Supplementary: Few-Shot Domain Adaptive Object Detection for Microscopic Images

Sumayya Inayat, Nimra Dilawar, Waqas Sultani, Mohsen Ali

Intelligent Machines Lab, Department of Artificial Intelligence,
Information Technology University, Pakistan
{sumayya.inayat, phdcs21003, waqas.sultani, mohsen.ali}@itu.edu.pk

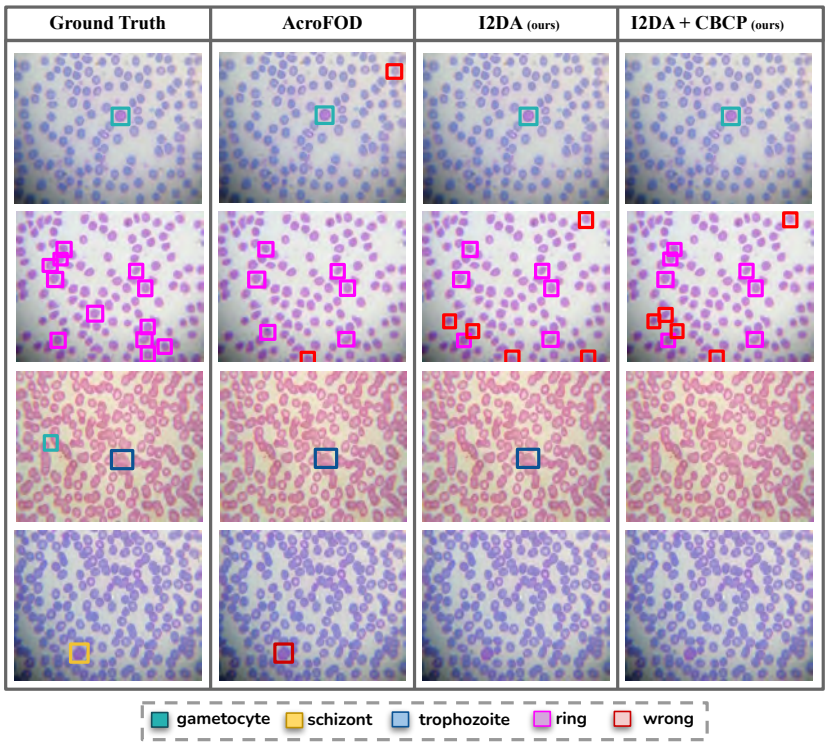

Fig. 1: Qualitative results of failed examples of LCM Malarial defected regions after adaptation. From Left to right, Ground Truth, AcroFOD, I2DA(ours) without augmentation, I2DA+cbcp(ours) with augmentation.

YOLOv5 return best.pt and last.pt weights of the trained model as per the validation data. Paper contains the last.pt results.

Table 1: Malaria test set - mAP@50(%) on best.pt.

Malaria-HCM-1000x → Malaria-LCM-1000x															
Method	mAP@50(%)			Gametocyte			Schizont			Trophozoit			Ring		
Shots	2	3	5	2	3	5	2	3	5	2	3	5	2	3	5
AsyFOD	26.0	28.0	29.7	14.9	17.1	23.0	1.2	0.8	1.8	59.4	**62.5**	63.6	28.7	31.8	30.3
AcroFOD	27.2	28.7	40.4	16.6	22.7	60.5	0.9	1.1	7.8	62.5	59.2	62.1	**29.1**	31.8	31.2
Ours	**38.7**	**38.8**	**48.9**	**57.6**	**54.2**	**68.2**	**2.4**	**6.6**	**30.4**	**65.7**	61.7	**65.7**	28.9	**32.6**	**31.5**

Table 2: Raabin-WBC test set - mAP@50(%) on best.pt.

Raabin-WBC-HCM → Raabin-WBC-LCM															
Method	mAP50(%)			Large Lymph			Neutrophil			Small Lymph			Monocyte		
Shots	2	3	5	2	3	5	2	3	5	2	3	5	2	3	5
AsyFOD	36.2	38.7	32.2	41.0	45.8	49.8	62.6	53.3	56.4	**40.7**	**45.2**	22.1	0.3	10.4	0.3
AcroFOD	54.5	49.2	64.5	**75.0**	63.0	83.0	**92.3**	88.9	**94.3**	34.1	38.6	**66.2**	16.7	6.30	14.4
Ours	55.2	58.8	70.7	71.9	**68.0**	**86.2**	80.1	**91.0**	93.8	37.3	37.1	44.2	**31.6**	**39.0**	**58.7**

Table 3: best.pt results in mAP@50(%) on M5 and RWBC test sets on 8 random few-target images.

Data	Malaria					Raabin-WBC				
Sets	mAP@50	Gamet.	Schizo.	Troph.	Ring	mAP@50	L-Lymp.	Neutro.	S-Lymp.	Mono.
AsyFOD	26.2	4.5	**6.8**	62.3	**31.4**	30.2	28.9	50.5	40.3	0.9
AcroFOD	34.7	50.3	5.4	60.6	22.5	50.8	59.1	59.5	**58.1**	26.3
Ours	**42.2**	**68.8**	3.0	**65.8**	31.3	**55.7**	63.5	79.6	45.0	**34.5**

Algorithm 1 Get Increment Stats

Input: $\mathcal{O}_c, \mathcal{I}_f$
$inc_{stats} \leftarrow \{\}$
$max \leftarrow \max(\mathcal{O}_c)$
for each c in keys of $\mathcal{O}_c$ **do**
  if $max > \mathcal{O}_c[O]$ **then**
    $\mathcal{S}^c \leftarrow \frac{max - \mathcal{O}_c[c]}{\mathcal{O}_c[c]}$
    $total_{i_f} \leftarrow \frac{\mathcal{S}^c \times \mathcal{O}_c[c]}{\text{len}(\mathcal{I}_f)}$
    if $total_{i_f} < 0$ **then**
      $total_{i_f} \leftarrow 1$
    end if
    $times_{i_f} \leftarrow \frac{\mathcal{S}^c}{total_{i_f}}$
    $inc_{stats}[c] \leftarrow [\mathcal{O}_c[c], times_{i_f}, total_{i_f}]$
  end if
end for
return inc_{stats}