# Yüksek Düzey Programlama Dönem Proje Ödevi

**Ali Osman Öztürk – 202013171069 – Bilgisayar Mühendisliği - 4. Sınıf - 1. Öğretim**

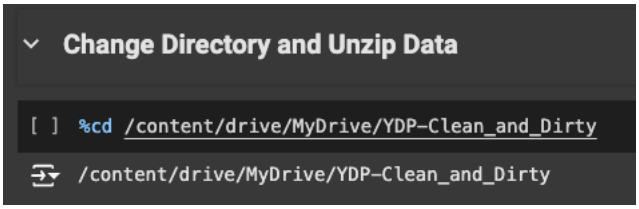
**Seçilen Veri Seti:** Cleaned vs Dirty

Bu veri setindeki tahminleme işlemleri görüntü işleme algoritmaları ile yapılabilmektedir. Görüntü işleme algoritmalarından YOLO-v11'i tercih ettim. Çünkü güncel bir teknoloji olarak en son nesil nesne tespiti yeniliklerini içermektedir. Hız ve doğruluk açısından optimize edilmiş bir mimariye sahiptir ve gerçek zamanlı uygulamalarda etkili performans sağlar.

## Kod açıklaması

Algoritmamın eğitimi için daha önceden kullanımına aşina olduğum Google Colab platformunu tercih ettim. Bu platformda T4 GPU'lu çalışma zamanını kullandım. GPU'lu bir sistem tercih etmemin nedeni, YOLO gibi görüntü işleme algoritmaları için CPU'ya oranla GPU'nun çok daha performanslı olmasıdır.

---




Çalışmalarımın kaybolmaması adına veri setini Google Drive'ıma yerleştirdim. Sonrasında Colab çalışma dizinimi veri setimin olduğu YDP-Clean_and_Dirty klasörüme ayarladım.

---


Veri setimin olduğu plates.zip dosyasını unzip komutu ile ZIP arşivinden çıkardım.

**Create classification dataset**

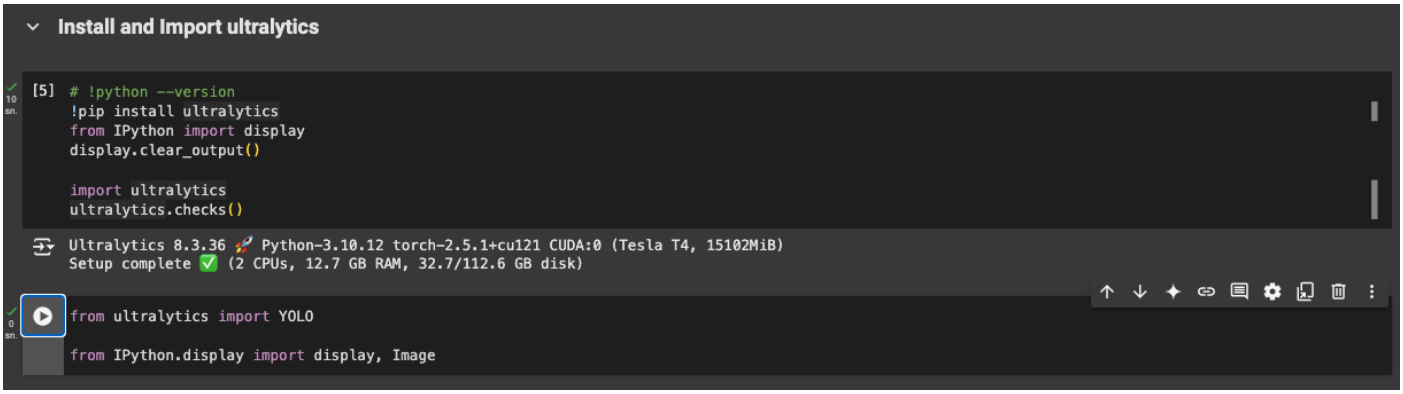
```
[ ] import os
    import random
    from shutil import copyfile

    def split_data(input_folder, output_folder, split_ratio=(0.8, 0.1, 0.1)):
        # Create output folders if they don't exist
        for split in ['train', 'test', 'val']:
            folder_path = os.path.join(output_folder, split)
            os.makedirs(folder_path, exist_ok=True)

        # Get a list of all subfolders in the input folder
        subfolders = [f for f in os.listdir(input_folder) if os.path.isdir(os.path.join(input_folder, f))]

        # For each subfolder, split its contents into train, test, and val
        for subfolder in subfolders:
            subfolder_path = os.path.join(input_folder, subfolder)
            output_subfolder_path_train = os.path.join(output_folder, 'train', subfolder)
            output_subfolder_path_test = os.path.join(output_folder, 'test', subfolder)
            output_subfolder_path_val = os.path.join(output_folder, 'val', subfolder)

            os.makedirs(output_subfolder_path_train, exist_ok=True)
            os.makedirs(output_subfolder_path_test, exist_ok=True)
            os.makedirs(output_subfolder_path_val, exist_ok=True)

            # Get a list of all image files in the subfolder
            image_files = [f for f in os.listdir(subfolder_path) if f.lower().endswith(('.jpg', '.jpeg', '.png'))]
            total_images = len(image_files)

            # Calculate the number of images for each split
            train_size = int(split_ratio[0] * total_images)
            test_size = int(split_ratio[1] * total_images)

            # Shuffle the list of image files
            random.shuffle(image_files)

            # Copy files to the respective folders
            for i, file_name in enumerate(image_files):
                source_path = os.path.join(subfolder_path, file_name)
                if i < train_size:
                    destination_path = os.path.join(output_subfolder_path_train, file_name)
                elif i < train_size + test_size:
                    destination_path = os.path.join(output_subfolder_path_test, file_name)
                else:
                    destination_path = os.path.join(output_subfolder_path_val, file_name)

                #print(f"Copying {source_path} to {destination_path}")

                try:
                    copyfile(source_path, destination_path)
                except Exception as e:
                    print(f"Error copying file: {e}")

    # Example usage:
    input_folder = r"/content/drive/MyDrive/YDP-Clean_and_Dirty/plates/train"
    output_folder = r'/content/drive/MyDrive/YDP-Clean_and_Dirty/working'
    split_data(input_folder, output_folder)
```

Buradaki **split_data** fonksiyonumuz ile veri setini **eğitim**, **test** ve **doğrulama** için ayırıyoruz.

- input_folder: Verilerin bulunduğu ana klasörün yolu (Örneğin az önce ZIP arşivinden çıkardığımız dosyalar).
- output_folder: Eğitim, test ve doğrulama için oluşturulacak yeni klasörlerin kaydedileceği yol.
- split_ratio: Eğitim (%80), test (%10) ve doğrulama (%10) setlerine ayrılacak oranlar (varsayılan (0.8, 0.1, 0.1)).

Fonksiyon öncelikle train, test ve val adlı klasörler oluşturur. Daha sonra klasördeki görüntü dosyalarını filtreler, toplam dosya sayısını hesaplar ve belirlenen oranlara göre dosyaları sırasıyla bu setlere ayırır. Dosyalar karıştırılarak rastgelelik sağlanır ve her biri belirlenen setlere atanır. Tüm bu işlem sonucunda veriler, working klasörüne kaydedilir. Veri setinde “cleaned” ve “dirty” sınıfları zaten ayrılmış olduğundan, bu işlem sırasında elle bir ayrım yapılmaz. İşlem sonunda klasör yapısı şu şekilde görünür:

```
/working/
  train/
    clean/
    dirty/
  test/
    clean/
    dirty/
  val/
    clean/
    dirty/
```

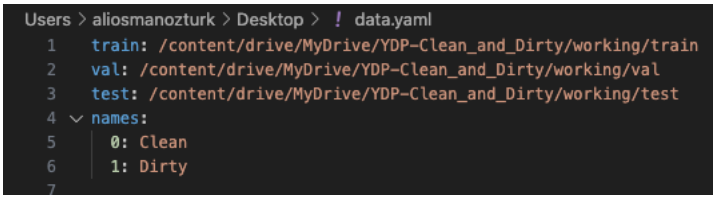



Ultralytics kütüphanesini indirip import ediyoruz; bu kütüphane adını YOLO algoritmasının yapımcılarından almıştır. Ardından YOLO sınıfını import ederek model yükleme ve eğitim gibi işlemler için hazır hale getiriyoruz. Ayrıca, Jupyter Notebook ortamında görselleri ve içerikleri görselleştirmek amacıyla kullanılan IPython.display kütüphanesini de import ediyoruz.

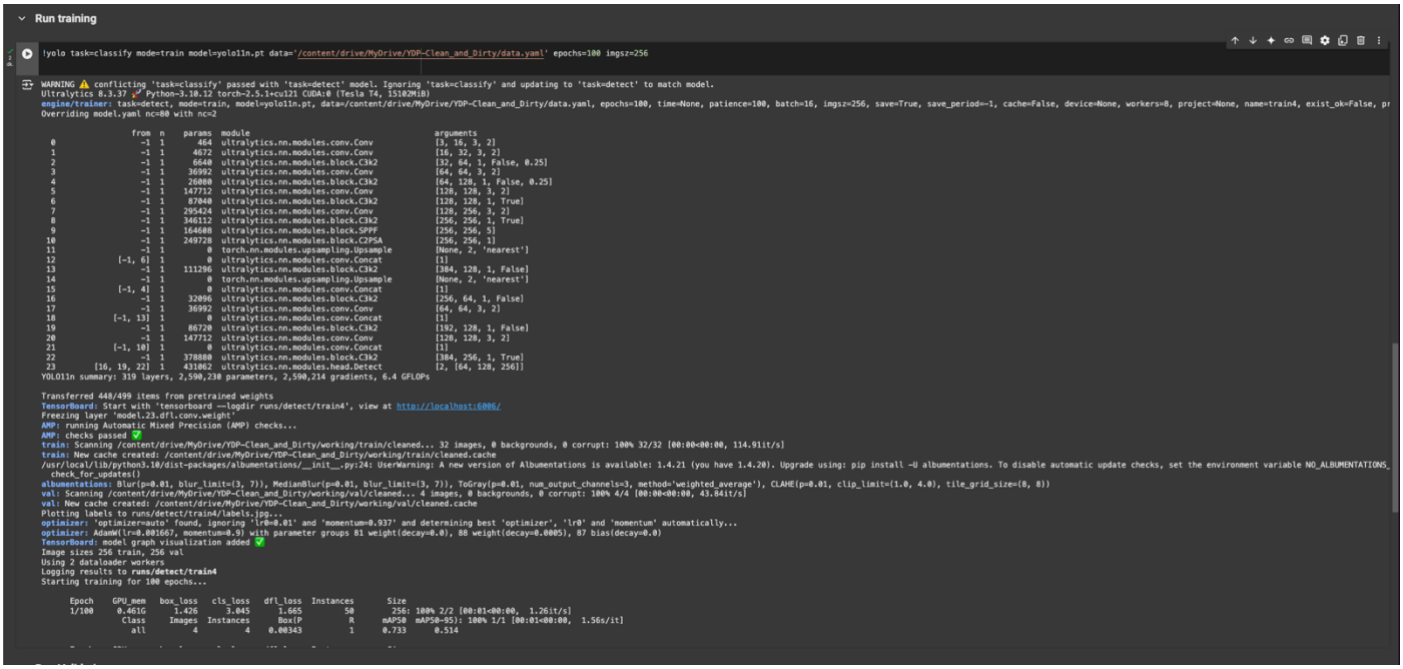
---


Bu kod, görüntülerin bulunduğu klasörlere göre her bir resim için otomatik olarak etiket dosyası (.txt) oluşturur. Klasör isimlerini sınıf etiketi olarak kullanır ve etiket dosyasına bu sınıf ID'sini yazar. Veri setimiz zaten halihazırda cleaned ve dirty olarak geldiğinden dolayı böyle bir yöntem izleyebiliyoruz.

---




Train, val ve test klasörlerinin yolları data.yaml dosyası aracılığıyla YOLO'ya tanıtılmıştır. Clean 0, Dirty 1 etiketli olarak belirlenmiştir.




Burada eğitim sürecini hızlandırmak için hafif bir model olan yolo11n.pt dosyası kullanılmıştır. Epoch sayısının 100 olarak belirlenmesi, modelin yeterince öğrenmesini sağlarken aynı zamanda eğitim süresini makul seviyede tutarak hız ve doğruluk arasında bir denge kurulmasını sağlamıştır.

**Çıktı**:

```
0000 cleaned
0001 dirty
0002 dirty
0003 dirty
0004 dirty
0005 cleaned
0006 cleaned
0007 dirty
0008 dirty
0009 dirty
0010 dirty
0011 dirty
0012 dirty
0013 dirty
0014 dirty
0015 cleaned
0016 cleaned
0017 cleaned
0018 dirty
0019 cleaned
0020 dirty
0021 dirty
0022 dirty
0023 cleaned
0024 cleaned
0025 cleaned
0026 cleaned
0027 dirty
0028 cleaned
0029 dirty
0030 dirty
0031 dirty
0032 dirty
0033 dirty
0034 dirty
0035 dirty
0036 dirty
0037 cleaned
0038 dirty
0039 dirty
0040 cleaned
0041 dirty
0042 dirty
0043 dirty
0044 dirty
0045 dirty
0046 cleaned
0047 cleaned
0048 cleaned
0049 dirty
0050 dirty
0051 dirty
0052 cleaned
0053 dirty
0054 dirty
0055 dirty
0056 dirty
0057 cleaned
0058 cleaned
0059 cleaned
0060 dirty
0061 dirty
0062 cleaned
0063 cleaned
0064 dirty
0065 dirty
0066 dirty
0067 dirty
0068 dirty
0069 dirty
0070 dirty
0071 dirty
0072 cleaned
0073 dirty
0074 dirty
0075 dirty
0076 dirty
0077 dirty
0078 dirty
0079 dirty
0080 dirty
0081 dirty
0082 dirty
0083 dirty
0084 dirty
0085 dirty
0086 dirty
0087 dirty
0088 cleaned
0089 dirty
0090 cleaned
0091 dirty
0092 cleaned
0093 dirty
0094 dirty
0095 cleaned
0096 dirty
0097 dirty
0098 cleaned
0099 dirty
0100 dirty
0101 dirty
0102 dirty
0103 dirty
0104 dirty
0105 cleaned
0106 dirty
0107 dirty
0108 dirty
0109 dirty
0110 cleaned
0111 dirty
0112 dirty
0113 dirty
0114 dirty
0115 cleaned
0116 cleaned
0117 dirty
0118 dirty
0119 cleaned
0120 cleaned
0121 cleaned
0122 dirty
0123 dirty
0124 dirty
0125 dirty
0126 dirty
0127 dirty
0128 dirty
0129 cleaned
0130 cleaned
0131 dirty
0132 cleaned
0133 dirty
0134 cleaned
0135 dirty
0136 dirty
0137 dirty
0138 dirty
0139 dirty
```

0140 dirty
0141 dirty
0142 dirty
0143 dirty
0144 cleaned
0145 dirty
0146 dirty
0147 cleaned
0148 dirty
0149 cleaned
0150 dirty
0151 cleaned
0152 dirty
0153 dirty
0154 dirty
0155 dirty
0156 cleaned
0157 dirty
0158 cleaned
0159 dirty
0160 cleaned
0161 cleaned
0162 dirty
0163 dirty
0164 dirty
0165 dirty
0166 cleaned
0167 dirty
0168 dirty
0169 dirty
0170 dirty
0171 dirty
0172 dirty
0173 dirty
0174 dirty
0175 dirty
0176 dirty
0177 dirty
0178 dirty
0179 dirty
0180 dirty
0181 dirty
0182 dirty
0183 dirty
0184 dirty
0185 cleaned
0186 dirty
0187 dirty
0188 dirty
0189 cleaned
0190 dirty
0191 dirty
0192 cleaned
0193 dirty
0194 cleaned
0195 cleaned
0196 cleaned
0197 cleaned
0198 cleaned
0199 cleaned
0200 cleaned
0201 dirty
0202 dirty
0203 cleaned
0204 cleaned
0205 dirty
0206 cleaned
0207 dirty
0208 dirty
0209 dirty
0210 cleaned
0211 dirty
0212 dirty
0213 cleaned
0214 cleaned
0215 cleaned
0216 dirty
0217 dirty
0218 cleaned
0219 cleaned
0220 dirty
0221 dirty
0222 dirty
0223 cleaned
0224 dirty
0225 dirty
0226 dirty
0227 dirty
0228 cleaned
0229 dirty
0230 cleaned
0231 dirty
0232 dirty
0233 dirty
0234 dirty
0235 dirty
0236 cleaned
0237 cleaned
0238 dirty
0239 dirty
0240 dirty
0241 dirty
0242 dirty
0243 cleaned
0244 dirty
0245 dirty
0246 dirty
0247 dirty
0248 cleaned
0249 cleaned
0250 cleaned
0251 dirty
0252 dirty
0253 dirty
0254 dirty
0255 cleaned
0256 cleaned
0257 dirty
0258 cleaned
0259 dirty
0260 cleaned
0261 cleaned
0262 dirty
0263 dirty
0264 cleaned
0265 dirty
0266 cleaned
0267 cleaned
0268 dirty
0269 cleaned
0270 dirty
0271 cleaned
0272 dirty
0273 cleaned
0274 cleaned
0275 dirty
0276 dirty
0277 dirty
0278 cleaned
0279 dirty
0280 dirty
0281 dirty
0282 dirty
0283 dirty
0284 dirty
0285 dirty
0286 dirty
0287 dirty
0288 dirty
0289 dirty
0290 dirty
0291 dirty
0292 dirty
0293 dirty
0294 dirty
0295 dirty
0296 cleaned
0297 dirty
0298 cleaned
0299 cleaned
0300 cleaned
0301 dirty
0302 dirty
0303 cleaned
0304 dirty
0305 cleaned
0306 dirty
0307 cleaned
0308 dirty
0309 dirty
0310 dirty
0311 cleaned
0312 cleaned
0313 dirty
0314 dirty
0315 dirty
0316 dirty
0317 dirty
0318 cleaned
0319 dirty
0320 dirty
0321 dirty
0322 cleaned
0323 dirty
0324 dirty
0325 cleaned
0326 dirty
0327 dirty
0328 dirty
0329 cleaned
0330 dirty
0331 dirty
0332 dirty
0333 cleaned
0334 dirty
0335 dirty
0336 dirty
0337 dirty
0338 dirty
0339 dirty
0340 cleaned
0341 dirty
0342 dirty
0343 dirty
0344 cleaned
0345 cleaned
0346 dirty
0347 cleaned
0348 dirty
0349 dirty
0350 dirty
0351 dirty
0352 cleaned
0353 cleaned
0354 cleaned
0355 cleaned
0356 dirty
0357 dirty
0358 dirty
0359 cleaned
0360 cleaned
0361 dirty
0362 cleaned
0363 dirty
0364 cleaned
0365 dirty
0366 dirty
0367 dirty
0368 dirty
0369 cleaned
0370 dirty
0371 cleaned
0372 dirty
0373 cleaned
0374 dirty
0375 dirty
0376 dirty
0377 cleaned
0378 dirty
0379 dirty
0380 dirty
0381 cleaned
0382 cleaned
0383 dirty
0384 cleaned
0385 dirty
0386 dirty
0387 dirty
0388 dirty
0389 dirty
0390 dirty
0391 dirty
0392 cleaned
0393 dirty
0394 cleaned
0395 cleaned

0396 dirty
0397 dirty
0398 dirty
0399 dirty
0400 dirty
0401 dirty
0402 dirty
0403 dirty
0404 dirty
0405 cleaned
0406 cleaned
0407 dirty
0408 cleaned
0409 cleaned
0410 dirty
0411 dirty
0412 dirty
0413 dirty
0414 dirty
0415 cleaned
0416 cleaned
0417 dirty
0418 dirty
0419 dirty
0420 cleaned
0421 dirty
0422 dirty
0423 cleaned
0424 dirty
0425 dirty
0426 dirty
0427 cleaned
0428 cleaned
0429 dirty
0430 dirty
0431 dirty
0432 dirty
0433 dirty
0434 dirty
0435 dirty
0436 dirty
0437 cleaned
0438 cleaned
0439 dirty
0440 cleaned
0441 dirty
0442 dirty
0443 dirty
0444 dirty
0445 cleaned
0446 dirty
0447 dirty
0448 dirty
0449 dirty
0450 cleaned
0451 dirty
0452 cleaned
0453 dirty
0454 dirty
0455 cleaned
0456 dirty
0457 dirty
0458 cleaned
0459 dirty
0460 cleaned
0461 dirty
0462 dirty
0463 dirty
0464 dirty
0465 cleaned
0466 dirty
0467 dirty
0468 cleaned
0469 dirty
0470 dirty
0471 dirty
0472 cleaned
0473 dirty
0474 dirty
0475 dirty
0476 cleaned
0477 dirty
0478 cleaned
0479 dirty
0480 dirty
0481 dirty
0482 dirty
0483 dirty
0484 dirty
0485 dirty
0486 cleaned
0487 dirty
0488 dirty
0489 cleaned
0490 cleaned
0491 dirty
0492 dirty
0493 dirty
0494 dirty
0495 cleaned
0496 dirty
0497 cleaned
0498 cleaned
0499 dirty
0500 dirty
0501 cleaned
0502 dirty
0503 cleaned
0504 cleaned
0505 cleaned
0506 cleaned
0507 dirty
0508 dirty
0509 cleaned
0510 dirty
0511 dirty
0512 dirty
0513 cleaned
0514 dirty
0515 cleaned
0516 dirty
0517 dirty
0518 dirty
0519 dirty
0520 cleaned
0521 dirty
0522 dirty
0523 dirty
0524 dirty
0525 dirty
0526 dirty
0527 cleaned
0528 dirty
0529 dirty
0530 dirty
0531 cleaned
0532 dirty
0533 cleaned
0534 dirty
0535 cleaned
0536 dirty
0537 dirty
0538 cleaned
0539 cleaned
0540 dirty
0541 dirty
0542 cleaned
0543 dirty
0544 dirty
0545 dirty
0546 dirty
0547 dirty
0548 cleaned
0549 dirty
0550 dirty
0551 dirty
0552 dirty
0553 dirty
0554 cleaned
0555 dirty
0556 dirty
0557 dirty
0558 cleaned
0559 dirty
0560 dirty
0561 dirty
0562 dirty
0563 dirty
0564 cleaned
0565 cleaned
0566 dirty
0567 cleaned
0568 dirty
0569 dirty
0570 dirty
0571 dirty
0572 dirty
0573 cleaned
0574 cleaned
0575 dirty
0576 cleaned
0577 dirty
0578 dirty
0579 cleaned
0580 dirty
0581 dirty
0582 dirty
0583 cleaned
0584 dirty
0585 cleaned
0586 cleaned
0587 cleaned
0588 dirty
0589 dirty
0590 dirty
0591 dirty
0592 dirty
0593 cleaned
0594 dirty
0595 dirty
0596 dirty
0597 cleaned
0598 dirty
0599 dirty
0600 dirty
0601 dirty
0602 dirty
0603 dirty
0604 dirty
0605 cleaned
0606 dirty
0607 dirty
0608 cleaned
0609 dirty
0610 cleaned
0611 dirty
0612 dirty
0613 dirty
0614 cleaned
0615 cleaned
0616 dirty
0617 dirty
0618 dirty
0619 dirty
0620 cleaned
0621 dirty
0622 dirty
0623 cleaned
0624 dirty
0625 cleaned
0626 dirty
0627 dirty
0628 cleaned
0629 cleaned
0630 dirty
0631 cleaned
0632 dirty
0633 dirty
0634 dirty
0635 dirty
0636 cleaned
0637 dirty
0638 dirty
0639 dirty
0640 dirty
0641 dirty
0642 dirty
0643 cleaned
0644 dirty
0645 cleaned
0646 dirty
0647 cleaned
0648 dirty
0649 dirty
0650 dirty
0651 dirty

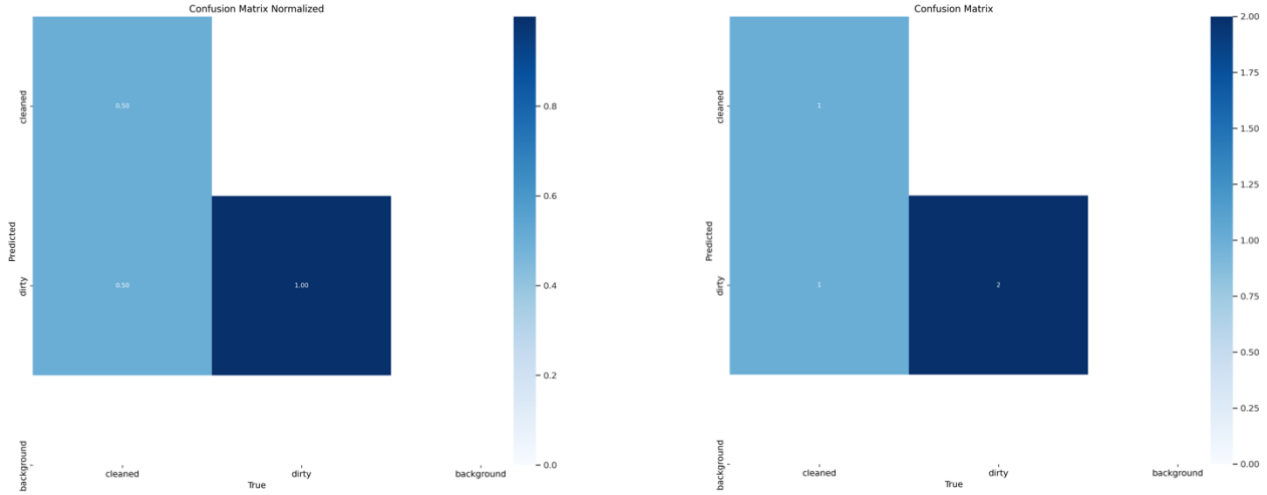
0652 dirty
0653 dirty
0654 cleaned
0655 dirty
0656 cleaned
0657 dirty
0658 dirty
0659 cleaned
0660 cleaned
0661 dirty
0662 dirty
0663 dirty
0664 cleaned
0665 dirty
0666 cleaned
0667 dirty
0668 cleaned
0669 dirty
0670 dirty
0671 dirty
0672 dirty
0673 dirty
0674 dirty
0675 cleaned
0676 dirty
0677 dirty

0678 cleaned
0679 cleaned
0680 cleaned
0681 dirty
0682 dirty
0683 dirty
0684 dirty
0685 dirty
0686 cleaned
0687 cleaned
0688 cleaned
0689 dirty
0690 dirty
0691 dirty
0692 dirty
0693 dirty
0694 dirty
0695 dirty
0696 dirty
0697 cleaned
0698 dirty
0699 dirty
0700 dirty
0701 dirty
0702 dirty
0703 dirty

0704 dirty
0705 dirty
0706 cleaned
0707 cleaned
0708 dirty
0709 cleaned
0710 dirty
0711 dirty
0712 dirty
0713 cleaned
0714 dirty
0715 cleaned
0716 dirty
0717 dirty
0718 cleaned
0719 dirty
0720 dirty
0721 dirty
0722 dirty
0723 dirty
0724 dirty
0725 dirty
0726 dirty
0727 dirty
0728 dirty
0729 dirty

0730 dirty
0731 dirty
0732 dirty
0733 dirty
0734 cleaned
0735 dirty
0736 dirty
0737 dirty
0738 dirty
0739 dirty
0740 dirty
0741 cleaned
0742 cleaned
0743 dirty

Bunun sonucunda aşağıdaki karmaşıklık matrisleri elde edilmiştir.




Bu matrisler sonucunda bu veri setinin zorlu bir yarışma, challenge olduğunu daha net anlayabiliyoruz. Veri setindeki clean ve dirty veri sayısı eşit olmasına rağmen train için yalnızca 38 veri olması, test için 744 veri olması bu veri setiyle doğru sonuç almayı epey zorlaştırmaktadır. 38 veriyle ne kadar iyi eğitim yapılırsa yapılsın, 744 test verisi üzerinden pek başarılı bir sonuç üretemeyecektir.