

AI Modelleri

1. Convolutional Neural Networks (CNNs)

- **Doğruluk:** CNN'ler genellikle yüksek doğruluk oranlarına sahiptir. Özellikle büyük veri kümeleri üzerinde eğitildiğinde başarılı sonuçlar verir.
- **Hız:** Basit CNN modelleri hızlı olabilir, ancak model derinliği arttıkça işlem süresi uzayabilir. Çok katmanlı modellerde hız, düşük donanımda sınırlı olabilir.
- **Hafıza Kullanımı:** Daha az katmanlı CNN'ler nispeten hafif hafıza kullanımı sağlar. Fakat katman sayısı ve parametreler arttıkça hafıza kullanımı da artar.

2. ResNet (Residual Networks)

- **Doğruluk:** ResNet, derin katman yapısı ile yüksek doğruluk sağlar. 152 katmanlı versiyonları bile yüksek performans gösterir.
- **Hız:** ResNet modelleri, katman sayısına bağlı olarak yavaş olabilir, ancak residual bloklar işlemlerin hızlanmasını sağlar.
- **Hafıza Kullanımı:** Derin ağlar olduğundan, özellikle 101 veya 152 katmanlı versiyonlar yüksek hafıza kullanır.

3. YOLO (You Only Look Once)

- **Avantajları:** Gerçek zamanlı nesne algılamada çok hızlıdır. Tek geçişte hem nesneleri tanımlar hem de tespit eder, bu nedenle otonom sürüşte engeller ve yolların hızlı tespiti için uygundur.
- **Uygulama Alanları:** Trafik kameraları, drone'lar, güvenlik uygulamaları.
- **Uygun Veri Setleri:** COCO (Common Objects in Context), KITTI.
- **Doğruluk:** YOLO, hızlı olmasına rağmen genellikle yüksek doğruluğa sahip değildir. Özellikle küçük nesneleri algılamada zayıf olabilir.
- **Hız:** YOLO'nun en büyük avantajı gerçek zamanlı nesne algılama yeteneğidir. Milyonlarca parametreyi saniyeler içinde işleyebilir.
- **Hafıza Kullanımı:** YOLO oldukça verimlidir ve mobil cihazlarda bile düşük hafıza kullanarak çalışabilir.

4. Faster R-CNN

- **Doğruluk:** Nesne algılamada yüksek doğruluk sağlar. Büyük veri kümelerinde oldukça etkili sonuçlar verir.
- **Hız:** Faster R-CNN, seleflerinden daha hızlı olmasına rağmen, YOLO gibi gerçek zamanlı işleme hızına ulaşamaz. Yavaş ama doğruluğu yüksektir.
- **Hafıza Kullanımı:** Model büyüklüğüne bağlı olarak hafıza kullanımı yüksektir. Derin öğrenme ağı gerektirdiğinden güçlü donanım ihtiyacı vardır.
- **Uygulama Alanları:** Yol ve çevre segmentasyonu, nesne algılama.
- **Uygun Veri Setleri:** Pascal VOC, KITTI, Cityscapes.

5. Mask R-CNN

- **Doğruluk:** Mask R-CNN, nesne tespiti ve segmentasyon işlemlerinde çok yüksek doğruluk sunar. Özellikle piksel düzeyinde nesne segmentasyonu yapabilmesi büyük avantajdır.
- **Hız:** Segmentasyon işlemleri nedeniyle oldukça yavaştır. Özellikle yüksek çözünürlüklü görüntülerde hız sorunları olabilir.
- **Hafıza Kullanımı:** Nesne maskeleme işlemleri yoğun hafıza gerektirir. Bu model hafıza açısından maliyetlidir.
- **Uygulama Alanları:** Yaya tespiti, trafik işaretleri, yol ve engel segmentasyonu.
- **Uygun Veri Setleri:** COCO, Cityscapes.

6. EfficientDet

- **Doğruluk:** YOLO ve Faster R-CNN'e kıyasla daha dengeli bir doğruluk-hız oranı sunar. Verimlilik için optimize edilmiştir, fakat doğruluğu da oldukça yüksektir.
- **Hız:** EfficientDet, özellikle mobil cihazlarda ve düşük güçlü donanımlarda hızlıdır. Yüksek hız avantajı vardır.
- **Hafıza Kullanımı:** EfficientNet'in temelinde olduğu gibi, EfficientDet de hafızayı verimli kullanarak yüksek performans sağlar.
- **Uygulama Alanları:** Otonom araç sistemleri, akıllı şehir uygulamaları.
- **Uygun Veri Setleri:** COCO, BDD100K.

7. MobileNet

- **Doğruluk:** MobileNet, mobil cihazlar için optimize edilmiştir, bu nedenle doğruluk genellikle çok derin ağlar kadar yüksek değildir, ancak kabul edilebilir seviyededir.
- **Hız:** Hızlı ve hafif bir modeldir. Mobil cihazlarda gerçek zamanlı çalışabilir.
- **Hafıza Kullanımı:** Çok düşük hafıza kullanımıyla çalışır. Mobil cihazlar için uygun hale getirilmiştir.

8. DeepLab

- **Doğruluk:** Semantik segmentasyonda mükemmel doğruluk sağlar. Nesnelerin piksellerini doğru bir şekilde sınıflandırmada iyidir.
- **Hız:** Segmentasyon işlemleri hızlı değildir, ancak doğruluk avantajı sunar.
- **Hafıza Kullanımı:** Segmentasyon işlemlerinde yüksek hafıza kullanır.
- **Uygulama Alanları:** Otonom sürüş, haritalama, engel tespiti.
- **Uygun Veri Setleri:** Cityscapes, ADE20K, Mapillary Vistas.

Model	Doğruluk	Hız	Hafıza Kullanımı	Uygun Veri Setleri	Uygulama Alanları
CNN (Temel)	Yüksek, özellikle büyük veri kümelerinde	Basit modeller hızlı, derin modeller yavaş	Az katmanlılar hafif, derinler daha fazla hafıza kullanır	COCO, Pascal VOC, KITTI	Genel nesne tanıma, sınıflandırma
ResNet	Çok yüksek, derin katman yapısı ile	Katman sayısına bağlı olarak yavaş	Derin ağlar yüksek hafıza kullanır	COCO, Cityscapes, ADE20K	Derin öğrenme, nesne tespiti
YOLO	Orta, küçük nesneleri algılama zayıf	Çok hızlı, gerçek zamanlı	Düşük hafıza kullanımı, mobil cihazlarda da çalışır	COCO, KITTI	Gerçek zamanlı nesne algılama, otonom sürüş
Faster R-CNN	Yüksek, nesne algılamada başarılı	Orta, YOLO kadar hızlı değil	Yüksek hafıza kullanımı	Pascal VOC, KITTI, Cityscapes	Yol ve çevre segmentasyonu, nesne algılama
Mask R-CNN	Çok yüksek, piksel bazlı segmentasyon	Yavaş, özellikle yüksek çözünürlükte	Yoğun hafıza kullanımı	COCO, Cityscapes	Yaya tespiti, trafik işaretleri, yol segmentasyonu
EfficientDet	Dengeli, yüksek doğruluk ve hız	Yüksek hız, mobil cihazlar için optimize	Verimli hafıza kullanımı	COCO, BDD100K	Otonom araç sistemleri, akıllı şehir uygulamaları
MobileNet	Kabul edilebilir, yüksek değil	Çok hızlı, mobil cihazlarda gerçek zamanlı	Çok düşük hafıza kullanımı	COCO, Pascal VOC	Mobil cihazlar için nesne tanıma, sınıflandırma
DeepLab	Çok yüksek, semantik segmentasyon	Orta, segmentasyon işlemleri yavaş	Yüksek hafıza kullanımı	Cityscapes, ADE20K, Mapillary Vistas	Otonom sürüş, haritalama, engel tespiti

DATASETS

1. COCO (Common Objects in Context)

- **Kapsam:** COCO, 330.000'den fazla görüntü içeren büyük bir veri setidir. 80 nesne sınıfına ayrılmış ve 1.5 milyon nesne etiketi içerir. Nesnelerin etraflarındaki bağlarla birlikte tanımlanmasını sağlar.
- **Veri Türleri:** Nesne tespiti, segmentasyon (hem nesne hem de piksel düzeyinde), anahtar nokta tespiti (insan pozları için) ve görüntü açıklaması içerir.
- **Sınıflar:** İnsanlar, hayvanlar, araçlar, mobilyalar, ev eşyaları, elektronik cihazlar, yiyecekler gibi çok çeşitli nesneleri kapsar.
- **Uygulama Alanları:** Genel nesne algılama ve tanıma, görüntü açıklaması oluşturma, otonom sürüş, robotik.
- **Artılar:**
 - **Bağlam Zenginliği:** Nesnelerin sadece varlığını değil, birbirleriyle olan ilişkilerini de kapsayan zengin bir veri seti.
 - **Geniş Kapsam:** Çok geniş sınıf ve veri içeriği ile pek çok farklı uygulamaya uygundur.
 - **Desteklenen Görevler:** Nesne tespiti, segmentasyon, anahtar nokta tespiti ve daha birçok farklı görevde kullanılabilir.
- **Eksiler:**
 - **Çeşitli ve karmaşık veri:** Bu zenginlik, model eğitimi ve işlemesi açısından bazen karmaşıklık yaratabilir.

2. KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute)

- **Kapsam:** Otonom sürüş sistemlerine yönelik 7481 eğitim görüntüsü ve 80.256 nesne etiketi içeren bir veri setidir. Lidar (lazer tabanlı) ve stereo kamera görüntüleri sunar.
- **Veri Türleri:** Nesne tespiti, izleme, yol algılama, stereo kamera görüntüleri ve Lidar verileri içerir.
- **Sınıflar:** Araçlar, yayalar, bisikletliler gibi trafikle ilişkili nesneler.
- **Uygulama Alanları:** Otonom sürüş, araç içi nesne tespiti, trafik izleme ve analiz.
- **Artılar:**
 - **Otonom Sürüş için Optimize Edilmiş:** Gerçek trafik sahneleri ve şehir yolları içerir, bu da otonom araç projeleri için mükemmel bir veri seti sağlar.

- **Lidar ve Stereo Görüntüler:** 3D algılama görevlerinde güçlüdür.
 - **Gerçek dünya verisi:** Şehir trafiği içinde veri toplandığı için gerçek dünya koşullarını yansıtır.
 - **Eksiler:**
 - **Sınırlı Sınıf Sayısı:** Sınıf sayısı COCO veya Pascal VOC kadar geniş değildir, sadece otonom araçlarla ilgili sınıfları içerir.
 - **Lidar İşleme Zorluğu:** Lidar verileri karmaşık olduğu için, veri işleme daha fazla kaynak ve zaman gerektirebilir.
-

3. Cityscapes

- **Kapsam:** 5.000'den fazla görüntü içeren bir semantik segmentasyon veri setidir. Veriler Avrupa'daki çeşitli şehirlerden toplanmıştır. Her görüntü piksel bazlı olarak anotlanmıştır.
 - **Veri Türleri:** Semantik segmentasyon, nesne algılama ve piksel tabanlı sınıflandırma.
 - **Sınıflar:** Yollar, binalar, yayalar, araçlar, bisikletliler, trafik işaretleri, bitkiler, gökyüzü gibi şehirde karşılaşılabilecek nesneler.
 - **Uygulama Alanları:** Otonom sürüş, şehir planlama, akıllı şehirler için semantik segmentasyon, trafik analizi.
 - **Artılar:**
 - **Şehir Segmentasyonu için İdeal:** Kentsel alanlardaki nesnelerin piksellere dayalı olarak segmentasyonunu sağlar.
 - **Detaylı Anotasyon:** Piksel tabanlı anotasyon, yollar, binalar ve trafik unsurlarını detaylı olarak analiz etmeyi mümkün kılar.
 - **Eksiler:**
 - **Kapsam Dar:** Kırsal alanları veya doğal ortamlardaki nesneleri içermediği için şehir dışında kullanımı sınırlıdır.
 - **Yüksek İşlem Gücü Gerektirir:** Piksel tabanlı veri işleme yüksek hesaplama gücü gerektirir.
-

4. Pascal VOC (Visual Object Classes)

- **Kapsam:** 20 sınıfta nesne algılama, sınıflandırma, segmentasyon ve eylem tanıma görevlerini içeren bir veri setidir. Görüntülerdeki nesnelerin sınır kutularıyla anotlanmıştır.
- **Veri Türleri:** Nesne algılama, sınıflandırma, semantik segmentasyon.

- **Sınıflar:** Araçlar, hayvanlar, insanlar, mobilyalar, yiyecekler gibi yaygın nesneler.
 - **Uygulama Alanları:** Genel nesne tespiti ve sınıflandırma, semantik segmentasyon, görüntü tanıma.
 - **Artılar:**
 - **Çeşitli Nesne Türleri:** Yaygın nesnelerden oluşan geniş bir sınıf yelpazesi sunar.
 - **Yıllara Göre Veriler:** Her yıl güncellenen veri setleri sayesinde performans gelişimleri değerlendirilebilir.
 - **Eksiler:**
 - **Veri Seti Boyutu Küçük:** COCO gibi daha büyük veri setleri ile karşılaştırıldığında boyut olarak küçüktür.
 - **Geniş Segmentasyon Verisi İçermez:** Piksel tabanlı segmentasyon yerine nesneye yönelik sınır kutuları içerir.
-

5. BDD100K (Berkeley DeepDrive)

- **Kapsam:** 100.000 video ve yaklaşık 10 milyon etiket içeren geniş bir veri setidir. Otonom sürüş sistemlerinin geliştirilmesi için tasarlanmıştır.
 - **Veri Türleri:** Nesne tespiti, yol izleme, çevresel koşul analizi (gece-gündüz, hava durumu) ve hareket analizi.
 - **Sınıflar:** Araçlar, yayalar, trafik işaretleri, yol işaretlemeleri, bisikletliler, hava durumu ve trafik koşulları.
 - **Uygulama Alanları:** Otonom sürüş, trafik izleme, çevresel durum analizi, araç içi güvenlik sistemleri.
 - **Artılar:**
 - **Çevresel Koşulların Analizi:** Gece ve gündüz, farklı hava durumları gibi koşulların tespiti, otonom araçlar için çok değerli.
 - **Çok Büyük Veri:** 100.000 video, gerçek dünyada karşılaşılan çok sayıda senaryoyu kapsar.
 - **Eksiler:**
 - **Yüksek Hesaplama Gereksinimi:** Çok büyük veri seti olması nedeniyle eğitim ve işlem süreleri uzayabilir.
 - **Veri Karmaşıklığı:** Çok çeşitli koşulların yer alması, eğitim sürecini karmaşık hale getirebilir.
-

6. Mapillary Vistas

- **Kapsam:** 25.000'den fazla görüntü ve 100'den fazla sınıf içerir. Görüntüler dünya çapında toplandığı için farklı coğrafi ve kültürel özelliklere sahip.
 - **Veri Türleri:** Semantik segmentasyon ve nesne algılama.
 - **Sınıflar:** Yollar, binalar, araçlar, bitkiler, işaretler, bisikletliler, hayvanlar ve diğer çevresel nesneler.
 - **Uygulama Alanları:** Otonom sürüş, haritalama, çevresel analiz, akıllı şehir projeleri.
 - **Artılar:**
 - **Dünya Çapında Çeşitli Görüntüler:** Farklı coğrafi bölgelerden veri içermesi, geniş kapsamlı uygulamalara olanak tanır.
 - **Detaylı Segmentasyon:** Piksel düzeyinde detaylı segmentasyon sağlar.
 - **Eksiler:**
 - **Yüksek Anotasyon Maliyeti:** Piksel tabanlı verilerin işlenmesi çok fazla zaman ve işlem gücü gerektirir.
-

7. ADE20K (A Large-Scale Scene Parsing Benchmark)

- **Kapsam:** 20.000 görüntü ve 150'den fazla sınıf içerir. Her görüntü piksellere ayrılarak detaylı şekilde anotlanmıştır.
- **Veri Türleri:** Semantik segmentasyon ve sahne anlama.
- **Sınıflar:** İnsanlar, hayvanlar, yapılar, doğa, yollar, mobilyalar ve diğer çevresel nesneler.
- **Uygulama Alanları:** Semantik segmentasyon, sahne analizi, robotik, otonom sürüş.
- **Artılar:**
 - **Geniş Sınıf Yelpazesi:** Pek çok farklı sahne ve nesne sınıfı içerir.
 - **Detaylı Segmentasyon:** Piksel tabanlı segmentasyon, sahne analizi için uygundur.
- **Eksiler:**
 - **Yüksek İşleme Gücü Gerektirir:** Yüksek çözünürlüklü ve detaylı görüntülerin işlenmesi zor olabilir.

Veri Seti	Kapsam/Alan	Görüntü Sayısı	Etiket Türleri	Sınıf Sayısı	Çözünürlük	Veri Türü	Kullanım Alanları
COCO	Genel nesne tanıma	330K+	Nesne, bölütleme, keypoint	80	640x480 - 1280x720	RGB	Nesne algılama, bölütleme, keypoint
KITTI	Otonom sürüş (araç içi kamera)	15K+	Nesne, derinlik, akış	8	1242x375	RGB, Stereo	Otonom araçlar, 3D algılama
Cityscapes	Kentsel sahne bölütleme	5K (fine)	Bölütleme, derinlik	30	2048x1024	RGB	Kentsel alan bölütleme, otonom sürüş
Pascal VOC	Genel nesne tanıma	11K+	Nesne, bölütleme	20	Değişken	RGB	Nesne tanıma, bölütleme
BDD100K	Otonom sürüş (genişletilmiş)	100K+	Nesne, bölütleme, yol durumu	10	1280x720	RGB	Otonom araçlar, nesne algılama
Mapillary Vistas	Kentsel sahne bölütleme	25K+	Bölütleme, trafik işaretleri	66	1920x1080+	RGB	Trafik işareti algılama, bölütleme
ADE20K	Genel sahne bölütleme	25K+	Bölütleme	150+	Değişken	RGB	Genel sahne ve nesne bölütleme

API ve SDK

1. TensorFlow

Genel Bilgi:

TensorFlow, Google tarafından geliştirilen açık kaynaklı bir makine öğrenimi kütüphanesidir. Geniş bir model yelpazesini destekler ve hem araştırma hem de üretim ortamlarında yaygın olarak kullanılır.

Desteklenen Modeller:

- **CNNs, ResNet, YOLO, Faster R-CNN, Mask R-CNN, EfficientDet, MobileNet, DeepLab**

Avantajlar:

- **Kapsamlı Dokümantasyon ve Topluluk Desteği:** TensorFlow, geniş bir kullanıcı ve geliştirici topluluğuna sahiptir. Öğrenme kaynakları, örnek projeler ve topluluk forumları mevcuttur.
- **TensorFlow Hub:** Önceden eğitilmiş modelleri kolayca entegre edebilirsiniz.
- **TensorFlow Lite:** Mobil ve gömülü cihazlar için optimize edilmiş modeller sunar.
- **Scalability:** Büyük ölçekli dağıtık eğitim ve üretim ortamları için uygundur.
- **Keras Entegrasyonu:** Yüksek seviyeli API (Keras) ile model geliştirme sürecini basitleştirir.

Dezavantajlar:

- **Öğrenme Eğrisi:** Özellikle yeni başlayanlar için karmaşık olabilir.
- **Performans:** Bazı durumlarda PyTorch kadar esnek olmayabilir.

Kullanım Kolaylığı:

Orta. Keras ile entegrasyon, model geliştirmeyi kolaylaştırırken, daha derin özellikler için ek öğrenme gerektirebilir.

2. PyTorch

Genel Bilgi:

PyTorch, Facebook tarafından geliştirilen açık kaynaklı bir makine öğrenimi kütüphanesidir. Dinamik hesap grafikleri ve Python ile derin entegrasyonu sayesinde araştırma ve geliştirme topluluğunda popülerdir.

Desteklenen Modeller:

- **CNNs, ResNet, YOLO, Faster R-CNN, Mask R-CNN, EfficientDet, MobileNet, DeepLab**

Avantajlar:

- **Dinamik Hesap Grafikleri:** Esnek ve dinamik model tanımlamayı destekler, bu da hata ayıklamayı ve araştırmayı kolaylaştırır.
- **Python Dostu:** Python ile doğal entegrasyonu, geliştirme sürecini hızlandırır.
- **TorchScript:** Modelleri üretime almak için optimize eder.
- **Geniş Model Ekosistemi:** torchvision, detectron2 gibi ek kütüphanelerle geniş model desteği sağlar.
- **Topluluk ve Akademik Destek:** Araştırma topluluğu tarafından yaygın olarak kullanılır, sürekli güncellenir ve desteklenir.

Dezavantajlar:

- **Dağıtık Eğitim:** TensorFlow kadar yerleşik değildir, ancak gelişmekte olan çözümler mevcuttur.
- **Mobil Desteği:** TensorFlow Lite kadar geniş değil, ancak PyTorch Mobile ile geliştirilmeye devam ediyor.

Kullanım Kolaylığı:

Yüksek. Dinamik yapısı ve Python ile entegrasyonu, özellikle araştırma ve prototipleme aşamalarında kullanıcı dostudur.

3. OpenCV

Genel Bilgi:

OpenCV (Open Source Computer Vision Library), bilgisayarla görme görevleri için geniş bir araç seti sunan açık kaynaklı bir kütüphanedir. Derin öğrenme entegrasyonları sayesinde çeşitli modelleri destekler.

Desteklenen Modeller:

- **YOLO, MobileNet, ResNet, EfficientDet, Faster R-CNN**

Avantajlar:

- **Hafif ve Hızlı:** Gerçek zamanlı uygulamalar için optimize edilmiştir.
- **Geniş Fonksiyon Seti:** Görüntü işleme, video analizi ve bilgisayarla görme görevleri için kapsamlı araçlar sunar.
- **DNN Modülü:** TensorFlow, Caffe, PyTorch gibi farklı framework'lerden modelleri destekler.
- **Platform Bağımsız:** Çeşitli işletim sistemlerinde ve cihazlarda çalışabilir.

Dezavantajlar:

- **Derin Öğrenme Özellikleri:** TensorFlow veya PyTorch kadar kapsamlı derin öğrenme desteği sunmaz.
- **Kısıtlı Model Eğitimi:** Genellikle model eğitimi yerine, önceden eğitilmiş modellerin uygulanması için kullanılır.

Kullanım Kolaylığı:

Orta. Bilgisayarla görme görevlerinde güçlüdür, ancak derin öğrenme modellerinin eğitimi için sınırlı araçlar sunar.

4. Detectron2

Genel Bilgi:

Facebook AI Research (FAIR) tarafından geliştirilen Detectron2, nesne algılama ve segmentasyon için optimize edilmiş bir PyTorch tabanlı kütüphanedir.

Desteklenen Modeller:

- **Faster R-CNN, Mask R-CNN**

Avantajlar:

- **Modüler ve Esnek:** Farklı model mimarilerini kolayca deneyebilirsiniz.
- **Hızlı Prototipleme:** Önceden yapılandırılmış konfigürasyonlar ve eğitim script'leri ile hızlı model geliştirme imkanı.
- **Topluluk ve Destek:** Aktif geliştirme ve geniş dokümantasyon.
- **Entegrasyon:** PyTorch ile doğal entegrasyon sunar, bu da kullanımını kolaylaştırır.

Dezavantajlar:

- **Öğrenme Eğrisi:** Kapsamlı özellikleri nedeniyle başlangıçta karmaşık olabilir.
- **Performans:** Özellikle çok büyük modellerde, optimize edilmesi gerekebilir.

Kullanım Kolaylığı:

Orta. PyTorch bilgisi gerektirir, ancak sunduğu araçlar ve dokümantasyon ile güçlü bir destek sağlar.

5. TensorFlow Lite

Genel Bilgi:

TensorFlow Lite, TensorFlow modellerini mobil ve gömülü cihazlarda çalıştırmak için optimize edilmiş hafif bir versiyonudur.

Desteklenen Modeller:

- **YOLO, MobileNet, EfficientDet, DeepLab, ResNet**

Avantajlar:

- **Mobil ve Gömülü Cihazlar için Optimize Edilmiş:** Düşük gecikme süresi ve düşük hafıza kullanımı.
- **Kolay Entegrasyon:** Android ve iOS için destek sağlar.
- **Edge TPU Desteği:** Google Coral gibi özel donanımlarda hızlandırma imkanı.
- **Quantization:** Model boyutunu küçültmek ve hızlandırmak için optimizasyonlar sunar.

Dezavantajlar:

- **Model Dönüşümü:** TensorFlow modellerinin TensorFlow Lite formatına dönüştürülmesi ek adımlar gerektirebilir.
- **Özellik Kısıtlamaları:** Tam TensorFlow özelliklerinin tümü desteklenmez.

Kullanım Kolaylığı:

Yüksek. TensorFlow ekosistemindeki araçlarla entegre edilmiştir ve mobil uygulamalara kolayca entegre edilebilir.

6. ONNX (Open Neural Network Exchange)

Genel Bilgi:

ONNX, farklı derin öğrenme framework'leri arasında model değişimini kolaylaştıran açık bir format ve ekosistemdir.

Desteklenen Modeller:

- **CNNs, ResNet, YOLO, Faster R-CNN, Mask R-CNN, EfficientDet, MobileNet, DeepLab**

Avantajlar:

- **Çapraz Platform Desteği:** PyTorch, TensorFlow, Keras gibi farklı framework'lerden modelleri destekler.
- **Hızlandırıcı Entegrasyonu:** ONNX Runtime ile CPU, GPU ve özel donanımlarda hızlı model çalıştırma imkanı.
- **Esneklik:** Modelleri kolayca dönüştürebilir ve çeşitli platformlarda kullanabilirsiniz.
- **Topluluk Desteği:** Geniş bir topluluk ve sürekli gelişen ekosistem.

Dezavantajlar:

- **Uyumluluk Sorunları:** Bazı özel veya nadir kullanılan katmanlar ve operasyonlar desteklenmeyebilir.

- **Model Dönüşümü Karmaşıklığı:** Bazı durumlarda model dönüşümü ek yapılandırma gerektirebilir.

Kullanım Kolaylığı:

Orta. ONNX formatına dönüştürme ve uyumluluk sorunlarını çözmek için ek adımlar gerekebilir, ancak esneklik ve platform bağımsızlığı büyük avantaj sağlar.

7. Hugging Face Transformers

Genel Bilgi:

Hugging Face, özellikle doğal dil işleme (NLP) alanında popüler olan açık kaynaklı bir kütüphanedir, ancak görsel modeller için de destek sunmaktadır.

Desteklenen Modeller:

- **YOLO, ResNet, EfficientDet, MobileNet, DeepLab** (Genişletilmiş olarak)

Avantajlar:

- **Kolay Kullanım:** Basit API'ler ve önceden eğitilmiş modeller sunar.
- **Topluluk ve Destek:** Aktif topluluk ve sürekli güncellenen modeller.
- **Entegrasyon:** TensorFlow ve PyTorch ile entegre çalışabilir.
- **Model Hub:** Geniş bir model havuzu sunar, farklı görevler için hızlı entegrasyon sağlar.

Dezavantajlar:

- **Odak Noktası:** Aslen NLP odaklıdır, görsel modeller için destek sınırlı olabilir.
- **Performans:** Özellikle büyük modellerde, performans optimizasyonu gerekebilir.

Kullanım Kolaylığı:

Yüksek. Basit API'ler ve geniş model yelpazesi sayesinde hızlı prototipleme imkanı sağlar.

8. NVIDIA DeepStream SDK

Genel Bilgi:

NVIDIA DeepStream SDK, video analizi ve akıllı kamera uygulamaları için optimize edilmiş bir çözüm sunar. GPU hızlandırmalı akış işleme sağlar.

Desteklenen Modeller:

- **YOLO, ResNet, Faster R-CNN, Mask R-CNN, EfficientDet, MobileNet, DeepLab**

Avantajlar:

- **GPU Hızlandırma:** NVIDIA GPU'ları ile yüksek performanslı akış işleme.

- **Entegre Çözümler:** Video akışı, nesne algılama, izleme ve analiz için entegre araçlar sunar.
- **Düşük Gecikme:** Gerçek zamanlı uygulamalar için optimize edilmiştir.
- **DeepStream Marketplace:** Önceden optimize edilmiş modeller ve eklentiler sunar.

Dezavantajlar:

- **Donanım Bağımlılığı:** NVIDIA GPU gerektirir, bu da donanım maliyetlerini artırabilir.
- **Karmaşıklık:** Gelişmiş özellikler ve entegrasyonlar için öğrenme eğrisi bulunabilir.

Kullanım Kolaylığı:

Orta. NVIDIA ekosistemine aşina olan kullanıcılar için güçlü araçlar sunarken, yeni başlayanlar için karmaşık olabilir.

9. Microsoft Cognitive Toolkit (CNTK)

Genel Bilgi:

Microsoft tarafından geliştirilen açık kaynaklı bir derin öğrenme framework'üdür. Yüksek performanslı eğitim ve model oluşturma imkanı sunar.

Desteklenen Modeller:

- **CNNs, ResNet, YOLO, Faster R-CNN, Mask R-CNN, EfficientDet, MobileNet, DeepLab**

Avantajlar:

- **Yüksek Performans:** Verimli hesaplama ve paralel eğitim yetenekleri.
- **Windows Entegrasyonu:** Windows tabanlı sistemlerde iyi çalışır.
- **Esneklik:** Farklı model mimarilerini destekler.

Dezavantajlar:

- **Topluluk ve Destek:** TensorFlow ve PyTorch kadar geniş bir topluluk desteği yoktur.
- **Geliştirme Durumu:** Aktif olarak geliştirilmiyor, yerini diğer framework'lere bırakmıştır.

Kullanım Kolaylığı:

Orta. Esnek olmasına rağmen, daha popüler framework'lere kıyasla daha az destek ve kaynak mevcuttur.

10. FastAI

Genel Bilgi:

FastAI, PyTorch üzerine inşa edilmiş yüksek seviyeli bir derin öğrenme kütüphanesidir. Eğitim sürecini hızlandırmak için tasarlanmıştır.

Desteklenen Modeller:

- **CNNs, ResNet, YOLO, MobileNet, EfficientDet, DeepLab** (Doğrudan destek sınırlı olabilir)

Avantajlar:

- **Kullanım Kolaylığı:** Yüksek seviyeli API'ler ile hızlı model geliştirme.
- **Eğitim Kaynakları:** Detaylı eğitim materyalleri ve topluluk desteği.
- **Entegrasyon:** PyTorch ile tam uyumlu çalışır.

Dezavantajlar:

- **Model Desteği:** Özel nesne algılama modelleri için sınırlı olabilir.
- **Esneklik:** Derin özelleştirme gerektiren projelerde sınırlamalar olabilir.

Kullanım Kolaylığı:

Yüksek. Özellikle eğitim ve prototipleme aşamalarında hızlı sonuçlar almak isteyenler için idealdir.

API/SDK	Desteklenen Modeller	Kullanım Kolaylığı	Performans	Topluluk ve Destek	Öne Çıkan Özellikler
TensorFlow	Tüm listelenen modeller	Orta	Yüksek (özellikle büyük ölçek)	Çok Geniş	Keras entegrasyonu, TensorFlow Lite
PyTorch	Tüm listelenen modeller	Yüksek	Yüksek	Çok Geniş	Dinamik hesap grafikleri, TorchScript
OpenCV	YOLO, MobileNet, ResNet, EfficientDet, Faster R-CNN	Orta	Yüksek (gerçek zamanlı)	Geniş	Bilgisayarla görme, DNN modülü

Detectron2	Faster R-CNN, Mask R-CNN	Orta	Yüksek	Orta-Geniş	Modüler yapı, hızlı prototipleme
TensorFlow Lite	YOLO, MobileNet, EfficientDet, DeepLab, ResNet	Yüksek	Çok Yüksek (mobil optimizasyon)	Geniş	Mobil ve gömülü cihazlar için optimize
ONNX	Tüm listelenen modeller	Orta	Yüksek	Geniş	Çapraz platform, hızlandırıcı entegrasyonu
Hugging Face	YOLO, ResNet, EfficientDet, MobileNet, DeepLab	Yüksek	Orta-Yüksek	Çok Geniş	Basit API, geniş model havuzu
NVIDIA DeepStream	YOLO, ResNet, Faster R-CNN, Mask R-CNN, EfficientDet, MobileNet, DeepLab	Orta	Çok Yüksek (GPU hızlandırmalı)	Orta-Geniş	Video analizi, gerçek zamanlı akış işleme
CNTK	Tüm listelenen modeller	Orta	Yüksek	Sınırlı	Yüksek performanslı hesaplama
FastAI	CNNs, ResNet, YOLO, MobileNet, EfficientDet, DeepLab	Yüksek	Yüksek	Orta-Geniş	Hızlı prototipleme, eğitim odaklı

Model, Veri Seti ve SDK Seçimi

Model Seçimi: MobileNet

MobileNet, düşük hesaplama gücü ve hafıza gereksinimleri ile bilinen bir derin öğrenme modelidir. Bu model, gömülü sistemlerde ve mobil cihazlarda kullanım için optimize edilmiştir.

- **Hafif Yapı:** MobileNet, küçük model boyutu ve düşük işlemci gücü gereksinimi ile gerçek zamanlı uygulamalar için idealdir.
- **Doğruluk-Oran Dengesi:** MobileNet, yüksek doğruluk sağlamasına rağmen, büyük modeller kadar performans gerektirmemektedir. Bu, düşük donanım kaynağına sahip cihazlarda kullanılabilirliğini artırmaktadır.

SDK Seçimi: TensorFlow Lite

MobileNet modelinin hızlı ve verimli bir şekilde çalışması için TensorFlow Lite SDK önerilmektedir. TensorFlow Lite, mobil ve gömülü cihazlar için optimize edilmiş bir derin öğrenme framework'üdür. Aşağıdaki avantajları sunar:

- **Yüksek Hız:** TensorFlow Lite, düşük gecikme süreleri ile hızlı çıkarım yapabilmektedir, bu da engel algılama gibi gerçek zamanlı sistemlerde önemli bir avantaj sağlar.
- **Mobil ve Gömülü Cihazlara Uygun:** TensorFlow Lite, donanım hızlandırma desteğiyle gömülü sistemlerde performansı artırmaktadır.
- **Kullanım Kolaylığı:** TensorFlow Lite, TensorFlow modellerini optimize ederek gömülü cihazlarda kolayca çalıştırılmasını sağlar.

Veri Seti Seçimi

Engel algılama sisteminin eğitimi ve test edilmesi için geniş kapsamlı veri setlerine ihtiyaç duyulmaktadır. Bu bağlamda, **COCO** ve **Pascal VOC** veri setleri, engel algılama için yaygın olarak kullanılmaktadır. Veri setlerinin özellikleri şu şekildedir:

- **COCO Veri Seti:** Nesne algılama, segmentasyon ve anahtar nokta algılama için geniş bir veri kümesidir. Engellerin ve yaygın nesnelerin tanınmasında yüksek doğruluk sağlar.
- **Pascal VOC Veri Seti:** Nesne algılama üzerine zengin bir veri seti olan Pascal VOC, çeşitli nesnelerle ilgili sınıflandırma ve tanımlama görevlerinde kullanılabilir.