

Test Projesi Raporu

**ÖZET**

|  |
| --- |
| Bu rapor, derin öğrenme modelleri kullanarak insan ve araba tespiti yapabilen bir uygulama geliştirme konusunda odaklanmaktadır. Nesne tespiti algoritmalarının çalışma süresi açısından pahalı olduğu bir gerçektir. Bu nedenle, bu projede nesne tespiti için farklı bir yaklaşım kullanılacaktır. İlk olarak, giriş imgesindeki nesneler segmente edilecek ve sonrasında segmentasyon sonucundaki blobların bounding boxları tespit edilecektir. Son olarak, bulunan bounding boxların sınıflandırılması ile hangi nesne tipinin tespit edildiği belirlenecektir. Bu yaklaşım, nesne tespiti problemini daha etkili ve hızlı hale getirebilir. |

**AMAÇ VE HEDEFLER**

|  |
| --- |
| Bu proje, insan ve araba tespiti yapabilen bir uygulamanın geliştirilmesi için segmentasyon ve sınıflandırma derin öğrenme modellerinin kullanılmasını hedeflemektedir. Geleneksel nesne tespiti algoritmalarının aksine, bu proje öncelikle giriş imgesindeki nesnelerin segmentasyonunu yaparak, sonrasında bu segmentlerin bounding boxlarının tespit edilmesini amaçlamaktadır. En son aşamada ise sınıflandırıcı yardımıyla bounding boxlar içindeki nesnelerin türlerinin belirlenmesi hedeflenmektedir. Bu yaklaşım, daha hızlı ve verimli bir nesne tespiti çözümü sunmayı amaçlamaktadır. |

**YÖNTEM VE UYGULAMALAR**

|  |
| --- |
| 1) Bir görüntü veri setini işlemek, diğer veri setleri için olduğu gibi ön işleme adımları gerektirir. İşleme adımları aşağıdaki gibi olabilir:   1. **Veri setinin yüklenmesi**: Görüntü veri setinizi yükleyin ve uygun kütüphaneleri ve araçları kullanarak veri setinin formatını doğrulayın. 2. **Veri setinin incelenmesi:** Veri setinizi inceleyerek, boyutunu, özelliklerini, veri tipini ve eksik verileri tespit edin. Veri setinde kaç adet veri var, kaç adet sınıf var, verilerin boyutu nedir, gibi soruların yanıtlarını arayın.   **Bu veri seti, 495 görüntüden oluşuyor ve "araba-insan" adlı bir etiketle birlikte verilmiş. Veri seti, eğitim, doğrulama ve test setleri olarak bölünmüştür. Eğitim seti, 480 görüntüden oluşurken, doğrulama seti 10 görüntü ve test seti 5 görüntü içerir.**  **Veri seti önişleme adımlarından "Auto-Orient" uygulanmıştır. Ayrıca, her eğitim örneği için üç çıktı vermek üzere "90° Rotate" ve "Bounding Box" gibi çeşitli artırma yöntemleri kullanılmıştır. Buna ek olarak, "Brightness" artırma işlemi de -25% ila +25% arasında değişmektedir.**  **Bu veri setinde "araba-insan" etiketi kullanılarak 495 görüntü toplanmıştır. Ancak, insan etiketi çok az sayıda görüntüde bulunmaktadır. Bu, insan etiketi ile ilgili modellerin eğitimi için yeterli veri olmadığı anlamına gelir. Bu durum, insan tespiti gibi belirli görevler için model eğitiminin zayıf kalmasına neden olabilir. Bu nedenle, veri seti kullanılarak gerçekleştirilecek bazı görevlerde insan etiketi ile ilgili sonuçların doğruluğu düşük olabilir. İnsan etiketi sayısının artırılması, model eğitimi için daha sağlam bir veri seti sağlayabilir ve sonuçların doğruluğunu artırabilir.**   1. **Veri normalizasyonu:** Görüntü veri setinizdeki piksel değerleri farklı ölçeklerde olabilir. Bu durumda, veri normalleştirme işlemi yapmanız gerekebilir. Piksel değerleri 0-255 arasında ölçülen bir görüntü veri seti, örneğin 0-1 arasına ölçeklenmelidir.   **Not:** Normalizasyon işlemi, veri setinizdeki görüntülerin piksel değerlerini aynı ölçekte tutarak daha iyi sonuçlar elde etmenize yardımcı olabilir. Ancak, bazı durumlarda normalizasyon işlemi, veri setinizdeki önemli özellikleri kaybetmenize neden olabilir. Bu nedenle, normalizasyon işlemi yapmadan önce veri setinizi dikkatli bir şekilde incelemeniz önerilir.   1. **Veri dönüştürme:** Bazı durumlarda, görüntülerin boyutları farklı olabilir veya görüntüler farklı formatta olabilir. Bu nedenle, görüntüleri aynı boyuta veya formata dönüştürmeniz gerekebilir. Bu işlem, özellikle bir derin öğrenme modeli kullanıyorsanız, modelin iyi performans göstermesi için önemlidir.   **Kitti veri seti genellikle 1242x375 veya 1241x376 boyutlarında görüntüler içerir. Bu boyutlar, LiDAR sensörleri ile elde edilen verilerin yatay ve dikey açılarının çözünürlüğüne uygun olarak seçilmiştir.**  **Segmentasyon işlemleri için, görüntü boyutunun sınırları, işlem yapmak istediğiniz alana bağlıdır. Örneğin, yüksek hassasiyetli detaylar için daha yüksek çözünürlüklü bir görüntü seçmek isteyebilirsiniz. Ancak, bu, eğitim süresini artırabilir ve modelinizi daha az kullanışlı hale getirebilir.**  **Geleneksel olarak, segmentasyon işlemleri için uygun boyutlar 512x512 veya 1024x1024 gibi kare boyutlarıdır. Bu boyutlar, segmentasyon işlemlerinde kullanılan birçok derin öğrenme modeli için standart olarak kullanılır.**   1. **Veri genişletme:** Görüntü veri setiniz yeterince büyük değilse, veri genişletme işlemi yapabilirsiniz. Veri genişletme, mevcut veri setindeki görüntülerin dönüştürülerek yeni görüntülerin oluşturulması işlemidir. Bu işlem, özellikle veri setinizdeki sınıflar arasında dengesizlik varsa, modelin performansını artırabilir. 2. **Veri bölme:** Veri bölme, veri setini eğitim, test ve doğrulama kümeleri olarak ayırmaktır. Eğitim kümesi, modelin eğitiminde kullanılırken, test kümesi ve doğrulama kümesi modelin performansını ölçmek için kullanılır.   2) Görüntü segmentasyonu, bir görüntüyü farklı bölümlere veya segmentlere ayırmak için kullanılan bir bilgisayar görüşü teknolojisidir. Bu segmentler, nesne tanıma, görüntü sıkıştırma, görüntü restorasyonu, robotik görüş, tıbbi görüntüleme vb. birçok uygulamada kullanılabilir.  Görüntü segmentasyonu için kullanılabilecek birçok model vardır, ancak en yaygın olarak kullanılanları şunlardır:   1. Piksel tabanlı segmentasyon: Görüntünün her pikselini bölümlere ayırmak için kullanılır. Bu model, her pikseli doğrudan belirli bir sınıfa atar. 2. Bölgesel segmentasyon: Bu model, benzer özelliklere sahip pikselleri birleştirerek bölümlere ayırır. Bölgesel segmentasyon, nesne tabanlı segmentasyonda da kullanılır. 3. Nesne tabanlı segmentasyon: Bu model, belirli nesneleri tanımak için kullanılır. İlk adımda nesneleri algılar ve ardından bu nesnelerin sınırlarını bulur. 4. Eşikleme: Bu model, bir görüntüyü farklı parçalara ayırmak için bir eşik değeri kullanır. Eşikleme, özellikle medikal görüntüleme gibi uygulamalarda kullanışlıdır.   Bir veri kümesini segmente etmek için hangi modelin kullanılacağı, veri kümesinin özelliklerine, uygulama amacına ve segmentasyonun amaçlarına bağlıdır. Veri kümesi büyükse, geniş bölümlere ayrılmak için bölgesel segmentasyon veya nesne tabanlı segmentasyon tercih edilebilir. Eşikleme daha küçük veri kümeleri için kullanışlıdır ve piksel tabanlı segmentasyon, belirli renklerin veya özelliklerin vurgulanması gerektiğinde kullanılabilir.  Burada, bazı popüler segmentasyon modelleri aşağıda listelenmiştir:   1. **U-Net:** Piksel tabanlı segmentasyon için popüler bir modeldir. Birçok medikal görüntüleme uygulamasında kullanılır. 2. **Mask R-CNN:** Nesne tabanlı segmentasyon için kullanılan bir modeldir. Görüntülerdeki nesnelerin sınırlarını belirlemek için kullanılır. 3. **Fully Convolutional Network (FCN):** Piksel tabanlı segmentasyon için kullanılan bir modeldir. Her pikseli belirli bir sınıfa atamak için kullanılır.   YOLOv8, bir nesne algılama ve sınıflandırma modelidir. YOLOv8 modeli, PyTorch Instance Segmentation kullanarak nesne tespiti ve sınıflandırmayı gerçekleştirir. YOLOv8, önceki sürümleri olan YOLOv5 ve YOLOv4'ten esinlenerek geliştirilmiştir.  YOLOv8 modelinin avantajları şunlardır:  Hız: YOLOv8 modeli, hızlı bir nesne algılama ve sınıflandırma modelidir. YOLOv8, GPU'ları etkili bir şekilde kullanarak gerçek zamanlı nesne tespiti ve sınıflandırma sağlar.  Yüksek doğruluk: YOLOv8 modeli, yüksek doğruluk sağlayan bir modeldir. YOLOv8, diğer nesne algılama ve sınıflandırma modellerinden daha yüksek bir doğruluk sağlar.  Çoklu sınıf destek: YOLOv8 modeli, çoklu sınıfları destekleyen bir modeldir. Bu, bir görüntüde farklı nesne sınıflarını algılamak için kullanılabilir.  Verimli bellek kullanımı: YOLOv8 modeli, verimli bellek kullanımı sağlar. Bu, yüksek boyutlu görüntüleri işlerken bellek kullanımını azaltır.  Ölçeklenebilirlik: YOLOv8 modeli, ölçeklenebilir bir modeldir. Bu, modelin farklı boyutlardaki görüntülerde kullanılabilmesini sağlar.  Kolay eğitim: YOLOv8 modeli, kolay eğitim sağlayan bir modeldir. Bu, modelin kolayca eğitilip kullanılabilmesini sağlar.  Sonuç olarak, YOLOv8 modeli, hızlı, yüksek doğruluğa sahip, çoklu sınıf desteği olan, verimli bellek kullanımı sağlayan, ölçeklenebilir ve kolay eğitim sağlayan bir modeldir. Bu özellikleri sayesinde, diğer nesne algılama ve sınıflandırma modellerine göre daha iyi performans gösterir.  Segmentasyon, bir görüntüdeki nesneleri piksel düzeyinde belirlemek ve ayrıntılı olarak tanımlamak için kullanılan bir makine öğrenimi tekniğidir. PyTorch Instance Segmentation kullanarak segmentasyon yapmak, özellikle görüntüdeki nesnelerin yerlerini ve sınıflarını tespit etmek için kullanılan bir yöntemdir.  PyTorch Instance Segmentation, iki aşamalı bir işlem kullanır: önceden eğitilmiş bir nesne algılama modeli kullanarak nesnelerin konumlarını tespit eder ve ardından nesnelerin kesin sınırlarını ve ayrıntılarını belirlemek için piksel seviyesinde bir segmentasyon işlemi uygular.  Segmentasyon, her bir pikselin bir nesneye ait olup olmadığını belirlemek için kullanılan bir piksel tabanlı bir sınıflandırma işlemidir. Bu işlem, her bir pikselin bir sınıfa ait olup olmadığını belirlemek için derin öğrenme algoritmaları kullanarak gerçekleştirilir. Sonuç olarak, her piksel belirli bir nesneye veya arkaplana ait olarak etiketlenir ve nesnelerin sınırları kesin bir şekilde belirlenir.  PyTorch Instance Segmentation, segmentasyon işlemini gerçekleştirmek için Convolutional Neural Networks (CNN) tabanlı birçok farklı model kullanabilir. Bu modeller, önceden eğitilmiş model ağırlıkları kullanılarak transfer öğrenme yöntemiyle özelleştirilebilir. Bu, modeli daha spesifik bir uygulama için eğitmek için gerekli veri sayısını azaltabilir ve eğitim süresini önemli ölçüde kısaltabilir. |

**BELİRTMEK İSTEDİĞİNİZ DİĞER KONULAR**

|  |
| --- |
| ***Bu rapor projenin giriş kısmını özetlemiştir. Projenin detayları ile ilgili github da açıklamalarda bulunulmuştur. Ama yapılan calısamların tam anlaşılması için yeterli değildir. Bireysel yapılan bir proje olduğu için yoruma cok acık bir proje haline geldi benim için umut ediyorum ki bir sonraki aşamada bunları anlatmak isterim.*** |

**7. EKLER**

**KAYNAKLAR**

[1]

[2]

[3]

[4]

[5]