

BURSA TEKNİK ÜNİVERSİTESİ MÜHENDİSLİK VE DOĞA
BİLİMLERİ FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ
BÖLÜMÜ SEMİNER DERSİ PROJESİ



Bilgisayar Bilimleri ve Robotik Cerrahi

Abdulkadir Yıldırım-20360859017

2023-2024

Önsöz

Bu çalışmanın amacı, robotik cerrahi ve yapay zeka teknolojilerini anlamak, bu teknolojilerin tıp ve sağlık sektörüne olan etkilerini değerlendirmek ve gelecekteki potansiyellerini tartışmaktır. Rapor, bu konulara ilgi duyan araştırmacılar, sağlık profesyonelleri ve öğrenciler için önemli bir kaynak olacaktır.

İÇİNDEKİLER

1) Önsöz

2) Özet

3) Robotik Cerrahinin Tanımı ve Gelişimi

- Tanımı ve Tarihçesi
- Kullanım Alanları

4) Yapay Zeka Kullanımı

- Derin Öğrenme
- Görüntü İşleme
- Simülasyon

5) Kullanılan Algoritmalar

- Kinematik ve Dinamik Algoritmalar
- Görüntü İşleme Algoritmaları
- Path Planning Algoritmaları
- Haptik Geribildirim Algoritmaları
- Simülasyon ve Eğitim Algoritmaları

6) Sistem Güvenliği

7) Uygulama

8) Kaynakça

Özet

Bu rapor, robotik cerrahi ve yapay zeka alanlarında yaşanan son gelişmeleri ele alarak bu teknolojilerin tıp alanındaki önemini vurgulamaktadır. İlk olarak, robotik cerrahinin tanımı, tarihçesi ve kullanım alanları incelenir. Ardından, yapay zeka teknolojilerinin robotik cerrahideki önemi üzerinde durulur; derin öğrenme, görüntü işleme ve simülasyon gibi temel konular detaylı bir şekilde açıklanır. Daha sonra, robotik cerrahide kullanılan çeşitli algoritmalar incelenir; kinematik ve dinamik algoritmalar, görüntü işleme algoritmaları, path planning algoritmaları, haptik geribildirim algoritmaları ve simülasyon ve eğitim algoritmaları bu başlık altında detaylı bir şekilde ele alınır. Sonraki bölümde, sistem güvenliği ve veri koruma önlemleri vurgulanır.

Raporun pratiğe dönük yanı, uygulama alanları üzerinde durarak, teorik bilgilerin gerçek dünya uygulamalarına nasıl dönüştürüldüğünü gösterir. Son olarak, raporun kaynakça bölümünde kullanılan kaynakların listesi sunulur. Bu rapor, robotik cerrahi ve yapay zeka alanlarındaki son gelişmeleri inceleyerek okuyuculara kapsamlı bir bakış sunmayı ve bu teknolojilerin tıp alanında nasıl kullanıldığını anlatmayı amaçlamaktadır.

Robotik Cerrahinin Tanımı ve Gelişimi

1) Tanımı ve Tarihçesi:

Robotik cerrahi, cerrahi müdahalelerin gerçekleştirilmesinde robot teknolojisinin kullanıldığı bir tıp alanıdır. Bu alanda cerrahlar, cerrahi robotlar veya robotik cerrahi sistemleri aracılığıyla hastalara minimal invaziv müdahalelerde bulunabilirler.

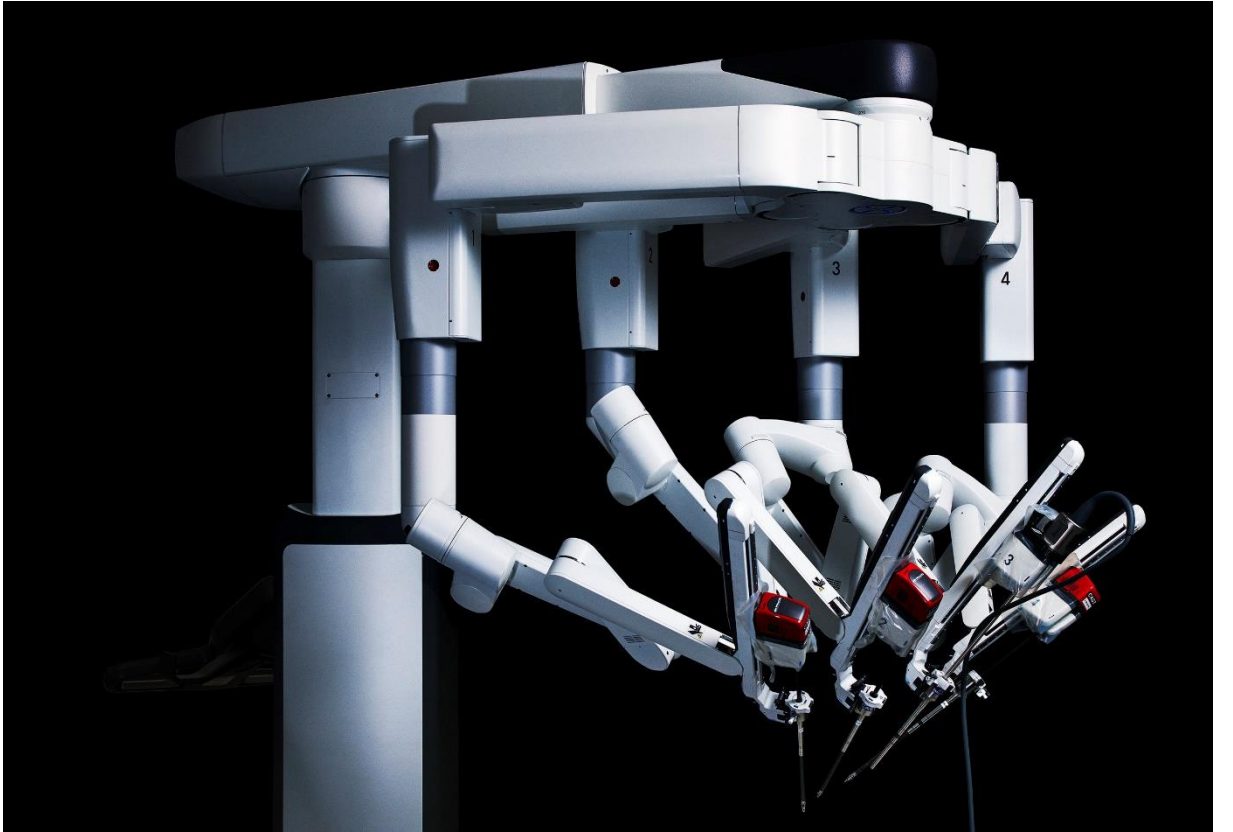
Robotik cerrahi, cerrahi operasyonların hassaslığını ve doğruluğunu artırarak, hastalara daha güvenli ve etkili tedavi seçenekleri sunar. Ayrıca operasyon sırasındaki hassasiyet ve dikkatin sonucu olarak hastanın iyileşme süresi de gözle görülür bir biçimde kısalmıştır.



Robotik cerrahinin tarihçesi, 1980'lerin ortalarına dayanır. İlk cerrahi robotlar, endoskopik ve laparoskopik cerrahi gibi minimal invaziv yöntemlerde kullanılmaya başlanmıştır.

1990'ların başlarından itibaren robotik cerrahi sistemleri, cerrahların ameliyatları daha hassas ve doğru bir şekilde gerçekleştirmelerine olanak sağlamıştır.

2000'li yılların başlarından itibaren da Vinci Surgical System gibi gelişmiş robotik cerrahi sistemleri piyasaya sürülmüş ve cerrahi alanında bir devrim yaratmıştır. Bu sistemler, cerrahlara 3D görüntüleme, yüksek hassasiyetli hareketler ve ergonomik avantajlar sunarak operasyonların etkinliğini artırmıştır.

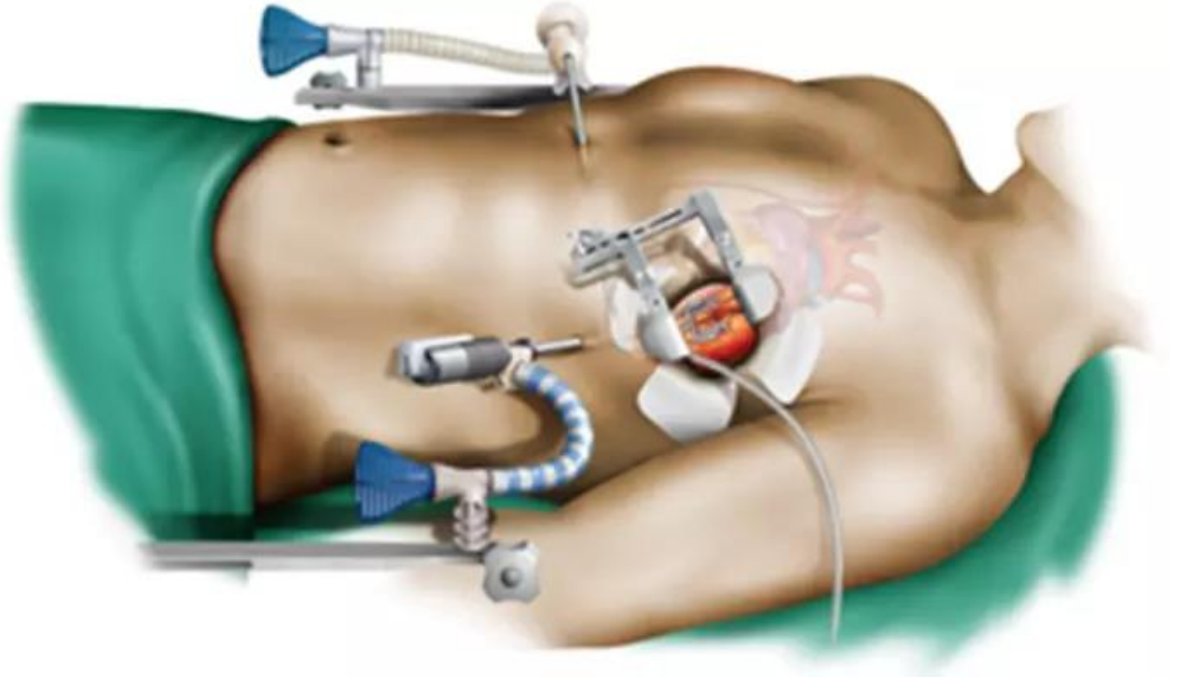


da Vinci Surgical System

2) Kullanım Alanları:

Robotik cerrahi, birçok farklı kullanım alanına sahiptir ve aşağıda bazı önemli kullanım alanlarına örnekler verilmiştir:

- **Üroloji:** Prostat kanseri, böbrek kanseri ve mesane kanseri gibi durumlarda robotik cerrahi tercih edilir.
- **Kardiyak Cerrahi:** Kalp kapakçıklarının onarımı, bypass ameliyatları ve diğer karmaşık kalp operasyonları için kullanılır.
- **Nöroloji:** Tümör çıkarma, epilepsi cerrahisi ve Parkinson hastalığı tedavisi gibi alanlarda robotik cerrahi önemli bir rol oynar.
- **Ortopedi:** Eklem replasmanı, omurga cerrahisi ve kemik kırıkları gibi ortopedik operasyonlarda da kullanılır.



Yapay Zeka Kullanımı

1) Derin Öğrenme:

Karmaşık yapıları otomatik olarak öğrenmeye dayalı bir makine öğrenimi yöntemidir.

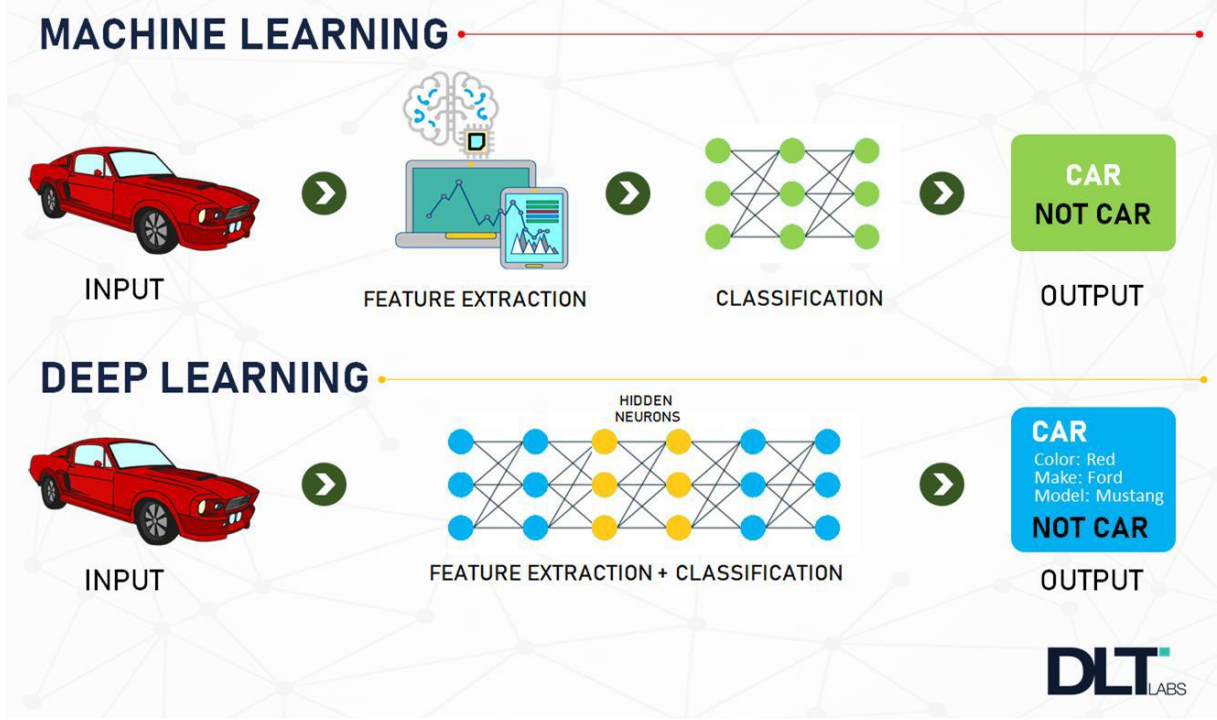
Bu teknik, yapay sinir ağları gibi karmaşık yapıli modellerin kullanılmasıyla veri analizinde ve örüntü tanıma gibi konularda son derece başarılı sonuçlar elde etmeyi amaçlar. Derin öğrenme, büyük veri setleri üzerinde karmaşık yapıları ve ilişkileri keşfetmek için kullanılır.

Kısacası eldeki verilerle yeni veriler elde etmektir.

Robotik cerrahideki kullanımı;

- a. **Robot Kontrolü:** Derin öğrenme, cerrahi robotların hareketlerini optimize etmek ve daha hassas ve doğru manipölasyonlar yapabilmelerini sağlamak için kullanılabilir.
- b. **Hasta Analizi:** Derin öğrenme, hastaların tıbbi verilerini analiz ederek cerrahların daha iyi tedavi planları oluşturmalarına yardımcı olabilir. Örneğin, hastaların genetik özelliklerini veya tıbbi geçmişlerini analiz ederek, cerrahlar tedavi sürecini kişiselleştirebilir ve daha etkili sonuçlar elde edebilirler.

- c. Komplikasyon Tahmini:** Derin öğrenme, cerrahi operasyonlar sırasında olası komplikasyonları tahmin etmek için kullanılabilir. Örneğin, ameliyat sırasında alınan verileri analiz ederek, derin öğrenme modelleri cerrahlara olası riskleri önceden bildirebilir ve önlemler almalarına yardımcı olabilir.



Makine Öğrenmesi ve Derin Öğrenme Farkı

2) Görüntü İşleme

Görüntü işleme, dijital görüntülerin analiz edilmesi, iyileştirilmesi ve yorumlanmasıyla ilgilenen bir bilgisayar bilimi dalıdır. Bu teknoloji, görüntülerdeki bilgiyi çıkarmak, desenleri tanımak, nesneleri algılamak ve görüntüleri anlamlandırmak için çeşitli algoritmaları kullanır.

Aşamaları;

I. Görüntü Edinme:

Bu ilk aşama, çeşitli dijital görüntülerin toplanmasını içerir.

II. Ön İşleme:

Elde edilen görüntülerdeki gürültülerin azaltılması, kontrastın artırılması, keskinlik düzeltmeleri yapılması gibi temel işlemler gerçekleştirilir.

III. Görüntü İyileştirme:

Bu aşamada görüntülerin kalitesini artırmak için çeşitli teknikler kullanılır. Bu teknikler arasında histogram eşitleme, kontrast artırma, kenar tespiti ve bulanıklık giderme gibi işlemler yer alır.

IV. Özellik Çıkarımı:

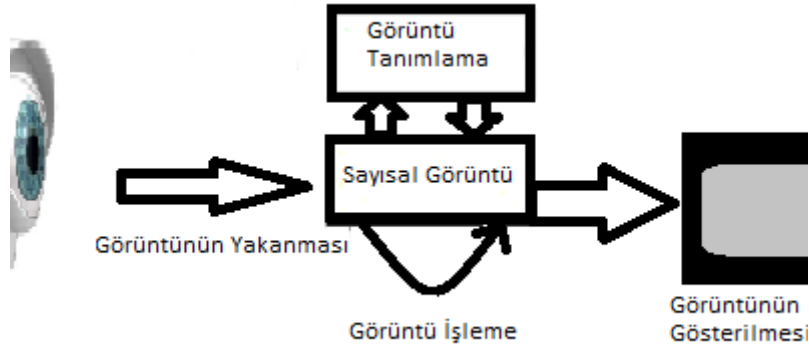
Özellik çıkarımı, görüntülerdeki önemli özellikleri tanımlamak ve bu özellikler üzerinden analiz yapabilmek için kullanılır.

V. Özellik Seçimi:

Özellik seçimi aşamasında, özellik çıkarımı sonucunda elde edilen özellikler arasından en anlamlı ve işe yarar olanları seçilir.

VI. Sonuçların Analizi ve Yorumlanması:

Son aşamada, elde edilen sonuçlar analiz edilir, yorumlanır ve istenilen hedeflere ulaşıp ulaşılmadığı değerlendirilir.



Robotik Cerrahide Kullanımı;

a. Cerrahi Navigasyon:

Görüntü işleme, cerrahların ameliyat sırasında iç organları ve dokuları daha net bir şekilde görmelerini sağlar. Özellikle 3D görüntüleme ve stereoskopik görüntüleme tekniklerini kullanmak, cerrahların ameliyatı doğru yerde ve doğru yönde gerçekleştirilmesini sağlar.

b. Doku Tanıma:

Görüntü işleme algoritmaları, cerrahi sırasında elde edilen görüntülerdeki dokuları tanımak ve sınıflandırmak için kullanılabilir. Örneğin, kanserli dokuları normal dokulardan ayırmak için renk analizi veya desen tanımlama teknikleri kullanılabilir.

c. Hareket Denetimi:

Görüntü işleme, cerrahi robotların hareketlerini kontrol etmek ve manipülasyonları gerçekleştirmek için kullanılabilir. Özellikle, cerrahi robotlarla gerçekleştirilen karmaşık operasyonlarda, görüntü işleme algoritmaları robotların doğru hareketleri yapmasına yardımcı olabilir.

3) Simülasyon

Simülasyon, gerçek dünya koşullarını taklit eden sanal ortamlar oluşturarak çeşitli durumları modelleme ve analiz etme sürecidir.

Robotik cerrahide simülasyon, cerrahi operasyonların sanal ortamda modellenmesi ve gerçekleştirilmesi anlamına gelir. Bu simülasyonlar, cerrahların yeni teknikleri öğrenmeleri, mevcut becerilerini geliştirmeleri ve karmaşık operasyonları planlamaları için önemli bir araçtır.



Pick-and-Place



Peg Transfer



Da Vinci SKILLS Simulator



Bladder Flap creation



Robotic Hysterectomy

Da Vinci Skills Simulator, Intuitive Surgical tarafından geliştirilen ve robotik cerrahi eğitiminde kullanılan bir simülasyon sistemidir. Bu sistem, cerrahların da Vinci cerrahi robotunu kullanma becerilerini geliştirmek ve yeni cerrahi teknikleri öğrenmek için tasarlanmıştır.

Robotik Cerrahide Kullanımı;

a. Farklı Senaryolar:

Simülatör, farklı cerrahi senaryoları sunar ve cerrahların çeşitli operasyon tekniklerini deneyimlemesine olanak tanır.

b. Operasyon Planlama:

Cerrahlar, simülasyonlar aracılığıyla operasyon stratejilerini belirleyebilir ve planlarını uygulamaya geçirmeden önce farklı senaryoları test edebilirler.

c. Gerçek Zamanlı Geri Bildirim:

Simulator, cerrahların performansını gerçek zamanlı olarak izler ve geri bildirim sağlar.

d. Risk Azaltma:

Simülasyonlar, cerrahların gerçek hastalar üzerinde deneyim kazanmadan önce riskleri minimize etmelerine yardımcı olur. Bu sayede cerrahlar, operasyon sırasında karşılaşılabilecekleri potansiyel riskleri önceden belirleyebilir ve bu riskleri yönetmek için stratejiler geliştirebilirler.

Kullanılan Algoritmalar

1. Kinematik ve Dinamik Algoritmalar

i. Kinematik Algoritmalar:

Bu algoritmalar, robotik cerrahide robot kollarının hareketini ve pozisyonunu hesaplamak için kullanılır. Özellikle, robotik cerrahi sistemlerindeki eklemlerin konumunu belirlemek ve bu eklemlerin hareketlerini optimize etmek için kinematik modeller kullanılır.

- #### **ii. Dinamik algoritmalar,**
- robot kollarının hareketini ve yüksek hızlı manipülasyonları optimize etmek için kullanılır. Bu algoritmalar, robotun hareketi sırasında oluşabilecek titreşimleri minimize etmek ve hassas manevralar gerçekleştirmek için dinamik dengelemeler sağlar.

2. Görüntü İşleme Algoritmaları

Görüntü işleme algoritmaları, ameliyat sırasında elde edilen görüntüleri analiz etmek ve cerrahi müdahaleyi yönlendirmek için kullanılır. Raporun daha önceki sayfalarında da bu konu detaylı bir şekilde ele alınmıştır.

3. Path Planning Algoritmaları

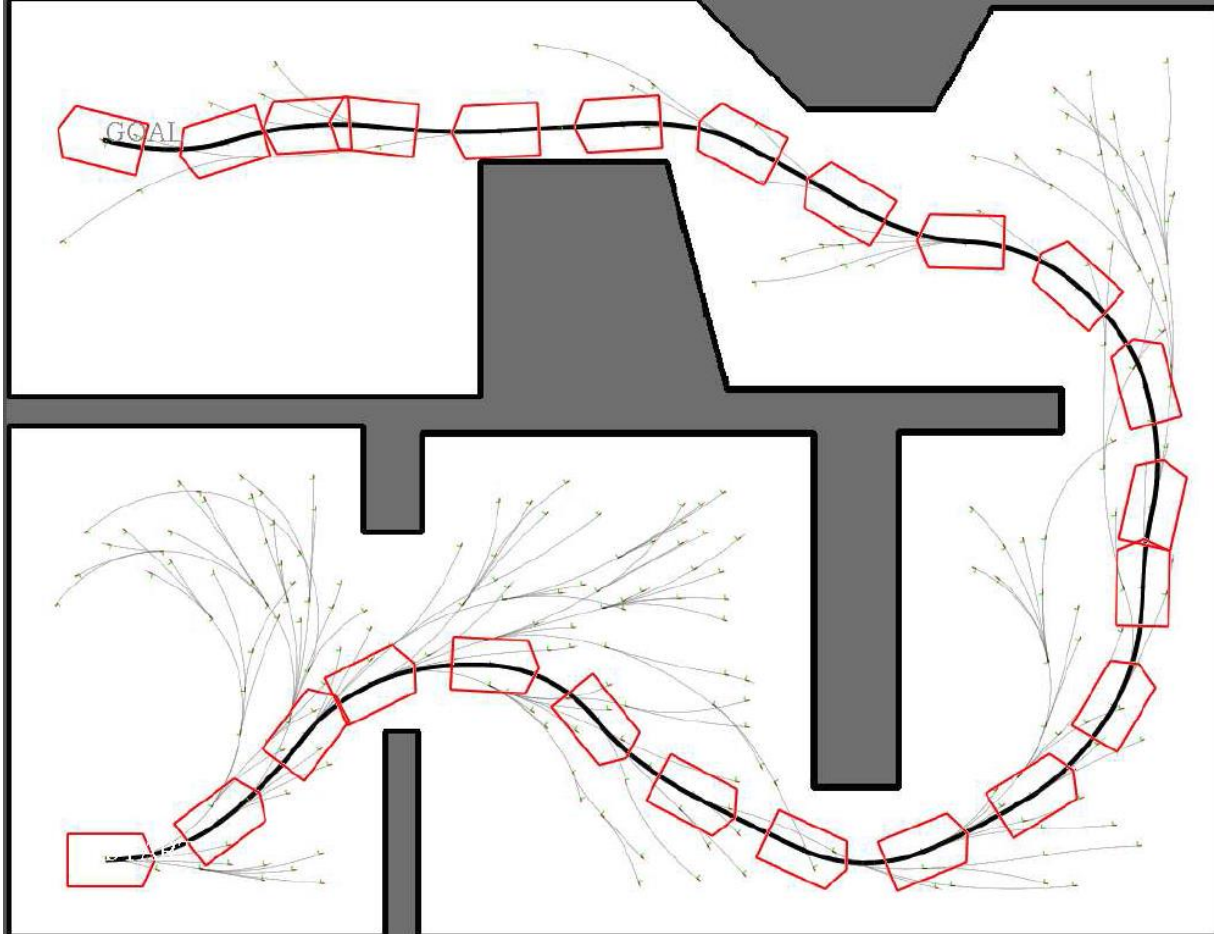
Path planning (yol planlama) algoritmaları, robot kollarının optimal hareket yollarını belirlemek için kullanılır. Bu algoritmalar, cerrahi robotun belirli bir hedefe ulaşması için en uygun ve güvenli yolu hesaplar. Yol planlama algoritmaları, çevresel engelleri dikkate alarak robotun çarpışma riskini minimize eder.

4. Haptik Geri Bildirim Algoritmaları

Haptik geribildirim algoritmaları, cerrahi robotun operasyon sırasında dokularla temasını simüle etmek için kullanılır. Cerrah, haptik geribildirim sayesinde dokuların sertlik, elastikiyet gibi özelliklerini hissedebilir. Bu da cerrahın operasyon sırasında daha hassas ve doğru müdahale yapmasına yardımcı olur.

5. Simülasyon ve Eğitim Algoritmaları

Simülasyon ve eğitim algoritmaları, cerrahların robotik cerrahi becerilerini geliştirmek ve yeni teknikleri öğrenmek için kullanılır. Bu algoritmalar, sanal ortamlarda gerçekleştirilen operasyonları simüle ederek cerrahların pratik yapmalarını sağlar. Ayrıca, geri bildirim mekanizmalarıyla cerrahların performansını değerlendirir ve eğitim sürecini optimize eder.



Path Planning Algoritması

Sistem Güvenliđi

Robotik cerrahide sistem güvenliđi, cerrahi robotlar ve ilgili altyapıların kullanımı sırasında güvenlik risklerini minimize etmeyi amaçlayan bir dizi önlemi içerir. Bu önlemler, hem cerrahi robotunun doğru ve güvenli bir şekilde çalışmasını sağlamak hem de hastaların ve cerrahların güvenliđini korumak için gereklidir.,

1) Güvenlik Eğitimleri ve Protokolleri:

Kullanıcılar, cerrahi robot sistemlerinin güvenli kullanımı konusunda eğitilir ve güvenlik protokolleri belirlenir. Acil durumlar için hazırlıklı olmak ve doğru tepkileri verebilmek için eğitimler düzenlenir.

2) Kimlik Doğrulama ve Erişim Kontrolü:

Cerrahi robot sistemlerinde, kullanıcıların kimlik doğrulaması ve erişim kontrolleri sağlanır. Sadece yetkili ve eğitilmiş personelin sisteme erişim sağlaması ve operasyon yapabilmesi sağlanır.

3) Sistem Tasarımı ve Güvenlik Standartları:

Cerrahi robotların tasarımı, güvenlik standartlarına ve yönetmeliklere uygun olarak yapılır. Bu standartlar, robotun donanım ve yazılım bileşenlerinin güvenliđini sağlamak için belirlenir.

4) Hata ve Güvenlik Alarm Sistemleri:

Cerrahi robotlar, sistem hatalarını algılayabilen ve güvenlik önlemleri alabilen alarm sistemleriyle donatılır. Örneđin, robotun anormal bir durumda otomatik olarak durması veya cerrahi uyarıcı mesajlar göndermesi gibi özellikler bulunabilir.

5) Risk Deđerlendirmesi ve Sürekli İyileştirme:

Robotik cerrahi sistemlerinde sürekli olarak risk deđerlendirmesi yapılır ve güvenlik önlemleri sürekli olarak gözden geçirilir. Bu sayede olası güvenlik açıkları tespit edilir ve gerekli iyileştirmeler yapılır.

Uygulama

OpenCV: OpenCV (Open Source Computer Vision Library), bilgisayarla görme ve makine öğrenimi alanlarında kullanılan açık kaynaklı bir yazılım kütüphanesidir. OpenCV, görüntü işleme ve analiz, video işleme, obje tanıma, yüz tanıma, hareket izleme ve diğer bilgisayarla görme uygulamaları için kapsamlı bir araç seti sunar.



Kurulum: Öncelikle Python’u bilgisayarımıza kurmuş olmamız gerekmektedir.

Python ile birlikte PIP(Python Package Manager)’i de kurduğunuzdan emin olun.

Daha sonra terminali açarak “`pip install opencv-python`” komutunu yazın.

İsteğe bağlı olarak “`pip install numpy`” komutu ile numpy’i de kurabilirsiniz.

En son olarak ise projemizin başında “`import cv2`” yazarak projemize ekleyebiliriz.

OpenCV Yetenekleri:

1) Görüntü İşleme:

OpenCV, görüntü işleme işlemleri için geniş bir fonksiyon yelpazesi sunar. Bu fonksiyonlar arasında filtreleme, kenar tespiti, histogram eşitleme, dönüşümler (çevrim, döndürme, ölçekleme) ve daha birçok işlem bulunur.

2) Objekt Tanıma ve Takip:

OpenCV, çeşitli algoritmalar kullanarak nesnelerin tanınması ve takibi için araçlar sağlar. Bu, yüz tanıma, hareket izleme, renk tabanlı obje takibi gibi uygulamaları içerir.

3) 3D Rekonstrüksiyon:

OpenCV, stereo görüş ve yapıdan hareket (structure from motion) teknikleriyle 3D modelleme ve rekonstrüksiyon yapabilir.

4) Makine Öğrenimi:

OpenCV, makine öğrenimi ve derin öğrenme algoritmaları için destek sunar. Bu, sınıflandırma, kümeleme, regresyon ve derin sinir ağları gibi işlemleri içerir.



A) Sobel Operatörü ile Kenar Tespiti:

Sobel operatörü, görüntü işleme ve bilgisayarla görme alanlarında kullanılan bir kenar algılama tekniğidir. Sobel operatörü, görüntüdeki kenarları tespit etmek için kullanılan bir gradyan tabanlı yöntemdir. Bu operatör, görüntünün yatay ve dikey gradyanlarını hesaplayarak kenarları belirler.

- 1) Projemize cv2 ve numpy kütüphanelerini ekliyoruz.

```
import cv2
import numpy as np
```

- 2) Kullanmak istediğimiz görüntüyü gri bir şekilde projeye ekliyoruz.

```
# Görüntüyü yükle
image = cv2.imread('ball.jpg', cv2.IMREAD_GRAYSCALE)
```

- 3) Sobel operatörü sayesinde görüntünün x ve y eksenindeki türevlerini alarak noktaların yoğunluklarını belirliyoruz.

```
# Sobel operatörüyle kenarları tespit et
sobelx = cv2.Sobel(image, cv2.CV_64F, 1, 0, ksize=3) # X yönünde türev
sobely = cv2.Sobel(image, cv2.CV_64F, 0, 1, ksize=3) # Y yönünde türev
```

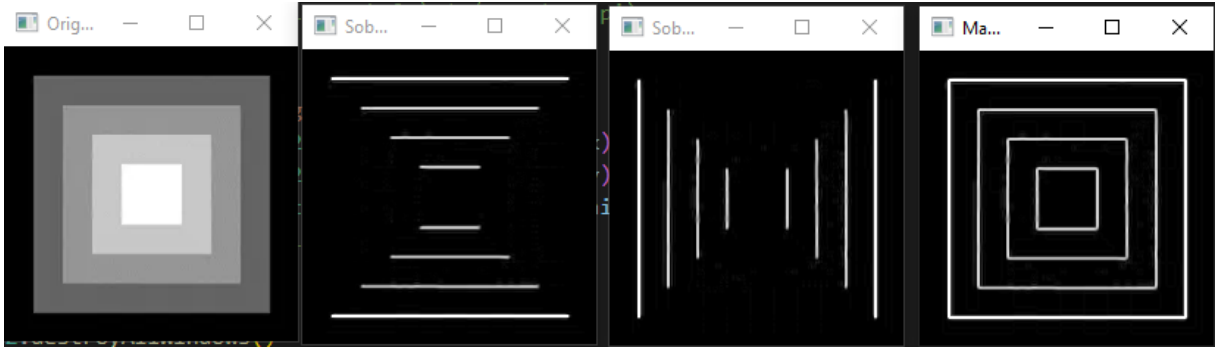
- 4) Kenarları tespit ediyoruz.

```
# Kenarların gücünü ve yönünü hesapla
magnitude = np.sqrt(sobelx**2 + sobely**2)
#angle = np.arctan2(sobely, sobelx) * (180 / np.pi)
```

- 5) Orijinal görüntü, x ve y eksenindeki kenarlar ve son olarak resmin tüm kenarlarını açılan pencerede gösteriyoruz.

```
# Görüntüyü göster
cv2.imshow('Original Image', image)
cv2.imshow('Sobel X', cv2.convertScaleAbs(sobelx))
cv2.imshow('Sobel Y', cv2.convertScaleAbs(sobely))
cv2.imshow('Magnitude', cv2.convertScaleAbs(magnitude))
#cv2.imshow('Angle', angle.astype(np.uint8))
```

Sonuç:



Kullanılan bu algoritma sayesinde yüklenen görüntünün kenarları net olarak tespit edilebilmiştir.

Robotik cerrahide de kullanılan bu algoritma çok daha karmaşık görüntüleri cerrahlar için algılayıp anlaşılması daha rahat hale getirmektedir.

B)Desen Eşleme Örneği:

Desen eşleme (pattern matching), bilgisayarla görme ve görüntü işleme alanlarında bir görüntüde belirli bir desenin veya modelin bulunmasını ve konumunun belirlenmesini amaçlayan bir tekniktir. Desen eşleme, genellikle endüstriyel otomasyon, robotik, yüz tanıma, nesne izleme ve medikal görüntüleme gibi çeşitli uygulamalarda kullanılır.

- 1) Projemize cv2 Kütüphanesi dahil ediyoruz.

```
import cv2
```

- 2) Görüntüyü renkli bir biçimde ekliyoruz. Eğer eklenemediyse hata mesajını koyuyoruz.

```
# Model görüntüsünü renkli olarak yükle
model_image = cv2.imread('2.jpg', cv2.IMREAD_COLOR)

# Model görüntüsü yoksa hata mesajı ver ve çık
if model_image is None:
    print("Model görüntüsü yüklenemedi!")
    exit()
```

- 3) Kamerayı açıyoruz ve görüntü yakalıyoruz.

```
# Kamera y Loading...
cap = cv2.VideoCapture(0) # 0: ilk kamera
```

- 4) Görüntü içindeki anahtar noktaları belirliyoruz.

```
# ORB (Oriented FAST and Rotated BRIEF) özellik dedektörü ve eşleştirici oluştur
orb = cv2.ORB_create()
matcher = cv2.BFMatcher(cv2.NORM_HAMMING, crossCheck=True)

# Model görüntüsündeki ORB özelliklerini bul
model_keypoints, model_descriptors = orb.detectAndCompute(model_image, None)
```

- 5) Sonsuz bir döngü içerisinde kamera üzerinden anahtar noktaları belirliyoruz.

```
while True:
    # Görüntü yakalaması yap
    ret, frame = cap.read()

    # Görüntüdeki ORB özelliklerini bul
    gray_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    frame_keypoints, frame_descriptors = orb.detectAndCompute(gray_frame, None)
```

- 6) Eşleşen özellikleri bulup bunları bir listeye ekliyoruz. Daha sonra en iyi eşleşmeleri yakalamak için onları sıralıyoruz.

```
# Eşleşen özellikleri bul
matches = matcher.match(model_descriptors, frame_descriptors)

# Listeye dönüştür
matches = list(matches)

# En iyi eşleşmeleri sırala
matches.sort(key=lambda x: x.distance)
```

- 7) En iyi N sayıdaki eşleşmeyi alıp çizgiler çekiyoruz.

```
# İlk N eşleşmeyi al (örneğin, en iyi 10 eşleşme)
top_matches = matches[:10]

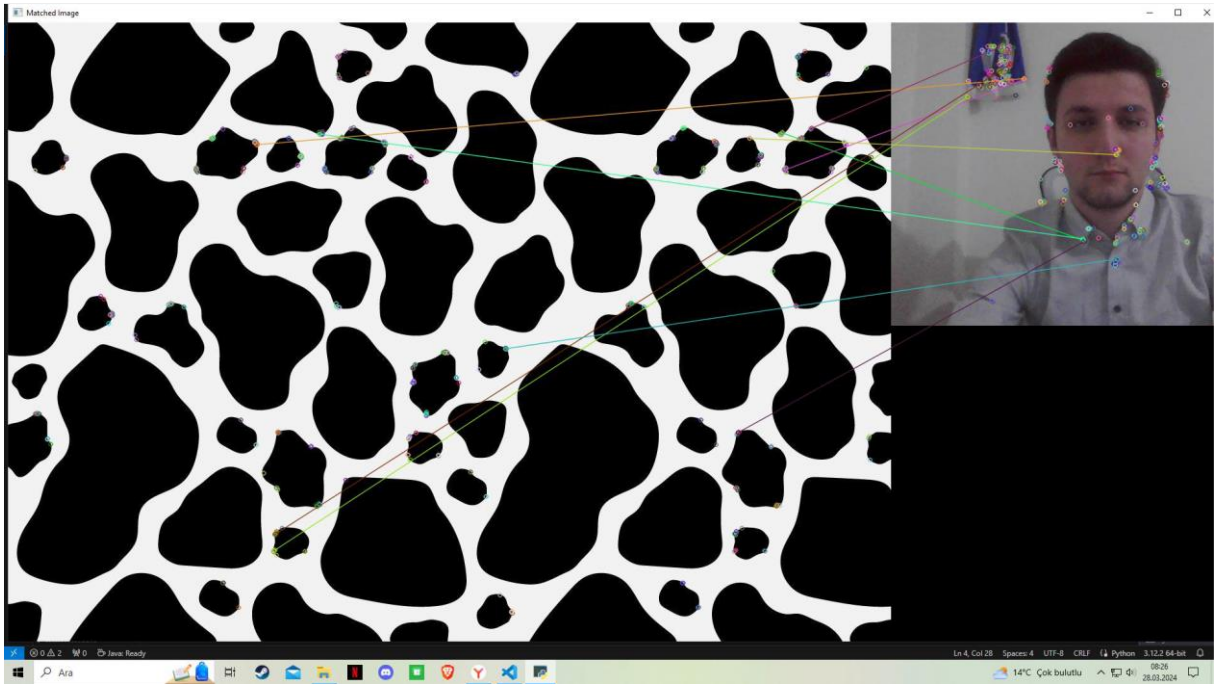
# Eşleşme çizgilerini çiz ve ekrana yazdır
matched_image = cv2.drawMatches(model_image, model_keypoints, frame, frame_keypoints, top_matches, None)
cv2.imshow('Matched Image', matched_image)
```

8) Çıkmak için klavyedeki 'q' tuşunu kullanıyoruz.

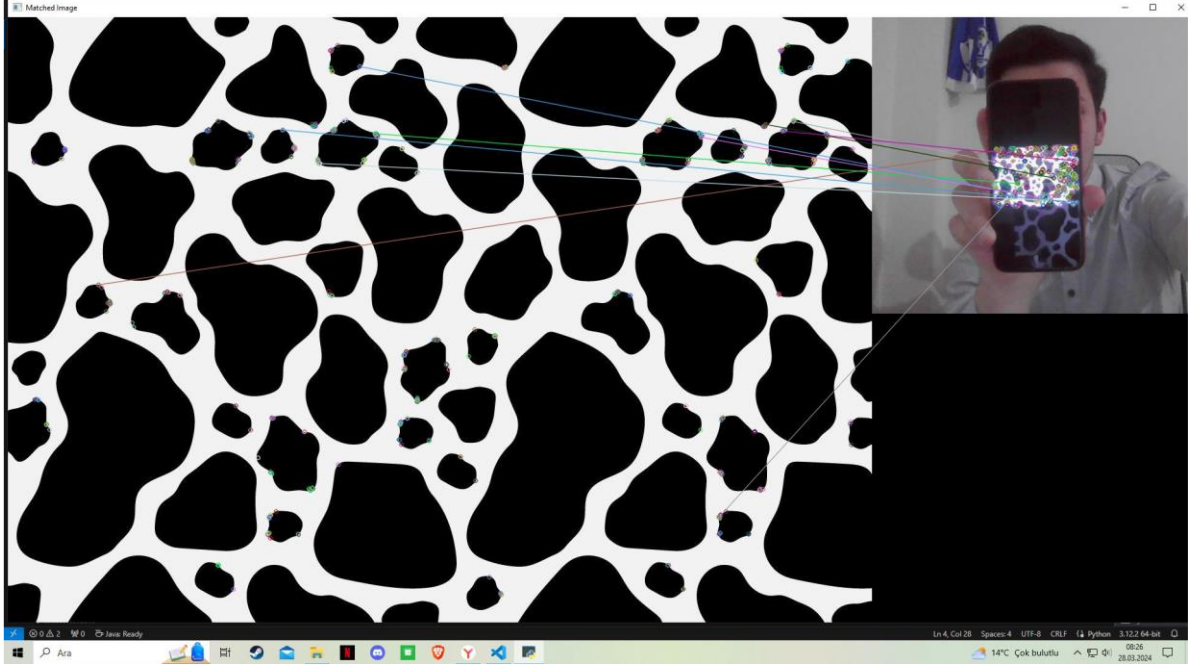
```
# Çıkış için 'q' tuşuna basılmasını bekle  
if cv2.waitKey(1) & 0xFF == ord('q'):  
    break
```

Sonuç:

İlk ekran görüntüsünde eşleşen öge olmadığı için çizgiler çok dağınık bir halde yayıldı.



İkinci ekran görüntüsünde ise desen tam olarak eşleşip oklar aynı noktada toplandı ve desen eşlemesi tamamlandı.



Kullanılan bu algoritma sayesinde ilk başta verilen görüntü devamında açılan kameradaki görüntüyle eşleştirilmeye çalışıldı. Kullanılan oklar sayesinde eşleşmenin başarılı olup olmadığı anlaşıldı.

Sağlık sektöründe de kullanılan bu algoritma sayesinde tümör, kanser gibi hastalıklı dokular görüntü işleme sayesinde çok daha net bir biçimde tespit edilebilmekte ve hastanın tedavisine çok daha hızlı bir şekilde başlanabilmektedir.

Projenin Github Linki

<https://github.com/kadiryildirim1745/openCv-Goruntu-Isleme>

Kaynakça

1. <https://www.acibadem.com.tr/acibadem-de/robotik-cerrahi/>
2. <https://www.davincicerrahisi.com/>
3. <https://medium.com/botsupply/a-beginners-guide-to-deep-learning-5ee814cf7706>
4. <https://chat.openai.com/>
5. <https://paradisetechsoftsolutions7.medium.com/basics-of-open-cv-open-cv-in-python-language-open-cv-images-e96a5d04fd2c>
6. <https://www.intuitive.com/en-us/products-and-services/davinci/learning/simnow>