

# ELM463– DÖNEM PROJESİ

## BİTKİ YAPRAKLARINDA HASTALIK TESPİTİ

Merve Tutar  
1901022050  
m.tutar2019@gtu.edu.tr

### ABSTRACT (ÖZET)

Bitki yapraklarındaki hastalıklar, nüfus artışına bağlı olarak gıda talebinin artmasıyla birlikte tarım sektörünün temel bir sorununu oluşturur. Bu hastalıkların erken ve doğru bir şekilde tespit edilememesi, bitkiler üzerinde ciddi etkiler yaratır ve ürün kalitesini, miktarını ve verimini olumsuz yönde etkiler. Geliştirilen görüntü işleme teknikleri ile bitki yapraklarındaki hastalıkların otomatik olarak tespit edilmesi, tarım verimliliğini artırarak gıda güvenliği ve ekonomik sürdürülebilirlik açısından önemli bir rol oynar.

### ANAHTAR KELİMELELER

Hastalık Tespiti, Görüntü işleme, Gürültü Azaltma, Kenar Tespiti, Segmentasyon

## 1. Giriş

Bitki yapraklarındaki hastalıkların görüntü işleme ile tespiti, tarım sektöründe önemli avantajlara sahiptir. Tarlalardaki bitki hastalıkları otomatik olarak tespit edilebilir. Geleneksel yöntemlerle hastalık tespiti genellikle uzun süreli ve işgücü yoğun bir süreçtir. Erken tespit sayesinde hızlı müdahale mümkün olur. Bu yöntem, iş gücü ve maliyetleri azaltırken, sürdürülebilir tarım uygulamalarının yaygınlaşmasına katkıda bulunarak gıda güvenliğini artırır ve insan sağlığına olumlu etkiler sunar. Görüntü işleme ile bitki hastalıklarının tespiti, kimyasal mücadele yöntemlerine olan ihtiyacı azaltabilir. Bu da sürdürülebilir tarım uygulamalarının benimsenmesini teşvik eder, çevresel etkileri azaltır ve ekosistem dengesini korur. Erken tespit ve hızlı müdahale, bitki hastalıkları nedeniyle oluşabilecek ürün kayıplarını minimize eder. Bu, daha sağlıklı ve güvenli tarım ürünleri elde edilmesine katkıda bulunur, gıda güvenliği standartlarını artırır.

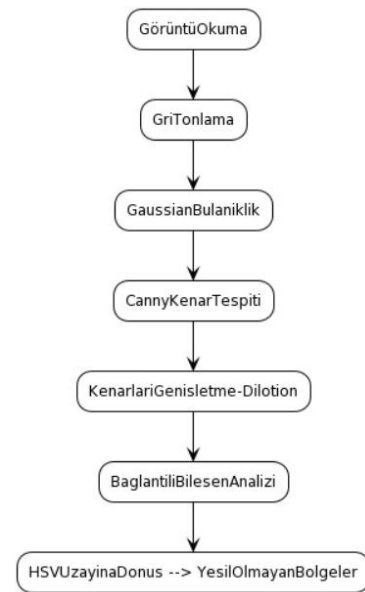
## 2. Deneyler ve Analiz

Bilgisayar ortamında bitki yapraklarında hastalık tespiti için, Python Programlama Dili'nin bir aracı olan Jupyter Notebook kullanılmıştır. Tespitin yapılabilmesi için, .png,

.jpg, .jpeg formatındaki görüntüler Jupyter Notebook ortamına eklenmiştir.

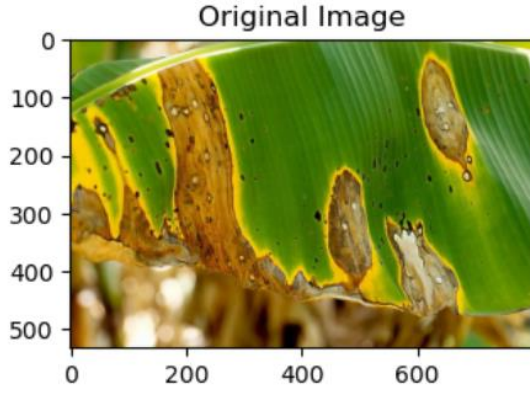
Görüntü işleme algoritması, sayısal görüntü işleme tekniklerini içeren bir fonksiyon içermektedir. İlk olarak, verilen bir görüntü dosyası OpenCV kütüphanesi kullanılarak okunmuş gri tonlamaya çevirilip ardından Gauss filtresi ve Canny kenar dedektörü uygulanmıştır. Daha sonra, kenarları birbirine bağlamak için genişletme işlemi (dilation) gerçekleştirilmiştir. Bağlantılı bileşen analizi kullanılarak görüntüdeki bağlantılı bileşenler tespit edilmiştir ve en büyük bileşen (genellikle yaprak) belirlenmiştir. Belirlenen bileşen maskeleme işlemiyle orijinal görüntüden izole edilmiş ve HSV renk uzayına dönüştürülmüştür. Bu işlemin ardından, yeşil renk aralığına uygun bir maske oluşturulmuş ve bu maske tersine çevirilerek yeşil olmayan bölgeleri vurgulanmıştır. Bu bölgeler hastalıklı kısımlar olarak tespit edilmiştir. Sonuçları görselleştirmek için matplotlib kütüphanesi kullanılarak işlemin her adımı yan yana gösterilmiştir.

Algoritmanın akış şeması Şekil 1.'de bulunmaktadır.



Şekil 1. Akış Şeması

- **Algoritma aşamaları**



Şekil 2. Orjinal "Leaf-spot-diseases.jpg" görüntüsü

Görüntü, renkli formatından gri tonlamaya dönüştürülmekte ve ardından 5x5 boyutunda bir Gauss filtresi ile işlenmektedir. Bu adım, görüntüdeki potansiyel gürültüyü azaltmaya yardımcı olur.

Gauss filtresi uygulanmış görüntü üzerine Canny kenar dedektörü uygulanarak belirgin kenarlar tespit edilmektedir.

Canny kenar tespiti sonuçlarını birleştirmek ve bağlamak amacıyla 2 iterasyon boyunca kenarlar genişletilir (dilation).

Bağlantılı Bileşen Analizi genişletilmiş kenarları kullanarak bağlantılı bileşenleri analiz eder, farklı bileşenler etiketlenir. En büyük bağlantılı bileşen seçilir ve buna ait maske oluşturulur.

Orijinal görüntüye bu maske uygulanarak sadece en büyük bileşeni içeren bölge elde edilir. İşlenmiş görüntü HSV renk uzayına dönüştürülür.

Hue (Renk Tonu): 30 ila 90 arasındaki değerler, yeşil renk tonunu kapsar. Renk döngüsü üzerindeki bu değerler, yeşil tonları temsil eder.

Saturation (Doygunluk): 40 ila 255 arasındaki değerler, renklerin canlılığını ifade eder. Bu aralık geniş bir doygunluk aralığını temsil eder.

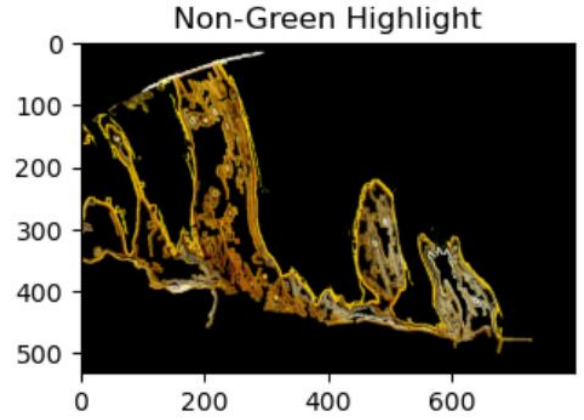
Value (Değer): 40 ila 255 arasındaki değerler, renklerin parlaklığını ifade eder. Bu aralık renklerin ne kadar parlak olduğunu kontrol eder.

HSV uzayında belirlenen yeşil renk aralığına uygun bir maske oluşturulur, bu maske tersine çevrilir ve işlenmiş görüntüye uygulanarak yeşil olmayan bölgeler vurgulanır.

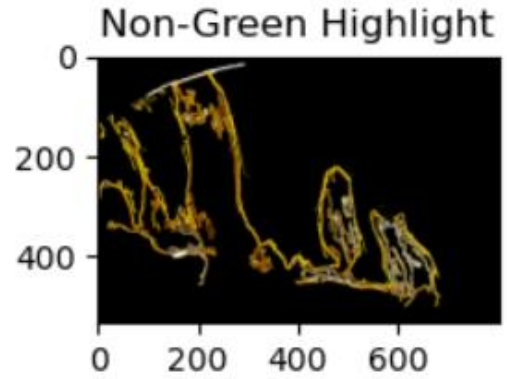
#### Geliştirme Aşamaları:

Bu algoritmayı geliştirirken farklı yöntemler denenip en son bu algoritmaya karar verilmiştir. Örneğin Şekil 2.'deki görüntüyü bulanıklaştırmak için 7x7 Gauss

uygulandığında görüntünün son halinde aşağıdaki gibi birçok detay kaybolmuştur. Bu sebeple 5x5 filtre denenmiş ve daha başarılı olmuştur.

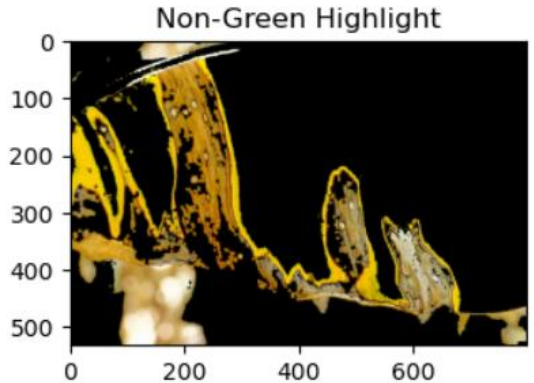


Şekil 3. 5x5 gauss sonucu hastalık tespiti



Şekil 4. 7x7 gauss sonucu hastalık tespiti

Canny kenar tespitinden önce çeşitli threshold yöntemleri denenmiştir. Ancak bu yöntemler bağlantılı bileşen analizi ile yaprağı tespit etmekte yeterince başarılı olamamıştır. Aşağıdaki görüntü Otsu's threshold yöntemi kullanılarak elde edilmiş olan son hastalık tespiti görüntüsünü vermektedir.



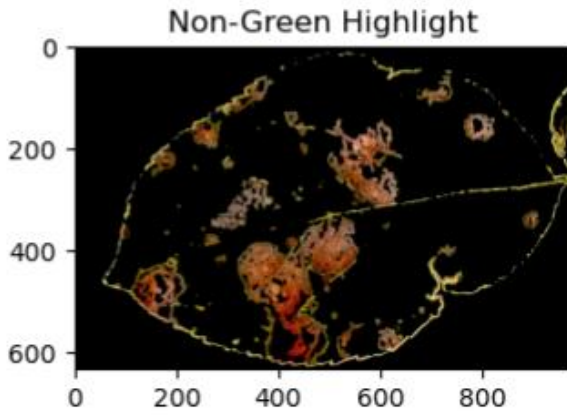
Şekil 5. Otsu's threshold sonucu hastalık tespiti

Görüntü sol tarafta arka planı da hastalık olarak almış ve başarısız olmuştur. Birçok görüntüde bu gibi sonuçlar verdiği için threshold yerine yaprağın daha iyi tespiti için canny kenar tespiti kullanılmasına karar verilmiştir.

HSV renk uzayı için yeşil renk aralığı aşağıdaki gibi belirlenmiştir.

```
lower_green = np.array([30, 40, 40])  
upper_green = np.array([90, 255, 255])
```

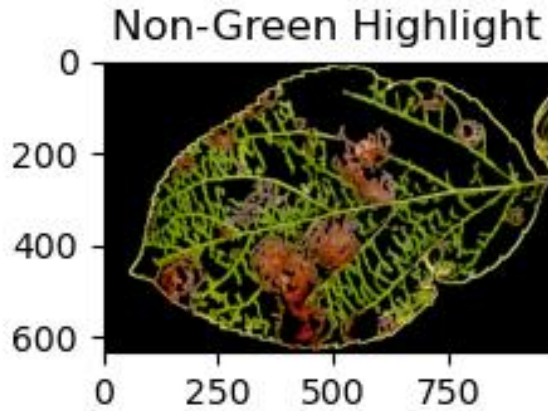
Bu geniş aralık, bir dizi yeşil tonunu içerir ve genellikle bitki yapraklarının renk tonunu kapsayacak şekilde seçilmiştir. Kod dosyasındaki "frogeye.jpg" görüntüsünde bu aralık kullanılarak elde edilen sonuç görüntüsü Şekil 6.'daki gibidir.



Şekil 6. Geniş HSV aralığı

Bundan önce aşağıdaki gibi daha dar bir yeşil renk aralığı seçilmiştir. İlk renk aralığına kıyasla daha dar bir aralık temsil eder. Bu, belirli bir yeşil tonunu daha spesifik bir şekilde tanımlamak amacıyla seçilmiş olabilir.

```
lower_green = np.array([40, 50, 50])  
upper_green = np.array([80, 255, 255])
```



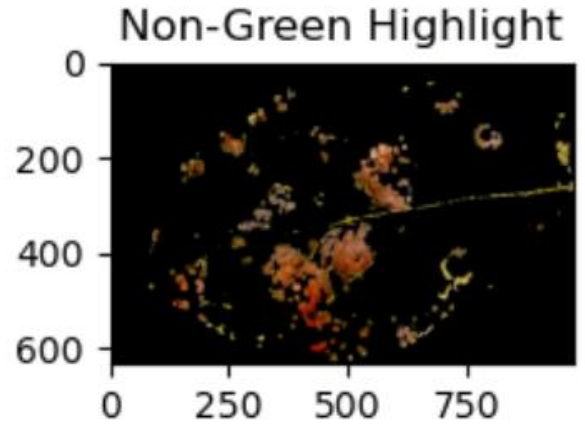
Şekil 7. Dar HSV aralığı

Ancak her görüntüde yeşil yaprakların tonu aynı olmadığı için biraz daha geniş bir aralık seçilmesine karar verilmiştir. Şekil 6.'nın Şekil 7.'ye göre daha iyi bir sonuç verdiği gözlemlenmiştir.

Son elde edilen non-green highlight görüntülerinin bazılarında Şekil 6.'daki gibi yaprak damarları da hastalık olarak tespit edilmiştir.

Opening (Açma) ve Closing (Kapama), morfolojik işlemler olarak bilinen görüntü işleme teknikleridir. Bu işlemler, görüntü üzerindeki küçük detayları gidermek veya nesneleri birleştirmek amacıyla kullanılır.

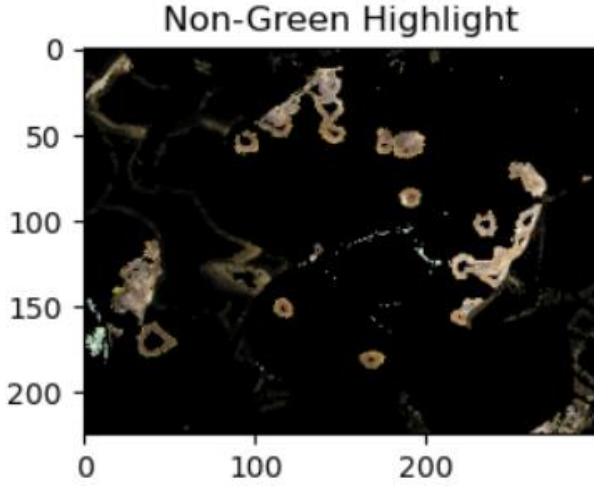
Bu damarlardan kurtulmak için opening ve closing gibi morfolojik operatörlerin kullanılması denenmiştir ancak bazı görüntülerde hastalıklı kısımlar çok küçük olduğu için bu yöntemler kullanıldığında Şekil 8.'deki gibi hastalıklı kısımlarda veri kaybedilmiştir. Ya da görüntüde bileşenler fazla yayıldığı için closing hastalığa ait olmayan bileşenlerin hastalık gibi tespit edilmesine sebep olmuştur. Bu sebeple bu operatörlerin kullanılmamasına karar verilmiştir.



Şekil 8. Opening uygulanmış olan görüntü

Algoritmanın görüntülerde verdiği çıktılar aşağıdaki gibidir:

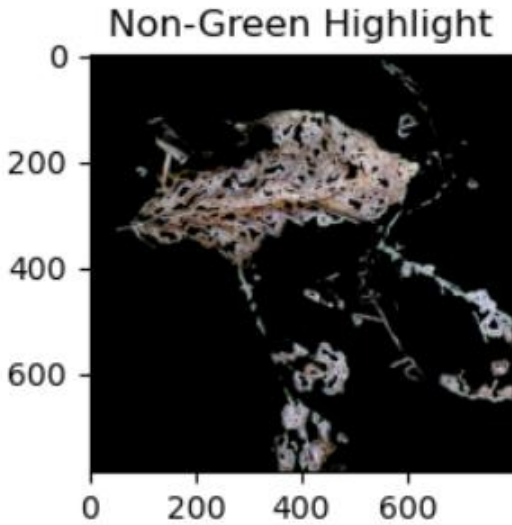
- "bacterial.png" görüntüsü için



Şekil 9. "bacterial.png" hastalıkları

Hastalıkların başarılı bir şekilde tespit edildiği görülmektedir.

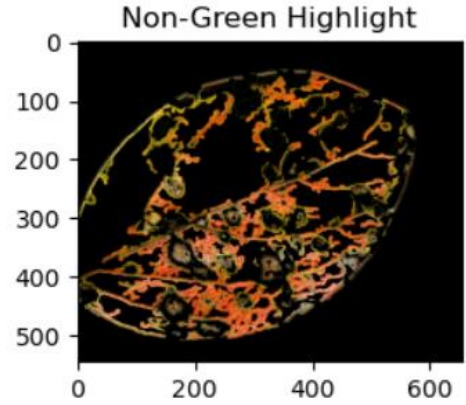
- "plant.jpg" görüntüsü için



Şekil 10. "plant.jpg" hastalıkları

Şekil 10.'daki sonuç nispeten hastalıkların tespit edilmesinin zor olduğu bir görüntüye aittir. Çünkü görüntüde iç içe geçmiş birden fazla görüntü vardır. Sağ alt kısıma doğru yapraklardaki damarların belirgin olması hataya sebep olmuştur.

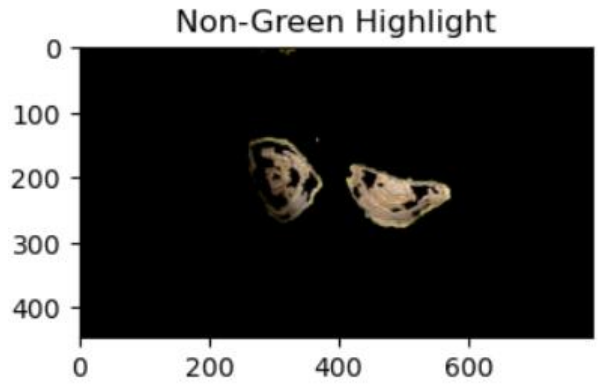
- "red.jpg" görüntüsü için



Şekil 11. "red.jpg" hastalıkları

Şekil 11. in giriş görüntüsü algoritmaya uygun bir görüntü değildir, algoritma yeşil yapraklarda etkili çalışır. Bu sebeple bu görüntüde iyi bir sonuç elde edilememiştir. Yaprak turuncu, kırmızı gibi renklere sahiptir bu sebeple HSV ile elde edilen maske bu kısımları içermez.

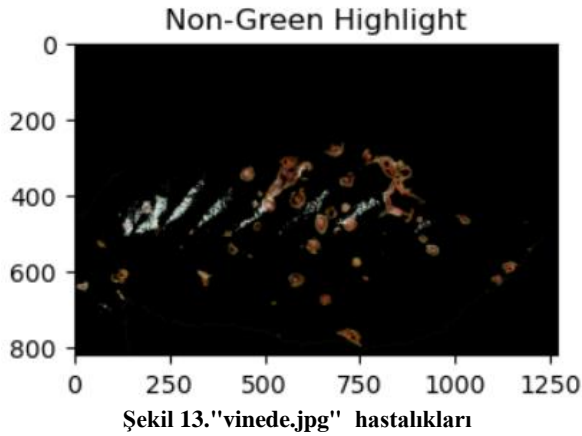
- "hole.jpg" görüntüsü için



Şekil 12. "hole.jpg" hastalıkları

Şekil 12.'de temiz bir çıktı görülmektedir. Ancak giriş görüntüsünde ("hole.jpg") yaprağın üst kısmında kıvrılmış bir hastalıklı kısım vardır ve tespit edilememiştir. Bunun sebebi bağlantılı bileşen analizinde yaprağın tamamının tespit edilememesidir. Yaprak yerine ortadaki hastalıklı kısım tespit edilmiştir ve daha sonraki işlemler bu görüntü üzerinden yapılmıştır. Yaprığın tespit edilememe sebebi de şudur; yaprak zaten giriş görüntüsünde bir bütün olarak bulunmamaktadır. Görüntüde yaprağın tamamı yoktur bu sebeple algoritma iyi çalışmamıştır.

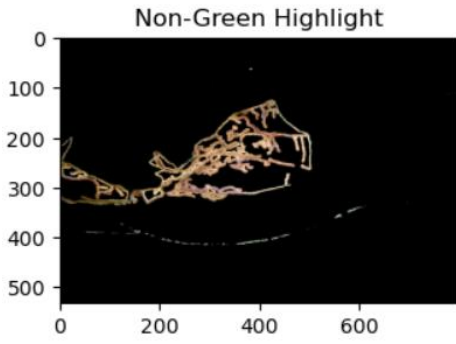
- "vinede.jpg" görüntüsü



Şekil 13. "vinede.jpg" hastalıkları

Şekil 13.'te çoğu hastalıklı bölge iyi tespit edilmiştir, yalnızca giriş fotoğrafına ışık yansımaları olan ortada solda olan kısım hatalı bir şekilde hastalık olarak tespit edilmiştir.

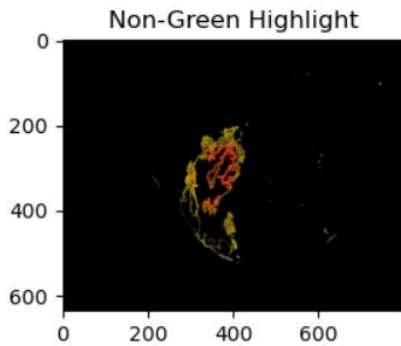
- "leaf.jpeg" görüntüsü



Şekil 14. "leaf.jpeg" hastalıkları

Şekil 14.'te hastalıklı bölge büyük oranda doğru tespit edilmiştir. Yaprığın uzun keskin kenarlarından biri hastalık olarak tespit edilmiştir.

- "peach.jpeg" görüntüsü



Şekil 15. "peach.jpeg" hastalıkları

Şekil 15.'te hastalıklı bölge doğru tespit edilmiştir.

### 3. Sonuç ve Yorum

Bu proje sürecinde, OpenCV ve NumPy kütüphanelerini kullanarak öğrendiğim görüntü işleme tekniklerini uygulama fırsatı buldum. Canny kenar dedektörü ve bağlantılı bileşen analizi gibi teknikleri uygulayarak nesne tespiti ve segmentasyon yeteneklerimi geliştirdim. Ayrıca, HSV renk uzayını kullanarak yeşil renk tonlarını belirleme ve tanıma konusunda deneyim kazandım. Morfolojik işlemler, özellikle opening ve closing, nesne tespiti ve gürültü azaltma konusunda becerilerimi artırdı. Matplotlib ile görselleştirme yeteneklerimi kullanarak algoritmaların çıktılarını anlamak ve açıklamak konusunda pratik kazandım. Bu proje, bitki hastalıkları veya objelerin tanınması gibi uygulamalara yönelik problem çözme yeteneklerimi ve deneme-yanılma yetilerimi geliştirmeme olanak sağladı.

Projedeki algoritma, bitki hastalıklarını veya belirli objeleri tanımak için görüntü işleme tekniklerini kullanmaktadır. Bu önerilen algoritmanın avantajları ve kısıtları şu şekildedir:

Avantajlar:

Bu algoritma, bağlantılı bileşen analizi ve HSV renk tanıma gibi etkili görüntü işleme tekniklerini kullanarak bitki hastalıklarını veya belirli objeleri tanıma yeteneğine sahiptir. Bağlantılı bileşen analizi, nesnelerin segmentasyonunu geliştirirken, HSV renk tanıma bitki yapraklarının yeşil tonlarını başarılı bir şekilde belirleyebilir.

Dezavantajlar:

Özellikle renk aralığına dayalı bir tanıma yöntemi kullanıldığından, değişen ışık koşulları veya farklı bitki türleri gibi faktörler, renk hassasiyetini etkileyebilir. Bu durum, algoritmanın genelde ölçekleme ve çeşitlilik konularında sınırlı olmasına neden olabilir. Ayrıca, gürültü ve arka plan detayları, bağlantılı bileşen analizi ve renk tanıma süreçlerini olumsuz etkileyebilir.

Geliştirme İçin Öneriler:

Bu algoritmayı geliştirmek için renk aralığı parametrelerinin dinamik olarak ayarlanabilir hale getirilmesi, ölçekleme ve çeşitlilik sorunlarının ele alınması, gürültü azaltma tekniklerinin iyileştirilmesi ve belki de makine öğrenmesi yöntemlerinin entegre edilmesi gibi adımlar düşünülebilir. Bu geliştirmeler, çözümün daha geniş bir uygulama yelpazesi ve daha güçlü bir performans sergilemesine katkıda bulunabilir.

### Kaynaklar

- [1] <https://www.geeksforgeeks.org/python-opencv-morphological-operations/>
- [2] <https://www.gardenia.net/disease/leaf-spot-diseases>
- [3] <https://www.gardenia.net/disease/botrytis-blight-of-peony4>
- [4] <https://www.entofito.com/yaprak-delen-cil-hastaligi/>