

CV Cheat Sheet

Key Concepts and Definitions

Object Detection

Identifying and locating objects within an image.

Bounding Box

The rectangle that surrounds the detected object in an image.

Intersection over Union (IoU)

A metric used to evaluate the accuracy of an object detector. It is the overlap between the predicted bounding box and the ground truth.

Formulae

IoU Calculation

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

Precision

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Recall

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

R-CNN

Regions with CNN features. It involves region proposal followed by CNN feature extraction and classification.

YOLO

You Only Look Once. It divides the image into a grid and predicts bounding

SSD

Single Shot Multibox Detector. It uses a single neural network to predict bounding boxes and class scores directly from full images in one pass.

Methodologies

Installation & Usage Examples

Diagrams

Troubleshooting

Hyperparameter Tuning

Adjust learning rates, batch sizes, and other parameters to improve model performance.

Transfer Learning

Fine-tune pre-trained models on your specific dataset to leverage existing knowledge.

Data Augmentation

Use techniques like flipping, rotation, and scaling to increase dataset variability.

Intersection over Union (IoU)



Object Detection Pipeline



Further Reading

<https://www.geeksforgeeks.org/how-to-install-keras-in-windows/>

<https://www.geeksforgeeks.org/computer-vision-libraries-for-python-features-applications-and-suitability/>

<https://chatgpt.com/>

<https://www.nvidia.com/en-us/glossary/tensorflow/>

TensorFlow Install

```
pip install tensorflow
```

TensorFlow Usage

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
```

```
# Create a simple neural network model
```

```
model = Sequential([
    Dense(64, activation='relu', input_shape=(784,)),
    Dense(64, activation='relu'),
    Dense(10, activation='softmax')
])
```

PyTorch Install

```
pip install torch torchvision
```

PyTorch Usage

```
import torch
import torch.nn as nn
import torch.optim as optim

# Define a simple neural network
class SimpleNN(nn.Module):
    def __init__(self):
        super(SimpleNN, self).__init__()
        self.fc1 = nn.Linear(784, 64)
        self.fc2 = nn.Linear(64, 64)
        self.fc3 = nn.Linear(64, 10)
    def forward(self, x):
        x = torch.relu(self.fc1(x))
        x = torch.relu(self.fc2(x))
        x = torch.softmax(self.fc3(x), dim=1)
        return x
```

Reflection

During this assignment, we learned about what is object detection, and how it encompasses techniques and algorithms to locate and classify objects within images and videos, making it a very useful task in computer vision. We discovered the common challenges that can arise like handling occlusions, varying scales, and complex backgrounds within visual data, but also gained some quick and simple tips to resolve the issues like adding methods that help improve the accuracy and robustness of object detection models. We also learned about architectures like Faster R-CNN, SSD, and YOLO, which are good to know so they can be applied to your machine, and how libraries can be installed to benefit from their uses. We now have a deeper understanding of additional definitions that are related to object detection and can say we gained more knowledge within this assignment to put to the test in the process of using object detection. Having this cheat sheet is a great tool of knowledge for beginners who are entering a machine learning career, because it serves as a valuable reference, providing quick access to key concepts, algorithms, and best practices. It helps streamline the process of selecting the adequate model architecture, helps understand various evaluation metrics, and provides preprocessing techniques. By having this cheat sheet, it enhances efficiency during model development and optimizes performance in future object detection work.