

# TransVOD ve imgaug Entegrasyonu

## Özet

TransVOD [1] modelinde varsayılan olarak bulunan data augmentation yerine daha detaylı ve kaliteli olan imgaug [2] kütüphanesi kullanılmıştır. Bu amacı yerine getirmek için TransVOD [1] kaynak kodunda küçük bir değişiklik yapılmıştır.

## IMGAUG [2] Kütüphanesi

Imgaug, data augmentation yapmayı sağlayan detaylı ve kaliteli sonuçlar verebilen bir Python kütüphanesidir. Gerekli olan Kütüphanelerde bir hata bulunduğu için orijinali yerine bir GitHub forku [3] kullanılmıştır.

Kütüphane güzelce dökümente edilmiş olup akla gelebilecek birçok augmentation yöntemine sahiptir. Bütün augmentation yöntemleri grplara ayrılmıştır. Detaylı bilgiye GitHub sayfasından ve sayfada bulunan yönlendirmelerden ulaşılabilir olmuştur.

GitHub sayfasında bulunan örnek augmentation sonuçları (Fig.1) gösterilmiştir.



Fig.1: Örnek data augmentation sonuçları

Örnek verilere ek olarak bir uçak videosu verisi kullanılarak kütüphane denenmiştir. Bu deneme sonucunda tüm data augmentation kütüphanelerinde bulunan **rotate**, **scale**, **translate** gibi yöntemlere ek olarak **cloud**, **rain** gibi hava durumu ekleyen augmentation yöntemlerinin de fazlasıyla kaliteli sonuçlar verdiği görülmüştür. Bulut (Fig.2), yağmur (Fig.3) ekleme sonuçları gösterilmiştir. Daha fazlasına GitHub sayfasından ulaşılabilir.



Fig.2: Cloud augmentation örneği



Fig.3: Rain augmentation örneği

Ayrıca kütüphane, farklı augmentation yöntemlerini birleştirmeyi sağlayan çeşitli yöntemler de kullanmaktadır. **Sequential**, **SomeOf**, **OneOf** ve **Sometimes** en çok işe yarayan birleştirme yöntemleridir. Herhangi bir tanesi kullanılarak yapılan birleştirme sonucu tekrardan diğer birleştirme yöntemleri içerisinde tek bir augmentation değeriymiş gibi kullanılabilmektedir.

**Sequential** kullanılarak farklı augmentation yöntemleri sırasıyla veya rastgele sırayla uygulanabilmektedir.

**SomeOf** kullanılarak parametre olarak verilen iki değer arasında bir değer kadar augmentation rastgele seçilerek uygulanabilmektedir. Mesela 3 ve 5 parametreleri verilirse seçilen augmentation listesi içerisinde 3, 4 veya 5 tanesi seçilmiş uygulanmaktadır.

**OneOf** kullanılarak seçilen augmentation listesi içindeki biri uygulanmaktadır.

**Sometimes** kullanılarak verilen bir olasılık değerine göre augmentation uygulanabilmektedir. Opsiyonel olarak olasılık gerçekleşmemesi durumunda başka bir augmentation da uygulanabilmektedir.

Örnek olarak kompleks bir augmentation (Fig.4) verilmiştir.

```
seq = iaa.Sequential([
    iaa.AddToHueAndSaturation((-10, 10)),
    iaa.Fliplr(0.5),
    iaa.Affine(
        rotate=(-20, 20),
        translate_percent={"x": (-0.2, 0.2), "y": (-0.2, 0.2)},
        scale={"x": (0.7, 1.3), "y": (0.7, 1.3)},
    ),
    iaa.SomeOf((0, 1), [
        iaa.Cutout(size=0.2),
        iaa.Dropout(p=(0, 0.2)),
        iaa.SaltAndPepper(0.1, per_channel=True),
    ]),
    iaa.OneOf([
        iaa.Clouds(),
        iaa.Fog(),
        iaa.Rain(drop_size=(0.10, 0.20)),
        iaa.Snowflakes(flake_size=(0.2, 0.7), speed=(0.007, 0.03)),
        iaa.imgcorruptlike.Spatter(severity=2),
    ]),
    iaa.Sometimes(
        0.5,
        iaa.Sequential([
            iaa.MotionBlur(k=5, angle=[-30, 30]),
            iaa.AdditiveGaussianNoise(scale=(10, 40)),
        ], random_order=True),
        iaa.pillike.FilterDetail(),
    )
], random_order=True)
```

Fig.4: Kompleks bir augmentation örneği

Verilen augmentation örneği (Fig.4) için augmentation sonuçları (Fig.5) verilmiştir.

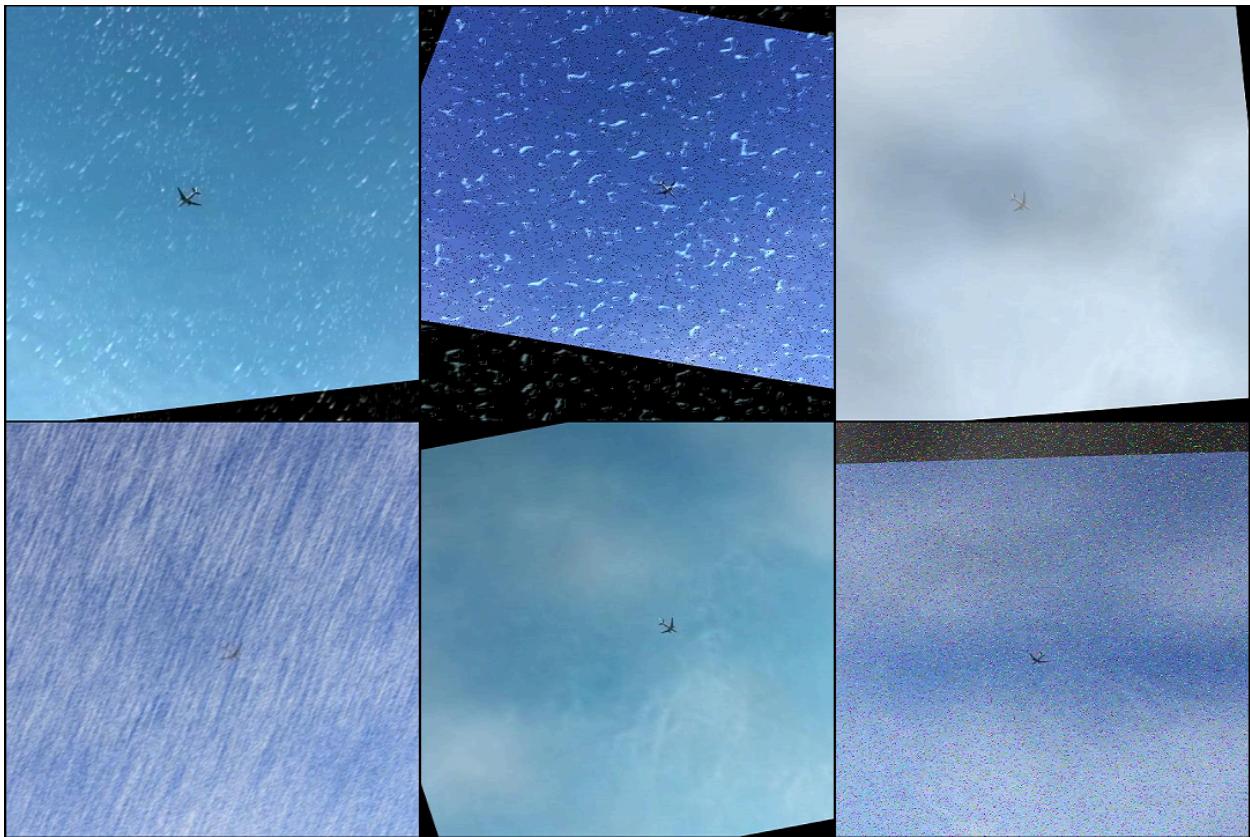


Fig.5: Kompleks augmentation sonuçları

Renk ve kontrast değişiklikleri için de birçok farklı augmentation yöntemi ve parametreleri bulunmaktadır ancak denenmemiştir.

## TransVOD Entegrasyonu

Imgau g kütüphanesini TransVOD modeline entegre etmek için kaynak kodunda ufak bir değişiklik yapmamız gerekmektedir. Tüm değişiklikler **vid\_single.py** dosyasında yapılmaktadır.

Öncelikle varsayılan data augmentation fonksiyonunun çağrııldığı yerin hemen üzerinde kendi yazdığımız imgaug data augmentation fonksiyonunu çağrırmamız gerekmektedir. Ayrıca bütün dönüşümler bittikten sonra test amacıyla fotoğrafları kaydeden bir fonksiyon eklenmiştir.

Varsayılan data augmentation fonksiyonu kaldırılmıştır. Nedeni, içerisinde normalization yapılmasıdır.

Kendi yazdığımız augmentation fonksiyonu içerisinde öncelikle Fig.4'te gösterildiği gibi bir augmentation tanımlanmıştır. Ayrıca veri tipi değişiklikleri yapılması da gerekmıştır. Değişiklikler, TransVOD'un kullandığı Tensor dizileri ve PIL resim formatı ile imgaug'un ihtiyaç duyduğu kendi bbox formatı ve Numpy dizisi arasında gerçekleşmiştir.

Eklenen son kod olan fotoğraf kaydetmeyi sağlayan fonksiyon sadece test etme amaçlı eklenmiştir. TransVOD çalıştırılırken hiçbir kullanımı yoktur ve yorum satırı olarak bırakılmıştır.

## Referanslar

- [1] <https://github.com/SJTU-LuHe/TransVOD>
- [2] <https://github.com/aleju/imgaug>
- [3] <https://github.com/marcown/imgaug>