

IEE YTU CAS MARİNE ÇK
YÜKSEK SEVİYELİ YAZILIM ÖDEV 3

AD SOYAD:Tolga AVCI

ÜNİVERSİTE:Yıldız Teknik Üniversitesi

BİRİM:Yüksek Seviyeli Yazılım

ÖDEV 3

THRESHOLDİNG NEDİR VE GÖRÜNTÜ İŞLEME ALANINDA NASIL KULLANILIR?

-Türkçe karşılığı eşikleme olan thresholding işlemi, görüntüdeki piksellerin gri seviye değerlerini bir eşik ile karşılaştırarak iki sınıfa ayırma işlemidir. Bunu yaparkenki amacımız görüntüyü basitleştirmek ve istediğimiz kısımları ön plana çıkarmaktır. Piksel değerleri eşikten büyük ise 255(beyaz) değerine, eşikten küçük ise 0(siyah) değerine eşitlenir.

-Eşikleme, görüntü işlemede bir çok alanda kullanılır. Bunlardan bazıları:

1-Nesne Tespiti: Görüntüdeki nesneleri arka plandan ayırmada kullanılır. Örneğin endüstride hatalı nesne tespitiinde kullanılır.

2-Belge ve Metin İşleme: Bize verilen ödeve benzer olarak taraması yapılan belge ve yazılarda metni arka plandan ayırmada kullanılır.

3-Tıbbi Görüntüleme: MR, CT veya röntgen görüntülerinde dokuları ön plana çıkarmak için eşikleme yapılır .Tıp alanında tümör, damar gibi yapıların daha iyi incelenmesini sağlar.

4-Biyometri ve Güvenlik: Yüz tanıma ve parmak izi sistemlerinde thresholding işlemi kullanılır.

5-Otonom Sistemler: Otonom sistemlerde, robotlarda çevre tanınması için görsel basitleştirmede kullanılır. Çizgi izleyen robotlarda yolun takibi thresholding işlemiyle gerçekleştirilir.

Kısaca thresholding işlemi görseli basitleştirerek birçok alanda kullanılır ve işte büyük bir kolaylık sağlar.

-Thresholding işlemi tek bir çeşit değildir. Birden fazla şekilde bu işlem yapılabilir. Temel olan birkaçından bahsetmek istersek:

1-Global Thresholding: Tüm görüntü için sadece bir tane eşik değeri seçilir. Eşik değerine göre pikseller 0 veya 255 değerine eşitlenir .Basit ve hızlı çalışmasına karşın ışık dengesi kötü olan görüntülerde iyi çalışmayabilir.

2-Adaptive Thresholding: Görüntü küçük parçalara bölünerek eşikleme işlemi uygulanır. Dolayısıyla ışık dengesi düzensiz olan fotoğraflarda verimli çalışır.

3-Otsu Thresholding: İstatiksel bir yöntem olarak piksel yoğunluğu dağılımını analiz ederek ona göre bir eşik değeri seçilir. Kullanıcı manuel olarak eşik değeri girmesine gerek kalmaz. Özellikle nesne ve arka plan kontrastının belirgin olduğu görsellerde etkilidir.

4-Renkli Thresholding: Gri seviye yerine renkli görsellerde kullanılır. HSV uzayında belirli bir renk aralığı seçilerek sadece o renkteki nesneleri çıkarmamızı sağlar. Renk tabanlı nesne tespitinde(denizaltı balık tespiti vb.) kullanılır.

EDGE DETECTION NEDİR, YÖNTEMLERİ VE NASIL KULLANILIR?

-Türkçede kenar tespiti olarak adlandırılan edge detection, ani parlaklık değişimlerini tespit etme işlemidir.Kenarları, nesnelerin sınırlarını, yapılarını ve şekillerini ortaya çıkarır.Görüntü sadeleşir ve önemli bilgiler korunur ve analiz daha da kolay hale gelir.

-Görüntünün üzerine çeşitli filtreler veya matematiksel türevler(gradyen) uygulanarak kenarlar açığa çıkartılır.Kullanılan adımların sırası genelde şu şekildedir:Görüntüyü griye çevrilir.Gürültü azaltmak için GaussianBlur vb. filtreler uygulanır.Sobel, Canny vb. kenar bulma algoritmaları

kullanılır.Sonuç olarak da kenarların siyah-beyaz çizgiler halinde belirtildiği yeni bir görsel oluşturulur.

-Edge detection metodunu kullanabileceğimiz çeşitli yöntemler vardır:

1-Sobel Operatörü: Görüntünün gradyanını(parlaklık değişim oranı) hesaplar.3x3 boyutunde Kernel kullanır.Bu Kernelin bir tanesi x yönünde bir tanesi y yönünde çalışır.Kenarın büyüklüğü bu iki yönün birleşmesiyle elde edilir.Basit ve hızlı bir şekilde temel kenar bilgisi vermesine karşın gürültüye karşı hassastır, çok net olmayan kenarlarda verdiği sonuçlar zayıf olabilir.

2-Prewitt Operatörü: Sobel operatörüne benzer fakat Kernel değerleri daha hafiftir.Aynı şekilde x ve y yönünde çalışır.Matematiksel olarak hafif olduğundan hızlı çalışır ve temel işlemlerde tercih edilir.

3-Robert Cross Operatörü: 2x2 boyutlu çok küçük Kernel kullanır. Diyagonal yönlerdeki değişimi tespit eder. Kenarların yerini çok hızlı tespit eder fakat gürültüye aşırı duyarlı olduğundan günümüzde tercih edilmez.

4-Laplacian Operatörü(İkinci Türev): Görüntünün ikinci türevini alarak parlaklık değişimlerini tespit eder. Yönü bağımsızdır, tüm yönlere aynı anda çalışır. Sonuç ince parlak çizgiler şeklinde temsil edilir. Gürültü çok arttığında sahte kenarlar çıkmasına sebep olabildiğinden genelde GaussianBlur ile beraber kullanılır.

5-Canny Edge Detection: Çok aşamalı bir algoritmadır ve diğer algoritmalara göre daha iyi sonuçlar elde etmemize olanak sağlar. GaussianBlur ile gürültü azaltılır. Sobel benzeri bir algoritma ile gradyan hesaplanır. Kalın kenar çizgileri inceltir ve sonuçta en güçlü piksel kalır. Güçlü kenarlar korunur ve ince kenarlar daha

sonra kontrol edilir. Zayıf kenarlar güçlü kenarlara bağlı olduğu takdirde korunur değilse silinir. Çok popüler olan doğru sonuç veren bir yöntemdir. Gürültüye dayanıklıdır ve net ince kenarlar üretir. Ama işlem yükü diğer algoritmalarından daha fazladır.

UYGULAMA 2 RAPOR

Öncelikle and görselini shape fonksiyonu ile inceledim 300x300 bir görsel olduğunu gördüm. Görsele baktığımda bir daire bir kare ve iki üçgen görünmekte. Görselde görüldüğü şekilde kare ve dairede herhangi bir kesişim boyanmamış direkt birleştirilmiş gibi. İçine oluşturulan üçgenlerde kesişimleri alınarak siyaha boyanmış gibi görünüyor. Görsele tekrardan baktığımızda dairenin kenarları tam görsel kenarlarına teğet yani çapı 300 piksel.

Setmousecallback kullanarak karenin kenarlarının dosyanın kenarlarına paralel ve 25 piksel uzaklığında olduğunu buldum. Aynı fonksiyonla üçgenlerin koordinatlarını da buldum. Sonra bu şekilleri çizdirdim. Ardından bitwise işlemlerine geçtim. Görsele baktığımızda hepsi birleştirilip ardından üçgenlerin kesiştiği kısım siyaha boyanmış görünüyor yani direkt kare ve daireyi birleştirip tümünden üçgenleri çıkartırsak görseli elde ederiz. Öncelikle çember ve kareyi U da OR ile birleştirdim. Sonra T de iki üçgeni birleştirdim. Ardından AND ile U ile not(T) nin kesişen kısımlarını aldığımızda görseli elde ettim.

KAYNAKÇA

<https://chatgpt.com>

https://tr.wikipedia.org/wiki/Görüntü_eşikleme

<https://medium.com/@emreozguruoglu/image-tresholding-görüntü-eşikleme-nedir-neden-kullanılır-b7594f67cb87>

https://blog-roboflow-com.translate.goog/edge-detection/?_x_tr_sl=en&_x_tr_tl=tr&_x_tr_hl=tr&_x_tr_pto=tc