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

Açıklama otomatik olarak oluşturuldu

Bu satır, edge fonksiyonu kullanılarak Canny kenar tespiti algoritması uygulanarak ikili bir görüntü elde eder. canny parametresi, Canny kenar tespiti algoritmasının kullanılmasını belirtir. [0.04 0.1] aralığındaki değerler, kenarları belirleme hassasiyetini kontrol eden eşik değerleridir. 4, kenar tespiti sırasında kullanılacak Gauss filtresinin boyutunu belirtir.

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

Açıklama otomatik olarak oluşturuldu

Bu satır, Hough dönüşümü uygular. Hough dönüşümü, doğru çizgilerin matematiksel temsilini elde etmeye yönelik bir tekniktir. H matrisi, Hough uzayındaki değerleri içerirken, theta ve rho ise dönüşüm parametrelerini temsil eder.

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

Açıklama otomatik olarak oluşturuldu

Bu satırlar, Hough dönüşümü sonucunda elde edilen matristeki tepe noktalarını belirler. Bu tepe noktalar, genellikle doğru çizgilerin temsil edildiği noktalardır.

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

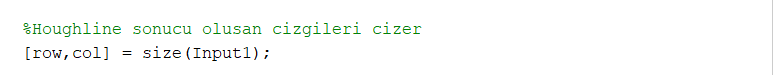
Açıklama otomatik olarak oluşturuldu

Bu satır, belirli bir boşluk ve uzunluktaki çizgileri çıkarır. FillGap parametresi, farklı çizgi parçalarını birleştirmek için kullanılır. MinLength parametresi ise kabul edilecek minimum çizgi uzunluğunu belirtir.

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

Açıklama otomatik olarak oluşturuldu

Bu döngü, LinesIm yapısındaki her çizgi segmentinin uç noktalarını alır (point1 ve point2), bunları ab ve cd adlı değişkenlere atar, sonra bu noktaları Input1 ve PointsGroup matrislerine ekler. Her çizgi segmenti için iki satır olmak üzere, Input1 matrisine iki noktayı birleştirirken, PointsGroup matrisine başlangıç ve bitiş noktalarını ve bu noktalara karşılık gelen çizgi segmentinin indislerini ekler.



Bu satır, Input1 matrisinin boyutlarını alır ve row ile satır sayısını, col ile sütun sayısını elde eder. Bu bilgiler daha sonra çizgi çizme işlemi sırasında kullanılabilir.

Bu işlemler genellikle Hough dönüşümü sonucu elde edilen çizgi segmentlerinin analizi veya görselleştirilmesi için kullanılır. Çizgi segmentlerinin uç noktalarını matrislere kaydetmek, daha sonra bu noktalar üzerinde çeşitli analizler yapmak veya çizimler gerçekleştirmek için kullanılabilir.

**Başlangıç ve bitiş noktalarını neye göre belirleriz ?**

**Başlangıç ve bitiş noktaları, Hough dönüşümü sonucunda tespit edilen çizgi segmentlerini tanımlayan iki uç noktadır. Bu noktalar, genellikle çizgi segmentinin görüntüdeki başlangıç ve bitiş noktalarını temsil eder. Çizgi segmentinin bu iki noktasını belirleme süreci, çizgi segmentinin başlangıç ve bitiş noktalarını bulma ve bu noktalara karşılık gelen matrislere eklemek adına önemlidir.**

**Belirli bir çizgi segmentinin başlangıç ve bitiş noktalarını belirlerken göz önünde bulundurmanız gereken bazı önemli faktörler şunlardır:**

**Hough Dönüşümü Parametreleri: Çizgi segmentlerini bulma süreci, Hough dönüşümü sırasında kullanılan parametrelere bağlıdır. Bu parametrelerin doğru bir şekilde ayarlanması, çizgi segmentlerinin doğru şekilde tespit edilmesine yardımcı olabilir.**

**Çizgi Segmentinin Uzunluğu ve Yönü: Belirli bir uygulamaya bağlı olarak, çizgi segmentinin uzunluğu ve yönü önemli olabilir. Örneğin, eğer uzun bir çizgi segmenti bulunuyorsa, başlangıç ve bitiş noktalarının çizgi segmentinin iki ucunu doğru bir şekilde temsil etmesi önemlidir.**

**Uygulama Bağlamı: Çizgi segmentlerinin başlangıç ve bitiş noktalarını belirlerken, uygulamanızın gereksinimlerini düşünmelisiniz. Hangi özelliklere odaklanmak istediğiniz, başlangıç ve bitiş noktalarını nasıl belirlediğinizde etkili olacaktır.**

**Görüntü Nitelikleri: Görüntü üzerindeki gürültü düzeyi, çizgi segmentlerinin düzgünlüğü ve diğer özellikler de başlangıç ve bitiş noktalarını belirleme sürecini etkileyebilir.**

**Ayrıca, başlangıç ve bitiş noktalarını belirlerken kullanılan algoritmanın doğruluğu, çizgi segmentlerinin doğru bir şekilde temsil edilmesi açısından kritiktir. Bu noktalarda bir hata, çizgi segmentinin doğru bir şekilde belirlenememesine ve uygulama sonuçlarını olumsuz etkileyebilir.**

**metin, yazı tipi, çizgi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Bu satır, DifferenceArray adlı bir matrisin ilk sütununu oluşturan pad adlı sıfır matrisini tanımlar.

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

Açıklama otomatik olarak oluşturuldu

Bu döngü, Input1 matrisindeki her noktanın diğer noktalara olan uzaklıklarını hesaplar. difference değişkenine, iki nokta arasındaki öklidyen uzaklık formülü kullanılarak hesaplanan uzaklık değeri atanır. Hesaplanan uzaklık değerleri, DifferenceArray adlı matrise sırasıyla eklenir. İkinci döngü içinde ColIndex, her bir satırdaki uzaklık değerlerini saklamak için kullanılır.

Bu satır, pad matrisini DifferenceArray matrisinin soluna ekler. Bu işlem, ilk sütunda sıfırlar içeren bir matris oluşturarak, daha sonra bu matrise uzaklık değerlerinin eklenebilmesini sağlar.

Sonuç olarak, DifferenceArray matrisi, Input1 matrisindeki noktalar arasındaki uzaklıkları içeren bir matristir. Bu tür bir matris, genellikle noktalar arasındaki mesafelerin analizinde veya benzeri uygulamalarda kullanılabilir.

**DifferenceArray adlı bir matrisin ilk sütununu oluşturan pad adlı sıfır matrisi neden yapılır?**

**Bu işlem, bir matrisin ilk sütununu belirli bir değerle (bu durumda sıfır) doldurmanın amaçlandığı durumları ele alır. pad adlı sıfır matrisi, daha sonra DifferenceArray matrisine eklenerek, her satırın ilk sütununa sıfırlardan oluşan bir sütun eklenir.**

**Bu tip bir işlemin amacı, genellikle matrisin belirli bir sütununu başlangıç değerleriyle başlatmak ve bu değerlere dayalı olarak diğer sütunların işlenmesini kolaylaştırmaktır. İlgili satırda, DifferenceArray matrisinin ilk sütununu pad matrisi ile başlatarak, bu matrisin diğer sütunlarında çeşitli değerleri depolayabilmek ve işleyebilmek için bir temel oluşturulur.**

**Bu durumda, pad matrisi zeros(row-1, 1) ifadesi ile tanımlanmış, yani ilk sütunu sıfırlardan oluşan bir sütun vektörüdür. Bu sütun vektörü, DifferenceArray matrisine eklenerek, her bir satırın ilk sütununa sıfırlar eklenmiş olur. Bu işlem genellikle matris işlemlerini daha düzenli hale getirmek, matrisin her bir satırındaki ilgili bilgilerin sütunlarda daha belirgin bir şekilde düzenlenmesini sağlamak amacıyla yapılır.**

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu iç içe geçmiş döngüler, DifferenceArray matrisindeki her bir elemanı kontrol eder. Eğer bir eleman belirli bir eşik değerinden küçükse ve sıfırdan farklıysa, o zaman bu noktanın benzer olduğunu düşünerek, SamePoint matrisinin ilgili elemanını 1'e (true) ayarlar. Aksi durumda, SamePoint matrisinin ilgili elemanını 0'a (false) ayarlar.

Bu işlem, noktalar arasındaki farkları belirli bir eşik değeriyle karşılaştırarak benzer noktaları belirlemek için kullanılır. Bu benzerlik değeri (70 olarak belirlenmiş) eşik değeri olarak adlandırılır ve kullanıcının uygulama bağlamına bağlı olarak ayarlanabilir. SamePoint matrisi, benzer noktaları belirten bir ikili matris olacaktır: 1, benzer noktayı; 0, benzer olmayan noktayı temsil eder.

**Kodun içinde belirtilen 70 değeri, benzer noktaları belirlemek için kullanılan bir eşik değeridir. Bu eşik değeri, iki nokta arasındaki farkların karşılaştırılması sırasında belirli bir sınırı ifade eder.**

**Eğer iki nokta arasındaki fark (örneğin, öklidyen uzaklık) bu eşik değerinden küçükse, bu noktalar "benzer" olarak kabul edilir ve SamePoint matrisindeki ilgili eleman 1 (true) olarak işaretlenir. Aksi durumda, yani fark eşik değerinden büyükse veya sıfıra eşitse, SamePoint matrisindeki ilgili eleman 0 (false) olarak işaretlenir.**

**Bu eşik değeri, uygulamanın gereksinimlerine ve veri özelliklerine bağlı olarak seçilmiş olabilir. Özellikle, iki nokta arasındaki farkların ne tür bir ölçü birimi veya metriği temsil ettiği, bu eşik değerinin seçilmesinde önemli bir faktördür. Eşik değeri seçimi, genellikle uygulama bağlamına, veri setine ve problemin doğasına göre belirlenir.**

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu iç içe geçmiş döngüler, SamePoint matrisini tarar ve değeri 1 (true) olan her bir eleman için, ilgili indis bilgilerini Input2 matrisine ekler. Her bir benzer noktanın indis bilgileri, DifferenceIndex\_1 ve DifferenceIndex\_2 değerleridir. k değişkeni, her bir benzer noktanın iki indis bilgisini sırasıyla eklemek için kullanılır.

Sonuç olarak, Input2 matrisi, benzer noktaların indis bilgilerini içeren bir matris olacaktır. Her bir satır, bir benzer noktanın indis bilgilerini içerir ve bu bilgiler sırasıyla DifferenceIndex\_1 ve DifferenceIndex\_2 değerleridir.

**K neden 2’şer artar?**

**İçteki döngünün her adımında k değişkeni ikişer adımlı olarak arttırılır. Bu, her bir benzer noktanın iki indis bilgisini sırasıyla Input2 matrisine eklerken, k'nın sürekli olarak 1, 3, 5, ... gibi tek sayı değerlerini almasını sağlar. Bu adım, DifferenceIndex\_1 değeri her değiştiğinde, k'nın sıfırlanarak tekrar ikişer adımlı artışa geçmesini sağlar.**

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

Açıklama otomatik olarak oluşturuldu**

Bu döngü, Temp değişkeni aracılığıyla B\_Input2 matrisinden fazla elemanları ve tekrarları kaldırmak için bir kontrol sağlar. Her iki döngü adımında, Temp'in tek veya çift olup olmadığına ve flag değişkeninin durumuna bağlı olarak sütunları siler.

Eğer Temp tekse (mod(Temp, 2) == 1) ve flag 0 ise, B\_Input2 matrisinden ilgili sütunu siler ve flag'i 1 yapar.

Eğer Temp çiftse (mod(Temp, 2) == 0) ve flag 1 ise, aynı işlemi tekrarlar ve flag'i 0 yapar.

Bu işlem, B\_Input2 matrisindeki fazla elemanları ve tekrarları kaldırmak için kullanılır. flag değişkeni, işlem sırasında hangi durumda olduğunu takip etmek için kullanılır. Bu tip bir işlem genellikle veriyi düzenlemek veya analiz etmek için yapılır ve matrisin içeriğini daha düzenli bir hale getirmeye yöneliktir.

**Tek veya çift durumuna neden bakarız?**

**Bu durum, Temp değişkeninin tek veya çift olup olmamasına dayalı olarak sütunları silme işlemi yapılmasını sağlamak içindir. Bu özel durum, Input2 matrisinde ikişerli benzer nokta çiftlerini temsil etmek amacıyla kullanılan bir düzenleme işlemine yöneliktir.**

**İki benzer nokta arasındaki ilişkiyi temsil etmek için her bir benzer nokta çiftini iki satıra yerleştirmiş olabilirsiniz. Bu durumda, her iki satır birbirine bağlıdır ve bir çifti temsil eder. İki satır arasındaki bağlantıyı sürdürebilmek için, her iki benzer nokta çifti için bir satırın başında veya sonunda ekstra bir sütun kullanılabilir.**

**Bu nedenle, tek veya çift durumuna bakılarak sütunları silme işlemi yapılır. mod(Temp, 2) ifadesi, Temp değişkeninin 2'ye bölümünden kalanı verir. Eğer bu değer 1 ise, Temp tek bir sayıdır; eğer 0 ise, Temp çift bir sayıdır.**

**Eğer Temp tekse ve flag 0 ise, bu durumda sütun silme işlemi yapılır ve flag 1 yapılır. Bu durum, ikişerli benzer nokta çiftlerinden birinci satırın durumu olabilir.**

**Eğer Temp çiftse ve flag 1 ise, bu durumda yine sütun silme işlemi yapılır ve flag 0 yapılır. Bu durum, ikişerli benzer nokta çiftlerinden ikinci satırın durumu olabilir.**

**Bu şekilde, ikişerli benzer nokta çiftlerini temsil etmek üzere düzenlenmiş bir matris oluşturulabilir ve bu matrisin içeriği daha okunabilir bir hale getirilebilir.**

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu iç içe geçmiş döngüler, B\_Input2 matrisini tarar ve her bir elemanı kontrol eder. Eğer bir eleman, belirli koşulları sağlıyorsa (bu koşullar b < ceil(row1/2) && b ~= DifferenceIndex\_1 && b ~= 0 olarak belirlenmiştir), o eleman DelNum matrisine eklenir. Bu koşullar genellikle ikişerli benzer nokta çiftleri için belirlenmiştir.

b < ceil(row1/2): b değeri, ikişerli benzer nokta çiftlerinin sayısından küçük olmalıdır.

b ~= DifferenceIndex\_1: b, o anki döngüdeki satır numarasına eşit olmamalıdır.

b ~= 0: b, 0 (sıfır) olmamalıdır.

Bu işlem sonucunda, DelNum matrisi, silinecek indis bilgilerini içeren bir vektördür.

metin, yazı tipi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu döngü, DelNum matrisindeki indisleri kullanarak B\_Input2\_2 matrisinden ilgili satırları çıkarır. DelIndex değişkeni, DelNum matrisindeki her bir indis için döner. Her bir adımda, DelNum(DelIndex) ile belirtilen satır, B\_Input2\_2 matrisinden çıkarılır.

DelRow değişkeni, çıkarılan satırların sayısını takip eder. Çünkü her satır çıkarıldığında, bir önceki çıkarılan satırdan sonra gelen satırların indisleri bir azalır. Bu nedenle, DelRow değişkeni, doğru indisleri belirlemede kullanılır.

Bu işlem, belirli indislerin bulunduğu satırları B\_Input2\_2 matrisinden çıkararak, matrisi günceller. Bu tür bir işlem genellikle gereksiz veya tekrarlayan verileri temizleme amacını taşır.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu satır, B\_Input2\_2 matrisindeki her bir benzer nokta çifti için karşılık gelen noktayı PointsGroup matrisinden bulur ve bu bilgiyi BuffFounder değişkeninde saklar. B\_Input2\_2(row\_i, col\_i) ile belirtilen değer, PointsGroup matrisindeki sütun indeksini temsil eder ve bu sütunun konumu B\_Input2Transpose\_2 matrisinde aranır. Bulunan konum, karşılık gelen noktanın sütununu temsil eder. Bu konumun sıfırdan farklı olması durumunda (BuffFounder ~= 0), bu değer B\_Input2\_3 matrisine eklenir.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu iç içe geçmiş döngüler, B\_Input2\_2 matrisindeki her bir benzer nokta çifti için karşılık gelen noktayı PointsGroup matrisinden bulur ve B\_Input2\_3 matrisine ekler.

B\_Input2\_2(row\_i, col\_i) ~= 0: Bu koşul, B\_Input2\_2 matrisindeki geçerli elemanın sıfır olmamasını kontrol eder.

BuffFounder Değişkeni: Benzer nokta çiftlerine karşılık gelen noktanın sütununu bulmak için kullanılır. floor((find(B\_Input2Transpose\_2 == PointsGroup(B\_Input2\_2(row\_i, col\_i), 3))) / col5 + 1) ifadesi, ilgili sütunun indisini bulmak için bir dizi işlem içerir.

BuffPointChosen: Karşılık gelen noktanın değerini temsil eder.

B\_Input2\_3(row\_i, col5 + 1 + col\_i) = BuffPointChosen: B\_Input2\_3 matrisine karşılık gelen noktanın değeri eklenir.

Bu işlem, benzer nokta çiftlerine karşılık gelen noktaları B\_Input2\_3 matrisine ekleyerek, bu bilgileri tutar ve işlemin sonunda bu matris güncellenmiş bir şekilde elde edilir.

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

Açıklama otomatik olarak oluşturuldu

Bu ifade, B\_Input2\_2 matrisindeki belirli bir benzer nokta çiftine ait indis değerini alır ve bu indise karşılık gelen PointsGroup matrisindeki üçüncü sütunu temsil eder. Bu sütun, benzer nokta çiftine karşılık gelen bir noktanın indisini içerir.

PointsGroup matrisinin üçüncü sütunu, benzer nokta çiftlerine karşılık gelen noktanın indisini temsil eder. Bu matriste her bir satır, bir benzer nokta çiftine karşılık gelir. Şu şekilde bir açıklama yapabiliriz:

PointsGroup matrisinin her bir satırı bir benzer nokta çiftini temsil eder.

Bir benzer nokta çiftine ait ilk iki sütun, çizgi segmentinin başlangıç ve bitiş noktalarını içerir.

Üçüncü sütun ise bu çizgi segmentine karşılık gelen bir noktanın indisini içerir.

Dolayısıyla, B\_Input2\_2 matrisindeki bir benzer nokta çiftine ait indis bilgisini almak için üçüncü sütunu kullanıyoruz. Bu indis, orijinal görüntüdeki çizgi segmentine karşılık gelen bir noktanın konumunu belirtir. Bu bilgi, çizgi segmentlerini birleştirmek için kullanılır.



Bu ifade, B\_Input2Transpose\_2 matrisinde, önceki adımda bulunan sütun indisine (PointsGroup(B\_Input2\_2(row\_i, col\_i), 3)) eşit olan değerlerin indekslerini bulur. Bu işlem, B\_Input2\_2 matrisindeki benzer nokta çiftine karşılık gelen noktanın sütununu B\_Input2Transpose\_2 matrisinde bulmaya yöneliktir.



Bu ifade, find fonksiyonunun sonucunu, col5 değerine bölerek indisi düzelten ve ardından floor fonksiyonu ile tam sayıya çeviren bir ifadedir. Bu adımlar, benzer nokta çiftine karşılık gelen noktanın sütununu temsil eden bir indis değerini bulmaya yöneliktir.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu iç içe geçmiş döngülerde, B\_Input2\_3 matrisindeki belirlenen noktalar arasında döngü yapılır. Her bir benzer nokta çiftine karşılık gelen noktaların koordinatları (Output1\_x, Output1\_y ve Output2\_x, Output2\_y) alınarak, bu noktalar arasına mavi renkte çizgiler (drawline fonksiyonu kullanılarak) çizilir.