Algorithm 1 Calculation of anchors hyperparameters for EfficientDet

Require: $B_1, B_2, .., B_n$ - list of n bounding boxes, T_{32}, T_{64}, T_{128}, T_{256}, T_{512} - template anchor sizes for each feature map.

Ensure: $A_1, A_2, .., A_k$ - list of k anchors hyperparameters

1: Calculate the k shapes of anchors using K-means with the Jaccard distance metric on the list of bounding boxes.
2: Calculate the list of aspect ratios R by dividing the height of each shape with its width.
3: **for** each shape s_i in the list of k shapes **do**
4: Calculate $s_i = \max(width(B_i), height(B_i))$
5: Calculate $A_i = T_{A_i} \times \max(width(B_i), height(B_i))/size(T_{A_i})$, where T_{A_i} is the template anchor size that best matches the calculated shape s_i.
6: **end for**
7: Merge the similar scales in A to decrease the number of anchors.
8: Return the list of anchors hyperparameters $A_1, A_2, .., A_k$

Algorithm 2 Anchor calculation for EfficientDet

Require: A list of bounding boxes B with corresponding image sizes I

Ensure: A list of anchor boxes A with aspect ratios and scales

1: Calculate the k shapes of anchors using K-means with the Jaccard distance metric on the list of bounding boxes B
2: Convert the obtained anchor widths and heights into a list of aspect ratios R and scales S
3: **for** i in range(k) **do**
4: $s_i = \max(width(B_i), height(B_i))$
5: $T_{A_i} = \arg\min |size(T_j) - s_i|$ where $T_j \in 32, 64, 128, 256, 512$
6: $S_i = \max(B_i)/T_{A_i}$
7: **end for**
8: Merge the scales in S that are similar
9: Return the final list of anchor boxes A

Algorithm 3 Calculation of EfficientDet anchors' hyperparameters

Require: List of bounding boxes $B_1, B_2, \ldots, B_n$

Ensure: List of aspect ratios and scales for EfficientDet anchors

Calculate the k shapes of anchors using K-means with the Jaccard distance metric on the list of bounding boxes:

    $IOU(B_i, A_j) = \frac{area(B_i \cap A_j)}{area(B_i \cup A_j)}$

    $D_{ij} = 1 - \frac{area(B_i \cap A_j)}{area(B_i \cup A_j)}$

    $A_1, A_2, \ldots, A_k \leftarrow$ K-means($B_1, B_2, \ldots, B_n$, D, k)

Calculate aspect ratios:

    $aspectratio_j = \frac{height(A_j)}{width(A_j)}$

Calculate scales:

    $s_i = \max(width(B_i), height(B_i))$

    $T_{A_i} = \arg\min |size(T_j) - s_i|$, where $T_j \in 32, 64, 128, 256, 512$

    $scale_i = \frac{\max(width(A_i), height(A_i))}{size(TA_i)}$

Merge similar scales:

    $S = scale_1, scale_2, \ldots, scale_k$

    Sort S in increasing order j $\leftarrow 1$

    while $j < |S|$:

        $q \leftarrow j + 1$

        while $q \leq |S|$ and $\frac{S_q}{S_j} < merge\_threshold$:

            $q \leftarrow q + 1$

        $scale_j \leftarrow \frac{\sum_{i=j^{q-1}} S_i}{q-j}$

        $j \leftarrow q$

return $aspectratio_1, aspectratio_2, \ldots, aspectratio_k$, scale$_1, scale_2, \ldots, scale_k$