**影像處理(Image Processing) Final Project**

NE6114011 人工智慧科技碩士學位學程碩一 楊雲翔

Questions

本次的作業的目標是要檢測在金屬3D列印雷射過程中，由雷射粉體燒融(Selective laser melting, SLM)產生的瑕疵。首先，需要讀入一張含有金屬外觀的影像，進行物件偵測，偵測出瑕疵的位置，瑕疵分為三個種類，分別為powder uncover, powder uneven, scratch，接著在進行影像分割，分割出瑕疵的更精確位置。

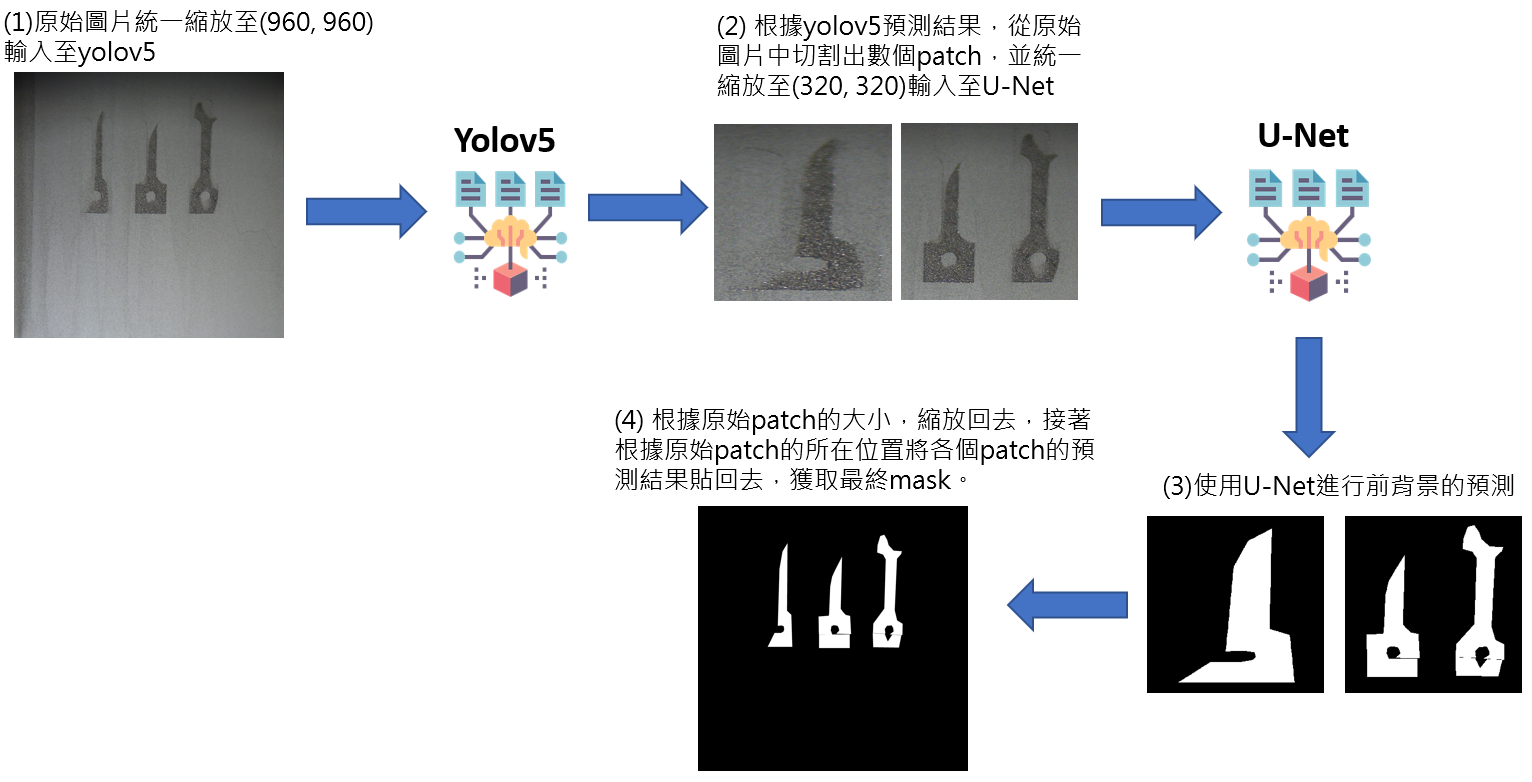
Methods

整個瑕疵檢測的過程分為兩步：

1. 物件偵測：輸入原始影像，偵測出瑕疵的位置及種類
2. 影像分割：根據物件偵測的結果，從原始圖片中分割出數張子圖片，再根據子圖片進行影像分割，更進一步找出瑕疵的具體位置

其中物件偵測的部分，我採用了yolov5來實現，yolov5是yolo系列中的一員，為一one-stage object detection model，也就是將候選框定位與分割結合為一步來完成。一個one-stage object detection model通常由Backbone + Neck + Head三部分組成，其中backbone用於提取特徵，neck用於特徵融合，head則負責最後的候選框位置回歸與候選框的分類，yolov5的backbone為CSP Net(Cross Stage Partial Network)，neck由FPN(Feature Pyramid Networks) + PAN(Path Aggregation Network)，head則是由三個檢測模型組成，用於偵測不同尺度的物件。此外，由參數量多至少，yolov5又可以分為四個版本：Yolov5s、Yolov5m、Yolov5l、Yolov5x，我採用的版本為Yolo5m。

另外，影像分割的部分，我採用了unet來實現，unet利用全卷積模型來實現影像分割，此外，Unet可以視為Autoencoder的一種變體，因為模型結構類似U型結構而得名，一個autoencoder會由一個編碼器(encoder)與一個解碼器(decoder)所組成，編碼器負責萃取特徵，隨著不斷的萃取，輸出的特徵圖尺寸會愈來愈小，之後，解碼器(decoder)會利用這些變小的特徵來重建圖像，與傳統的autoencoder相比，Unet在原有的編碼器與解碼器上增加了skip connection，使每一段編碼器的輸出能與對應的解碼器相連接，使重要資訊不會遺失。雖然unet可以用於多分類問題，即為每個像素都分配一種類別，但在本次作業中，我只將unet作為區分前背景，瑕疵的分類則交由yolov5完成。



Results

**資料集說明**

資料集由三個類別組成，分別為powder uncover, powder uneven, scratch，分為training set與test set，以下是各類別資料的分布情形。

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Total** | **Training Set** | **Test Set** |
| Powder Uncover | 150 | 100 | 50 |
| Powder Uneven | 150 | 100 | 50 |
| Scratch | 150 | 100 | 50 |

為了實驗的準確度，我採用了4-fold cross validation來進行驗證，將training set拆分為4個fold，每次使用3個fold進行訓練，1個fold作為驗證，每組參數設定總共進行4次的訓練

**實驗設定**

* 硬體

|  |  |
| --- | --- |
| **Type** | **Name** |
| GPU | Nvidia GeForce RTX 3080 Ti |
| CPU | Intel i5-12400 |

* 執行環境：Windows10 + Python 3.7.15 + torch 1.12.1
* 模型設定

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **Batch size** | **Input Image Size** | **Epochs** | **Optimizer** | **Learning Rate** |
| Yolov5 | 16 | (960, 960) | 300 | SGD | 1e-2 |
| UNet | 16 | (320, 320) | 100 | RMSProp | 1e-5 |

**後處理**

1. 經由segmentation的結果修正bounding box

由於U-Net只負責前背景的區分，任務較為簡單，相較於yolov5，U-Net可以學到更加通用的特徵，因此，我透過U-Net的分類結果更進一步修改yolov5的候選框，若某候選框經過U-Net後，大部分像素點皆為背景，則該候選框將被視為錯誤的分類，將其捨去，下面是實際的例子。

1. 後處理前：在yolov5的預測結果(左圖)中，錯誤預測了兩個框

一張含有 文字 的圖片

自動產生的描述

1. 經後處理後：捨去了在segmentation中背景像素佔大多數的候選框

一張含有 文字 的圖片

自動產生的描述

**實驗結果**

以下實驗數據皆為使用Test Set中圖片的驗證結果

欄位說明:

1. **：**分別屬於powder uncover, powder uncover或scratch之圖片的平均dice coefficient
2. **：**所有圖片的平均dice coefficient
3. **：**分別屬於powder uncover, powder uncover或scratch之圖片的平均AP50
4. **mAP：**所有類別平均AP

**\*表最高者**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **(%)** | **(%)** | **(%)** | **(%)** | **(%)** | **(%)** | **(%)** | **mAP**  **(%)** |
| Fold1 | **80.5\*** | 92.9 | 95.1 | **89.5\*** | 68.3 | **99.5\*** | **99.5\*** | 89.1 |
| Fold2 | 76.1 | 92.3 | 95.0 | 87.7 | 76.6 | **99.5\*** | **99.5\*** | 91.9 |
| Fold3 | 79.1 | **93.7\*** | 94.1 | 89.0 | **79.0\*** | 99.1 | **99.5\*** | **92.5\*** |
| Fold4 | 76.0 | 93.0 | **97.4\*** | 88.8 | 69.0 | **99.5\*** | **99.5\*** | 89.3 |
| Average | 77.9 | 93.0 | 95.4 | 88.8 | 73.2 | 99.4 | 99.5 | 90.7 |

Discussion

觀察實驗結果後可以發現，在物件偵測中，powder uncover的準確度往往較另外兩個類別低，

主要有兩個因素所導致，

1. Powder uncover瑕疵的形狀相較於powder uneven與scratch來說更加不規則，瑕疵大小由大至小皆有。而眾所皆知的是，yolo在小目標偵測上往往效果不佳。
2. 原始的label有些標註非常奇怪，舉例來說：有些標籤彼此overlap，或是明明是一整塊的標籤，卻拆分為數個小標籤

一張含有 文字, 白板, 室內 的圖片

自動產生的描述一張含有 牆, 室內 的圖片

自動產生的描述

圖1：標籤之間彼此overlap 圖2：一整塊的標籤拆分為數個小標籤

Conclusion