Annexure I

Image (I) An image $I \in \mathbb{R}^{m\times n}$ can be defined as a set of pixels that contains projections of objects belonging to a set of object classes $C \in \mathbb{R}^{N\times 1}$. An image $I_0 \in \mathbb{R}^{p\times q}$ can be called a subimage of I if: (i) $p \leq m$, $q \leq n$, and (ii) $I(i+l, j+k) = I_0(l,k)$ for all $(l = 1,\ldots,p), (k = 1,\ldots,q)$ and $(i = 1,\ldots,m), (j = 1,\ldots,n)$.

Bounding Box (BB) Corresponding to each object belonging to class $C_i \in C; i = 1,\ldots,N$ in I, there exists a set of sub-images that include the object. Among them, the minimal sub-image that entirely covers the object and provides its location and area information in I is considered the bounding box of the object. A bounding box of one object can be a sub-image of another, but it will not be the bounding box of that object if it is not the minimal sub-image enclosing that object.

Object Detection (OD) Object detection in an image can be defined as a mapping between a set of bounding box sub-images $I_s \subset I$ and a set of object classes C present in I. It can be parameterized by P, a set of position vectors of each element of I_s with respect to I, and C^*, a set of class vectors for each element of P. Thus, an object detector is a function of some parameter θ that takes an image I and produces a set of predictions for C^* and P, i.e.,

$$f_\theta(I) = \hat{Y} = (\hat{C^*}, \hat{P})$$

where $\hat{C^*}$ and $\hat{P}$ denote the predicted class and position vectors, respectively.

CNN-Based Object Detector (CNNOD) A CNN-based detector is composed of multiple layers of 2-D convolution filters (or kernels). In the first layer, a set of filters are convolved with each pixel of the image to extract attributes such as edges, textures, or shapes—known as features. Thus, for a given image I and a filter $K \in \mathbb{R}^{H_k\times W_k}$, the output feature for the pixel at location (m,n) is:

$$\phi(m,n) = \sum_{i=0}^{H_k-1}\sum_{j=0}^{W_k-1} I(m+i, n+j)\cdot K(i,j)$$

where H_k, W_k are the height and width of the filter.

The set of all output features from a layer is called the *feature map*, i.e.,

$$\Phi_i = \{\phi(i,j) \mid i \in I, j \in J\}$$

where I and J are index sets.

For the subsequent layers in the CNN detector, the feature map from the previous layer is taken as input to generate another feature map as output:

$$\Phi_i = f_{\theta_i}(\Phi_{i-1})$$

where $\theta_i \in \theta$ is the parameter subset of the ith layer, and θ is the set of all filter parameters of the detector, tuned using a dataset $D = \{I_D, Y_D\}$. The dataset contains N images of size $m\times n$ with c color channels, i.e., $I_D \in \mathbb{R}^{N\times m\times n\times c}$, and corresponding ground-truth annotations $Y_D \in \mathbb{R}^{N\times(n(C)+|P_i|)\times M}$, where $n(C)$ is the cardinality of class set C, $|P_i|$ is the number of attributes of each element of P, and M is the maximum number of detectable objects per image.

Annotation Function To evaluate prediction error, the output of f_θ is compared with actual data—sets of ground-truth position vectors P and class vectors C^*. The actual data is produced by the annotation function $A(.)$, which takes an image I as input and outputs ground-truth position and class vectors for each element of I_s:

$$A(I) = Y = \{(C_i^*, P_i) \mid i \in \{1, \dots, Q\}\}$$

where Q is the number of objects in I.

Loss Function The error between predicted and ground-truth data is computed using a loss function $\mathcal{L}(.)$. Since the detector $f_\theta(.)$ produces two output sets, the total loss is a weighted sum of the individual losses:

$$\mathcal{L}(f_\theta(I), A(I)) = \alpha\, \mathcal{L}(\hat{C}^*, C^*) + \beta\, \mathcal{L}(\hat{P}, P)$$

where α, β are weight coefficients.

For a given image I, the goal of tuning θ is to minimize the error between predicted and actual outputs:

$$f_{\theta^*}(I) = \arg\min_\theta \mathcal{L}(f_\theta(I), A(I))$$

Precision and Recall For any ground-truth annotation subset $Y_i(C_i^*, P_i) \in Y_D$, the prediction $\hat{Y}_i(\hat{C}_i^*, \hat{P}_i) \in \hat{Y}_D$ by the tuned model f_{θ^*} is said to be a true positive if the Intersection over Union (IoU) of the ground-truth and predicted bounding boxes exceeds a threshold TH:

$$\text{True Positive} = \frac{(P_i \cap \hat{P}_i)}{(P_i \cup \hat{P}_i)} > TH$$

For a given image $I_i \in I_D$,

$$\text{Precision} = \frac{\text{True Positives}}{|\hat{P}_i|}, \quad \text{Recall} = \frac{\text{True Positives}}{|P_i|}$$

Mean Average Precision (mAP) For a given class $C_i \in C$, the Average Precision (AP) is the area under the precision–recall curve obtained from all elements of Y_D, i.e.

$$P = \{P_i \mid (C_i, P_i) \in Y_D, i \in \{1, 2, \dots, Q\}\}$$

The mean of APs across all classes is the mean Average Precision (mAP), which serves as a key performance metric for object detectors.