Object Detection
Object Detection

- Sparse labeling
  - Boxes around objects
  - Pose/Keypoint estimation
- Basis of other tasks
Datasets

- Pascal VOC (2007)
  - 11k images with 27k objects in 20 classes
- MS COCO (2014)
  - 200k images with 1.5m objects in 80 classes
  - Including 250k people with pose data
- Driving datasets
  - nuScenes, BDD100k, ApolloScape, etc.
  - Usually contain 3D and temporal data
Simulated Datasets

- GTA, Carla, Habitat
- Makes labeled data cheap
- Limited relation to real-world scenarios
Region-Based Convolutional Neural Network (RCNN)

Better performance than existing models in 2013
Very slow – ~1 min per image
Fast RCNN

RCNN: pick regions here

Convolutional Layers

Fast RCNN: pick regions here

Linear Layers

Needs some processing to fit cropped activations into the linear layers: RoIPooling and RoIAlign

50-100x faster than RCNN

Use a neural network for region proposal

For several predetermined box sizes NxM, train a classifier to predict whether an NxM box is interesting or not based on only the center location

Run those classifiers at every spatial location
Faster RCNN

Convolutional Layers → Proposal Network

Convolutional Layers → Non-Maxima Suppression

Convolutional Layers → Classification

Convolutional Layers → Bounding Box Regression

All the networks can be trained together
RetinaNet

Stage 1: Proposal

Proposal Network → RoIPooling

Stage 2: Evaluation

Classification → Bounding Box Regression

RetinaNet: Combine these two stages

Focal Loss

\[ p_t = \begin{cases} 
  p & \text{if } y = 1 \\
  1 - p & \text{otherwise}
\end{cases} \]

\[ \ell_{CE}(p_t) = -\log(p_t) \]

\[ \ell_{FL}(p_t) = -(1 - p_t)^\gamma \log(p_t) \]

Reduces the weight of high-confidence samples