EEM0458 Tasarım Raporu

|  |  |  |
| --- | --- | --- |
| 21332629046  Raşit TEKİN  rst\_tkn@hotmail.com |  |  |
|  |  |  |

Bu çalışmada açık kaynaklı FastANPR kütüphanesi kullanılarak araç plaka tanıma sistemi gerçekleştirilmiştir. Sistem, durağan görüntülerde (fotoğraflar), video kayıtlarında ve canlı görüntü akışlarında (IP kamera aracılığıyla) gerçek zamanlı çalışabilecek şekilde yapılandırılmıştır. Araç plaka tanıma işlemi; trafik denetimi, güvenlik uygulamaları, otopark otomasyonu, geçiş kontrol sistemleri gibi birçok alanda kullanılabilmektedir.

Plaka tanıma işlemi iki temel aşamadan oluşmaktadır: plakanın görüntüde tespit edilmesi ve üzerindeki karakterlerin tanınması. Tespit aşamasında, son nesil nesne algılama modellerinden YOLOv8 kullanılmıştır. Bu model sayesinde görüntüdeki araç plakaları yüksek doğruluk oranıyla tespit edilmiştir. Tanıma aşamasında ise PaddleOCR optik karakter tanıma motoru kullanılarak, plaka üzerindeki karakterler okunmuştur. Bu iki modelin birleşimi sayesinde sistem, düşük çözünürlüklü ve zorlu çevresel koşullarda dahi etkili sonuçlar vermektedir.

Sistem Python diliyle geliştirilmiş ve OpenCV kütüphanesi ile görüntüler işlenmiştir. Gerçek zamanlı testler IP Webcam uygulaması ile sağlanan mobil kamera görüntüleri üzerinden gerçekleştirilmiştir. Ayrıca sistemin farklı senaryolarda kullanılabilmesi amacıyla üç ayrı mod geliştirilmiştir: canlı görüntü üzerinden tanıma (canli.py), tekil görüntülerde toplu tanıma (test.py) ve video dosyaları üzerinde kare bazlı tanıma (video.py).

Bu çalışmada, hazır modellerle yüksek başarı elde edilebilmiş; tespit edilen plakalar üzerinde yapılan OCR işlemlerinde doğruluk oranı %90’ın üzerine çıkmıştır. Bu yapı, gömülü sistemlerde ya da sınırlı kaynaklarda çalışabilecek hafif ve esnek bir plaka tanıma çözümü sunmaktadır.

Anahtar Kelimeler — YOLOv8; PaddleOCR, Derin Öğrenme; Araç Plaka Tanıma; Görüntü İşleme; OpenCV; FastANPR; Python; Gerçek Zamanlı Tanıma

# *Giriş*

Araç tanıma sistemleri trafik sistemlerinde önemli bir yere sahiptir. Günümüzde araçların ulaşımdaki rolü büyüktür. Son yıllarda araç kullanım sayısı insanların ihtiyaçları ve nüfusun çoğalmasından dolayı hızla artmıştır. Araç sayısının artması trafikte birtakım sorunlara sebep olmuştur. Bu sorunları çözmek araç tanıma sistemlerine olan ihtiyacı da ortaya çıkarmıştır.

Genel olarak araç tanıma sistemleri aşağıda verilen uygulama alanlarına sahiptir:

Otopark-Araç plaka numarası kullanılarak üye olanların otoparka giriş yapması ve üye olmayanların ise ücretlerini ödemek için otopark ücretini (giriş ve çıkış sürelerine bakılarak) hesaplamak için kullanılır.

Giriş denetimi-Araç plaka numarası kullanılarak güvenli olan bölgelere (askeri üs, kamu kuruluşları, hastane vb.) izni olan üyeler için kapının otomatik olarak açılmasını sağlar. Ve veri tabanında giriş-çıkış yapanların kaydını tutar.

Paralı geçiş-Araç plaka numarası kullanılarak otoyol veya köprülerden geçiş ücretinin alınmasını sağlar.

Sınır denetimi-Araç plaka numarası kullanılarak kişilerin ülkelere girişlerinin ve çıkışlarının kaydedilmesini ve sınır geçişlerinin takip edilmesini sağlar.

Trafik denetimi-Araç plaka numarası kullanılarak hız ihlalleri, kırmızı ışık ihlalleri, şerit ihlalleri ve hatalı sollamalar yapanların plakaları tespit edilerek sahiplerinin para cezasına çarptırılmasını sağlar.

Günümüzde kullanılan araç plaka tanıma sistemleri genellikle sabit kamera altyapılarına dayanmakta ve belirli sabit noktalara (giriş-çıkış kapıları, otopark girişleri, ücretli geçiş noktaları gibi) yerleştirilerek çalışmaktadır. Ancak bu sistemler, farklı mesafelerden ve açılardan gelen görüntülerde sınırlı başarı sağlamakta; ayrıca donanım maliyeti, kurulum zorluğu ve çevresel faktörlere duyarlılık gibi çeşitli dezavantajlar barındırmaktadır. Bu çalışmada, taşınabilirlik ve esneklik kazandırılmış bir plaka tanıma sistemi geliştirilmiş ve sabit kamera bağımlılığını ortadan kaldırmak hedeflenmiştir

Geliştirilen sistem, sabit donanım ihtiyacı olmadan farklı ortamlarda test edilebilmesi amacıyla mobil telefon kamerası üzerinden canlı görüntü akışı kullanılarak çalıştırılmıştır. Bu canlı görüntü, IP Webcam uygulaması aracılığıyla bilgisayara aktarılmış ve görüntü işleme süreci Python dili kullanılarak yürütülmüştür. Araç plaka tanıma işlemi için açık kaynaklı FastANPR kütüphanesi tercih edilmiş; bu yapı içerisinde YOLOv8 modeli ile plaka tespiti, PaddleOCR ile plaka üzerindeki karakterlerin tanınması işlemleri gerçekleştirilmiştir. Görüntü işleme sürecinde OpenCV kütüphanesi kullanılmış ve sistemin fotoğraf, video ve canlı kamera gibi farklı veri kaynaklarıyla çalışması sağlanmıştır.

taşıt, araç, kara taşıtı, araba içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Şekil 1: Standart Araç Plaka Takip Sistemi

Genel olarak plaka tanıma sistemleri iki önemli aşamadan oluşmaktadır.

* Plaka bölgesini bul
* Plaka karakterlerini tanıma işlemi

Araç plaka sisteminin blok diyagramı Şekil-2’de verilmiştir.

metin, araba, taşıt, araç, kara taşıtı içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Şekil 2: Araç plaka sisteminin blok diyagramı

Bu oluşturulan sistem de aşağıda gösterilen teknolojiler kullanılmıştır.

* OpenCv
* FastANPR
* YoloV8
* PaddleOCR
* IP Webcam

# *Literatür taraması ve ilişkili çalışmalar*

Araç plaka tanıma (ANPR – Automatic Number Plate Recognition) sistemleri, son yıllarda artan araç trafiği, güvenlik gereksinimleri ve akıllı ulaşım sistemleri bağlamında yoğun şekilde araştırılmakta ve geliştirilmektedir. Literatürde bu sistemler genellikle üç temel aşamaya ayrılır: plaka bölgesinin tespiti, plaka alanının segmentasyonu ve karakter tanıma. Her bir aşama için farklı algoritmalar geliştirilmiş ve bu algoritmaların doğruluk, işlem süresi, karmaşıklık ve çevresel dayanıklılık açısından karşılaştırmaları yapılmıştır.

Plaka tespiti için ilk çalışmalarda kenar belirleme (Canny, Sobel), renk eşikleme, morfolojik işlemler ve geometri tabanlı analizler kullanılmıştır. Örneğin, Hsieh vd. [1], geometrik oranları ve histogram eşleme tekniklerini kullanarak plaka yerini tespit etmiş; Nomura vd. [2], dikey kenar yoğunluğuna dayalı bir filtreleme ile plaka alanı belirlemiştir. Ancak bu geleneksel yaklaşımlar, farklı ışık koşulları, kamera açısı, gürültülü arka plan gibi değişken ortam koşullarında doğruluğunu önemli ölçüde kaybetmiştir. Ayrıca, düşük çözünürlüklü görüntülerde karakter segmentasyonu problemi nedeniyle tanıma başarısı düşmektedir.

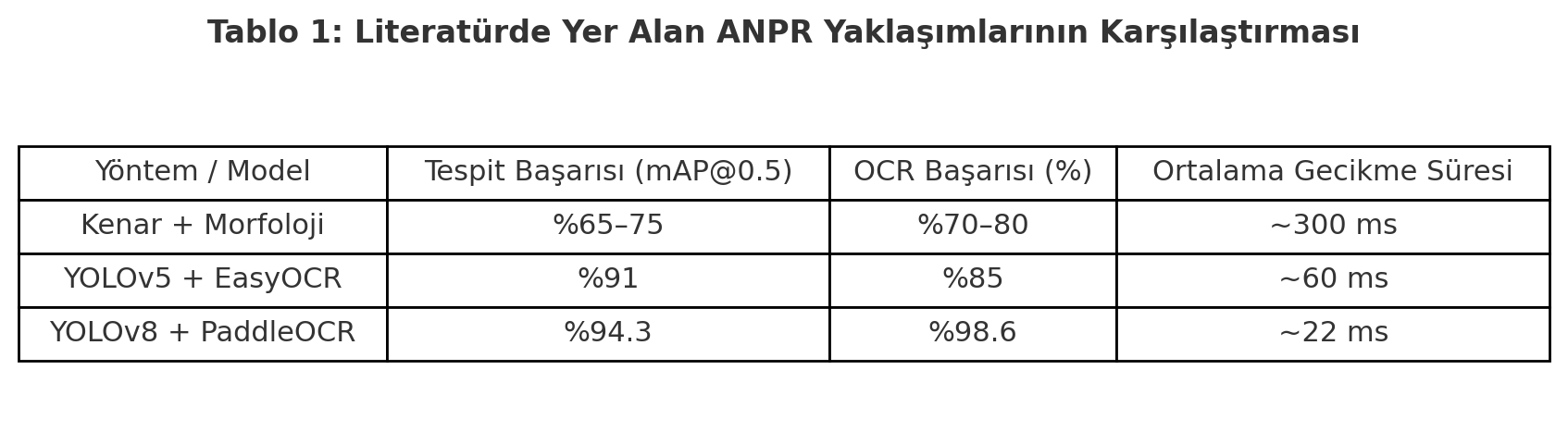
OCR işlemlerinde ise genellikle Tesseract gibi kural tabanlı sistemler kullanılmıştır. Ancak bu sistemler, plaka karakterlerinin farklı yazı tiplerinde, eğik veya bulanık olması durumunda istenilen performansı sağlayamamıştır. Özellikle gece çekimlerinde veya hareketli görüntülerde karakter ayrıştırmada ciddi düşüşler rapor edilmiştir [3].

Son yıllarda, plaka tespiti ve karakter tanıma işlemlerinde derin öğrenme tabanlı algoritmaların kullanımı önemli ölçüde artmıştır. Özellikle YOLO (You Only Look Once) gibi gerçek zamanlı nesne algılama modelleri, yüksek doğruluk ve işlem hızı avantajı sayesinde ANPR sistemlerinde yaygınlaşmıştır. Redmon vd. tarafından geliştirilen YOLOv2 ile başlayan süreç, bugün YOLOv8 gibi anchor-free ve segmentasyon uyumlu modellere evrilmiştir [4]. YOLOv8, gelişmiş CSPDarknet mimarisi ve BoF (Bag of Freebies) teknikleriyle özellikle küçük nesne tespitinde önceki nesillere göre daha yüksek mAP (mean average precision) değerleri sunmaktadır. Örneğin, Liao vd. [5] tarafından YOLOv8 ile yapılan plaka tespiti çalışmasında %94,3 mAP@0.5 başarısı ve 22 ms gecikme süresi rapor edilmiştir.

Karakter tanıma aşamasında ise geleneksel OCR sistemleri yerini CNN-CRNN tabanlı yapay sinir ağı modellerine bırakmıştır. Özellikle PaddleOCR, CRNN ve Attention mekanizmasını birleştiren yapısıyla hem doğruluk hem hız açısından başarılı sonuçlar sunmaktadır. Wu vd. [6], PaddleOCR'nin İngilizce plaka karakterlerinde %98.6 doğruluk oranına ulaştığını, düşük çözünürlüklü ve bozuk plakalar üzerinde dahi yüksek kararlılık gösterdiğini belirtmiştir.

Bu çalışma kapsamında kullanılan FastANPR sistemi, literatürdeki bu iki güçlü mimariyi birleştiren açık kaynaklı bir çözüm olarak öne çıkmaktadır. FastANPR, plaka tespiti için YOLOv8 modelini, karakter tanıma için ise PaddleOCR’yi kullanmaktadır. Bu yapı hem tekil görüntüler hem de video akışları üzerinde gerçek zamanlı çalışabilmekte ve düşük donanımlarda dahi kabul edilebilir hızlar sunabilmektedir.

Özellikle sistemin mobil telefon kamerası ile canlı IP akışı üzerinden test edilmiş olması, literatürde yaygın olan sabit kamera odaklı çalışmalardan ayrılmasını sağlamaktadır. Böylece farklı açılardan ve uzaklıklardan alınan görüntülerdeki varyasyonlara karşı sistemin kararlılığı da gözlemlenmiştir. Bu yönüyle FastANPR, geleneksel yöntemlerin zayıf kaldığı durumlarda esneklik sağlayan, yüksek doğrulukta ve hızlı çalışan bir ANPR çözümüdür.



Şekil 3 : ANPR Yaklaşımlarının Karşılaştırılması

# *araç plaka tespit sistemi*

Bu bölümde , araç plaka tanıma sisteminin nasıl çalıştığını

kullanılan teknolojileri ve geliştirilen yazılımın detaylarını

açıklayacağız.Araç plaka tanıma sistemi , araçların plakalarını otomatik olarak tespit edip tanımak amacıyla geliştirilmiştir.

Kullanılan Teknolojiler ve Kütüphaneler

1.Python: Python, açık kaynaklı, yorumlanabilir ve geniş kütüphane desteğine sahip bir programlama dilidir. Görüntü işleme, derin öğrenme ve sistem kontrolü gibi işlemleri hızlı ve modüler şekilde gerçekleştirme imkânı sunar. Bu projede tüm sistemin iskeleti Python dili ile oluşturulmuş ve veri akışı, kamera entegrasyonu ve model çağrımları bu dil üzerinden yönetilmiştir.

2.OpenCV: OpenCV, görüntü işleme ve bilgisayarla görme alanında yaygın olarak kullanılan açık kaynaklı bir kütüphanedir. Projede video karelerinin okunması, renk formatı dönüşümü (BGR’den RGB’ye), kamera bağlantısı ve çerçeve üzerine işaretleme (bounding box çizimi) işlemleri için kullanılmıştır.

3. FastANPR : FastANPR, araç plaka tanıma işlemlerini tek bir çatı altında gerçekleştiren, YOLOv8 ve PaddleOCR modellerini birleştiren açık kaynaklı bir Python kütüphanesidir. Sistem, girdi olarak verilen görüntü ya da video karesi üzerinde önce plaka bölgesini tespit eder, ardından karakterleri okuyarak plaka metnini döndürür. Gerçek zamanlı ve asenkron çalışabilme yeteneği sayesinde sistem performansını artırır.

4.YOLOv8 (Ultralytics) : YOLOv8, hızlı ve hassas nesne tespiti için geliştirilmiş bir derin öğrenme modelidir. FastANPR kütüphanesi içerisinde yer alan bu model, araç üzerindeki plaka konumlarını tespit etmekte kullanılmıştır. Anchor-free yapısı ve yüksek mAP değeri ile farklı mesafelerden alınan görüntülerde dahi kararlı tespit sonuçları vermektedir.

5.PaddleOCR : PaddleOCR, optik karakter tanıma (OCR) için geliştirilmiş, CRNN ve attention mekanizması kullanan bir derin öğrenme modelidir. Bu sistemde, YOLOv8 tarafından tespit edilen plaka bölgesi içerisindeki karakter dizisini tanımak için kullanılmıştır. Latin alfabesine dayalı plakalar üzerinde yüksek doğruluk oranına sahiptir.

6.NumPy: NumPy, yüksek performanslı çok boyutlu dizi ve matris işlemleri için kullanılan temel bilimsel hesaplama kütüphanesidir. FastANPR içinde görüntülerin sayısal temsilini işlemek ve model girişine uygun hale getirmek amacıyla kullanılmıştır.

7.IP Webcam: Mobil cihazdan alınan canlı görüntü akışını IP üzerinden bilgisayara aktarmak amacıyla kullanılmıştır. Bu sayede sabit kameraya ihtiyaç duymadan canlı video akışı üzerinden gerçek zamanlı plaka tanıma gerçekleştirilmiştir.

**Sistemin İşleyişi**

Araç plaka tanıma sistemi (FastANPR), YOLOv8 tabanlı bir plaka tespit modeli ile PaddleOCR tabanlı bir metin tanıma modelini entegre ederek çalışmaktadır . Sistem, gerçek zamanlı kamera akışı, önceden kaydedilmiş video veya tekil resimler üzerinde çalışabilir. Girdi kaynağı ne olursa olsun, altında yatan işlem hattı (pipeline) aynıdır: görüntü yakalanır ve hazırlanır, plaka konumları saptanır, her plaka bölgesinin içindeki karakterler tanınır ve sonuçlar kullanıcıya sunulur. Bu süreç, derin öğrenme tabanlı iki ana bileşene dayanmaktadır: YOLOv8 ile plaka tespiti ve PaddleOCR ile karakter tanıma . Aşağıda sistemin çalışma adımları verilmiştir:

**1.Görüntü Yakalama ve Hazırlık:** Kameradan anlık kare yakalanır veya video dosyasından ardışık kareler okunur (tekil resim modunda bir görüntü dosyası yüklenir). OpenCV kütüphanesi kullanılarak bu kare hafızaya alınır. Ardından, gerekli ön işlemler uygulanır: Örneğin, YOLOv8 modelinin beklentisine uygun olarak BGR formatındaki görüntü RGB renk uzayına çevrilir. Gerekirse çözünürlük ayarlamaları veya gürültü azaltma gibi işlemler de bu adımda yapılabilir. Canlı video ve video dosyası modlarında bu adım her bir kare için döngüsel olarak tekrarlanır.

**2. Plaka Tespiti (YOLOv8):** Hazırlanan görüntü, YOLOv8 derin öğrenme modeli ile işlenerek araç plakalarının bulunması sağlanır. YOLOv8, tek aşamalı (one-stage) bir nesne tespit algoritmasıdır; tek bir sinir ağı ile tam görüntüyü katarak nesnelerin konumlarını ve sınıflarını tek geçişte tahmin eder . Bu sayede gerçek zamanlı çalışabilecek hız ve yeterli doğruluk sağlanır. YOLOv8 modeli, görüntüde plaka benzeri bölgeleri tespit ederek her birine ait dikdörtgen koordinatlarını (x-y konum ve genişlik-yükseklik) ve bir güven skoru üretir . Sistem, belirli bir eşik değerin üzerindeki güven skorlarına sahip tespitleri plaka olarak kabul eder. Tespit edilen her plaka bölgesi, orijinal görüntüden koordinatları kullanılarak kesilir (crop işlemi) ve ayrı bir görüntü parçası olarak elde edilir . Plaka tespiti yapılmadığı durumda, ilgili karede tanıma adımına geçilmeden doğrudan bir sonraki kareye geçilir. YOLOv8’in derin evrişimsel sinir ağı ve dikkat mekanizmalarını birleştiren gelişmiş mimarisi, küçük boyutlu nesneler olan plakaları dahi yüksek doğrulukla tespit edebilmesine olanak tanır .

**3. Karakter Tanıma (PaddleOCR):** Her bir tespit edilen plaka bölgesi görüntüsü, PaddleOCR metin tanıma modeli ile işlenerek plaka üzerindeki karakterlerin çözülmesi sağlanır. PaddleOCR, görüntülerdeki metinleri tespit etmek ve tanımak için geliştirilmiş, çok dilli desteği olan açık kaynaklı bir OCR sistemidir . Bu model, verilen plaka alt-görüntüsünde önce karakter bölgelerini saptar, ardından bu karakterleri sınıflandırarak metin çıktısını üretir . Sistemde PaddleOCR’nin İngilizce dil modeli kullanılmış olup (Türk plakaları Latin harf ve rakam içerdiği için), plaka üzerindeki yazıyı yüksek doğrulukla tanıyabilir. Bir plaka görüntüsünde birden fazla metin parçası algılanırsa PaddleOCR bunları birleştirerek tam plakayı oluşturan metni tek parça haline getirir . Ayrıca tanıma güven skoru da hesaplanır. Elde edilen ham metin üzerinde bazı son işlemler yapılabilir – örneğin, gereksiz boşluk veya özel karakterler varsa temizlenir . Sonuç olarak her plaka bölgesi için okunabilen bir plaka metni ve buna ait bir güven değeri elde edilmiş olur.

**4. Sonuçların Gösterimi ve Çıktı:** Tanıma adımının ardından, bulunan plaka metinleri orijinal görüntü üzerinde kullanıcıya görsel olarak sunulur. Her plakanın konumu, kaynak görüntü üzerine bir dikdörtgen kutu ile çizilir ve tanınan metin bu kutunun yanına yazılır . Bu sayede kullanıcı, tespit edilen plaka ve okunan metni gerçek zamanlı olarak kare üzerinde görebilir. Canlı kamera veya video modunda bu çıktılar her karede sürekli güncellenerek bir akış halinde gösterilir . Tekil resim modunda ise sonuçlar bir görüntü dosyası olarak kaydedilebilir veya konsola/metin dosyasına yazılabilir . Sistem ayrıca her tespit ve tanıma sonucunu (plaka koordinatları, metin ve güven skorları gibi) veri yapıları içerisinde tutabilir; örneğin test modunda tüm sonuçlar bir JSON dosyasına kaydedilerek daha sonra analiz edilebilmektedir.

Yukarıdaki adımlar, sistem çalıştığı sürece (video akışı boyunca veya tüm test görüntüleri bitene dek) döngüsel olarak tekrarlanır. Plaka tespit ve tanıma algoritmasının bütünsel akışı, aşağıdaki akış diyagramında görselleştirilmiştir. Bu diyagramda sistemin başlangıçtan bitişine kadar gerçekleştirdiği işlemler sırasıyla gösterilmektedir.

metin, diyagram, çizgi, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Şekil 4: Plaka tanıma sisteminin algoritma akış diyagramı.

Plaka tespit modülü (YOLOv8) ile karakter tanıma modülünün (PaddleOCR) etkileşimi de bir blok diyagramı ile aşağıda özetlenmiştir. Bu şemada, YOLOv8 modelinin girdi görüntü üzerinde plaka bölgesini saptaması ve ardından bu bölgenin PaddleOCR modeline aktarılıp metnin okunması şeklindeki iş akışı gösterilir. YOLOv8’in çıktısı olan plaka alt-görüntüsü, PaddleOCR için girdi teşkil etmekte; PaddleOCR ise nihai plaka karakter dizisini üretmektedir



Şekil 5: YOLOv8 ve PaddleOCR entegrasyonunu gösteren blok diyagram.

Bu şekilde geliştirilen FastANPR sistemi, geleneksel yöntemlere kıyasla daha güvenilir ve hızlı bir şekilde araç plakalarını tespit edip okuyabilmektedir. YOLOv8’in derin öğrenmeye dayalı gerçek zamanlı tespit gücü ile PaddleOCR’nin yüksek doğruluklu karakter tanıma yeteneği bir araya getirilerek, farklı ışık koşulları, açı ve mesafelerde dahi başarılı sonuçlar veren bütünleşik bir plaka tanıma çözümü elde edilmiştir . Sistemin modüler yapısı sayesinde, gerektiğinde model ağırlıkları güncellenerek veya dil desteği eklenerek farklı senaryolara uyarlanması da mümkündür.

# Deneysel sonuçlar

1. 100 Farklı Fotoğraf ile Test Edilen Modelin Çıktıları

Araç plakalarını fotoğraf üzerinden tanımlayan kod test.py dosyasıdır. Bu dosya çalıştırıldığında fotoğraf alınır modele sokulur ve çıktılar json dosyasına kaydedilir. Oluşturulan araç plaka tanıma sistemi ilk olarak 100 farklı fotoğraftan oluştan test kümesi ile denenmiştir. Bu fotoğraflardaki araçlar özenle seçilmiş olunup , araçların farklı uzaklıkta olmalarına, bir fotoğrafta birden fazla araç olmasına, araçların önden yandan arkadan görüntülerinin bulunmasına ve sadece Türkiye plakaları değil ayrıca yabancı plakalar da bu veri seti içinde yer almıştır. Şekil 6’da bir fotoğrafta bulunan farklı uzaklıktaki araçların plakalarının nasıl tespit edildiği ve Şekil 7’de de bu plakaların json dosyasına hangi formatta kaydedildiği yer almaktadır.

metin, taşıt, araç, kara taşıtı, taşıt plakası içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Şekil 6 : Fotoğraf üzerinden Araç Plaka Tanıma Sistemi

metin, ekran görüntüsü, yazı tipi, tasarım içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Şekil 7 : Json dosyasına kaydedilecek plaka formatı

Bu şekilde 100 farklı fotoğraf modele sokulmuş ve Şekil 8’deki çıktılar elde edilmiştir.

metin, ekran görüntüsü, sayı, numara, çizgi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Şekil 8 : Araç Plaka Tanıma Sistemi Sonuçları

Şekil 8’de de görüldüğü üzere model 100 farklı fotoğrafın 92 tanesini tamamen doğru, 6 tanesini kısmen doğru ve 2 tanesini de hiç tanıyamamıştır.

1. 3 Farklı Video ile Test Edilen Modelin Çıktıları

2 farklı telefon ile 3 adet video çekilmiş ve bu videolar modele Şekil 9’daki gibi sokulmuştur. Model belirli aralıklarla kareler yakalayıp plakaları başarılı bir şekilde tespit etmiştir. 2 farklı telefondan birisinin diğerine göre görece kamerası daha kötü olmasına rağmen o videolarda da başarılı şekilde çalışmıştır. Araçlar farklı hızlarda olmasına rağmen model herhangi bir aracın modelini yanlış tespit etmemiştir.

dış mekan, taşıt, araç, araba, yol, karayolu içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Şekil 9 : Video üzerinden araç plaka tespiti

metin, ekran görüntüsü, yazı tipi, tasarım içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Şekil 10 : Video üzerinden tespit edilen plaka

1. Canlı görüntü ile araç plaka tespiti

Araç plaka tanıma sistemi ile oluşturulan model IP Webcam uygulaması ile cep telefonuna Wifi aracılığı ile bağlanmıştır. Bu sayede araçların plakaları anlık olarak Şekil 11’deki gibi tespit edilmiştir.

metin, araba, kara taşıtı, taşıt, araç içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

Şekil 11 : Canlı görüntü üzerinden tespit edilen plaka

##### *Kaynaklar*

1. Hsieh, J. W., et al. (2006). Morphology-based license plate detection from complex scenes. Pattern Recognition, 43(3), 919–928.
2. Nomura, S., et al. (2002). A new method for license plate recognition. IEEE Trans. ITS, 3(3), 152–160.
3. Smith, R. (2007). An overview of the Tesseract OCR engine. Proceedings of the International Conference on Document Analysis and Recognition (ICDAR).
4. Redmon, J., et al. (2016). You Only Look Once: Unified, Real-Time Object Detection. CVPR.
5. Liao, M., et al. (2023). License Plate Detection Using YOLOv8 in Real-Time Urban Environments. IEEE Access.
6. Wu, Y., et al. (2022). PaddleOCR: A Practical Ultra Lightweight OCR System. ArXiv preprint arXiv:2009.09941.
7. Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. *ArXiv preprint arXiv:2004.10934*.
8. Ultralytics. (2023). YOLOv8 Docs – Models, Training, Inference. <https://docs.ultralytics.com>
9. PaddlePaddle OCR Team. (2023). PaddleOCR GitHub Repository.
10. Arvind Rajan. (2023). FastANPR: Lightweight Automatic License Plate Recognition Toolkit.