A09 ITAI 1378 CV Cheat Sheet

by massontl via cheatography.com/208718/cs/44777/

Basics of Object Detection

Object Detection. A computer vision technique used to identify and locate objects in an image or video with bounding boxes.

Annotations. Labeled data (images and their corresponding bounding boxes) used to train object detection models.

Intersection over Union (IoU). Measures the overlap between the predicted bounding box and the ground-truth bounding box.

Bounding Boxes. Rectangles drawn around detected objects, represented as (x_min, y_min, x_max, y_max).

Confidence Scores. Probability score representing the model's confidence in detecting an object.

IoU = Area of Overlap / Area of Union
Higher IoU indicates better accuracy in object localization.

Challenges in Object Detection

Objects may vary in size, position, and rotation.	Part of the object might be hidden or blocked by another object.
Some classes may have fewer examples than others, affecting model performance.	Trade-off between detection accuracy and speed for real-time applications.
Model performs well on training data but poorly on unseen data.	

Common Object Detection Algorithms

R-CNN (Region-based Convolutional Neural Network). Uses Selective Search to generate region proposals. Extracts features using CNN and applies classifiers to each region.

Fast R-CNN. Improves R-CNN by using a shared CNN feature map for all region proposals. Introduces the RoI (Region of Interest) pooling layer.

Faster R-CNN. Replaces Selective Search with a Region Proposal Network (RPN). Achieves faster region proposal generation.

SSD (Single Shot MultiBox Detector). Detects objects in a single forward pass. Uses feature maps from multiple layers for detecting objects of various sizes.

YOLO (You Only Look Once). Treats object detection as a single regression problem. Divides the image into a grid and predicts bounding boxes and class probabilities directly. Known for speed and real-time performance.

Tools and Libraries Overview (1/2)

Tensorflow	Keras
Installation: pip install tensorflow	Installation: pip install keras
Widely used for creating custom object detection models.	Provides high-level APIs to build and train models.
	Example: model.fit(x_train, y_train, epochs=10)

Object Detection Workflow

Step 1: Data Collection and Annotation.

Step 2: Data Preprocessing (resizing, normalization).

Step 3: Model Selection (R-CNN, SSD, YOLO, etc.).

Step 4: Model Training (using frameworks like TensorFlow or PyTorch).

Step 5: Model Evaluation (using metrics like IoU, precision, recall).

Step 6: Model Optimization (hyperparameter tuning, model pruning).

Step 7: Deployment (integrate the model into applications).

Tools and Libraries Overview (2/2)

OpenCV	PyTorch
Installation: pip install opencv-python	Installation: pip install torch torchvision
Useful for image processing tasks like resizing and augmentations.	Known for dynamic computational graphs, making it flexible for research.
Example: cv2.imread('image.jpg')	Example: torch.nn.Module for creating custom models.