Scale-Equivalent Distillation for Semi-Supervised Object Detection

Qiushan Guo¹, Yao Mu¹, Jianyu Chen², Tianqi Wang¹, Yizhou Yu¹, Ping Luo¹
¹The University of Hong Kong ²Tsinghua University
{qsguo, ymu, tqwang, yzyu, pluo}@cs.hku.hk  jianyuchen@tsinghua.edu.cn

A. Appendix

A.1. Implementation of SED on DETR.

Our method can be extended to DETR, a single feature map detector based on anchor-free label assignment rule. We match the predictions of input in different views according to Hungarian algorithm, where the pair-wise matching cost is defined as: 

\[ C_{match} = D_{JS}(p_1, p_2) + \lambda L_{IoU}(b_1, b_2) \]

where \( D_{JS} \) is JS-Divergence between the probability vectors and \( L_{IoU} \) is GIoU loss [10]. The python-style pseudo-code of matching algorithm is provided in Alg. 1.

Algorithm 1 Matching Pseudocode, PyTorch-like

```python
def hungarian_match(cls_score_1, cls_score_2, bbox_pred_1, bbox_pred_2, cls_weight, iou_weight):
    # cls_score: [bs, num_query, c]
    # bbox_pred: [bs, num_query, 4]
    cls_dist = JSCost(cls_score_1, cls_score_2)
    iou_dist = IoUCost(bbox_pred_1, bbox_pred_2)
    cost = cls_dist*cls_weight + iou_dist*iou_weight
    bs = cost.shape[0]
    col_inds = []
    for i in range(bs):
        col_ind = linear_sum_assignment(cost[i])
        col_inds.append(col_ind)
    return col_inds
```

A.2. Stronger augmentations.

Geometric augmentations are common image data augmentations. Therefore, we further conduct experiments with stronger augmentations: color + geometric augmentations, to demonstrate the extendability of SED. We simply adopt the same geometric transformations in RandAug [3], including RandRotate, RandTranslation and RandShear. We set the rand level to 5 and select only 1 transformation to apply. The results in Tab. 1 show that additional geometric augmentations lead to incremental improvement.

A.3. Implementation and Training Details.


**Training Details.** The weights of the backbone are first initialized by the corresponding ImageNet-Pretrained model, which is a default setting in existing works [6,8,11,14]. All the models are trained with learning rate starting at 0.01. The learning rate drops by 0.1 at the 120k and 160k iteration for 180k training schedule as default. We set the weight decay to 0.0001, batch size to 16, and the momentum is 0.9 for SGD optimizer. Like [8], we separate 5k/10k/12k/90k iterations from the whole process as the burn-in phase for 5%/10%/35k/100% data protocols. For verifying the effectiveness of our method, we simply set the \( \lambda_s \) and \( \lambda_d \) in Eq. 1 as 0.5 and 1 separately. The EMA update rate starts with 0.99 and steps to 0.9 at the 120k iteration, aligned with the learning rate decay policy.

**Data Augmentation.** As shown in Tab. 2, the weak data augmentation only contains random resize from (1333, 640) to (1333, 800) and random horizontal flip with a probability of 0.5. The strong data augmentation is composed of random Color Jittering, Grayscale, Gaussian Blur, and Cutout [4], without any geometric augmentation.
<table>
<thead>
<tr>
<th>Process</th>
<th>Probability</th>
<th>Parameters</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color Jittering</td>
<td>0.8</td>
<td>brightness, contrast, saturation = 0.4, 0.4, 0.4</td>
<td>Brightness factor is chosen uniformly from [0.6, 1.4], Contrast factor is chosen uniformly from [0.6, 1.4], Saturation factor is chosen uniformly from [0.6, 1.4]</td>
</tr>
<tr>
<td>Grayscale</td>
<td>0.2</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>GaussianBlur</td>
<td>0.5</td>
<td>$\sigma \sim U(0.1, 2.0)$</td>
<td>Gaussian filter kernel size is 23</td>
</tr>
<tr>
<td>Cutout 1</td>
<td>0.7</td>
<td>scale=(0.05, 0.2), ratio=(0.3, 3.3)</td>
<td>Randomly selects a rectangle region in an image</td>
</tr>
<tr>
<td>Cutout 2</td>
<td>0.5</td>
<td>scale=(0.02, 0.2), ratio=(0.1, 6)</td>
<td>Randomly selects a rectangle region in an image</td>
</tr>
<tr>
<td>Cutout 3</td>
<td>0.3</td>
<td>scale=(0.02, 0.2), ratio=(0.05, 8)</td>
<td>Randomly selects a rectangle region in an image</td>
</tr>
</tbody>
</table>

Table 2. Details of data augmentations.

References


