SSD: Single Shot MultiBox Detector

Abstract

- Method for detecting objects in images using a single deep neural network.
- Applies a default number of bounding boxes over different aspect ratios and scales per feature map location.
- At prediction time, the network generates scores for the presence of each object category in each defauly bos and produces adjustments to the box to better match the object shape.
- Combines predictions from multiple feature maps with different resolutions to naturally handle objects of various sizes.
- Eliminates proposal generation and subsequent pixel or feature resampling stages and encapsulates all computation in a single network.

Introduction

- Most methods make bounding box proposals, then resample features and pixels for each box and apply a high quality classifier to obtain detection. These have been too computationally intensive for embedded systems and too slow for real-time applications.
- Even Faster R-CNN operates at only 7fps.
- The fundamental improvement in speed from SSD is the elimination of bounding box proposals and the subsequent pixel or feature resampling stage.
- SSD, improves on previous methods that use the same approach by:
  - using a small convolutional filter to predict object categories and offsets in bounding box locations.
  - Using separate filters for different aspect ratio detections and applying these filters to multiple feature maps from the later stages of a network in order to perform detection at multiple scales.

The Single Shot Detector (SSD)

Model

- Produces a **fixed-size collection of bounding boxes** and scores for the presence of objects in those boxes followed by a non-max supression step to produce the final detections.
- The early network layers are based on a standard architecture(base network) used for high quality image classification(without the final fully connected layers used for classification). Auxiliary structures is added to the network to produce detections.
- **Multi-scale feature maps for detection**:
  - Convolutional layers are added to the end of the truncated base network.

  - These layers decrease in size progressively and allow predictions of detections at **multiple scales**.
  - **The convolutional model for predicting detections is different for each feature layer.**
- **Convolutional predictors for detection**:
  - Each added feature layer, or an existing feature layer from the base network can produce a fixed set of detection predictions using a set of convolutional filters. For a feature layer of size m x n with p channels, a kernel of 3 x 3 x p is used as the basic element for predicting parameters of a potential detection.
  - This kernel can either produce a score for a category or a shape offset relative to the default box coordinates.
  - The bounding box offset output values are measured relative to a default box position relative to each feature map location.
- **Default boxes and aspect ratios**:
  - A set of default bounding boxes is associated with each feature map cell for multiple feature maps.
  - The defauly boxes tile the feature map in a convolutional mannger so that the position of each box relative to its corresponding cell is fixed.
  - At each feature map cell, bounding box offsets relative to the default box shapes in the cell as well as the per-class scores are predicted.
  - For each such default box out of k default boxes applied at each location, class scores for the c classes and the 4 tuple for the offset is computed which results in $(c+4)kmn$ outputs for a mxn feature map.
  - Default boxes similar to the notion of anchors in Faster R-CNN. Applying them to several feature maps allows for different resolutions.

NOTE:

In short, for each convolution layer(added after the base network), the output feature layer is fed into a detector network that generates $(c+4)kmn$ outputs for the k default boxes considered at each of the mn cells, providing c probability values for the classification task and 4 values corresponding to the regression box offsets.

Training

- Difference between training a typical detector that uses proposals and SSD is that ground truth information needs to be assigned to specific outputs in the fixed set of detector outputs.
- **Matching Strategy**:
  - During training, it needs to be determined which of the default boxes correspond to a ground truth detection and traing the network accordingly.
  - However, this can be slightly difficult as, for each ground truth box,

a corresponding "best match" default box needs to be selected from default boxes that vary over location, aspect ratio and scale.
  - This is done by:
    * first matching each ground truth box to the default box with the best jaccard overlap.
    * then, matching default boxes to any ground truth box with jaccard overlap higher than a threshold of 0.5
  - The second type of matching simplifies the learning problem allowing the network to predict high scores for multiple overlapping default boxes rather than to pick one with the maximum overlap. The question of maximum overlap can be handled by a per-class non-max suppression.
- **Training objective**:
  - The SSD training objective is derived from the MultiBox objective but is extended to handle multiple object categories.
  - It is the average taken over the number of matched default boxes of the regression loss and the classification loss.
  - The regression loss follows the same smooth L1 loss as used in Faster R-CNN with offsets taken with respect to the centre of the default box. The loss considers only default boxes that are matched to a ground truth box.
  - The classification loss is the softmax loss over the classes confidences.
  - As with Faster R-CNN, a parameter alpha is used to balance the loss contributions between classification and bounding box regression and is set to 1.
- **Choosing scales and aspect ratios for default boxes**:
  -