

Karamanoğlu Mehmetbey Üniversitesi

Mühendislik Fakültesi

Bilgisayar Mühendisliği Bölümü

**Sayısal Görüntü İşleme Dersi
Proje Formu**

Proje Başlığı

Plaka Tanıma Sistemi

Öğrenci Bilgileri

Öğrenci No 211312073

Öğrenci Ad Soyad Hasan Batuhan AKAY

DersSorumlusu Unvan,Ad,Soyad

Dr.Öğr.Üyesi.İbrahimÇINAROĞLU

**2024-2025 Güz Dönemi
Karaman**

Proje Raporu Hazırlama Kuralları

Her öğrenci bitirme projesini aşağıda belirlenen kurallara göre hazırlayacaktır.

1. Rapor Bölümleri

- Kapak
- İçindekiler
- Projenin Amacı ve Hedefi
- Projede kullanılan materyaller ile ilgili bilgiler
- Projenin yapım aşamaları, uygulanacak metotlar ve elde edilen sonuçlar
- Kaynaklar

2. Konu Anlatımı

- Etik kurallara uygun olarak, öğrenci konuyu kendi cümleleri ile sade bir şekilde anlatmalıdır. Yararlanılan kaynaklar belirtilmelidir.

3. Sayfa Düzeni

- Kenar Boşlukları: 2.5 cm
- Sayfa Numarası: Sağ Alt Köşede Projenin Önemi sayfasından itibaren başlamalıdır.

4. Metin Özellikleri

- Paragraf Girintisi Yok
- Metin iki yana yaslı
- Tek satır aralıklı
- 12 Punto ve Normal
- Paragraflar arasında 1 boşluk
- Ana Başlıklar ve alt Başlıklar: 12 Punto ve Kalın
- Tüm Metin Fontu: Times New Roman

5. Şekiller ve Tablolar

- Şekiller sayfa içerisinde ortalı olmalıdır.
- Şekil açıklama metni şekil altında tek satıra sığıyorsa ortalı, sığmıyorsa iki yana yaslı olmalıdır.
- Tablolar sayfa içerisinde ortalı olmalıdır.
- Tablo açıklama metni tablo üstünde tek satıra sığıyorsa ortalı, sığmıyorsa iki yana yaslı olmalıdır.

6. Kaynaklar

- Bitirme projesinde hazırlanmasında yararlanılan kaynaklar, metin içerisinde kullanım sırasına göre rapor sonunda ve “Kaynaklar” başlığı altında sıra numaraları verilerek listelenecektir.

Proje çalışmalarınızda başarılar dilerim

İçindekiler

1. Projenin amacı ve hedefi	4
2. Projede kullanılacak materyaller ile ilgili bilgiler	4
2.1. Yazılımlar / Kütüphaneler	4
2.2. Veri Seti	4
2.3. Donanımlar	5
3. Projenin yapım aşamaları, uygulanacak metotlar ve elde edilen sonuçlar	5
Kaynaklar (References)	5

1. Projenin amacı ve hedefi (varsa özgün değeri)

Plaka tanıma sistemi, araçların plaka numaralarını otomatik olarak algılayarak dijital ortama aktarmak ve bu veriyi çeşitli uygulamalarda kullanmak amacıyla geliştirilmiş bir sistemdir. Bu tür sistemler, genellikle güvenlik, trafik yönetimi ve otopark kontrolü gibi birçok farklı alanda kullanılmaktadır. Projenin temel amacı, araçların plakalarını otomatik bir şekilde tanımlamak, plaka bilgilerini veri tabanına kaydetmek, plaka kontrolü yapmak ve araçların takibini kolaylaştırmaktır.

Bu projede, görüntü işleme teknikleri ve optik karakter tanıma (OCR) yöntemleri kullanarak araç plakalarını doğru ve hızlı bir şekilde belirlemek ve tanımak hedeflenmektedir. Projenin bir diğer önemli avantajı, sistemin farklı alanlarda kolayca adapte edilebilmesi ve ölçeklenebilir bir yapıya sahip olmasıdır. Örneğin, otopark giriş-çıkış kontrolünde kullanılabileceği gibi, şehir içi trafik izleme, otoyol geçiş sistemleri ya da kamuya açık alanlarda güvenlik kontrol noktaları için de kullanılabilir. Özellikle güvenlik güçleri ve trafik denetim sistemleri için, belirli araçların plakalarının hızlı bir şekilde tespit edilip sistemdeki verilerle karşılaştırılması büyük önem taşımaktadır.

Bu plaka tanıma sistemi, kullanıcı dostu bir arayüz ve verimli veri işleme algoritmalarıyla, geniş bir uygulama alanına sahip olacak şekilde tasarlanacaktır.

2. Projede kullanılacak materyaller ile ilgili bilgiler

2.1. Yazılımlar / Kütüphaneler

- OpenCV: Görüntü işleme için temel kütüphane.
- Python: Kütüphaneleri entegre etmek ve algoritmayı geliştirmek için kullanılacak programlama dili.
- Datetime: İşlem zaman damgalarının oluşturulması için.
- Time: Fonksiyonu saniye cinsinden zaman ölçümü yapmak için kullanılır.
- Numpy: Görüntü verileri üzerinde işlemler için sayısal hesaplama işlemlerini sağlar.
- Imutils: Görüntü işleme sırasında konturları işlemek için yardımcı fonksiyonlar sağlar.
- Save: Tanımlı bir dosyaya veri yazma işlemlerini gerçekleştirir.
- Easyocr: Görüntüden metin tanıma işlemlerini yapar.

2.2. Veri Seti

Plaka tanıma için çeşitli araç plakalarını içeren bir veri setine ihtiyacımız olacak. Açık veri setleri kullanabiliriz veya kendi görüntülerimizi toplayabiliriz. Örnek olarak birkaç veri setleri:

- OpenCV

2.3. Donanımlar

- Kamera: Plaka görüntülerini almak için kullanılacak.
- Bilgisayar: Görüntü işleme ve analiz işlemlerini yapmak için gerekli.

3. Projenin yapım aşamaları, uygulanacak metotlar ve elde edilen sonuçlar

Projemın Yapım Aşaması Ve Uyguladığım Metotlar:

Neden Canny Kullandım?

1. Yüksek Doğruluk:

- a. Canny kenar algılama yöntemi, güçlü ve zayıf kenarları algılayarak daha kesin kenar haritaları oluşturur.
- b. Plaka tanıma gibi uygulamalarda kenarların doğru belirlenmesi kritik bir önem taşır. Bu nedenle Canny yöntemi tercih edilmiştir.

2. Gürültüye Karşı Dayanıklılık:

- a. Görüntü işleme aşamasında gürültü, yanlış kenar tespitine neden olabilir.
- b. Canny yöntemi, önceden uygulanan **pürüzsüzleştirme işlemleri** (örneğin **bilateral filter**) sayesinde gürültülerin etkisini azaltarak net ve kesin kenarların belirlenmesini sağlar.

3. Çift Eşikleme:

- a. Canny yönteminde çift eşikleme kullanılır. Bu, yalnızca anlamlı ve bağlantılı kenarların tespit edilmesine olanak tanır.
- b. Örneğin, plaka çerçevesine benzer belirgin kenarların doğru seçilmesini sağlar.

4. Performans Avantajı:

- a. Canny yöntemi, diğer kenar algılama yöntemlerine göre daha hızlı çalışır.
- b. Gerçek zamanlı uygulamalarda (canlı video akışında plaka tanıma gibi) performans açısından büyük avantaj sağlar.

Neden EasyOCR Tercih Ettim?

1. Basit ve Hızlı:

- a. EasyOCR, eğitim gerektirmeyen hazır bir OCR çözümüdür.
- b. Bu sayede plaka tanıma sistemi için hızlı bir şekilde kullanılabilir ve geliştirme süreci hızlanır.

2. Doğruluk:

- a. OCR işlemlerinde dil seçeneği ile doğruluk artırılabilir. Örneğin 'en' seçeneği ile İngilizce karakterler üzerinde işlem yapmak, plaka üzerindeki karakterlerin daha doğru bir şekilde tanınmasını sağlar.

3. Geniş Destek:

- a. EasyOCR, rakamlar ve harfler için yüksek performans gösterir.
- b. Bu, plaka üzerindeki karakterlerin hızlı ve etkili bir şekilde tanınmasına yardımcı olur.

Neden Maskeleye Yaptım?

1. Plaka Alanını İzolasyon:

- a. Görüntü üzerinde **screenCnt** adı verilen plaka bölgesinin sınırları belirlenerek bu alan izole edildi.
- b. Böylece yalnızca plaka alanına işlem yapılmak hedeflendi.

2. İhtiyaç Dışı Alanların Dışlanması:

- a. Görüntü üzerinde plaka dışında yer alan arka plan ve diğer gereksiz detaylar maskeleye işlemi sayesinde dışlanarak sadece hedef bölge üzerinde işlem yapıldı.

Neden Kesme Yaptım?

1. Sadece Plaka Alanını İşleme:

- a. Görüntü üzerinde plaka bölgesinin sınırlarını belirledikten sonra bu bölge kesildi. Bu, yalnızca plaka bölgesinde işlem yapılmasını sağlar.

2. OCR için Odaklanma:

- a. Kesme işlemi ile plaka bölgesindeki yalnızca gerekli bilgi bölgesine odaklanıldı.
- b. Bu işlem OCR için gerekli gürültüsüz ve net bir giriş oluşturdu. Böylece OCR sisteminin doğruluğu artırıldı.

Projemden Elde Ettiğim Sonuçlar

1. Plaka Tanıma İşlemlerinde Doğruluk Sağlandı:

- a. Kullanılan **Canny Kenar Algılama** yöntemi sayesinde, plaka çerçevelerinin kenarları doğru bir şekilde algılanarak başarılı bir plaka bölgesi tespiti gerçekleştirildi.

2. Gerçek Zamanlı Plaka Tanıma:

- a. Canlı video akışı üzerinde yapılan işlemler sonucunda, sistemin gerçek zamanlı olarak çalıştığı

doğrulandı.

- b. Bu sayede kullanıcıların anlık olarak araç plakalarını görüntüleyebilmesi sağlandı.

3. OCR İşlemlerinin Başarılı Sonuçları:

- a. **EasyOCR** kullanılarak yapılan karakter tanıma işlemlerinde, plaka üzerindeki harf ve rakamlar başarıyla algılanarak analiz edildi.
- b. Bu sayede plaka tanıma işlemlerinde yüksek doğruluk oranına ulaşılmış oldu.

4. Plaka Bölgesinin Doğru Bir Şekilde İzolasyonu:

- a. Görüntü işleme yöntemleri ve maskeleye işlemleri kullanılarak plaka bölgesi doğru bir şekilde izole edildi.
- b. Bu, plaka dışında gereksiz arka plan detaylarının sistemden çıkarılmasını sağladı ve işlem yükü azaltıldı.

5. Kullanıcı Deneyimi Odaklı Arayüz ve İşlem Akışı:

- a. Video akışında plaka tespiti sonucunda kullanıcıya gerekli geri bildirimler sunuldu.
- b. Kullanıcıların işlem yapmasını kolaylaştıracak bir yapı oluşturuldu.

6. Kenar Algılama ve Gürültü Yönetimi:

- a. Canny kenar algılama yöntemi ile kenar tespitinde yüksek hassasiyet sağlandı.
- b. Gürültü azaltma ile sistemin farklı ortam koşullarında performans sorunu yaşamaması önendi.

7. Plaka Tanıma Sürecinin Başarıyla Gerçekleştirilmesi:

- a. Plaka bölgesi algılama, maskeleye, kesme ve OCR adımlarının başarılı bir şekilde entegre edilmesiyle plaka tanıma işlemleri başarıyla tamamlandı.

8. Hızlı ve Gerçek Zamanlı İşlem:

- a. Kullanılan yöntemlerin ve algoritmaların optimize edilmesi sayesinde, plaka tanıma işlemlerinin hızlı bir şekilde gerçekleştirilebildiği doğrulandı.

Kaynaklar (References)

OpenCV Documentation: <https://docs.opencv.org/>

EasyOCR: <https://github.com/JaidedAI/EasyOCR>

Datetime(Python Standart Kütüphanesi): <https://docs.python.org/3/library/datetime.html>

NumPy (numpy): <https://numpy.org>

Imutils: <https://github.com/PyImageSearch/imutils>