CSCI 8980 Think Deep Learning

Applications of CNNs in Computer Vision
Applications of CNNs in Computer Vision

- Object Detection
- Segmentation
Object Detection


Paper from UMN IRVLab: http://irvlab.dl.umn.edu/
What is Object Detection?

2 main tasks:

- Localizing one or more objects in the Image
- Classifying each object in the image


Paper from UMN IRVLab: http://irvlab.dl.umn.edu/
Object Detection Network Input and Output

(Li et al., Detection and Segmentation 2020)
Object Detection Network: Training

(Treat localization as a regression problem!

(Li et al., Detection and Segmentation 2020)
Object Detection Network: Training

(Li et al., Detection and Segmentation 2020)
Multiple Objects: Multiple outputs

CAT: (x, y, w, h)

DOG: (x, y, w, h)
DOG: (x, y, w, h)
CAT: (x, y, w, h)

DUCK: (x, y, w, h)
DUCK: (x, y, w, h)

....

(Li et al., Detection and Segmentation 2020)
Multiple Objects: Initial Solution

Dog? NO
Cat? NO
Background? YES

(Li et al., Detection and Segmentation 2020)
Multiple Objects: Initial Solution

Dog? NO
Cat? NO
Background? YES

Dog? YES
Cat? NO
Background? NO

(Li et al., Detection and Segmentation 2020)
Multiple Objects: Heavy computational cost

Apply a CNN to many different crops of the image, CNN classifies each crop as object or background.

Dog? NO
Cat? YES
Background? NO

Problem: Need to apply CNN to huge number of locations, scales, and aspect ratios, very computationally expensive!

(Li et al., Detection and Segmentation 2020)
Solution: 4-step Object-detection framework

1. **Region Proposal**: identify regions of interest (RoI) for potential locations of objects
2. **Feature extraction**: extract visual features within each RoI for classification
3. **Non-maximum Suppression**: avoid repeated detections
4. **Evaluation metrics**: evaluate performance of model
1. Region Proposal:

Identification of Regions of Interest (RoIs)

- General procedures for Region Proposal:
  - Generate thousands of bounding boxes (BBs)
  - Classify BBs as foreground or background based on 'objectness score'
  - Pass only foreground through rest of network

- One common approach is using Selective Search
  - Fast algorithm, ~200 region proposals in a few seconds on CPU

(Elgendy, 2020)
Selective Search

Greedy search algorithm for region proposal

Step 1:

- Segmentation
- Defining ‘blobs’ that Could be objects

(Elgendy, 2020)
Selective Search

Input image  Proposed regions  After the first iteration  After a few iterations

(Elgendy, 2020)
2. Feature Extraction & Classification in RoI

Using a pretrained CNN network,

- Extract features using a pretrained CNN

- Make 2 predictions using additional layers:
  - Bounding box prediction (x, y, width, height)
  - Class prediction (softmax function predicting the class probability
    For each object

(Elgendy, 2020)
3. Remove Duplicate Object Detections

Non-maximum suppression (NMS):
A 4-step technique for eliminating duplicate detections of objects.

1. Discard BBs with predictions below a confidence threshold.
2. Select the BB with the highest probability.
3. Calculate the overlap of all remaining boxes with the same class prediction.
4. Suppress any box with an IoU smaller than a threshold (NMS threshold, usually 0.5).
3. Remove Duplicate Object Detections

Non-maximum suppression (NMS): A 4-step technique for eliminating duplicate detections of objects.

1. Discard BBs with predictions below a confidence threshold.
2. Select the BB with the highest probability.
3. Calculate the overlap of all remaining boxes with the same class prediction.
4. Suppress any box with an IoU smaller than a threshold (NMS threshold, usually 0.5).

(Elgendy, 2020)
4. Evaluation Metrics for Detector Performance

Once an object detector has been developed, it is typically evaluated using two main metrics:

1. Frames per second (FPS) - detection speed
2. Mean Average Precision (mAP) - network precision

mAP calculated from a Bounding Box’s object score and the precision-recall curve

(Elgendy, 2020)
4. Evaluation of Detector Performance: IoU

Intersection over Union: IoU

\[
\text{IoU} = \frac{B_{\text{ground truth}} \cap B_{\text{predicted}}}{B_{\text{ground truth}} \cup B_{\text{predicted}}}
\]

Figure 7.5 The IoU score is the overlap between the ground truth bounding box and the predicted bounding box.

(Elgendy, 2020)
4. Evaluation of Detector Performance: Precision-Recall

Precision-Recall (PR):

\[
\text{Recall} = \frac{TP}{TP + FN}
\]

\[
\text{Precision} = \frac{TP}{TP + FP}
\]

(Elgendy, 2020)
State of the Art Object Detection CNNs

- R-CNNs
- SSD
- YOLO
R-CNNs: Region-based CNNs

R-CNN family of networks:

- R-CNN
- Fast-RCNN
- SOTA: Faster-RCNN

R-CNN architecture ->

(Elgendy, 2020)
R-CNNs: Region-based CNNs

R-CNN:

1. Selective search algorithm is used to extract RoIs from the input image.
2. Extracted regions are warped before being fed to the ConvNet.
3. Forward each region through the pretrained ConvNet to extract features.
4. The network produces bounding-box and classification predictions.

(Elgendy, 2020)
Fast R-CNN

Improves on R-CNN in both detection speed and accuracy.

Architecture changes:

- CNN feature extractor first applied to entire image, region proposals performed afterward
  - Only run one CNN instead of ~2000 CNNs on overlapping RoIs

- CNN performs both the classification and feature extraction
  - SVM machine replaced with a softmax layer

(Elgendy, 2020)
Faster R-CNN: State of the Art

Architecture

- Same overall structure as Fast R-CNN except for Region proposal algorithm
- Selective search replaced with Region proposal network, which Outputs
  - Objectness score
  - Bounding box location

(Elgendy, 2020)
Multi-stage vs Single-Stage Detectors

Multistage detectors:

- First Identifies Regions of interest & objective score
- Later outputs final Bounding boxes

Single-stage detectors:

- Identifies Regions of Interest and final bounding boxes together
SSD Single-Shot Detector

- Pretrained network extracts features; cut off before Classification
- Convolutional layers allow for detections at different scales
- Non-maximum suppression Eliminates overlapping BBs to keep one detection per object
- Outputs a set number of BBs prior to NMS.
SSD Feature Extraction

- Anchors overlaid over image
- Bounding boxes created with Anchors at their center
- Network considers each BB a separate image
- If feature extractor found boat Features in BB, BB sent on to NMS layer
Multi-scale Feature Layers

Convolutional feature layers after the base network
SSD Single-Shot Detector

- Output of SSD

An example visualization of the output prediction when we have four classes in our problem. The convolutional layer predicts the bounding box coordinates, objectness score, and four class probabilities: $C_1$, $C_2$, $C_3$, and $C_4$. 

(Elgendy, 2020)
YOLO: Real-time Object Detection

- No region proposal
  Network

- Performs predictions
  Based on a grid of cells

- Each cell directly predicts
  BBs and classification

- NMS yields final prediction

(Elgendy, 2020)
Applications of CNNs in Computer Vision

- Object Detection
- Segmentation
Segmentation


Paper from UMN IRVLab: http://irvlab.dl.umn.edu/
What is Semantic Segmentation?

FISH, DIVER, BACKGROUND, AQUATIC PLANTS, SEAFLOOR

Training data paired:
Each pixel labeled with a semantic category.

During test, classify each pixel of the new image.

-Paper from UMN IRVLab http://irvlab.dl.umn.edu/
Semantic Segmentation: Sliding Window

Impossible to classify without context!

How do we include context?

Paper from UMN IRVLab  http://irvlab.dl.umn.edu/
Semantic Segmentation: Sliding Window

(ISlam et al., "Semantic Segmentation of Underwater Imagery: Dataset and Benchmark," 2020)

Paper from UMN IRVLab http://irvlab.dl.umn.edu/

(Li et al., Detection and Segmentation 2020)
Semantic Segmentation: Sliding Window

Paper from UMN IRVLab  http://irvlab.dl.umn.edu/
(Li et al., Detection and Segmentation 2020)
This is very inefficient! It re-identifies shared features for each overlapping patch.
Intuition: encode the entire image with a CNN, then do semantic segmentation at the end.

Challenge: Classification architectures reduce feature sizes as they go deeper into the network; Semantic segmentation requires output size == input.

Paper from UMN IRVLab  http://irvlab.dl.umn.edu/
(Li et al., Detection and Segmentation 2020)
Semantic Segmentation Idea: Convolution

Eliminate downsampling in the network:

Input: \(3 \times H \times W\)

Convolutions: \(D \times H \times W\)

Scores: \(C \times H \times W\)

Predictions: \(H \times W\)

Paper from UMN IRVLab http://irvlab.dl.umn.edu/
(Li et al., Detection and Segmentation 2020)
Semantic Segmentation Idea: Convolution

Issue: very computationally expensive!

Input: \(3 \times H \times W\)

Convolutions: \(D \times H \times W\)

Scores: \(C \times H \times W\)

Predictions: \(H \times W\)

Paper from UMN IRVLab  http://irvlab.dl.umn.edu/
(Li et al., Detection and Segmentation 2020)
Semantic Segmentation Idea: Convolution

Solution: add both downsampling and upsampling inside network!

(Paper from UMN IRVLab http://irvlab.dl.umn.edu/)
(Li et al., Detection and Segmentation 2020)
Semantic Segmentation Idea: Convolution

Downsampling:
Pooling, strided convolution

Upsampling: from 1st half of lecture

Input: 3 x H x W

High-res: D_1 x H/2 x W/2

Med-res: D_2 x H/4 x W/4

Low-res: D_3 x H/4 x W/4

High-res: D_1 x H/2 x W/2

Predictions: H x W


Paper from UMN IRVLab http://irvlab.dl.umn.edu/

(Li et al., Detection and Segmentation 2020)
Semantic Segmentation: Summary

PAPER FROM UMN IRVLab  http://irvlab.dl.umn.edu/
(Li et al., Detection and Segmentation 2020)
Semantic Segmentation: Summary

Goal: label each pixel in the image with a category label. Don’t differentiate between different instances of the same class of object; only care about the pixel-level.

<table>
<thead>
<tr>
<th>Object category</th>
<th>RGB color</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background (waterbody)</td>
<td>000</td>
<td>BW</td>
</tr>
<tr>
<td>Human divers</td>
<td>001</td>
<td>HD</td>
</tr>
<tr>
<td>Aquatic plants and sea-grass</td>
<td>010</td>
<td>PF</td>
</tr>
<tr>
<td>Wrecks or ruins</td>
<td>011</td>
<td>WR</td>
</tr>
<tr>
<td>Robots (AUVs/ROVs/instruments)</td>
<td>100</td>
<td>RO</td>
</tr>
<tr>
<td>Reefs and invertebrates</td>
<td>101</td>
<td>RI</td>
</tr>
<tr>
<td>Fish and vertebrales</td>
<td>110</td>
<td>FV</td>
</tr>
<tr>
<td>Sea-floor and rocks</td>
<td>111</td>
<td>SR</td>
</tr>
</tbody>
</table>

Paper from UMN IRVLab  http://irvlab.dl.umn.edu/
State of the Art
Segmentation
CNNs

Paper from UMN IRVLab  http://irvlab.dl.umn.edu/
UNET: Semantic Segmentation

- Applied to medical imaging to identify Tumors

- Main contribution: addition of "Shortcut connections:

- Compensates for information loss, Since not all features can be re-created By the decoder

(Ronneberger et al., 2015)
Mask R-CNN: Instance Segmentation

- Extension of Faster R-CNN
- Adds a masking network after the Output of Faster R-CNN
- Masking network outputs a Segmentation mask for each Object instance

Figure 1. The Mask R-CNN framework for instance segmentation.

(He et al., 2018)
Mask R-CNN: Instance Segmentation

Figure 2. Mask R-CNN results on the COCO test set. These results are based on ResNet-101 [19], achieving a mask AP of 35.7 and running at 5 fps. Masks are shown in color, and bounding box, category, and confidences are also shown.

(He et al., 2018)
Popular Datasets for Classification, Detection, and Segmentation

- COCO (172 classes, common benchmark dataset)
  - [http://cocodataset.org/#home](http://cocodataset.org/#home)

- Cityscapes (roads, lanes vehicles, objects on roads)
  - [https://www.cityscapes-dataset.com/](https://www.cityscapes-dataset.com/)

- Pascal Context (real-world; over 400 classes)
  - [https://cs.stanford.edu/~roozbeh/pascal-context/](https://cs.stanford.edu/~roozbeh/pascal-context/)

- Lits (medical imaging, CT scans)
  - [https://competitions.codalab.org/competitions/17094](https://competitions.codalab.org/competitions/17094)

- Inria Aerial Image Labeling
Acknowledgements


