

Proje istekler:

- 1) Projedeki sonuçların detaylı dökümanite edildiği ara dönem (vize haftası) ve proje sonuç raporu (final haftası)
- 2) Projedeki sonuçları görsel olarak özetleyen ve kodun nasıl çalıştırıldığını demo yapan bir sunum videosu google drive linki proje sonuç raporuna koyulacaktır
- 3) Projede raporlanan sonuçların notlandırma aşamasında çalıştırılmasına imkan sağlayacak github linki (proje kodunun nasıl çalıştırılacağı 1 nolu isterde detaylı olarak dökümanite edilecektir, çalıştırılmayan proje kodları notlandırılmayacaktır)
- 4) Her bir proje için minimum 3 adet Sinir Ağı modeli denenerek karşılaştırmaları yapılacaktır. Maksimum iki kişilik gruba izin verilir, iki kişilik gruplarda en az 6 adet modelin karşılaştırılması yapılmalıdır
- 5) **BONUS puan:** Projedeki anomali tespiti uygulaması için UI tasarımı yapan (QT, JavaScript, React gibi araçlar kullanarak) ve Frontend dizayn ve implementasyonu yapan projelere ekstra puan verilecektir.
- 6) **Proje teslim tarihini geçiren projeler değerlendirmeye kesinlikle alınmayacaktır**

Ahşap Verileri ile Anomali Tespiti ve Segmentasyonu

Görev Tanımı

Bu görevde, verilen ahşap yüzey verileri kullanılarak denetimsiz öğrenme (unsupervised learning) yaklaşımıyla bir anomali tespit modeli geliştirilecektir. Model, ahşap yüzeyinde kusurların veya düzensizliklerin tespit edilmesi için eğitilecektir.

Görev Kapsamı ve Hedefler

1. Ön İşleme Adımları ve Veri Hazırlığı

Arka Plan Çıkarma

- Arka planın temizlenmesi, modelin yalnızca ahşap yüzeyindeki anomali bölgelerine odaklanmasını sağlayacaktır.(Örneğin: Adaptive Thresholding, Otsu Thresholding...)

Görüntü Normalizasyonu ve Boyutlandırma

- Modelin sabit bir formatta eğitilmesi için:
 - Bütün görüntüler 256x256 piksel boyutuna getirilmelidir.
 - Piksel değerleri [0,1] aralığında normalize edilmelidir.

2. Denetimsiz Model Eğitimi

Etiketsiz Veri Kullanımı

- Model, anomaliye dair doğrudan etiket bilgisi olmadan eğitilecek ve anormal alanları kendisi belirleyecektir.

Model Çıktıları

1. Maske (Mask):

- Ahşap yüzeyindeki anomali bölgelerini belirleyen segmentasyon maskesi üretilecektir.
- Model çıktısı, gerçek anomalilerle karşılaştırılarak değerlendirilecektir.

2. Anomali Skoru (Anomaly Score):

- Model, her veri örneği için bir anomali skoru hesaplayacaktır.
- Skor belirli bir eşik değeriyle ikili (binary) sınıflandırma yapmak için kullanılacak.

3. Anomali Skorunun Kullanımı

Sınıflandırma

- Belirlenen eşik değeri (threshold) üzerinde olan anomaly score değerleri anomalili (kusurlu) olarak etiketlenecektir.
- Altında kalan değerler normal (kusursuz) olarak etiketlenecektir.
- Farklı eşik değerleri test edilerek en iyi sınıflandırma performansı elde edilmeye çalışılacaktır.

4. Model Değerlendirme Metrikleri

Modelin doğruluğunu ölçmek için iki temel metrik kullanılacaktır:

1. F1 Score – Anomali Skoruna Dayalı İkili Sınıflandırma Performansı

- Modelin belirlediği anomaly score eşik değeri kullanılarak F1 skoru hesaplanacaktır.
- Precision ve Recall dengesine göre anomali tespiti performansı değerlendirilecektir.

2. IoU (Intersection over Union) – Maske Kalitesinin Değerlendirilmesi

- Modelin ürettiği anomali maske çıktıları ile gerçek anomali bölgeleri arasındaki örtüşme hesaplanacaktır.
- IoU skoru (Jaccard Index) hesaplanarak modelin segmentasyon başarımı değerlendirilecektir.
- Yüksek IoU değeri, modelin maske çıktılarının gerçek anomalilere ne kadar yakın olduğunu gösterecektir.

5. Desktop Uygulaması Geliştirme

- Bir desktop uygulaması geliştirilmelidir.
- Bu uygulama üzerinden eğitim ve test işlemleri gerçekleştirilecektir.
- Test sonuçları, belirtilen F1 Score ve IoU metrikleri ile sunulacaktır.
- Model tarafından oluşturulan segmentasyon maskeleri, uygulama üzerinden görselleştirilerek kullanıcıya sunulacaktır.

Veri Seti Yapısı

Bu görevde kullanılacak veri seti, 3 ana klasörden oluşmaktadır: train, test, ve ground_truth. Her klasör, modelin eğitim, test ve değerlendirme aşamalarında kullanılacak verileri içermektedir. Veri setinin dizin yapısı şu şekilde düzenlenmiştir:

Wood_dataset/

|

|— wood/

| |— train/

```
| | └─ good/
| |
| └─ test/
| | └─ good/
| | └─ defect/
| |
| └─ ground_truth/
| └─ defect/
```

Klasörler ve İçerikleri:

1. train/good/
 - Bu klasörde, normal (kusursuz) ahşap yüzeyleri içeren görüntüler bulunmaktadır.
 - Bu görüntüler, modelin eğitim aşamasında kullanılarak, normal desenleri öğrenmesine yardımcı olacaktır.
2. test/good/
 - Bu klasörde, test aşamasında kullanılacak normal (kusursuz) ahşap yüzeyleri içeren görüntüler bulunmaktadır.
 - Modelin doğruluğu, bu klasördeki verilerle test edilecek ve normal bölgelerin doğru bir şekilde sınıflandırılması değerlendirilecektir.
3. test/defect/
 - Bu klasörde, test aşamasında kullanılacak kusurlu (defect) ahşap yüzeyleri içeren görüntüler bulunmaktadır.
 - Bu veriler, modelin anomali tespiti başarımını ölçmek için kullanılacaktır. Modelin, kusurlu yüzeyleri doğru bir şekilde tespit etmesi hedeflenmektedir.
4. ground_truth/defect/
 - Bu klasörde, kusurlu bölgelerin maskeleri bulunmaktadır.
 - Maskeler, her bir kusurlu (defect) yüzey için gerçek anomali bölgelerini belirtmektedir. Bu maskeler, modelin ürettiği anomali maskesi ile karşılaştırılarak, segmentasyon başarımı ölçülecektir.
 - Maskeler, modelin segmentasyon kalitesini IoU (Intersection over Union) metriği ile değerlendirmek için kullanılacaktır.

Veri Setinin Kullanımı:

- Eğitim (Training):
 - train/good klasöründeki görüntüler, modelin normal yüzeyleri öğrenmesi için kullanılacaktır. Bu veri, modelin kusursuz yüzeyler üzerinde eğitim yapmasına ve anomali tespit etme yeteneğini geliştirmesine yardımcı olur.
- Test (Testing):

- test/good klasöründeki görüntüler, modelin normal (kusursuz) yüzeyler üzerinde ne kadar doğru sınıflandırma yaptığını test etmek için kullanılacaktır.
- test/defect klasöründeki görüntüler, kusurlu yüzeylerin tespit edilmesi için kullanılacaktır. Bu görüntülerle modelin anomali tespiti başarımı değerlendirilir.
- Ground Truth (Gerçek Etiketler):
 - ground_truth/defect klasöründeki maskeler, test/defect klasöründeki her bir kusurlu görüntü için doğru anomali bölgelerini belirtir. Bu maskeler, IoU skoru ile modelin segmentasyon başarımını değerlendirmede kullanılacaktır.

Bu yapı sayesinde, modelin segmentasyon ve anomali tespiti yetenekleri doğru ve verimli bir şekilde test edilecektir.

Yardımcı Doküman ve Linkler

Denetimsiz Anomali Tespiti Modelleri ile ilgili makale ve kodlara aşağıdaki bağlantılardan ulaşabilirsiniz:

[Anomaly Detection on MVTec AD - Papers with Code](#)

Bu kaynaklar, ilgili modellerin makalelerine ve açık kaynak kodlarına erişmenizi sağlayacaktır.

Dikkat Edilmesi Gerekenler:

1. Veri Seti Yapısı Farklılığı:

MVTecAD veri seti birden fazla alt veri kümesinden (subdataset) oluşmaktadır.

Sizin kullanacağınız veri seti ise yalnızca Wood alt kümesini içermektedir.

Bu nedenle, repolardaki (depolardaki) kodların veri okuma bölümlerinde sadece Wood subsetinin kullanılacak şekilde düzenlenmesi gerekmektedir.

2. Dizin Yapısına Dikkat Edin:

MVTec veri setleri farklı dizin yapılarına sahip olabilir.

Paylaşılan kodların veri okuma bölümlerinde, MVTec'in genel yapısına göre işlem yapıldığı görülmektedir.

MVTech dizin yapısı:

```
mvtec_ad/  
|  
├── bottle/  
|   ├── train/  
|   |   └── good/  
|   └── test/  
|   └── good/
```

```

| | |— broken_large/
| | |— broken_small/
| | |— contamination/
| |— ground_truth/
| |— broken_large/
| |— broken_small/
| |— contamination/
|
|— wood/
| |— train/
| | |— good/
| |— test/
| | |— good/
| | |— scratch/
| | |— hole/
| |— ground_truth/
| |— scratch/
| |— hole/
|
|— carpet/
| |— train/
| |— test/
| |— ground_truth/
|
|— ...

```

Kodları kullanırken, yalnızca **Wood** alt veri kümesini seçmeniz gerekmektedir. Bu nedenle, ilgili yolları ve değişkenleri güncellemeniz önemlidir.

Örnek: Mevcut Kod

Bazı açık kaynak kodlarında veri kümesi tüm MVTec AD alt kümelerini destekleyecek şekilde yapılandırılmış olabilir. Örneğin:

```
_CLASSNAMES = [
    "carpet",
    "grid",
    "leather",
    "tile",
    "wood",
    "bottle",
    "cable",
    "capsule",
    "hazelnut",
    "metal_nut",
    "pill",
    "screw",
    "toothbrush",
    "transistor",
    "zipper",
]

IMAGENET_MEAN = [0.485, 0.456, 0.406]
IMAGENET_STD = [0.229, 0.224, 0.225]

class DatasetSplit(Enum):
    TRAIN = "train"
    TEST = "test"

class MVTecDataset(torch.utils.data.Dataset):

    def __init__(
        self,
        source,
        anomaly_source_path='/root/dataset/dtd/images',
        dataset_name='mvtec',
        classname='leather',
        resize=256,
        imagesize=256,
    ):

```

Düzenlenmiş Versiyon

Aşağıdaki gibi güncellenmelidir:

```
_CLASSNAMES = [  
    "wood",  
]  
  
IMAGENET_MEAN = [0.485, 0.456, 0.406]  
IMAGENET_STD = [0.229, 0.224, 0.225]  
  
class DatasetSplit(Enum):  
    TRAIN = "train"  
    TEST = "test"  
  
class CustomDataset(torch.utils.data.Dataset):  
  
    def __init__(  
        self,  
        source,  
        anomaly_source_path="/root/dataset/dtd/images",  
        dataset_name="wood_dataset",  
        classname="wood",  
        resize=256,  
        imagesize=256,  
    ):  
  
        ~
```

3. Dosya Formatı Uyumluluğu:

Mevcut veri setinizdeki görüntüler .jpg formatındadır.

Ancak bazı açık kaynak kodlarında .png formatı kullanılmış olabilir.

Bu tür hatalarla karşılaşmamak için, veri okuma kodlarınızı .jpg formatına uyumlu hale getirmelisiniz.

Bu düzenlemelere dikkat ederek modeli, kendi veri setinizle sorunsuz bir şekilde çalıştırabilirsiniz.

Kısıtlamalar

Çalışmada kullanılan modellerin büyük bir kısmı çok uzun eğitim sürelerine sahiptir. Bu nedenle, eğitim süreleri 20 dakika ile sınırlamanız gerekmektedir. Bu sınırlama, kaynak kullanımı ve verimlilik açısından önemli bir gereklilik olarak belirlenmiştir.