



2022-2023 Güz Dönemi
BM5113 Bilgisayarla Görme
Final Ödevi

RANSAC, Optik Akış, Konvolüsyonel Sinir Ağları

- 1- Kampüs içerisinde farklı konum ve açılardan çekilmiş fakat benzer içerik barındıran iki görüntüyü images/campus klasörü altından belirleyiniz. Mesela P1070507.jpg ve P1070509.jpg olabilir. Görüntüleri kampus1.jpg ve kampus2.jpg şeklinde kaydediniz. Karmaşayı azaltmak için görüntünün boyutunu öncesinde uygun ölçüde küçültebilirsiniz. Ardından kampus1.jpg görüntüsünü kampus2.jpg görüntüsüne çeviren homografi matrisinin parametrelerini RANSAC algoritması ile bulunuz. Bunun için kampus1.jpg ve kampus2.jpg görüntüleri üzerinde hazır SIFT, SURF veya ORBS algoritmasını çalıştırarak nokta eşlemelerini belirleyiniz. Belirlediğiniz nokta eşlemeleri ile her seferinde rasgele 4 eşlenmiş nokta çifti seçerek 100 tekrarla RANSAC algoritmasını çalıştırınız. Nihayetinde en yüksek sayıda uyumlu noktayı içeren homografi matrisini belirleyiniz. Bu homografinin uyumlu noktalarını ve aykırı noktalarını kampus1.jpg ve kampus2.jpg görüntüleri üzerine çizdirerek gösteriniz. kampus2.jpg görüntüsünü bulduğunuz bu homografi matrisi ile eğme (warping) işlemini çift doğrusal ara değerlendirme (bilinear interpolation) kullanarak gerçekleştiriniz. Eğilmiş kampus2.jpg ile kampus1.jpg görüntülerini daha büyük bir görüntü düzleminde harmanlayınız (blending).
- 2- images/bridge-1 video görüntüsünün tüm çerçevelerinde histogram eşitlemenin ardından peş peşe ortalama (running average) yöntemini kullanarak arka plan çıkarma gerçekleştiriniz. Bunun için ekte verilen avi_reader.py dosyası değiştirilebilir. İlk 50 çerçeveyi arka planı belirlemek için kullanınız. $t = 50$ anında okunan çerçeveden itibaren arka plan bilgisini çıkarıp tam sayı değerini elde ettikten sonra uygun bir eşik değeri kullanarak ön plan bilgisini elde ediniz. Arka plan matrisini her okuduğunuz çerçevede $B_{t+1} = \alpha I_t + (1 - \alpha)B_t$ denklemine göre güncelleyiniz. Denkleminde $\alpha = 0,005$ gibi küçük bir değer olabilir. Eşikleme neticesinde elde edilen ön plan görüntülerinde bağlı bileşen analizi yapıp bağlı piksel sayısı belirli bir büyüklüğün üzerinde olan bağlı bileşenleri seçerek hareketli bölgeye denk gelen kısımları belirlemeye çalışınız. Benzer bir çerçeve döngüsü içerisinde ardışık çerçeve çiftleri için 3×3 büyüklükte başlayan pencereler kullanarak optik akış algoritmasını uygulayınız ve her piksel için (u,v) kayma bilgisini elde ediniz. Her piksel için pencere büyüklüğünü her adımda ikişer arttırıp bir önceki adımda belirlenen (u,v) değeriyle benzer bir kayma elde edene kadar (örneğin iki (u,v) arasındaki mesafe 1 veya 2 piksel altında olana değin) tekrar ediniz. Uygun gördüğünüz iki çerçeve için arka plan ve ön plan görüntüleri ile sadece seçilen bağlı bileşen piksellerine ait akış genliği ve akış açısı görüntülerini hesaplatıp raporunuza ekleyiniz. Videoda ön planı bulma başarısı ve genlik ile açı görüntülerinin araçların hareket bilgilerini ne derece yansıttığı üzerine değerlendirmede bulununuz.
- 3- Edinburgh mutfak eşyaları veri setini kullanarak sınıflandırma amaçlı derin öğrenme modelleri eğitip başarılarını değerlendiriniz. Bunun için ekte verilen transferLearning.py dosyasından faydalanınız. Bu dosyadaki load_data() fonksiyonu verilen bir dizin içerisinde yer alan tüm RAW dosyalarını okuyup 3-1 oranında eğitim ve test olarak ikiye ayırmaktadır. Uygulamayı problemsiz çalıştırma için Tensorflow kütüphanesine ek olarak imutils ve scikit-learn kütüphanelerini şu komutlarla kurunuz.

```
conda install -c conda-forge imutils  
conda install -c conda-forge scikit-learn
```

- Uygulamanın ilk kısmı için temelde `get_transfer_model()` fonksiyonu tamamlanacaktır. Bu fonksiyonda nakil öğrenme (transfer learning) için VGG-16 modeli tüm bağlı (fully connected) katmanları `include_top = False` seçeneğiyle çıkartılarak imagenet için öğrenilmiş ağırlıklar ile giriş katmanı (224, 224, 3) boyutlu görüntüleri alacak şekilde yüklenmiştir. Yüklenen bu VGG-16 modelinin katmanlarında öğrenme devre dışı bırakılarak filtrelerin yeterince ayırt edici özellikler çıkarabiliyor olmasından faydalanılacaktır. Bu modele ek olarak uygun gördüğünüz `conv2D`, `AveragePooling2D`, `MaxPooling2D`, `Dense`, `Flatten`, `Dropout`, `BatchNormalization` vb. katmaları ekleyerek modeli önceden paylaşılan 10 mutfak araç ve gerecini sınıflandıracak şekilde dönüştürünüz. Modelin doğru bir şekilde oluşturulması durumunda `train_model` ve `evaluate_model` fonksiyonları çalışacaktır. Gerek görüldüğü takdirde `train_model` ve `evaluate_model` fonksiyonlarında değişiklik yapılabilir. **Not:** İsteyenler orijinal verisetini indirip 20 sınıf üzerinden de çalışabilir.
- Uygulamanın ikinci kısmı için `get_your_model()` fonksiyonu tamamlanacaktır. Bu kısımda sadece kendi üreteceğiniz bir derin öğrenme modeli oluşturulacaktır. Yine `train_model` ve `evaluate_model` fonksiyonları kullanılarak eğitim ve değerlendirme gerçekleştirilecektir. **Not:** Mevcut bilgisayar alt yapınız dolayısıyla eğitimin gerçekleşmesi zaman alabilir. Bunun için oluşturduğunuz modellerdeki parametre sayılarının fazla yüksek olmamasına, epoch değerinin çok büyük olmamasına ve sadece 10 sınıflı veri kümesi üzerinde çalışmaya dikkat edebilirsiniz. Fakat bunların modelin başarısını arttırmak için yeterli büyüklükte olması gerektiği de unutulmamalıdır. Alternatif olarak Google Collaboratory veya Kaggle benzeri platformların hesaplama kapasitelerinden yararlanabilirsiniz.
- Geliştireceğiniz bu iki derin öğrenme modelinin katmanlarını şematik olarak açıklayınız. Test verisi üzerinde nakil öğrenme ile mi yoksa kendi oluşturduğunuz model ile mi daha yüksek başarılar elde ettiğinizi değerlendiriniz. Değerlendirmenin hep aynı eğitim ve test kümesi kullanılarak gerçekleştirilmesini garantilemek için `train_test_split` fonksiyonundaki `random_state=42` seçeneğini değiştirmeyiniz.

Not: Ödev için gerektiği yerde derste referans olarak verilen kodları veya hazır kütüphane fonksiyonlarını kullanabilirsiniz.

Teslim:

- Yaptığınız çalışmadan tek bir pdf raporu hazırlayıp Classroom'a yükleyiniz.
- Raporda,
 1. orijinal görüntülerin işlenmesi ile elde edilen uyumlu veya aykırı noktalar, eğilmiş görüntüler, arka plan, ön plan ve akış görüntüleri, diğer ara görüntüler ve karmaşa matrisi benzeri bilgileri gerektiği yerde ekleyerek açıklayınız.
 2. kullandığınız algoritmalardan kısaca bahsediniz.
 3. her geliştirdiğiniz uygulama neticesinde elde ettiğiniz bilgi ve görüntüleri kısaca yorumlayıp tartışınız.
- Son olarak **kaynak kodlarını** açıklamalı olarak hazırlayıp aynı raporun sonuna ekleyiniz.
- Ödeve isminizi yazmayı unutmayınız.