**PROJE BAŞLIĞI:** KİŞİ SAYMA VE İZLEME SİSTEMİ

1. **GİRİŞ**

Günümüzde yapay zeka ve özellikle derin öğrenme, bilgisayar bilimleri ve yapay zeka alanında devrim niteliğinde ilerlemeler sağlamıştır. Yapay zeka, insan benzeri zekaya sahip sistemlerin tasarımı ve geliştirilmesi için kullanılan bir disiplindir. Bu alandaki en önemli kavramlardan biri de derin öğrenme olarak karşımıza çıkar. Derin öğrenme, büyük veri setlerinden öğrenme yeteneğine sahip yapay sinir ağları üzerine odaklanan bir alt dalıdır.

Derin öğrenme, yapay sinir ağlarının bir türüdür ve insan beyninin işleyişini taklit ederek karmaşık desenleri ve ilişkileri öğrenme yeteneğine sahiptir. Bu teknoloji, günümüzde nesne tanıma, görüntü sınıflandırma, konuşma tanıma, oyun stratejileri geliştirme gibi birçok alanda kullanılmaktadır.

Yapay zeka ve derin öğrenme, tarihleri boyunca çeşitli kilometre taşları ve gelişmelerle şekillenmiştir. Alanın öncüleri arasında bulunan Viola-Jones algoritması, görüntü tabanlı nesne algılama için temel bir adım olmuştur. Ancak, gerçek çığır açıcı gelişmeler, 2010'ların başlarından itibaren derin öğrenmeyle geldi. Hinton, LeCun, ve Bengio gibi araştırmacıların çalışmaları, derin sinir ağlarının yeniden popülerlik kazanmasını sağladı ve bu da görüntü işleme, doğal dil işleme ve daha birçok alanda devrim yarattı.

Derin öğrenmenin popüler hale gelmesi, büyük veri setlerinin ve güçlü işlemcilerin de desteğiyle gerçekleşti. Google'ın 2012'deki ImageNet yarışmasında AlexNet adlı derin öğrenme modeliyle büyük bir başarı elde etmesi, derin öğrenmenin etkisini daha da artırdı. Daha sonra, VGG, GoogLeNet, ResNet gibi derin ağlar geliştirildi ve nesne tanıma alanında olağanüstü sonuçlar elde edildi.

Bu çalışma, derin öğrenmenin sağladığı bu gelişmeleri nesne algılama ve takip uygulamalarına entegre ederek, gerçek zamanlı veri analizi ve nesne izleme konusundaki potansiyelini vurgulamaktadır. Ultralytics, OpenCV ve cvzone gibi güçlü kütüphaneler ve YOLO modeli gibi derin öğrenme algoritmaları, bu çalışmanın temelini oluşturarak nesne algılama ve takibini gerçekleştirmek için kullanılmıştır.

1. **LİTERATÜR**

1-Fırat Üniversitesi, Yazılım Mühendisliği Bölümü ve Trakya Üniversitesi, Bilgisayar Teknolojileri Bölümleriden Resul DAŞ , Berna POLAT ve Gürkan TUNA 2019 yılında Derin Öğrenme ile Resim ve Videolarda Nesnelerin Tanınması ve Takibi adında çalışma gerçekleştirmişlerdir. Yapılan bu çalışmada Google’ın açık kaynak kodlu makine öğrenmesi kütüphanesi olan TensorFlow kullanılmıştır. Ayrıca nesne takibi için Region Based Convolutional Networks kütüphanesinden Faster R-CNN modeli ele alınmıştır. Yapılan çalışma sonucunda bu kütüphaneler ile durağan görüntüler, video görüntüleri ve webcam görüntüleri üzerinde nesne tanıma işlemi gerçekleştirilmiş.

2-Fatih Bayram 2020 yılında Derin öğrenme tabanlı otomatik plaka tanıma adlı çalışma gerçekleştirmiştir ve Politeknik dergisinde yayınlanmıştır . Bu çalışma plaka tanıma sistemlerinin geliştirilmesinde üç temel adım üzerinde durmuştur. İlk olarak, 430 araç görüntüsünden oluşan bir veri seti manuel olarak etiketlenmiş ve Türk plaka sistemi için geçerli olmayan bazı özel karakterlerin filtrelenmesi yapılmıştır. Çalışmada, maskeli bölgesel evrişimsel sinir ağı kullanılarak derin öğrenme temelli bir plaka tanıma modeli tasarlanmış ve eğitilmiştir. Bu model, nesne konumlarını dikdörtgen koordinatlarından çok, sınıflara ve çokgen koordinatlara dayalı olarak belirlemek üzere etiketlenmiş plakaları kullanmıştır. Google Colab bulut sistemi üzerinde Python 3, Keras ve TensorFlow kullanılarak geliştirilen bu model, 130 araç görüntüsü üzerinde %98.46 plaka tanıma başarısı elde etmiştir. Ayrıca, bu işlem ortalama 0.4 saniye gibi kısa bir sürede gerçekleştirilebilmiştir. Bu çalışma, otomatik plaka tanıma sistemlerinde manuel etiketleme süreci, maskeli bölgesel evrişimsel sinir ağı kullanımı ve yüksek başarı oranıyla dikkat çekmektedir. Derin öğrenme tabanlı bu model, plaka tanıma işlemlerinde hız ve doğruluk açısından önemli bir potansiyel sunmaktadır.

3-Fırat Üniversitesinde Suat TORAMAN tarafından yapılan projenin adı Derin Öğrenme ile İnsansız Hava Aracı Görüntülerinden Yaya Tespitidir. Çalışma, Stanford Üniversitesi'nden alınan drone video görüntülerinden yola çıkarak özellikle yaya, bisikletli, araba, ağaç ve sokak lambalarını içeren bir veri seti oluşturmayı amaçlamıştır. Toplam 2040 resimlik veri seti, özellikle yaya tespiti üzerine odaklanarak derin öğrenme modelleri AlexNet ve VGG16 ile sınıflandırılmıştır. Bu modeller, farklı katman yapıları ve özellikler ile öne çıkmaktadır. Ardından, sınıflandırma için Destek Vektör Makinesi algoritması (DVM) kullanılmış ve bu algoritmanın farklı parametreleri incelenmiştir. MATLAB kullanılarak gerçekleştirilen çalışma, İHA'lardan elde edilen görüntüler üzerinde yapılmış ve nesne tanıma odaklı olarak ele alınmıştır.

4-Süleyman Demirel Üniversitesi Endüstri Mühendisliği bölümünde Kadir Hacıfazlıoğlu ve Erdal Aydemirin yaptığı Görüntü İşleme ile Kalite Kontrol Hatalarının Tespit Edilmesi adlı çalışmada otomotiv parçaları üzerindeki hataların tespiti için YOLOv5 derin öğrenme algoritması kullanılarak bir görüntü işleme modeli geliştirilmiştir. OpenCv , Torch , Numpy , Pandas kütüphanelerini kullanmıştır. Panasonic Lumix DMC-TZ80 kamera ile yaklaşık 2000 görsel veri toplanmış ve bu verilerin %80'i eğitim, %20'si ise test için ayrılmıştır. Eğitim için makeSense.ai gibi platformlar üzerinden elde edilen etiketlerle YOLOv5 algoritması kullanılarak modelin eğitimi gerçekleştirilmiştir. Eğitim süreci yaklaşık 15 saat sürmüştür.

Derin öğrenme modellerinde başarılı sonuçlar elde edebilmek için genellikle büyük ve çeşitli veri setlerine ihtiyaç duyulmaktadır. Bu çalışma kapsamında, kusurlu bölgelerin etiketlenmesi için çeşitli görsel ve video verileri kullanılmıştır. Eğitim sonucunda elde edilen model, Precision (P), Recall (R), ve mAP (mean Average Precision) gibi metriklerle değerlendirilmiştir.

Çalışmanın sonuçları, kusurlu ürünlerin tespit edilmesinde ortalama bir doğruluk sağlandığını göstermektedir. Ancak, modelin performansını artırmak adına daha fazla veri seti kullanımı, hata çeşitliliğinin detaylandırılması, eğitim sürecinde değişiklikler yapılması ve farklı veri artırım tekniklerinin uygulanması gibi geliştirmelerin projenin devamında yapılması öngörülmektedir. Bu adımların, modelin doğruluğunu ve genel performansını artırabileceği düşünülmektedir.

**3-TİCARİLEŞMİŞ LİTERATÜR**

**Waymo**: Waymo, otonom araçlar konusunda uzmanlaşmış bir şirkettir. Gerçek zamanlı olarak yayalar, bisikletliler ve diğer araçlar gibi nesneleri tespit etmek için YOLO'yu kullanıyorlar. Sistemleri halka açık yollarda test edilmiş ve engelleri tespit etme ve bunlardan kaçınma konusunda etkileyici sonuçlar göstermiştir.

**Tesla**: Tesla, yakın zamanda otonom araçlar üreten tanınmış bir elektrikli otomobil üreticisidir. Nesne tespiti için YOLO da dahil olmak üzere sensörler ve derin öğrenme algoritmalarının bir kombinasyonunu kullanmaktadırlar. Sistemleri halka açık yollarda test edilmiş ve nesneleri tespit etme ve hareketlerini tahmin etme konusunda umut verici sonuçlar göstermiştir.

**Uber**: Uber, otonom araçlara büyük yatırım yapan bir araç paylaşım şirketidir. Nesne algılama için YOLO'yu kullanıyorlar ve sistemleri dünya çapındaki şehirlerde kapsamlı bir şekilde test edildi. Elde ettikleri sonuçlar, YOLO'nun kalabalık kentsel ortamlarda bile nesneleri etkili bir şekilde algıladığını ve hareketlerini tahmin ettiğini göstermiştir.

YOLO, perakende sektöründe nesne algılama için etkili bir şekilde kullanılan bir derin öğrenme algoritmasıdır. Perakende sektöründe nesne algılama, ürünleri, müşterileri ve çalışanları bulmak için kullanılır. Bu bilgiler daha sonra stokları takip etmek, müşterilerin nasıl davrandığını incelemek ve mağazaların kurulum şeklini iyileştirmek için kullanılır. Nesne algılama, hırsızlık veya mağaza soygunu gibi sahtekarlıkları da tespit edebilir.

**Walmart**: Walmart dünyanın en büyük perakendecilerinden biridir ve mağazalarında nesne tespiti için YOLO'yu kullanmaktadır. YOLO'yu envanteri takip etmek ve müşteri davranışlarını analiz etmek için kullanıyorlar. YOLO, ürünleri etkili bir şekilde tanımladı ve mağaza boyunca hareketlerini izledi.

**Amazon**: Amazon, son zamanlarda fiziksel mağazalarla denemeler yapan bir çevrimiçi perakendecidir. Envanteri takip etmek ve müşteri davranışını analiz etmek için nesne algılamada YOLO'yu kullanıyorlar. YOLO, ürünleri etkili bir şekilde tanımlar ve mağaza boyunca hareketlerini izler.

**Zara**: Zara, mağazalarında nesne algılama için YOLO kullanan bir moda perakendecisidir. Müşteri davranışlarını analiz etmek ve mağaza düzenlerini iyileştirmek için YOLO'yu kullanıyorlar. YOLO, müşterileri etkili bir şekilde tanımladı ve mağaza boyunca hareketlerini izledi.

**Kullanılan Teknolojiler ve Araçlar:**

* **Programlama Dili:** Python
* **Kütüphaneler:** Ultralytics, OpenCV, cvzone
* **Model:** YOLO (You Only Look Once) nesne algılama modeli
* **Algoritmalar:** SORT (Simple Online and Realtime Tracking)
* **IDE:** Visual Studio Code
* **Dosyalar ve Kaynaklar:** Video dosyası, önceden eğitilmiş ağırlıklar (Yolo-Weights), mask.png gibi görsel kaynaklar
* **Video Kaynağı:** Belirli bir alanda ya da ortamda çekilmiş bir video kaynağı (örneğin, bir güvenlik kamerasından gelen video).
* **Belirli Bölgelerin Koordinatları:** İzleme veya sayım yapılacak bölgelerin koordinatları.

**Amaç:**

Bu projenin temel amacı, bir video dosyası üzerinde YOLO modeli ile nesne algılama yaparak insanları tespit etmek ve ardından bu tespit edilen insanları izlemek, takip etmek ve belirli bölgelerde geçişlerini saymak.

**Gerçekleştirilecek Adımlar:**

* **Video Girişi ve İşleme:** Projede kullanılacak video kaynağı belirlenir ve bu video kaynağı Python kullanılarak yüklenir. OpenCV kütüphanesi kullanılarak her bir kare alınır ve işlenir.
* **YOLO Modeli ile Nesne Algılama:** Video kareleri üzerinde YOLO modeli kullanılarak nesne algılama gerçekleştirilir. Bu adımda insanları temsil eden nesnelerin tespit edilmesi ve konumlarının belirlenmesi amaçlanır.
* **SORT Algoritması ile Takip:** Algılanan insanların takibi için SORT (Simple Online and Realtime Tracking) algoritması kullanılır. Her bir insan için bir kimlik (ID) atanır ve hareketleri izlenir.
* **Belirli Bölgelerdeki Geçişlerin İzlenmesi:** Projenin belirli amaçları doğrultusunda, belirlenmiş bölgelerdeki geçişler takip edilir ve sayılır. Örneğin, belirli bir çizgi veya alanın geçilmesi durumunda bu geçişlerin sayımı yapılır.
* **Görsel İşaretler ve Bilgilendirme:** Takip edilen insanların etrafına dikdörtgenler çizilir, takip ID'leri üzerlerine yazılır ve belirli geçiş bölgeleri görsel olarak işaretlenir. Bu sayım ve izleme süreci, video üzerinde görsel olarak temsil edilir.
* **Sonuçların Analizi ve Sunumu:** Proje sonuçları, belirlenen geçiş sayıları veya izlenen hareketler gibi verilerle sunulur. Bu veriler, proje amacına göre analiz edilir ve görsel veya sayısal olarak sunulur.

**MATERYAL VE YÖNTEM**

**Kullanılan IDE**

Visual Studio Code (VS Code), Microsoft tarafından geliştirilen ve çok popüler bir açık kaynaklı metin düzenleyicisi ve entegre geliştirme ortamıdır (IDE). Çeşitli programlama dilleri için destek sunar ve geniş bir kullanıcı tabanına sahiptir.

VS Code, kullanıcı dostu arayüzü, zengin özellik seti ve genişletilebilirliği ile dikkat çeker.

VS Code, web geliştirme, Python, Java, C#, JavaScript, TypeScript, ve birçok farklı programlama dili için yaygın olarak kullanılır. Eklenti deposu geniş bir seçenek yelpazesi sunar ve bu da farklı geliştirme ortamlarına ve gereksinimlere uygun olarak özelleştirilebilirlik sağlar. Hem yeni başlayanlar hem de deneyimli geliştiriciler için ideal bir geliştirme ortamı olabilir.

**Kullanılan Programlama Dili**

**PYTHON**

Python, kullanımı kolay ve genel amaçlı bir programlama dilidir. Basit ve anlaşılır sözdizimi sayesinde özellikle yeni başlayanlar için idealdir. Yüksek seviyeli bir dil olması, kodunun insanlar tarafından kolayca okunabilir ve anlaşılabilir olmasını sağlar.

Python, geniş bir kütüphane ve modül ekosistemiyle birlikte gelir. Bu sayede çeşitli alanlarda kullanılan pek çok işlevsellik, veri yapıları ve araçlar hazır olarak sunulur. Veri analizi, yapay zeka, web geliştirme, bilgisayarla görme, ağ programlama, oyun geliştirme gibi birçok alanda kullanılır.

Ayrıca, açık kaynaklı olması ve geniş bir topluluğa sahip olması, Python'un sürekli olarak gelişmesini sağlar. Bu da yeni kütüphanelerin ve araçların sürekli olarak ortaya çıkmasına ve dilin güncel kalmasına olanak tanır.

Python, taşınabilirliği ve çeşitli platformlarda (Windows, macOS, Linux) çalışabilme özelliğiyle de dikkat çeker. Yüksek seviyeli dil olması sayesinde, kodun hızlı bir şekilde yazılmasını ve işlevsel olmasını sağlar.

Genel olarak, Python'un basitliği, okunabilirliği, geniş ekosistemi ve çok yönlülüğü, birçok programcı, veri bilimcisi, mühendis ve araştırmacı için tercih edilen bir dil olmasını sağlar.

**Kullanılan Model**

**YOLO**

YOLO, "You Only Look Once" kısaltmasıyla bilinen bir nesne algılama (object detection) algoritmasıdır. Bu algoritma, nesneleri gerçek zamanlı olarak algılamak ve sınıflandırmak için kullanılır. YOLO, bir görüntüyü tek bir geçişle analiz eder ve görüntü içindeki nesnelerin konumlarını ve sınıflarını tahmin etmek için bir derin öğrenme ağı kullanır.

YOLO'nun ana avantajlarından biri hızıdır. Geleneksel nesne algılama yöntemlerine göre daha hızlı çalışır, çünkü bir görüntüyü birden çok kez işlemek yerine, tek bir geçişte nesneleri algılar. Ayrıca, YOLO nesnelerin konumlarını ve sınıflarını aynı anda tahmin ettiği için oldukça hızlıdır.

**Kullanılan Algoritma**

**SORT**

Sort, genellikle sıralama algoritmaları anlamına gelir. Sıralama algoritmaları, bir diziyi veya listeyi belirli bir düzene göre yeniden düzenleyen algoritmalardır. Bu düzenleme genellikle sayısal veya alfabetik sıraya göre olur.

Bir sıralama algoritması, bir dizi içindeki elemanları küçükten büyüğe veya büyükten küçüğe doğru sıralayabilir. Sıralama algoritmaları, veri analizi, veritabanları, bilgi teknolojisi gibi birçok alanda kullanılır.

**Kullanılan Kütüphaneler**

YOLO derin öğrenme modeli ile eğitimi gerçekleştirilen görsel veri seti Python programlama dilinde yazılımı gerçekleştirilerek görsel sonuç çıktılarına ulaşılmıştır. Bu sonuçların elde edilmesi için Ultralytics , OpenCV ve Cvzone kütüphaneleri kullanılmıştır.

**Ultraytics**

Derin öğrenme ve yapay zeka alanında çalışan bir yazılım şirketidir. YOLOv5 gibi nesne algılama ve sınıflandırma gibi alanlarda PyTorch tabanlı modeller geliştirirler. Kullanıcı dostu araçlar sunarlar ve derin öğrenme modellerini daha erişilebilir hale getirmeyi amaçlarlar

**OpenCv**

OpenCV, Intel tarafından 1999 yılında geliştirilen açık kaynak kodlu görüntü işleme kütüphanesidir. Bu kütüphane ile görsel veri setlerinin çekilerek, tespit edilmesi istenilen özelliklerin tespit edilmesi, analiz edilmesine yardımcı olan bir kütüphanedir. Görüntülerden nihai sonuç elde edilmeye çalışılır.

**Cvzone**

OpenCV'yi daha verimli bir şekilde kullanmak için geliştirilmiş bir yardımcı kütüphanedir. OpenCV üzerine inşa edilmiştir ve görüntü işleme projelerini daha kolay ve hızlı bir şekilde geliştirmek için ek işlevsellikler ve kolaylıklar sunar. Özellikle, kutular içinde metin veya nesne tespiti, el izleme gibi çeşitli özellikler sağlar.

**Yapılan Benzer Projeler:**

**Proje Adı :** Görüntü İşleme ile Kalite Kontrol Hatalarının Tespit Edilmesi

Üretim sistemlerinin temel amaçlarından biri olduğu gibi otomotiv parçaları üretim sektöründe üretilen ürünler, kusursuz bir şekilde üretilmeli ve olası kusurlu ürünlerin üretimin bir sonraki üretim hattına geçmemesi gerekmektedir. Üretim kalite kontrol süreci genellikle insan faktörü tarafından gerçekleştirilmektedir. Gelişen teknolojiler ile makine öğrenmesi ve görüntü işleme teknikleri gibi sistemler ile olası kusurların tespiti için derin öğrenme modelleri geliştirilebilmektedir. Bu teknolojilerin yardımı ile kalite kontrol birimlerinde çalışanlara yönelik yardımcı sistemler geliştirilmesi mümkün gözükmektedir. Bu çalışma, otomotiv parçaları üzerinde oluşan hataların tespiti için YOLOv5 algoritması yardımıyla yüzeyde oluşan kusurların tespiti üzerine gerçekleştirilmiştir. Ürünler üzerinde oluşan hatalar Panasonic Lumix DMC-TZ80 kamerası ile fotoğraflanarak bir görsel veri seti oluşturulmuştur. Bu veri seti çeşitli görsel veri artırım yaklaşımlarıyla YOLOv5 ile eğitilmiştir. YOLOv5 ile gerçekleştirilen derin öğrenmenin sonucunda OpenCv kütüphanesi kullanılarak PyCharm uygulamasında test işlemleri gerçekleştirilmiştir. Sonuçta, tanımlanan ürün kalite kontrol hataları görüntü işleme ile tespit edilmeye çalışılmıştır