



GEBZE TECHNICAL UNIVERSITY
ELECTRONIC ENGINEERING

ELM463
GÖRÜNTÜ İŞLEME

PROJE RAPORU

HAZIRLAYANLAR
ÖMER CEBECİ 171024007 ABDULLAH MEMİŞOĞLU 171024001

1. Giriş:

Savunma sanayi, medikal ve otomotiv gibi birçok sektörde nesne tespit algoritmaları ile geliştirilen yapay zeka örnekleri önemli yer tutmaktadır. Askeri alanda mayın tespit edilmesi [1], medikal alanda hastalıklı hücrelerin tespiti [2] ve otomotiv sektöründe yaya tespiti [3] yapay zeka örnekleri olarak gösterilebilir. Nesne tespitinde yüksek başarı oranı elde edilebilmesi için eğitim görüntüleri üzerinde arka planın yok edilmesi ve gürültü azaltma işlemlerinin, tespit edilmesi gereken nesneyi kaybetmeyecek şekilde yapılması gerekmektedir.

Bu projede veri setindeki görüntülerde bulunan ceviz, fındık ve leblebi nesnelerinin tespit edilmesi amaçlanmıştır. Literatür araştırması sonucunda, nesne tespiti için yüksek başarı hedefleyen projelerde makine öğrenmesi algoritması kullanıldığı gözlemlenmiştir. Makine öğrenmesi kullanılmadığı algoritmalarda test verisi için başarı oranı düşmektedir. Bu durum, farklı görüntülere karşı duyarlı algoritma geliştirmenin zor olmasından kaynaklanmaktadır. Yapay zeka kullanıldığı durumda sistem kendi kendini eğitebildiği için birçok farklı görselde istenilen nesnelerin tespiti sağlanabilmektedir.

2. Veri Seti ve Kurulan Algoritma ile Fonksiyonların Tanımlanması

2.1 Veri Seti:

Nesne tespiti için kullanılacak eğitim ve test verileri hazır kaynaktan alınmamıştır. Her bir grup için “basit”, “orta” ve “zor” olmak üzere üç farklı kategoriye ayrılarak fotoğraflar çekilmiştir. “Basit” kategoride bulunan görüntülerde, herhangi bir ışık problemi ve arka planları nesnelerin tespitini zorlaştıracı etmen bulundurmayacak şekilde çekilmiştir. “Orta” kategorisi, farklı açılardan çekilmiş ve arka planları nesnelerin tespitini zorlaştıracı etmen bulunduracak görüntülerden oluşmaktadır. “Zor” kategorisinde bulunan görüntüler ise ışık, gölge ve arka plan problemlerinin hepsini bulunduracak şekilde oluşturulmuştur. Oluşturulan veri setinde 5 eğitim 6 test verisi olmak üzere toplam 11 adet görüntü bulunmaktadır. Görüntüler dengeli bir şekilde; basit , orta ve zor, eğitim ve test verileri olarak ayrılmıştır. Görüntülerde tespit edilecek nesnelerin sayısı ile ilgili herhangi bir sınırlama yoktur. Fakat kullanılan öznetelikten dolayı her görüntüde en az bir adet ceviz bulunmaktadır.

2.2 Kurulan Algoritma ile Fonksiyonların Tanımlanması

Projede Python programlama dili kullanılmış ve görüntüler üzerinde işlem yapmak için birçok imkan sunan Open-CV kütüphanesinden yararlanılmıştır. Kurulan algoritma iki temel aşamadan oluşmaktadır:

2.2.1 Görüntünün Hazırlanması

Giriş görüntüsündeki arka plan, gölge ve ışık problemlerini çözerek nesne tespitinin kolay hale getirildiği aşamadır. Algoritmada ilk adım olarak, Figür-1’de verilen imread fonksiyonu ile görüntü okunmuştur [4].



```
cv2.imread(path, flag)
```

Figür-1

Görüntünün Bulanıklaştırılması

Bulanıklaştırma işlemi görüntüde bulunan gürültülerin azaltılması için yapılır. Open-CV kütüphanesi görüntü bulanıklaştırma için birçok farklı fonksiyon sunmaktadır [5]. Ortalama filtre, Gaussian filtresi, Ortanca filtre vb. Bu proje kapsamında BilateralFiltre ve Ortanca filtre kullanılmıştır. Okunan görüntü ilk olarak bilateral fonksiyonu ile bulanıklaştırılmıştır.

Görüntüde bulunan kenarları (edges) kaybetmeden gürültüleri azaltmak için Figür-2’de verilen bilateral filtre fonksiyonu kullanılmıştır. BilateralFilter fonksiyonu giriş parametreleri Tablo 1’de verilmiştir.

Parametreler	Açıklamalar
src	Giriş Görüntüsü
d	Filtreleme sırasında kullanılan her piksel komşusunun çapıdır. Kullanılan filtrenin büyüklüğünü belirler.
SigmaColor	Uygulanan her bir filtrede bulunan birbirlerinden uzak renklerin ne kadar karıştırılacağı hakkında bilgi verir.
sigmaSpace	Filtrenin nasıl işletileceği hakkında bir parametredir. Parametrenin büyük değeri alması, renkleri yeterince yakın olduğu sürece uzak piksellerin birbirlerini etkileyeceği anlamına gelir.

Tablo 1

*Genel kullanımda her iki sigma parametresi birbirleri ile aynı değeri alacak şekilde seçilir. Bu değerler 10 dan küçük ise filtrenin etkisi az, 150 den büyük ise filtrenin etkisi fazla olur.

*Filtrenin etkili bir şekilde çalışması için sigma değerleri 105 olarak verilmiştir. Filtre çapı ise eğitim verileri üzerinde yapılan çalışmalarda en iyi sonucu veren 35 değeri seçilmiştir.

cv.bilateralFilter(src, d, sigmaColor, sigmaSpace[, dst[, borderType]])

Figür-2

Eşikleme İşlemi:

Nesnelerin, arka plan ile ayrılmasını bu sayede tespitinin daha kolay yapılması amacıyla eşikleme işlemi gerçekleştirilmiştir. Eşikleme işlemi için Open-CV kütüphanesi iki temel fonksiyon sunmaktadır [6]. cv.threshold ve cv.adaptiveThreshold. cv.threshold fonksiyonu bütün bir görüntü için tek bir eşik değeri belirler. cv.adaptiveThreshold fonksiyonu ise görüntüde belli bölgelerde eşik değerleri seçerek eşikleme işlemini gerçekleştirir. Kullanılan görüntülerde ışık her bölgede aynı değilse cv.adaptiveThreshold fonksiyonunun kullanılması daha uygun olmaktadır. Bu proje kapsamında da görüntülerde ışık problemleri bulunması sebebi ile cv.adaptiveThreshold kullanılmıştır. Fonksiyon için giriş parametreleri Tablo 2 ‘de verilmiştir.

Parametreler	Açıklamalar
src	Giriş Görüntüsü
maxValue	Maksimum piksel değerini belirtir.Genel olarak 255 seçilir.
adaptiveMethod	Eşik değerlerin seçilmesinde iki yöntem kullanılır, ilgili blokta bulunan piksel değerlerin ortalaması (Mean) veya ağırlıklı gauss ortalaması (Gaussian) alınarak eşik değeri belirlenir.
thresholdType	thresholdType parametresi için iki seçenek mevcuttur; THRESH_BINARY ve THRESH_BINARY_INV. THRESH_BINARY piksel değeri eşik değerden büyük ise maksimum değeri alır, THRESH_BINARY_INV piksel değeri eşik değerden küçük ise maksimum değeri alır.
blockSize	Yerel eşik değeri hesaplamak için kullanılacak blokların, komşulukların, boyutu
C	C kat sayısı komşu piksellerin ortalama veya ağırlıklı toplamından çıkarılan bir sabittir.

Tablo 2

a

cv.adaptiveThreshold(src, maxValue, adaptiveMethod, thresholdType, blockSize, C)

Figür-3

ptiveMethod'u belirlemek için iki yöntem ile elde edilen sonuçlar karşılaştırılmıştır. Bunun sonucunda Gaussian yönteminin daha iyi sonuç verdiği gözlemlenmiştir. Eşikleme işlemi sonucunda belirli bir piksel değerinin altında kalan değerleri görüntüden silinmek istenmesi sebebiyle THRESH_BINARY yöntemi seçilmiştir. Blocksize ve C değeri ise eğitim verileri üzerinde en iyi sonucu verecek şekilde denemeler yapılarak ayarlanmıştır.

Morfolojik İşlemler

Veri seti içerisinde arka planı problemleri olan görüntülerde, eşikleme işlemi sonucunda istenilen başarı elde edilememiştir. Görüntülerde hâlâ arka planların etkisi ve gürültülerin var olduğu gözlemlenmiştir. Bu problemlerden kurtulmak için opening ve closing işlemleri sırasıyla yapılmıştır. Opening işlemi ile görüntüde arka plan yüzünden az belirgin bulunan nesneler ön plana çıkarılmıştır. Daha sonra closing işlemi uygulanarak gereksiz detaylar ve gürültüler görüntü üzerinden silinmiştir. Closing işlemi ile görüntü üzerinde kaybedilen kısımların tekrar elde edilememesi sebebiyle önce opening sonra closing işlemi gerçekleştirilmiştir. Opening ve closing morfolojik işlemlerinin gerçekleştirilmesi için yapılandırma elemanı seçilmesi gerekmektedir. Yapılandırma elemanı, tespit edilmek istenen nesneler daire şeklinde olduğu için elips seçilmiştir. Boyutu ise en ufak nesne olan leblebiyi bulabilecek şekilde (3,3) seçilmiştir. Kullanılan Open-CV fonksiyonu Figür-4'te verilmiştir.

cv.morphologyEx(src, op, kernel)

Figür-4

morphologyEx fonksiyonun giriş parametreleri Tablo 3'te verilmiştir.

Parametreler	Açıklamalar
src	Giriş Görüntüsü
op	Gerçekleştirilecek morfolojik işlem
kernel	Yapılandırma elemanı

Tablo 3

Open-CV'de erosion,dilation,opening,closing,morfolojik gradient,top hat ve black hat olmak üzere 7 adet morfolojik operatör mevcuttur [7]. Yukarıda bahsedilen sebeplerden dolayı bu proje kapsamında opening ve closing operatörleri kullanılmıştır. Open-CV de dikdörtgen, elips ve çapraz biçimli (haç şekli) Kernel, yapılandırma elemanları mevcuttur. Tespit edilmek istenen nesnelerin daire şeklinde olması sebebiyle kernel elips seçilmiştir.

Görüntü hazırlama başlığı altında yapılan son işlem medyan (ortanca) filtresi kullanılarak görüntü üzerinde kalan gürültülerin bastırılması olmuştur. Ortanca filtre, Merkez piksel ve filtre içerisinde kalan diğer piksel değerlerinden ortanca olanı alarak merkez piksel değerini günceller. Bu işlemi görüntüde ki her piksel için gerçekleştirir. Ortanca filtre için kullanılan Open-CV fonksiyonu Figür-5'te verilmiştir.

cv.medianBlur(src, ksize)

Figür-5

medianBlur fonksiyonun giriş parametreleri Tablo 4'te verilmiştir.

Parametreleri	Açıklamalar
src	Giriş Görüntüsü
ksize	Filtre Boyutu (tek ve >1 olmalı)

Tablo 4

Eğitim verileri üzerinde yapılan iyileştirme çalışmaları sırasında 11 sayısı gürültüleri en iyi bastıran değer olduğu gözlemlenmiştir. Bu yüzden filtre boyutu 11 seçilmiştir.

2.2.2 Nesne Tespiti

2.2.1 bölümünde görüntü üzerinde nesne tespiti için iyileştirmeler yapılmıştır. Bu bölümde ise 2.2.1 bölümünde elde edilen görüntüden yuvarlak şeklinde olan nesnelerin tespit edilmesi ve ceviz fındık ve leblebi sayılarının belirlenmesi gerçekleştirilmiştir. Nesne tespiti için “Hough Circle Transform” yöntemi kullanılmıştır. Bu yöntem ile görüntü üzerinde istenilen yarıçap büyüklüğünde daireler tespit edilir. Yarıçap uzunluğu değiştirilerek farklı büyüklüğe sahip daireler tespit edilebilir. Bu işlem için Open-CV’de kullanılan fonksiyon Figür-6’da verilmiştir[8].

cv.HoughCircles(image, method, dp, minDist, circles, param1, param2, minRadius, maxRadius)

Figür-6

HoughCircles fonksiyonun giriş parametreleri Tablo 5’te verilmiştir.

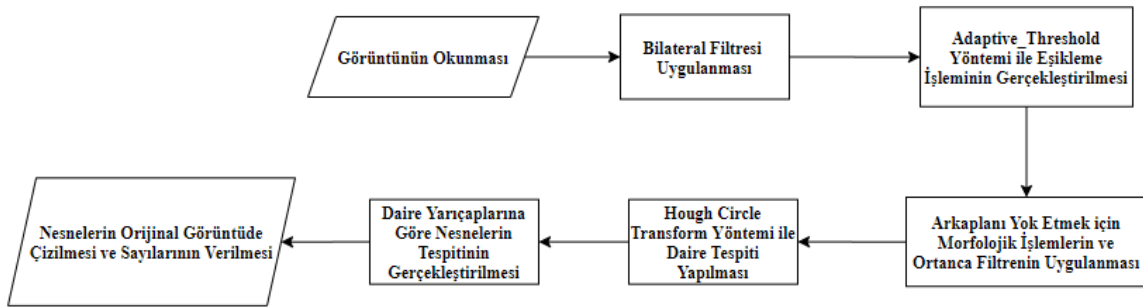
Parametreler	Açıklamalar
image	Giriş Görüntüsü
method	Tespit Yöntemi. HOUGH_GRADIENT ve HOUGH_GRADIENT_ALT olmak üzere iki adet tespit yöntemi mevcuttur.
dp	Accumulator çözünürlüğüdür. Eğer 1 olarak alınırsa giriş görüntüsü ile accumulator büyüklüğü aynı alınır.
minDist	Tespit edilen daireler arasındaki minimum mesafeyi gösterir. Bu değer çok küçük olması dar bir alanda çok fazla daire bulunmasına , çok büyük olması ise bazı dairelerin tespit edilememesine yol açar.
param1	Giriş görüntüsü için kullanılan eşik değeridir.
param2	Accumulator’un eşik değeridir. Çok küçük olması durumunda yanlış daire tespitleri yapılabilir.
minRadius	Tespit edilecek minimum daire yarıçap değeri
maxRadius	Tespit edilecek maksimum daire yarıçap değeri

Tablo 5

Yukarıda verilen parametreler eğitim verileri üzerinde en iyi sonucu verecek şekilde ayarlanmıştır. Yapılan denemelerde minimum yarıçap değeri yeterince küçük verilmemiş ise küçültülmüş. Gereksiz büyüklükte daireler bulunmuş ise maksimum yarıçap değeri küçültülmüştür. Param1 ve param2 değişkenleri ile eşik değerler değiştirilerek uygun yerlerde ki dairelerin tespit edilmesi sağlanmıştır. Literatür araştırması sonucunda yaygın olarak kullanılan metodun HOUGH_GRADIENT olduğu gözlemlenmiştir. Bu yüzden metot olarak “HOUGH_GRADIENT” seçilmiştir.

HoughCircles fonksiyonu için en uygun parametreler belirlendikten sonra görüntü üzerinde tespit edilen bütün dairelerin yarıçap ve koordinat bilgileri belirlenmiştir.

Algoritmada tespit edilen nesnenin ceviz, fındık veya leblebi olduğunun kararı yarıçap uzunluğuna göre verileceğinden, görüntülerde arka plan, ışık ve gürültü probleminin yanı sıra çekim mesafesi de sorun oluşturmaktadır. Eğitim verileri incelendiğinde çekim mesafeleri normal çekim, uzak çekim ve yakın çekim olmak üzere üç kategoriye ayrılmıştır: Normal çekim görüntülerde ceviz nesnesinin yarıçapı en az 60, uzak çekim görüntülerde en fazla 55 ve yakın çekim görüntülerde en az 140 olduğu gözlemlenmiştir. Görüntüde bulunan ceviz nesnesinin yarıçap uzunluğuna göre görüntü çekim mesafesinin hangi kategoride olduğuna karar verilip daha sonra nesnelerin etiketlenmesi işlemi gerçekleştirilmiştir. Nesne çeşitlerinin (ceviz, fındık, leblebi) sayıları ise her nesneye özel oluşturulan sayaçlar sayesinde belirlenmiştir. Belirlenen nesne ceviz olduğuna karar verildiğinde ceviz için belirlenen sayaç değeri bir artırılarak görüntüdeki ceviz sayısı tespit edilmiştir. Aynı şekilde tespit edilen nesne fındık veya leblebi olarak karar verildiğinde nesnelere özel olan sayaç değerleri artırılarak nesne sayıları belirlenmiştir. Algoritma için oluşturulan blok diyagram Figür-7’de verilmiştir.



Figür-7

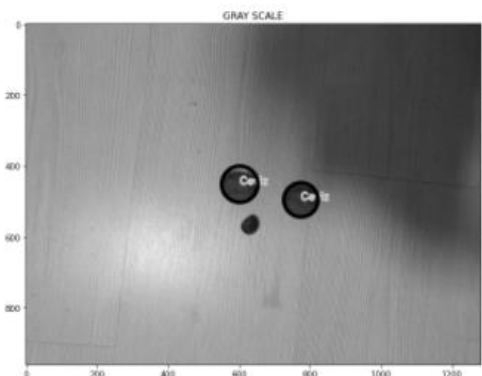
3 Eğitim Verilerinin Analizi

2. bölümde anlatılan fonksiyonlar ile algoritma geliştirilmiştir. Fonksiyonların parametreleri değiştirilerek eğitim verilerinde başarı oranı arttırılmaya çalışılmıştır. Algoritma Eğitim verileri ile geliştirildikten sonra başarısı test verileri üzerinde ölçülmüştür.

1.Eğitim Görüntüsü:

Train1 görüntüsü algoritmanın geliştirilmesi için üzerinde çalışılan ilk görüntüdür. Görüntüde çekim mesafesi problemi mevcuttur. Bölge bölge görüntü aydınlık veya karanlık olmuştur. Aynı zamanda arka planda bulunan düz çizgilerde görüntüde gürültüye sebep olmaktadır. Bahsedilen bu problemleri en aza indirmek için Figür-7’de verilen algoritma işletilmiştir.

Orijinal görüntünün bulanıklaştırılması ile eşikleme işlemini daha iyi yapılması sağlanmıştır (Train1_BilateralFilter). Eşikleme işleminden sonra nesneler arka plandan ayrılabilir arka plandan kaynaklı gürültüler hâlâ mevcuttur (Train1_ThresHold). Bu gürültüleri bastırmak ve arka plandan tamamen kurtulmak için opening ve closing işlemleri sırasıyla uygulanmıştır (Train1_Opening, Train1_Closing). Son adımda ortanca filtre uygulanarak görüntüde sadece istenilen nesneler kalmıştır (Train1_Median). Algoritmanın ikinci aşaması olan nesne tespiti işletildiğinde elde edilen sonuç Train1_Output görüntüsünde verilmiştir. Algoritma, görüntüde bulunan 3 nesneden 2’sini bulabilmiştir.



Görüntüde bulunan fındık nesnesinin bulunamamasının sebebi ortanca filtre sonucunda fındık nesnesinin diğer nesnelerde

Train1_Output

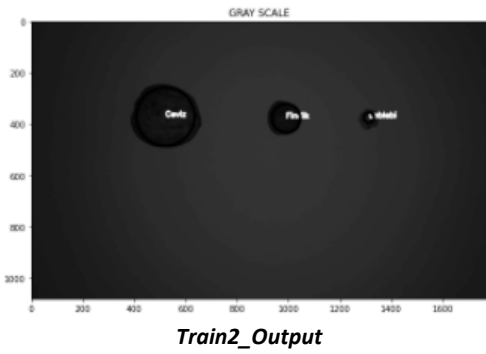
olduğu gibi etrafı çizilmiş tam bir daire oluşturamamış olması olarak gösterilebilir (Train1_Median).

Arka plan gürültüsü ve görüntü üzerinde bulunan ışık sorunları görüntü hazırlama aşamasında kullanılan fonksiyonlar ile çözülmüştür. Hough Circle Transform yönteminde parametre ayarlamaları ile en yüksek başarı oranı elde edilmiştir.

2. Eğitim Görüntüsü

Train2 basit kategorisine giren bir eğitim görüntüsüdür. Arka plan gürültüsüz beyaz olmasına rağmen karanlıktır.

Görüntünün arka planı nesnelerin tespitini zorlaştıracak biçimde değildir. Aynı zamanda görüntüde herhangi bir gürültü de yoktur. Bu yüzden eşikleme işlemi sonucunda tespit edilmek istenen nesneler elde edilmiştir (Train2_ThresHold). Görüntü de bahsedilen problemler bulunmadığı için ortanca filtre ile eşikleme işlemi sonucunda elde edilen görüntüler birbirleri ile benzer sonuçlar vermiştir (Train2_Opening, Train2_Closing, Train2_Median). Nesne tespiti gerçekleştirildiğinde elde edilen sonuç Train2_Output görüntüsünde verilmiştir.

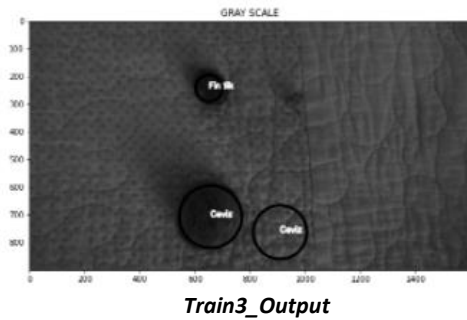


Basit kategoride bulunan Train2 görüntüsünde her nesne tespit edilmiştir.

3. Eğitim Görüntüsü:

Train3 Orta kategorisinde bulunan bir görüntüdür. Arka plan nesne şekilleri ile karışabilecek şekilde seçilmiştir. Train3 aynı zamanda karanlık bir görüntüdür.

Nesnelerin arka plandan ayrılma işleminin gerçekleştirilmesindeki en büyük problem arka planın nesneler ile karışabilecek motiflere sahip olmasıdır. Eşikleme işlemi yapıldığında nesneler ve arka planın karıştığı gözlemlenmiştir (Train3_ThresHold). Bu durum bir sonraki adım olan opening-closing işlemlerinin başarısını da azaltmıştır (Train3_Opening, Train3_Closing). Ortanca filtre ile arka plandan kaynaklı gürültüler azaltılmış olsa da nesnelerin şekillerinde kayıplar gözlemlenmiştir (Train3_Median). Daha iyi bir iyileştirme için eşikleme işleminde blok boyutları değiştirilse de başarılı bir sonuç elde edilememiştir. Yapılandırma elemanının boyutu arttırıldığı durumda ise arka plan ve gürültüler yok edilirken nesnelerin şekilleri de tamamen kaybedilmiştir. Nesne tespiti gerçekleştirildiğinde elde edilen sonuç Train3_Output görüntüsünde verilmiştir.

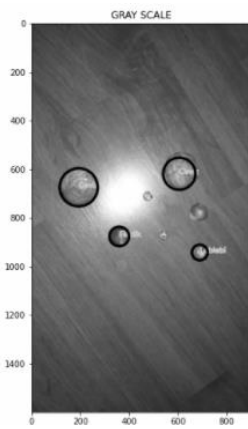


Algoritma görüntüde bulunan 3 nesneden 2'sini doğru bir şekilde tespit ederken, fazladan bir nesne daha tespit etmiştir. Bu fazladan tespit edilen nesne Train3_Median incelendiğinde arka planın tamamen atlamadığı ve gürültünün bulunduğu bölgede olduğu gözlemlenmiştir.

4. Eğitim Görüntüsü:

Train4 Zor kategorisinde bulunan bir görüntüdür. Görüntüde bulunan nesnelerin ortasında ışık problemi mevcuttur. Arka plan üzerinde bulunan ince çizgiler gürültüye sebep olurken arka plan ile nesnelerin piksel değerleri birbirleri ile yakındır. Aynı zamanda görüntü dikey açı ile çekilmemiştir

Görüntüye bilateral filtresi uygulandıktan sonra (Train4_BilateralFilter) yerel eşikleme gerçekleştirilmiştir. Yerel eşikleme sonucunda nesneler arka plandan ayırt edilebilecek duruma gelmiştir fakat arka planda bulunan çizgilerden kaynaklı gürültüler mevcuttur (Train4_ThresHold). Opening-closing işlemlerinin sırasıyla gerçekleştirilmesi ardından ortanca filtresinin uygulanması ile gürültüler büyük oranda azaltılmıştır (Train4_Opening, Train4_Closing, Train4_Median). Fakat eşikleme sonucunda elde edilen nesnelerin şekillerinde bozulmalar meydana geldiği gözlemlenmiştir. Nesne şekillerinin kaybedilmemesi için ortanca filtresinin boyutu azaltılmıştır fakat bu durumda gürültüler yeterince bastırılmadığı için nesne tespiti tam başarılı bir şekilde gerçekleştirilememiştir.



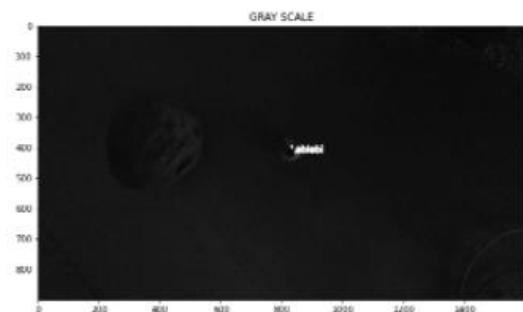
Train4_Output

Train4 görüntüsünde 7 nesneden 4'ü tespit edilirken 3'ü ise tespit edilememiştir. Bu nesnelerin tespit edilememelerinin sebebi yarıçaplarının HoughCircle fonksiyonunda verilen minimum yarıçap değerinden daha düşük olması olarak gözlemlenmiştir. Minimum yarıçap değeri azaltıldığı durumda ise yanlış yerlerde çok sayıda nesne tespiti gerçekleştiği görülmüştür. Tespit edilemeyen bir nesne (leblebi) ise görüntüde ışığa çok yakın bir konumda bulunmaktadır. Bu yüzden uygulanan morfolojik işlemler ve filtre sonucunda şeklini kaybetmiştir.

5. Eğitim Görüntüsü:

Train5 Zor kategorisinde bulunan bir görüntüdür. Arka plan karanlıktır. Aynı zamanda arka plan ile nesnelerin piksel değerleri birbirleri ile yakındır. Görüntünün sağ tarafında bulunan yarım daire şekli ise algoritmanın yanlış nesne tespiti yapmasına sebep olabilmektedir.

Train5 görüntüsüne yerel eşikleme işlemi uygulanması sonucu sol kısımda bulunan ceviz nesnesi kaybedilmeye başlanmıştır (Train5_ThresHold). Bunun sebebi arka plan ile ceviz nesnesinin piksel değerlerinin çok yakın olması olarak söylenebilir. Bu durum düzeltilmek için yerel eşikleme fonksiyonunda değişiklikler yapıldığında morfolojik işlemlerin başarısı azaldığı gözlemlenmiştir. Aynı zamanda diğer eğitim görüntülerinde de bozulmalar meydana geldiği görülmüştür. Bu yüzden eşikleme fonksiyonunda herhangi bir değişiklik yapılmamıştır. Daha sonra uygulanan filtre ve morfolojik işlemlerde de aynı durum gözlemlenmiştir (Train5_Opening, Train5_Closing). Ortanca filtre sonucunda sol kısımda bulunan ceviz nesnesinin tamamen kaybediliği görülmüştür (Train5_Median).



Train5_Output

Algoritmanın eğitim verilerini tespit etme başarısı Tablo 6’da verilmiştir.

Eğitim Görüntüsü	Doğru Tespit Sayısı	Yanlış Tespit Sayısı	Eksik Tespit Sayısı
Train1	2	0	1
Train2	3	0	0
Train3	2	1	1
Train4	4	0	3
Train5	1	0	1

Tablo 6

Tablo 6’da verilen “Yanlış Tespit Sayısı” sütunu, olmayan bir nesnenin veya var olan nesnenin çeşidinin yanlış tespit edilmesini ifade etmektedir.

4. Algoritmanın Test Edilmesi ve Sonuçlar

4.1 Algoritmanın Test Edilmesi

Eğitim verileri kullanılarak algoritmanın başarı oranı yükseltmek ve farklı özellikteki görüntülere karşı iyi sonuçlar vermesini sağlamak amaçlanmıştır. Test verileri yardımı ile de bu hedefin ne kadar gerçekleştiği ölçülmüştür. Test görüntülerinin her biri algoritma ile çalıştırılmış ve çıkan sonuçlar değerlendirilerek problemlerin sebepleri belirlenmiştir.

1. Test Görüntüsü

Test1 arka planı düz ve gürültü olmayan bir görüntüdür. Algoritmanın morfolojik işlemler sonucunda nesneleri tespit başarısı bundan dolayı yüksek olmuştur. Arka plan başarılı bir şekilde yok edildiği için nesne tespiti kısmında 3 nesne de başarılı bir şekilde tespit edilmiştir. Görüntünün arka planı ve nesnelerinin piksel değerleri birbirlerinden farklı olduğu için eşikleme işlemi sonucunda arka plan kolayca atılmıştır (Test1_ThresHold), bu yüzden morfolojik işlemler sonucu ile eşikleme işlemi sonucu benzer olduğu gözlemlenmiştir (Test1_Opening, Test1_Closing).

2. Test Görüntüsü

Test2 Orta kategorisinde bulunan bir görüntüdür. Görüntünün bazı bölgelerinde karanlık bazı bölgeleri ise aydınlıktır. Nesnelerin gölgeleri mevcuttur. Aynı zamanda arka planda bulunan kalın çizgiler nesne tespitini zorlaştıran etmenlerden biridir. Eşikleme işlemi yerel olarak yapılmasına rağmen karanlıkta kalan bölgelerdeki nesnelerin çoğunun kaybedildiği gözlemlenmiştir (Test2_ThresHold). Morfolojik işlemler ve ortanca filtre uygulanması sonucunda karanlık kısımda kalan nesneler tamamen arka plan veya gürültü Kabul edilerek yok edilmiştir (Test2_Opening, Test2_Closing, Test2_Median). Hough Circle Transform yöntemi iki adet nesne tespiti yapabilmıştır. Bu nesneler orijinal görüntüde aydınlık kısımda bulunan nesnelerdir (ceviz, leblebi).

3. Test Görüntüsü

Test3 basit kategorisinde bulunan bir görüntüdür. Arka planın pikselleri ile leblebi ve ceviz nesnelerinin piksellerinin değerleri yakındır. Bu durum arka planın düzgün bir şekilde atılmasını engelleyen etmenlerdendir. Eşikleme işlemi sonucunda arka plan neredeyse yok edilmiştir fakat nesnelerin etrafında gürültüler mevcuttur (Test3_ThresHold). Morfolojik işlemler ve ortanca filtre uygulamaları ile bu gürültüler de yok edilmiştir (Test3_Opening, Test3_Closing, Test3_Median). Nesne tespitinden önce elde edilen Median görüntüsü incelendiğinde 3 nesnenin de doğru bir şekilde tespiti hazır hale getirildiği gözlemlenir. Algoritmanın ikinci aşaması işletildiğinde 3 nesne de tespit edilmiştir. Fakat leblebi ve fındık nesneleri tam daire oluştururken ceviz nesnesi bazı bölgelerinde tam daire oluşturamamıştır bundan dolayı algoritma bu bozuk bölgelerde fazladan bir nesne (leblebi) tespiti gerçekleştirmiştir.

4. Test Görüntüsü

Test4 orta kategorisinde bulunan bir görüntüdür. Arka planda nesne tespitini zorlaştırabilecek şekiller (desenler) mevcuttur. Görüntü dikey açı ile çekilmemiştir. Nesnelerin gölgelerinin olmaması ise tespiti kolaylaştıran etmenlerdendir. Eşikleme İşlemi sonucunda nesneler doğru bir şekilde belirlenmiş fakat arka plan tam atılamadığı için gürültüler kalmıştır (Test4_ThresHold). Morfolojik İşlemler ve ortanca filtre uygulamaları ile gürültüler bastırılmıştır (Test4_Opening, Test4_Closing, Test4_Median). Görüntüde bulunan 2 leblebi nesnesi tespit edilememiştir. Bunun sebebi leblebi nesnelerinin yarıçap değeri, Hough Circle Transform ile belirlenen yarıçap değerinden küçük olmasıdır. Hough Circle Transform fonksiyonunda minimum yarıçap değeri azaltıldığı takdirde yanlış yerlerde nesne tespiti gerçekleştirildiği gözlemlenmiştir.

5. Test Görüntüsü

Test5 Zor kategorisinde bulunan bir görüntüdür. Arka planda bulunan bir çok çizgi gürültüye sebep olmaktadır. Nesnelerin birbirlerine olan mesafeleri az olması ve gölgelerinin bulunması nesne tespitini zorlaştırır. Aynı zamanda görüntünün çekilme açısı dik olmaması da tespiti zorlaştıran diğer etmenlerden biridir. Eşikleme İşlemi sonucunda nesnelerin birbirlerine yakın olması problem yaratmış, iki ceviz nesnesi tek bir nesneymiş gibi görünmeye başlamıştır (Test5_ThresHold). Arka plandan kaynaklı gürültüler morfolojik işlemler ve ortanca filtre uygulamaları ile belirli düzeyde bastırılrsa da görüntünün sol alt kısmında başarılı olunamamıştır. Bu durumda yanlış yerde nesne tespitinin yapılmasına yol açmıştır. Algoritma, nesne tespitinde kararını yarıçapa göre vermektedir. Çekim açısının farklı olması nesnelerin boyutları hakkında yanlış yorum yapılmasına sebep olmuştur. Bu yüzden aslında ceviz olan nesne fındık olarak tespit edilmiştir.

6. Test Görüntüsü

Test6 Zor kategorisinde bulunan bir görüntüdür. Arka planda gürültüye sebep olacak farklı piksel değerlerine sahip çizgiler mevcuttur. Aynı zamanda görüntüde ışık problemi de mevcuttur, bazı bölgeler karanlık bazı bölgeler aydınlıktır. Eşikleme İşlemi sonucunda nesneler başarılı bir şekilde belirlenmiş fakat çok fazla gürültü oluşmuştur (Test6_ThresHold). Bu gürültüleri bastırmak için yapılan morfolojik işlemler ve filtre uygulamaları sonucunda arka plan yok edilmiş gürültüler bastırılmıştır fakat çoğu nesnenin şekli de kaybedilmiştir (Test6_Opening, Test6_Closing, Test6_Median). Bundan dolayı sadece bir nesne doğru bir şekilde tespit edilebilmiştir.

Algoritmanın test verilerini tespit etme başarısı Tablo 7’de verilmiştir.

Test Görüntüsü	Doğru Tespit Sayısı	Yanlış Tespit Sayısı	Eksik Tespit Sayısı
Test1	3	0	0
Test2	2	0	5
Test3	3	1	0
Test4	5	0	2
Test5	2	3	5
Test6	1	1	5

Tablo 7

Tablo 7’de verilen “Yanlış Tespit Sayısı” sütunu, olmayan bir nesnenin veya var olan nesnenin çeşidinin yanlış tespit edilmesini ifade etmektedir.

4.2 Sonuçlar

Algoritmanın, eğitim verileri üzerinde 5 görüntüden 1'inde bütün nesneleri tespit etmesiyle başarı yüzdesi %20.00 olarak hesaplanmıştır, test verileri üzerinde 6 görüntüden 1'inde bütün nesneleri tespit etmesiyle başarı yüzdesi %16.66 olarak hesaplanmıştır. Başarı yüzdeleri hesaplanırken yanlış nesne tespiti yapılmayan fakat eksik nesne tespiti yapılan görüntüler de başarısız olarak kabul edilmiştir. Elde edilen bu değerler ile algoritmanın başarılı bir şekilde çalışmadığı sonucuna varılmıştır. Oluşturulan veri setindeki görüntülerde gölge ve gürültü sorunu başarıyı düşüren en önemli sebeplerden olmuşlardır. Nesnelerin gölgesinden kurtulmak için Top Hat, Black Hat gibi Morfolojik İşlemler , farklı yumuşatma filtreleri ve farklı kenar bulma algoritmaları (Canny) denenmiş fakat istenilen sonuç elde edilememiştir. Gürültü bastırma konusunda algoritma iyi çalışmıştır. Fakat arka planın ve nesnelerin piksel değerlerinin yakın olduğu görüntülerde nesneler de gürültüymüş gibi algılanarak yok edilmiştir. Başarı yüzdesini düşüren önemli etmenlerden bir diğeri de çekim mesafesidir. Nesne tespitinde karar verme mekanizması yarıçap özneliğine bağlı olarak karar vermektedir fakat uzaktan çekilen ve yakından çekilen görüntülerde elde edilen yarıçap değerleri birbirlerinden çok farklı çıkması sebebiyle yanlış karar verme şansı artmıştır. Çekim açısının değişmesi de kararın yanlış verilmesinde bir etmendir. Çekim mesafesi ve açısından kaynaklı bu problem çözmek için karar verme mekanizması yakından, uzaktan ve normal çekilen görüntülere göre 3 farklı şekilde davranış göstermektedir. Nesnelerin tespiti için kullanılan referans yarıçap değerleri çekim mesafelerine göre değiştirilmiştir.

5 Genel Yorumlar

Proje kapsamında ilk olarak eğitim verisinde bulunan görüntülerde başarılı bir şekilde çalışacak algoritma oluşturmak amaçlanmıştır. Eşikleme, Morfolojik Operatörler, Daire Tespiti, Filtreleme, Öznitelik Çıkartımı (Yarıçap) özellikleri kullanılmıştır. Algoritma iki kısımda kurulmuştur. Birinci kısımda arka plan atılması ve gürültü azaltma işlemleri yapılarak görüntü nesnelerin tespit edilmesi için hazır hale getirilmiştir. İkinci kısımda ise yarıçap uzunluğu öznitelik olarak Kabul edilerek ceviz leblebi ve fındık nesnelerinin tespit edilme işlemi gerçekleştirilmiştir. Eğitim verisinde bulunan görüntülerde ki gölge, gürültü, kontrast ve çekim mesafesi gibi problemleri en aza indirmek için fonksiyonların parametreleri ayarlanmıştır.

Eğitim ve test verilerindeki tam olarak bütün nesnelerin tespit edildiği görüntü sayısı azdır. Fakat görüntüler tek tek incelendiğinde nesnelerin yanlış olarak değil eksik tespit edildiği gözlemlenmiştir. Algoritmanın, bütün veri dikkate alındığında başarı yüzdesi düşük olmasına rağmen, her görüntü ayrı ayrı incelendiği takdirde başarı yüzdesi artmıştır.

Kaynaklar

[1] Venkataraman Padmaja, V. Rajendran, P. Vijayalakshmi, “Study on metal mine detection from underwater sonar images using data mining and machine learning techniques”

<https://doi.org/10.1007/s12652-020-01958-4>

[2] Nikos Petrellis, “A Smart Phone Image Processing Application for Plant Disease Diagnosis”

<https://doi.org/10.1109/MOCAS.2017.7937683>

[3] Constantine Papageorgiou, Theodoros Evgeniou, Tomaso Poggio, “A Trainable Pedestrian Detection System”

https://www.researchgate.net/profile/Tomaso-Poggio-2/publication/2467374_A_Trainable_Pedestrian_Detection_System/links/0fcfd50e6dd305e726000000/A-Trainable-Pedestrian-Detection-System.pdf

- [4] https://docs.opencv.org/3.4/d4/da8/group_imgcodecs.html
- [5] https://docs.opencv.org/4.x/d4/d13/tutorial_py_filtering.html
- [6] https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html
- [7] https://docs.opencv.org/3.4/d9/d61/tutorial_py_morphological_ops.html
- [8] https://docs.opencv.org/3.4/d4/d70/tutorial_hough_circle.html