The max function selects the larger value between 0 and the difference between margin m and distance.

SCALER

COMPUTER VISION CHEATSHEET

D. Embeddings Before and After Contrastive Loss

E. Triplet Loss

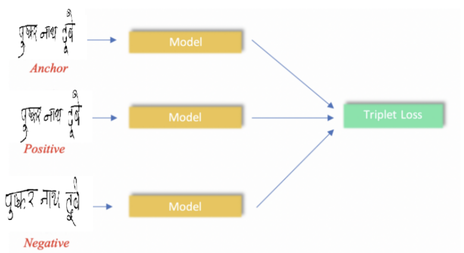
- A triplet includes an anchor, positive sample, and negative sample.
- The anchor and positive samples are similar, while the anchor and negative samples differ.
- The triplet loss function minimizes the difference in distances between anchor-positive and anchor-negative pairs, encouraging embeddings where the anchor-positive distance is less than anchor-negative by a margin.

F. Triplet Loss Equation

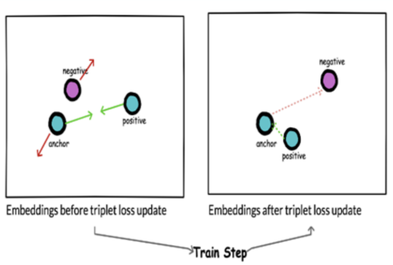
$$L(A,P,N) = max(0, d(A,P) - d(A,N) + \alpha)$$

- d(A,P) is anchor-to-positive-distance and d(A,N) is anchor-to-negative-distance and α is the margin

G. Triplet Loss



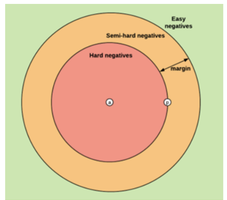
H. Embeddings Before and After Triplet Loss



I. Types of Triplet Mining

- Easy triplets: d(A,N)>d(A,P)+α, loss=0, nothing to learn.
- Hard triplets: d(A,N)<d(A,P), high loss, backpropagation.
- Semi-hard triplets: d(A,P)<d(A,N)<d(A,P)+α, positive loss.

J. Triplet Mining



K. Lossless Triplet Loss

$$\sum_{i=1}^{n}[-ln(-\frac{(f_i^a - f_i^p)^2}{\beta}) + 1 + \epsilon) - ln(-\frac{N-(f_i^a - f_i^n)^2}{\beta}) + 1 + \epsilon)]$$

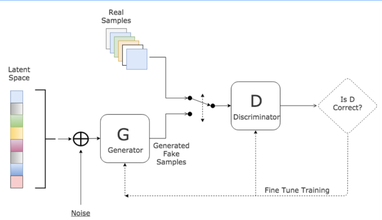
- N is the number of dimensions (Number of output of your network)
- β is the scaling factor
- ϵ is the margin
- $(f_i^a - f_i^p)$ is the distance between anchor and positive
- $(f_i^a - f_i^n)$ is the distance between anchor and negative

L11 : Introduction to GANs Cheat Sheet

A. What are GANs?

- Generative Adversarial Networks (GANs) are a type of deep learning architecture that uses two neural networks, a generator and a discriminator, to generate new data samples that are similar to a training set.
- The generator network creates new data samples while the discriminator network evaluates the generated samples and determines whether they are similar to the real training data.

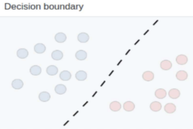
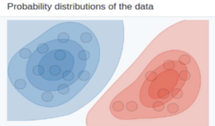
B. Architecture of GANs



C. What are the difficulties with training GANs?

- **Mode Collapse**: It occurs when the generator network produces only a limited number of distinct outputs rather than a diverse set of outputs.
- **Instability**: GANs are highly sensitive to the initial conditions and the choice of hyperparameters.
- **Convergence**: GANs can be difficult to train to convergence, meaning that the generator and discriminator networks may never reach an optimal solution.

D. Discriminative vs Generative model

	Discriminative model	Generative model
Goal	Directly estimate $P(y\|x)$	Estimate $P(x\|y)$ to then deduce $P(y\|x)$
What's learned	Decision boundary	Probability distributions of the data
Illustration		
Examples	Regressions, SVMs	GDA, Naive Bayes

D. What is Discriminative loss?

One common example of a discriminative loss function is binary cross-entropy loss, which is defined as:

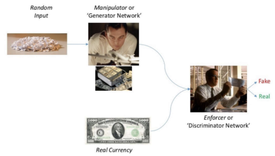
Loss = -[y * log(D(x)) + (1 - y) * log(1 - D(x))]

E. What is Generative loss?

The generative loss in GANs is a scalar value that represents the discrepancy between the generated samples and the real data. The goal of the generator is to minimize this loss so that the generated samples become as similar as possible to the real data.

Loss = -(1/m) ∑[y log(p) + (1-y) log(1-p)]

F. Simple working of GANs



G. Application of GANs

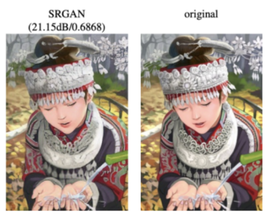
- Image Generation
- Image Translation
- Data Augmentation
- Text Generation
- Video Generation

H. GANs applications example - 1

CycleGAN: CycleGAN is a GAN that is used for image-to-image translation, where the goal is to transform an image from one domain to another, such as converting a horse into a zebra, or a photograph into a painting.

I. GANs applications example - 2

SRGAN is a Generative Adversarial Network used for single image super-resolution. It increases the resolution of an image, making it appear clearer and more detailed.



J. GANs applications example - 3

StyleGAN: StyleGAN is a Generative Adversarial Network (GAN) used for synthesizing new images of human faces that are highly realistic and diverse.