

# Study Case 1 — Klasik Görüntü İşleme ile Hareketli Bölgelerin Ayırıştırılması

Bu çalışmada, gerçek hayattaki kamera görüntülerinde sık görülen bir problemi ele almanı istiyoruz: hareketli bölgeleri (foreground) arka plandan güvenilir şekilde ayırmak ve bunu tek başına anlamlı çıktılara dönüştürmek. Bu case'in odağı görüntü işleme algoritmaları kullanarak kod geliştirme olmalıdır.

Bu case CDnet 2014 veri seti ile yapılacaktır. Bu veri seti gölge ve aydınlanma değişimleri, dinamik arka plan, ghosting / kalıcı izler, kamera jitter, düşük kalite / gürültü gibi zorlayıcı senaryolar içerir.

Alt küme seçimini (kategori/sekans sayısı vb.) sen yapabilirsin. Ancak çalıştığın sekansları repo içinde açıkça belirtmelisin.

Bu çalışma sonunda videodan ürettiğin sonuçların hem görsel hem de sayısal olarak anlaşılır olmasını bekliyoruz. Örnek çıktı türleri şunlar olabilir (zorunlu liste değil, yönlendirme amaçlıdır):

- Her kare için foreground mask benzeri ikili çıktı
- Maskeden türetilen bölge/nesne özetleri (örn. bounding box, connected components istatistikleri)
- Sonucu hızlı anlamamızı sağlayacak overlay görselleri veya kısa demo çıktıları
- Sonuçları özetleyen metrik/tablolara ve kısa yorum

Burada görev tarif etmiyoruz. Problemden doğru çıkarımı yapman ve çözümü kendin kurgulamalısın.

## İpuçları

Şu yaklaşımlar faydalı olabilir;

- “Neden hata oluyor?” sorusunu somut örneklerle incelemek (gölge, arka plan hareketi vb.)
- Çıktıyı daha kararlı hale getiren iyileştirmeler düşünmek (post-process fikirleri)
- Farklı sahne tiplerinde davranışı karşılaştırmak (bazı sekanslarda neden bozuluyor?)
- Sonucu yalnızca görsel değil, ölçülebilir metriklerle de özetlemek

Bizim için önemli olan: deneyerek öğrenmen, gözlem yapman ve kararlarını gerekçelendirmen.

## Teslim Formatı

### 1) Git Repository

Çalışmanı bir Git (Github/Gitlab vs.) reposu olarak paylaşmalısın. Repo temiz, çalıştırılabilir ve tekrar üretilebilir olmalı.

## 2) Repo formatı

- **README.md**
  - Projenin kısa özeti
  - Kurulum ve çalıştırma talimatı
  - Kullandığın dataset sekansları/kategorileri
  - Üretilen çıktılar nerede oluşuyor? (klasör yapısı)
- **requirements.txt** veya **environment.yml**
- Kod yapısı (tercihen **src/** altında)
  - Notebook kullanabilirsin
- **report.pdf** (öneri: 2–4 sayfa)
  - Yaklaşımın kısa özeti
  - Örnek overlay çıktılar
  - Metrik/tablolara ve yorum
  - Zorlandığın örnekler + kısa hata analizi

## 3) Çıktı artefaktları (repo içinde)

Repo içinde (veya Git release / link ile) şu tip çıktıları bekliyoruz:

- En az 1–2 sekans için kısa demo video/GIF (overlay'li)
- Sonuç özetini gösteren grafik/tablolara (PNG veya PDF)
- (Varsa) metrikleri içeren **results.csv** benzeri dosya

## 4) Çalıştırılabilirlik standardı

Tercihimiz:

- Tek bir giriş noktası (örn. **python run.py ...**)
- Çıktıları belirli bir klasöre yazma (**outputs/** gibi)