

一、Question

找出圖中物體鋪粉後的瑕疵檢測。

二、Method

利用 maskrcnn 的物件偵測與語意分割，找出物體鋪粉後的瑕疵。

Backbone : Resnet101

Anchor :

Scales=[4, 8, 16, 32, 48, 96, 216, 480, 640]

Ratios=[0.1, 0.2, 0.5, 1, 2, 5, 10, 25, 50, 60, 70]

將 anchor 的大小與長寬比例對不同的資料集做參數調整

Loss :

Classification = Cross-entropy loss

Bounding box = L1 loss

Mask = Cross-entropy loss

Epoch : 250

Batch size : 2

Image size : 800x800、640x640

Data augmentation : 50% 的機率隨機水平或垂直翻轉 (RandomFlip)

Optimizer :

```
optimizer = dict(type='SGD', lr=0.001, momentum=0.9,  
weight_decay=0.0001)
```

```
lr_config = dict(  
    policy='step',  
    warmup='linear',  
    warmup_iters=500,  
    warmup_ratio=0.001,  
    step=[100, 150, 200])
```

用 SGD 做 optimizer，learning rate 為 $10e-3$ ，並在第 100、150 與 200 epoch 時，learning rate 乘上 0.1

三、Result

4-fold :

Bounding box mAP 為 70.7%、73.9%、76.5% 與 78.6%

Segmentation mAP 為 69.5%、73.1%、74.2% 與 77.4%

平均 Bounding box mAP 是 74.925%

平均 Segmentation mAP 是 73.55%

Val data :

AP50 在 uncover、uneven 與 scratch 上分別是 89%、100% 與 100%

AP50 平均為 96.4%



四、Discussion

1. 資料差異性大

同一種的 `powder_uncover`，就有水滴狀、脊椎狀到鑰匙狀等等，不同種類的語意分割圖，因此在做 4-fold 訓練時，切割資料也需要把這些差異考量進去，則無法訓練出結果較好的模型。

2. Bounding box 標註問題

Bounding box 標註有點瑣碎，像三個鑰匙狀組成的語意分割圖竟需要 11 個 bounding box 做標註，因此造成模型容易學不起來，但藉由調整 anchor 可以解決此問題。

3. Maskrcnn 的 anchor 調整

Anchor-based 的物件偵測模型需要因應資料集的不同，調整 anchor 的大小與長寬比例，人工調整 anchor，使其大小與比例能涵蓋整個資料集的 bounding box 後，準確率有顯著的提升。

五、Conclusion

利用資料分析與模型 backbone、anchor 到 hyper-parameter 的調整，使 maskrcnn 能針對該資料集有較好的預測結果。