



**TARIM ALANLARINDA İHA GÖRÜNTÜLERİ
KULLANILARAK BİTKİ SIRALARININ VE
DAMLA SULAMA HATLARININ TESPİTİ**

Yüksek Lisans Tezi

Alper ONRAT

Eskişehir, 2024

**TARIM ALANLARINDA İHA GÖRÜNTÜLERİ KULLANILARAK BİTKİ
SIRALARININ VE DAMLA SULAMA HATLARININ TESPİTİ**

Alper ONRAT

Yüksek Lisans Tezi

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Yazılımı Bilim Dalı

Danışman: Dr. Öğr. Üyesi Selcan KAPLAN BERKAYA

Eskişehir

Eskişehir Teknik Üniversitesi

Lisansüstü Eğitim Enstitüsü

Ağustos 2024

JÜRİ VE ENSTİTÜ ONAYI

Alper ONRAT'ın TARIM ALANLARINDA İHA GÖRÜNTÜLERİ KULLANILARAK BİTKİ SIRALARININ VE DAMLA SULAMA HATLARININ TESPİTİ başlıklı çalışması 07/08/2024 tarihinde aşağıdaki jüri tarafından değerlendirilerek "Eskişehir Teknik Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği"nin ilgili maddeleri uyarınca, Bilgisayar Mühendisliği Anabilim dalında Yüksek Lisans Tezi olarak kabul edilmiştir.

Jüri Üyeleri

Unvan Adı Soyadı

İmza

Üye

: Dr. Öğr. Üyesi Selcan KAPLAN
BERKAYA

Üye

: Prof. Dr. Serkan GÜNAL

Üye

: Dr. Öğr. Üyesi Muhammet Yasin PAK

Prof. Dr. Semra KURAMA

Lisansüstü Eğitim Enstitüsü Müdürü

07/08/2024

DANIřMAN ONAYI

Daniřmanlıęını yrttęm Yksek Lisans ęrencisi Alper ONRAT, TARIM ALANLARINDA İHA GRNTLERİ KULLANILARAK BİTKİ SİRALARININ VE DAMLA SULAMA HATLARININ TESPİTİ bařlıklı tez alıřmasını tamamlamıřtır. Hazırlamıř olduęu tez tarafımca incelenmiř ve ęrencinin tez savunma sınavına alınması bilimsel ve etik aıdan uygun grlmřtir.

Tez Daniřmanı

Dr. ęr. yesi Selcan KAPLAN BERKAYA

ÖZET

TARIM ALANLARINDA İHA GÖRÜNTÜLERİ KULLANILARAK BİTKİ SIRALARININ VE DAMLA SULAMA HATLARININ TESPİTİ

Alper ONRAT

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Yazılımı Bilim Dalı

Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Ağustos 2024

Danışman: Dr. Öğr. Üyesi Selcan KAPLAN BERKAYA

Dünya genelinde tarımsal sulama, su tüketiminin büyük bir kısmını oluşturmaktadır. Damlama sulama sistemleri, tarımsal sulamada kullanılan suyun verimli kullanılmasını ve sudan tasarruf edilmesini sağlar. Damlama borularının doğru yerde olması bitkilerin yeterli miktarda su alabilmesi için kritik öneme sahiptir. Büyük ölçekli tarlalarda damlama boru hatlarının doğru sırada olup olmadığını kontrol etmek oldukça zaman alıcıdır. Bu tez çalışmasında, insansız hava aracı ile elde edilen görüntüler üzerinde bitki sıralarını ve damlama boru hatlarını tespit eden yeni bir yöntem önerilmektedir. İnsansız hava aracıyla mısır tarlalarından alınan görüntülerle yeni bir veri seti oluşturulmuştur. Oluşturulan bu veri setindeki görüntüler üzerinde, önerilen yöntemin ilk adımı olan bitki sıralarının bulunması için renk eşikleme ve çeşitli morfolojik görüntü işleme yöntemleri uygulanmıştır. Daha sonra elde edilen çizgi parçaları, açıları ve aralarındaki uzaklığa göre birleştirilerek bitki sırasına ait tüm hattın bulunması sağlanmıştır. Bitki sıralarının tespit edilmesinin ardından bitki sıraları arasında kalan alanlarda damlama borularının tespit edilmesi için farklı görüntü ön işleme yöntemleri ve çizgi parçaları arasındaki komşuluklardan yararlanılmıştır. Deneysel çalışmalar ile bitki sırası tespiti ve damlama boru hattı tespiti algoritmalarının performansları ayrı ayrı değerlendirilmiştir. Önerilen yöntem yukarıda belirtilen iki problem için sırasıyla %99.01 ve %86.54 doğrulukla çalıştığı tespit edilmiştir.

Anahtar Sözcükler: Bitki sırası tespiti, Damlama borusu tespiti, Görüntü ön işleme, Akıllı tarım, İHA.

ABSTRACT

DETECTION OF CROP ROWS AND DRIP IRRIGATION LINES USING UAV IMAGERY IN AGRICULTURAL FIELDS

Alper ONRAT

Department of Computer Engineering

Programme in Computer Software

Eskişehir Technical University, Institute of Graduate Programs, August 2024

Supervisor: Asst. Prof. Dr. Selcan KAPLAN BERKAYA

Agricultural irrigation constitutes a significant portion of water consumption worldwide. Drip irrigation systems enable the efficient use of water in agricultural irrigation and contribute to water conservation. The correct placement of drip irrigation lines is critical to ensure that plants receive an adequate amount of water. Checking the correct placement of drip irrigation lines in large-scale fields is quite time-consuming. In this thesis, a novel method is proposed to detect plant rows and drip pipelines on images obtained with an unmanned aerial vehicle. A new dataset was created with images taken from cornfields using an unmanned aerial vehicle. On the images in this dataset, color thresholding and various morphological image processing methods were applied to find crop rows, which is the first step of the proposed method. Then, the line segments obtained were combined based on their angles and distances to identify the entire line of the crop row. After detecting the crop rows, various image preprocessing methods and the relationships between neighboring line segments were utilized to identify drip irrigation lines in the areas between the crop rows. Experimental studies separately evaluated the performances of the algorithms for crop row detection and drip irrigation line detection. The proposed method demonstrated performance with 99.01% and 86.54% accuracy for the two problems mentioned above, respectively.

Keywords: Crop row detection, Drip irrigation line detection, Image preprocessing, Smart agriculture, UAV.

TEŞEKKÜR

Tez çalışmam boyunca danışmanlığımı yürüten, kıymetli bilgi birikimiyle bütün sorularımda bana yol gösteren, desteğini ve rehberliğini hiçbir zaman esirgemeyen değerli danışman hocam Dr. Öğr. Üyesi Selcan KAPLAN BERKAYA'ya teşekkür eder, saygılarımı sunarım.

Tüm hayatım boyunca maddi ve manevi emek vererek yanımda olan, beni her zaman destekleyen, sonsuz sevgi ve fedakarlıkları ile her zaman daha iyiye yönlendiren bu hayattaki en büyük şansım sevgili aileme teşekkür eder, minnettarlığımı sunarım.

Alper ONRAT



ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Eskişehir Teknik Üniversitesi tarafından kullanılan “bilimsel intihal tespit programı”yla tarandığını ve hiçbir şekilde “intihal içermediğini” beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Alper ONRAT

İÇİNDEKİLER

Sayfa

BAŞLIK SAYFASI	I
JÜRİ VE ENSTİTÜ ONAYI.....	II
DANIŞMAN ONAYI	III
ÖZET	IV
ABSTRACT.....	V
TEŞEKKÜR	VI
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ.....	VII
İÇİNDEKİLER.....	VIII
TABLolar DİZİNİ.....	X
ŞEKİLLER DİZİNİ.....	XI
SİMGELER VE KISALTMALAR DİZİNİ.....	XIII
1. GİRİŞ	1
2. VERİ SETİ	8
3. ÖNERİLEN YÖNTEM	12
3.1 Bitki Sıralarının Bulunması.....	12
3.1.1. Görüntü ön işleme	13
3.1.2. Komponentlerin oluşturulması	18
3.1.3. Komponent sıralarının oluşturulması	20
3.2. Damlama Borularının Tespit Edilmesi.....	30
3.2.1. Görüntü ön işleme	31
3.2.2. Komponentlerin oluşturulması	35
3.2.3. Komponent sıralarının oluşturulması	35
4. DENEYSEL ÇALIŞMALAR VE TARTIŞMA	38
5. SONUÇLAR	45
KAYNAKÇA.....	54

ÖZGEÇMİŞ



TABLÖLAR DİZİNİ

Sayfa

Tablo 4.1. Bitki sırası bulma yöntemi karışıklık matrisi.....	38
Tablo 4.2. Damlama boru hattı bulma yöntemi karışıklık matrisi	42
Tablo 4.3. Önerilen yöntemin performansı (%)	42
Tablo 4.4. Literatürde bitki sıralarını bulan benzer çalışmalar ile karşılaştırma	44
Tablo 4.5. Önerilen Yöntemin Çalışma Performansı.....	44
Tablo 5.1. Bitki sırası bulma algoritmasının performansı.....	47
Tablo 5.2. Damlama boru hattı bulma algoritmasının performansı	52



ŞEKİLLER DİZİNİ

Sayfa

Şekil 1.1. Damlama sulama sistemi kurulumu	2
Şekil 2.1. DJI Mini 2 İHA	8
Şekil 2.2. Bitki sıralarını içeren görüntü.....	9
Şekil 2.3. Damlama borularını içeren görüntü	9
Şekil 2.4. Damlama boruları görüntüleri, (a) mat, (b) parlak	10
Şekil 2.5. İlgilenilen bölge dışı alan içeren görüntü	10
Şekil 2.6. İlgilenilen alan (ROI)	11
Şekil 3.1. Önerilen yöntemin genel tasarımı	12
Şekil 3.2. Bitki sırası oluşturma adımları	12
Şekil 3.3. Görüntü ön işleme akış şeması	13
Şekil 3.4. (a) RGB renk uzayındaki orijinal görüntü, (b) HSV görüntü.....	14
Şekil 3.5. Renk eşikleme sonucunda elde edilen ikili görüntü	15
Şekil 3.6. Medyan bulanıklaştırma sonucunda elde edilen ikili görüntü.....	16
Şekil 3.7. Morfolojik erozyon işlemi sonucunda elde edilen ikili görüntü	16
Şekil 3.8. Morfolojik kapama işlemi sonucunda elde edilen ikili görüntü.....	17
Şekil 3.9. Bitki sırası bulma görüntü ön işleme sonucu elde edilen görüntü	18
Şekil 3.10. Merkezi hücre ve 8 bağlantı	19
Şekil 3.11. Bağlı bileşen oluşturma şeması	19
Şekil 3.12. Şekil 3.9'a bağlı bileşen oluşturma adımları uygulandıktan sonra elde edilen komponentler.....	20
Şekil 3.13. Komponent sırası oluşturma şeması.....	21
Şekil 3.14. Komşu atama şeması	25
Şekil 3.15. Sınır bölge	27
Şekil 3.16. Hata düzeltmeden önce komponentlerin bağlantısı.....	28
Şekil 3.17. Hataları düzeltilmiş komponent sıraları	29
Şekil 3.18. Orjinal görüntü üzerinde bitki sıraları	30
Şekil 3.19. Bitki sıralarının izdüşüm görüntüsü	30
Şekil 3.20. 2'li bitki sıraları	31
Şekil 3.21. Gerçek görüntünün HSV renk uzayına dönüşümü ve ardından renk eşiklemesinin uygulanması	32
Şekil 3.22. Gri renk uzayı ve yeşil renk eşikleme yapılmış görüntülerin işlenmesi.....	33

Şekil 3.23. GABOR filtresinin uygulanması ile elde edilen görüntü	33
Şekil 3.24. Renklerin ters çevrilmesi ile elde edilen görüntü	34
Şekil 3.25. Renk eşikleme sonucunda gürültüden arındırılmış görüntü	34
Şekil 3.26. Bitki sıra arası görüntüsü	36
Şekil 3.27. Komponentlerin komşuluk görüntüsü	36
Şekil 3.28. Komponent sırası görüntüsü	37
Şekil 3.29. Gerçek görüntü ile komponent sırasının örtüştürülmesi	37
Şekil 4.1. Bitki sırası bulunamayan görüntü	39
Şekil 4.2. Bitki sırasının bulunabildiği görüntü	39
Şekil 4.3. Hatalı bitki sırası bulunan görüntü	40
Şekil 4.4. Yabancı ot bulunan sıra arasının (a) görüntü ön işleme sonucu, (b) gerçek görüntüsü	40
Şekil 4.5. Hatalı sıra bulunan görüntüdeki bitki sıralarının izdüşüm görüntüsü	41
Şekil 4.6. Gölge içeren bitki sırasının (a) gerçek, (b) görüntü ön işleme, (c) komponent sırası görüntüleri	42

SİMGELER VE KISALTMALAR DİZİNİ

İHA	: İnsansız Hava Aracı
RGB	: Kırmızı, Yeşil, Mavi (ing. Red, Green, Blue)
HSV	: Renk, Doygunluk, Değer (ing. Hue, Saturation, Value)
ROI	: İlgilenilen Bölge (ing. Region of Interest)



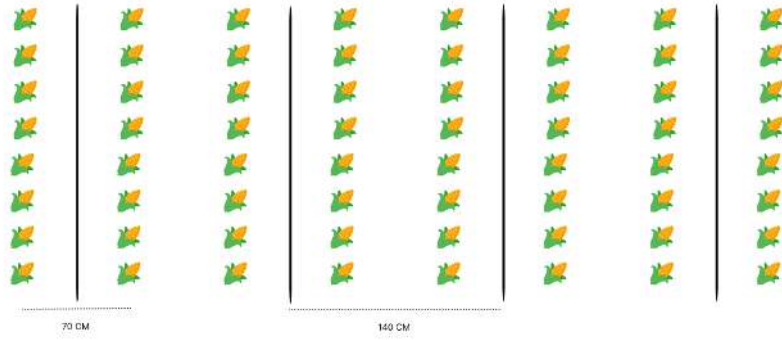
1. GİRİŞ

Kurak ve yarı kurak iklime sahip bölgelerde bitkisel ürünlerin yetiştirilmesi için sulamaya ihtiyaç duyulur. Sulama, bitkilerin ihtiyaç duyduğu ve yağışlarla karşılanamayan suyun çeşitli yöntemlerle bitkinin köklerine ulaştırılması olarak tanımlanır. Su, sürdürülebilir tarımın en önemli unsurlarındandır. Dünya su tüketiminde su kaynaklarının yaklaşık %70'i tarım, %19'u sanayi ve %11'i evsel amaçlı kullanılmaktadır (FAO Aquastat, 2013). Sürdürülebilir tarım için suyun verimli kullanılması gerekmektedir. Tarımda yaygın olarak kullanılan sulama yöntemleri; yüzey sulama, yağmurlama sulama ve damla sulama olmak üzere üçe ayrılmaktadır.

Yüzey sulama, suyun yüzeyden bitkilere ulaştırıldığı en yaygın ve eski sulama yöntemidir. Bu yöntem, suyun tarlanın yüzeyinde akarak bitki köklerine ulaşmasını sağlar. Maliyeti düşüktür ancak su tüketimi yüksektir.

Yağmurlama sulama, suyun bitkilere yağmur damlaları gibi püskürtülerek verildiği bir sulama yöntemidir. Bu yöntem, suyun belirli bir basınçla boru hattı üzerinden yağmurlama başlıklarına taşınmasını ve buradan tarlaya yayılmasını sağlar. Yağmurlama sulama, geniş alanların sulanmasında oldukça etkili ve yaygın olarak kullanılan bir yöntemdir. Su tüketimi yüzey sulamaya göre düşüktür. Ayrıca rüzgar ve sıcaklık sulama verimliliğini etkilemektedir.

Damla sulama, suyun doğrudan bitki köklerine düşük miktarlarda ve düzenli aralıklarla verilmesini sağlayan bir sulama yöntemidir. Bu yöntem, suyun damlalar halinde bitkilerin kök bölgesine uygulanmasını içerir. Damla sulama, suyu bitki köklerine doğrudan ileterek su israfını en aza indirir ve bitkilerin su ihtiyacını verimli bir şekilde karşılar. Suyla birlikte verilen gübrenin bitkinin doğrudan kök bölgesine ulaştırılmasını sağlayarak gübre kullanımında fayda sağlar (Smith vd., 2016). Bu şekilde düşük basınçlarda dahi sulama yapılabilir. Su tüketimi yağmurlama ve yüzey sulamaya göre oldukça düşüktür. Rüzgar ve sıcaklığa yağmurlama sulama sistemine göre daha toleranslıdır. Yüzey sulama yöntemine göre üretim verimliliğini önemli ölçüde artırır (Wonprasaid vd., 2022). Aynı zamanda su tüketimini %60 oranında azaltır (Karimi vd., 2011). Damla sulama sistemi, damlama borularının traktör yardımıyla bitki sıraları arasına serilmesi ile hazırlanır. Genellikle bitki sıraları arasına birer atlayarak (1 dolu 1 boş gelecek şekilde) serilir. Damla sulama sistemi kurulumu için örnek görsel Şekil 1.1'de verilmiştir.



Şekil 1.1. Damlama sulama sistemi kurulumu

Serme işlemi sırasında rüzgar, sel gibi çevresel faktörler veya işçilik hatasından dolayı borular doğru sıraya denk gelemeyebilir. Bu durumda tarla dengeli sulanamaz. Bazı bitkilere fazla su giderken bazılarına eksik su gidebilir. Bu problemin önüne geçmek için damlama borularının dağıtımını sırasında veya sonrasında boruların doğru sırada olup olmadığı kontrol edilmesi gerekir. Doğru sırada olmayan borular düzeltilerek doğru sıraya alınır. Bu işlem tarlayı dolaşarak ve tarlayı gözle tarayarak yapılır. Büyük tarlalarda bu iş oldukça zaman alır ve hataya açıktır.

Akıllı tarım uygulamalarının geliştirilmesi, tarımsal bilgi algısı, otonom navigasyon, değişken karar verme ve kontrol gibi hassas operasyon teknolojilerinin temel tarımsal kaynaklarla entegre edilmesi, verimi artırmanın ve tarımın yeşil kalkınmasını sağlamanın etkili bir yoludur (Zhang vd., 2024). Son yıllarda teknolojiye gelişmeler sayesinde birçok iş otonom veya yarı otonom yapılabilmektedir. Bu teknolojik gelişme tarımda kullanılan aletlerde de etkisini göstermektedir. Tarım alanını tarayarak yabancı otların tespit edilmesi (Pérez-Ortiz vd., 2015), gübre ve su ihtiyacının analiz edilmesi (Kamiński vd., 2019), otonom sürüş yapabilen traktörler ile tarla sürümü, ekim ve hasat (Furioli vd., 2021) yapılabilmektedir. Bu işlemler otonom bir şekilde daha hassas yapılabildiği için ürün verimini arttırırken kullanılan hammadde ve işçiliği azaltır. Tarlayı dolaşarak ve tarlayı gözle tarayarak yapılan manuel gözlem yerine bu sistemlerin kullanılması daha hızlı ve güvenilirdir. Bitki veya tarla analizi için geliştirilen otonom sistemlerde genellikle yapılan ilk işlem bitki sıralarının tespit edilmesidir. Bitki sıralarının tespitine yönelik literatürde birçok çalışma yer almaktadır.

(Ruan vd., 2023) çalışmasında yazarlar, YOLO-R'ye dayalı insansız bir tarım makinesinde çalışacak bitki sırası tespit yöntemi geliştirmişlerdir. Geliştirdikleri yöntem,

geleneksel görüntü işleme yöntemlerinin yabancı otlardan, ışıktan ve diğer faktörlerden kolayca etkilenmesi sorununu çözmeyi amaçlamaktadırlar. İlk olarak, YOLO-R nesne tespit algoritması kullanılarak bitki konum bilgisi elde edilmiştir. Daha sonra, görüntüdeki bitki sıra sayısı ve her bitki sırasındaki ürünler DBSCAN kümeleme algoritması ile belirlenmiştir. Son olarak, her bir bitki sırası için en küçük kareler yöntemi kullanılmıştır. Önerilen çalışmada veri seti olarak 7, 14 ve 21 günlük pirinç tarlasından alınan görüntüler kullanılmıştır. Bitki sırası tespit algoritmasının doğruluk değerleri 7, 14 ve 21 günlük pirinç tarlasından alınan görüntülerden elde edilen veri setlerinde sırasıyla %93.91, %95.87 ve %89.87 olarak bulunmuştur. Bu sonuçlar, önerilen algoritmanın bitki sıralarını etkili bir şekilde tanımlayabildiğini göstermektedir.

(De Silva vd., 2023) çalışmasında yazarlar, derin öğrenmeye dayalı geliştirdikleri sistemde düşük maliyetli kameralar kullanan ve değişken tarla koşullarına dayanabilen bir bitki sırası tespit algoritması geliştirmişlerdir. Bitki sırası tespitine yönelik mevcut veri setleri olası tüm tarla değişikliklerini içermedikleri için yeni bir veri seti oluşturmuşlardır. Veri seti içerisinde, farklı büyüme dönemlerinde şeker pancarı bitkileri, çeşitli ışık seviyeleri, farklı yabancı ot yoğunlukları, kavisli bitki sıraları ve süreksiz bitki sıralarını içeren 11 farklı tarla varyasyonlarını içermektedir. Bitki sırası tespiti için U-Net modeli kullanılmıştır. Robotun takip etmesi gereken merkezi bitki sırasının hat parametrelerini belirlemek için üçgen tarama yöntemi kullanmışlardır. Geliştirdikleri algorithmada 11 veri seti için %82.13 ile %90.00 oranında doğruluk değeri elde etmişlerdir.

(Hassanein vd., 2019) çalışmasında yazarlar, düşük maliyetli bir insansız hava aracı (İHA) kullanarak elde edilen tarla görüntülerinden geliştirdikleri algoritma ile bitki sıralarının tespit edilmesini amaçlamışlardır. İlk olarak RGB (Red, Green, Blue) görüntüleri HSV (Hue, Saturation, Value) renk uzayına dönüştürerek ve Hue görüntüsü çıkarılmıştır. Daha sonra bitki sıralarının yönelim açısını (θ) tespit etmişlerdir. Bunun için Temel Bileşen Analizi kullanmışlardır. Metodolojinin üçüncü ve son adımında bitki sıralarının tespit edilmesini sağlamışlardır. Bu adımda önerilen metodolojide, sapmanın tespit edilen bitki sırası yönüne (θ) eşit olduğu bir çizgi tarayıcı oluşturulmuştur. Çizgi tarayıcı, Hue görüntüsünü tarayarak, her çizginin Hue değerlerinin ortalamasını almışlardır. Bu değerlerden en yüksek ve en düşük olanların ortalamasını alarak Hue kanalındaki yeşil renk ile karşılaştırmış ve bitki sıralarını tespit etmişlerdir. Geliştirdikleri

algoritmayı kendi oluşturdıkları 10, 20, 30, 40, 60, 80, 100 ve 120 metre yükseklikten elde edilmiş 10 adet görüntü üzerinde çalıştırmış ve başarılı sonuçlar elde etmişlerdir. Ancak geliştirilen modelin dezavantajının işlem süresi olduğu vurgulanmıştır ve yüksek işlem süresinden dolayı gerçek zamanlı uygulamalar için uygun olmadığını değerlendirmişlerdir.

(Winterhalter vd., 2018) çalışmasında yazarlar, küçük bitkiler dahil olmak üzere bitki sıralarının tespit edilmesi için yeni bir algoritma geliştirmişlerdir. Bu algoritmanın farklı ortamlarda çalışabilmesi için ilk olarak sensör verilerinden öznitelik haritasını çıkartmışlardır. Çıkarılan harita, bitki sıralarının tespiti için algoritmaya girdi sağlamıştır. Öznitelik haritasını çıkarmak için vizyona dayalı segmentasyon ve lazer tabanlı segmentasyon yöntemleri kullanılmıştır. Vizyona dayalı segmentasyonda RGB formatındaki resim üzerinde yeşil renklere bakarak pikselin bitkiye ait olup olmadığına karar verilmiştir. Lazer tabanlı segmentasyonda 3 boyutlu lazer menzilli tarayıcılar kullanılarak 3 boyutlu nokta bulutları oluşturulmuştur. Bitki sırası tespiti için 2 boyutlu öznitelik haritası girdi olarak alınmıştır. Çoğu ekin sırası birbirine paralel ve eşit mesafede olduğu için ilk olarak öznitelik haritasından bu sıraların örüntüsünü çıkartmışlardır. Büyük bitkilerin bitki sıralarını örttüğü veya bitkilerin çok küçük olduğu durumlarda öznitelik haritalarını hesaplamak için kullanılan bitki örtüsü segmentasyonunun doğru çalışabilmesi için tüm sıraların tek seferde tespit edilmesi önerilmiştir. İlk yöntem olarak Line Hough dönüşümü önerilmiştir. İkinci yöntem, olan Dual Line Hough dönüşümünde, ilk yöntemden elde edilen sıraya sıra arası mesafe eklenerek paralel sıra tahmin edilmeye çalışılmıştır. Üçüncü yöntem olarak Pattern Hough dönüşümü yöntemi ile çalışılmıştır. Tasarlanan Pattern Hough dönüşümü modelin tüm sıraları tek seferde çıkartmıştır. Son olarak RANSAC modeli ile diğer yöntemler karşılaştırılmıştır. Modeller kanola, mısır, pırasa, orta büyüklükte şeker pancarı ve küçük şeker pancarı bitkileri üzerinde çalıştırılmıştır. Veri seti yaklaşık 5 cm uzunluğunda orta boy bitkiler, 1 cm uzunluğunda yeni çıkmış küçük şeker pancarları, kanola, mısır ve pırasa tarlalarından lazer verilerini içermektedir. Bu verileri bir robot vasıtasıyla elde etmişlerdir. Elde edilen sonuçlara göre bitki sırasındayken Pattern Hough dönüşümü yöntemi diğer modellere göre neredeyse tüm bitkilerde daha iyi sonuç vermiştir. Kanolada tüm Hough tabanlı algoritmalar RANSAC'tan daha yüksek başarı oranına sahipken, pırasada ise RANSAC, Hough ttabanlı algoritmalarından daha iyi sonuç verdiği gözlemlenmiştir.

(Pérez-Ortiz vd., 2015) çalışmasında yazarlar, İHA'lar tarafından sağlanan görüntüleri kullanarak yabancı ot haritalaması için bir sistem sunmuşlardır. Yabancı otların mahsul verimine etkisinin önemi vurgulanmış ve Avrupa Birliği'ndeki yaygın yöntem olan herbisit kullanımının maliyeti ve çevreye zararından bahsedilmiştir. Erken uyarı sistemi için pilotlu uçakların ve uyduların erken dönemde etkisiz kaldığını, bu iki yöntem yerine yüksek çözünürlükte görüntü sağlayan İHA'ların kullanılmasını önermişlerdir. Veri setini İHA kullanarak ayçiçek bitkisinin 4-6 yaprak döneminde oluşturmuşlardır. Elde edilen görüntüler ortorektifikasyon ve mozaikleme ile birleştirilmiştir. Orta-moziklenmiş görüntü doğru şekilde eşleştirildikten sonra hesaplama işlemi için 1000 x 1000 piksellik alt görüntülere parçalanmıştır. Bitki örtüsünün belirli bir özelliğini vurgulamak için bitki örtüsü indeksleri iki yaygın indeks ile hesaplanmış ve performansları karşılaştırılmıştır. Hough dönüşüm yöntemi ile bitki sıraları tespit edilmiştir. Bitki, yabancı ot ve toprak kümeleme tekniği, yarı denetimli teknik ve denetimli teknik ile sınıflandırılmıştır. Kümeleme tekniğinde k-ortalamlar algoritması kullanılmıştır. Yarı denetimli ve denetimli yöntemde SVM tabanlı yaklaşım seçilmiştir. Tekniklerin karşılaştırılmasında insansız hava aracının görüntüyü kaç metre yükseklikten çektiğinin önem arz ettiği gözlemlenmiştir. 30, 60, 100 metreden elde edilen görüntüler üzerinde farklı sınıflandırıcılar ile denetimsiz, yarı denetimli ve denetimli öğrenme yöntemi sonuçlarında 0.05 ile 0.30 aralığında MAE elde etmişlerdir.

(Zheng vd., 2014) çalışmasında yazarlar, şerit analizi ile çoklu bitki sırası tespit etmişlerdir. Bu işlem için ilk olarak orijinal görüntüyü yeşil bitkiler ve toprak arka planı arasında gelişmiş kontrasta sahip gri ölçekli bir görüntüye dönüştürmüşlerdir. Diğer olası renk kanalı kombinasyonlarıyla karşılaştırıldığında, bu yöntemin iyi sonuçlar verdiği ve mevcut görüntüler için iyi çalıştığını kanıtlamışlardır. Sıra tespit tekniklerine geçmeden önce görüntüleri yatay sıralara bölmüşlerdir. Sıra konumlarının tahmininden önce, sıraların merkezini gösteren bir dizi nokta belirlemişlerdir. Ekin sıralarının sınırlarını hesaplamışlardır ve her ekin sırasının ağırlıklarını bulmuşlardır. Ardından sıraların merkez noktalarını tespit etmişlerdir. Sıraların merkez çizgilerinin yönünü ve yanal konumunu en küçük kareler regresyon analizi ile tahmin etmişlerdir. Bitki büyümesinin doğallığı gereği doğru denebilecek bir nokta ve pozisyon olmadığı için sistemin test edilmesinin kolay olmadığı vurgulanmıştır. Hesaplanan sıra konumunu manuel olarak algılanan sıra konumuyla karşılaştırarak sıra algılama algoritmasının performansını test

etmişlerdir. Test sonucunda, sistemin sıra konumlarını yüksek doğruluk ile sınıflandırdığı ve hesaplama işleminin hızlı olduğu sonucuna varılmıştır.

(Romeo vd., 2012) çalışmasında yazarlar, mısır tarlasında gerçek zamanlı bitki sırası tespit eden yeni bir yöntem sunmuşlardır. Tarım aracının üzerine yerleştirdikleri kamera sistemiyle görüntüleri elde etmişlerdir. Görüntü işlemeyi iki ana sürece bölmüşlerdir. İlk olarak görüntü segmentasyonu yapmışlardır. Bu işlem için yeşil bitkileri topraktan ayırmak için renk eşiklemesi uygulamışlardır. İkinci adımda bitki sıralarının tespiti için, yeşil piksellerin maksimum birikimini arayan görüntü perspektif projeksiyonunu temel alan bir yöntem uygulamışlardır. Geliştirdikleri modeli, Hough Dönüşüm uygulamasıyla karşılaştırmışlardır. Başarı oranı ve çalışma zamanı bakımından Hough Dönüşüm yönteminden daha iyi sonuç elde etmişlerdir. Çeşitli görüntü çözünürlüğünde ortalama %97.00 oranında başarı sağlamışlardır.

Kapsamlı literatür taraması sonucunda, yapılan çalışmaların çoğunun bitki sırası tespitine yönelik çalışmalar olduğu görülmüştür. Yapılan çalışmalarda genellikle özel veri seti kullanılması bu alanda açık erişimli bir veri setinin eksikliğine neden olmuştur. Ayrıca tarımsal sulamada suyun daha verimli kullanılmasını sağlayan damlama sulama yöntemindeki problemlerden biri olan damlama boru hatlarının olması gerektiği sıra dışına taşma ve bunun sonucunda su israfı, mahsul verimindeki düşüş gibi problemlere hızlı ve güvenilir çözüm bulmak oldukça önemlidir. Bu çalışmada, yeni ve etkili bir bitki sırası tespiti algoritması geliştirilmesinin yanı sıra literatürdeki çalışmalardan farklı olarak İHA ile elde edilen görüntüler üzerinde bitki sıralarının ve damlama boru hatlarını tespit eden yeni bir yöntem önerilmektedir. Ayrıca İHA ile mısır tarlalarından alınan görüntülerle yeni bir veri seti oluşturulmuş ve diğer araştırmacıların erişimine açık hale getirilmiştir. Oluşturulan bu veri setindeki görüntüler üzerinde, önerilen yöntemin ilk adımı olan bitki sıralarının bulunması için renk eşiklemesi ve çeşitli morfolojik görüntü işleme yöntemleri uygulanmıştır. Daha sonra elde edilen çizgi parçaları, açıları ve aralarındaki uzaklığa göre birleştirilerek bitki sırasına ait tüm hattın bulunması sağlanmıştır. Bitki sıralarının tespit edilmesinin ardından bitki sıraları arasında kalan alanlarda damlama borularının tespit edilmesi için farklı görüntü ön işleme yöntemleri ve çizgi parçaları arasındaki komşuluklardan yararlanılmıştır.

Bu tez çalışması aşağıdaki şekilde organize edilmiştir: Bölüm 2’de çalışmada kullanılan veri seti hakkında ayrıntılı bilgi verilmektedir. Bölüm 3’te, bitki sıralarının ve

damlama borularının bulunması için önerilen kural tabanlı algoritma sunulmaktadır. Bölüm 4'te yapılan deneysel çalışmalar, algoritmanın oluşturulan veri seti üzerinde çalıştırılması ile elde edilen sonuçları ve literatür ile karşılaştırılması yer almaktadır. Son olarak Bölüm 5'te çalışma ile ilgili genel değerlendirmeler verilmiştir.



2. VERİ SETİ

Bu çalışmada, damlama borusu içeren, mısır tarlalarından alınan görüntülerle yeni bir veri seti (UAV-CRDID: UAV-Crop Row Drip Irrigation Dataset) oluşturulmuştur. Veri setinin tamamı <https://www.kaggle.com/datasets/alperonrat/uav-crdiduav-crop-row-drip-irrigation-dataset> adresinde tüm araştırmacıların erişimine açılmıştır. Bu veri setindeki görüntülerin elde edilmesi için mısır bitkisinin 4-8 yaprak arasında ve 10-20 cm olduğu dönemde çekimler yapılmıştır.

Veri seti içerisindeki görüntüler iki farklı mısır tarlasından çekilmiştir. Tarlalar üzerinde Şekil 2.1’de gösterilen DJI firmasının Mini 2 model İHA’sı ile 2-5 metre yükseklikten mp4 formatında 1920x1080 çözünürlükte video kayıtları alınmıştır. Bu video kayıtlarından 1 saniyelik periyodlarla 1920x1080 çözünürlükte jpg formatında görüntüler elde edilmiştir.

Tüm görüntülere ait bitki sırası özellikleri Ek 1’de, bazı görüntülerde bulunan damlama borularının özellikleri Ek 2’de yer almaktadır.

Veri setindeki bitki sırası ve damlama borusu sırası içeren örnek görüntüler sırasıyla Şekil 2.2-4’te verilmiştir. Şekil 2.2’de yeşil ile işaretlenenler bitki sıralarına, Şekil 2.3’te sarı ile işaretlenenler damlama borusu sıralarına karşılık gelmektedir.



Şekil 2.1. DJI Mini 2 İHA



Şekil 2.2. Bitki sıralarını içeren görüntü



Şekil 2.3. Damlama borularını içeren görüntü

Veri setini oluşturmak için çekilen videolar günün farklı saatlerinde kaydedildiği için bitki ve damlama borularının renkleri ve gölgeleri güneş açısına göre farklılık göstermektedir. Örneğin Şekil 2.5a'da damlama boruları daha mat bir renge sahipken Şekil 2.5b'de daha parlak bir renge sahiptir. Bu farklılık nesnelerin arka plandan ayrılması işlemlerinde kullanılacak eşik değerlerini belirlemede zorluğa neden olmaktadır.



(a)

(b)

Şekil 2.4. *Damlama boruları görüntüleri, (a) mat, (b) parlak*



Şekil 2.5. *İlgilenilen bölge dışı alan içeren görüntü*

Yukarıda bahsedilen bu durumlara çözüm olarak veri setindeki görüntülerin sıra sayısı ve damlama boru sayısı özelliklerini çıkarırken sadece sol ve sağ alt köşelerden içeriye dahil olan sıralar dikkate alınmıştır. Şekil 2.7’de görüntülerde, önerilen yöntemin kullandığı ilgilenilen alan gösterilmektedir.



Şekil 2.6. *İlgilenilen alan (ROI)*

3. ÖNERİLEN YÖNTEM

Bu çalışmada, damlama boru hatalarının tespiti için önerilen model tasarımında öncelikle görüntülerde bitki sıralarının bulunması gerekmektedir. Bitki sıraları bulunduktan sonra her sıra arasında damlama borusu varsa tespit edilir.



3.1 Bitki Sıralarının Bulunması

Bitki sıralarının tespit edilmesi damlama borularının tespit edilmesi için ilk adımdır. Bitki sıralarının tespit edilmesinin ardından elde edilen sıraların orta kısımlarında damlama boruları aranır. Bitki sıralarının doğru tespit edilmesi damlama borularının tespit edilmesi için ön koşuldur. Şekil 3.2’de bitki sıralarının bulunması için yapılacak işlemler gösterilmektedir.



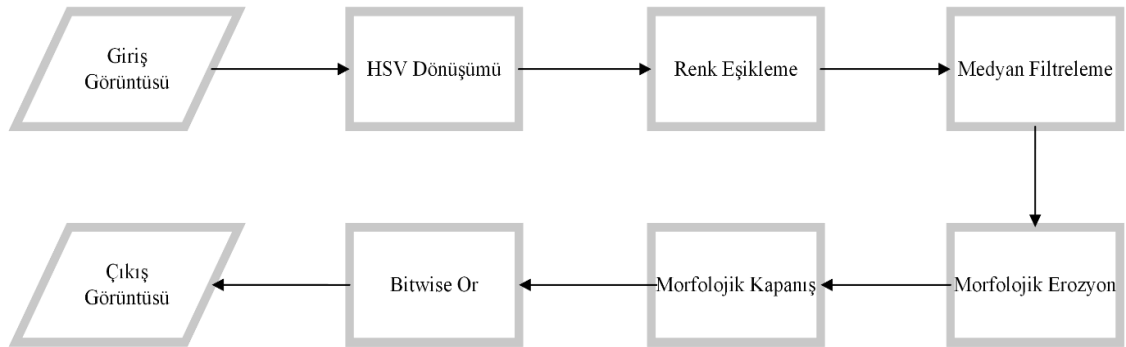
Şekil 3.2. Bitki sırası oluşturma adımları

Sıra tespitinde öncelikle görüntü ön işleme teknikleri kullanılarak bitkilerin arka plandan ayrılması hedeflenmektedir. Arka plandan ayrılmış bitkiler üzerinde sıra bulma algoritması koşturularak bitki sıralarının tespit edilmesi sağlanır.

3.1.1. Görüntü ön işleme

Görüntü işleme algoritmalarında model ne kadar iyi olursa olsun kullanılan veriler düşük kalitede ise model kötü performans gösterebilir. Bu yüzden modele uygun veriler kullanmak ve bu veriler üzerinde gürültü azaltma, renk ve boyut ayarlama, istenilen özellikleri ön plana çıkarma gibi çeşitli teknikler kullanılarak modelin daha iyi performans göstermesi sağlanır. Kullanılacak teknikler amaca göre değişkenlik gösterebilir. Bu çalışmada önerilen kural tabanlı algoritmanın doğru çalışması ve istenen sonuçları üretebilmesi için görüntünün ön işleminden geçirilmesi gerekir.

Ön işleme teknikleri Şekil 3.3'te gösterildiği gibi RGB'den HSV formatına renk dönüşümü yapılır. HSV görüntü üzerinde OpenCV'de yer alan inRange fonksiyonu ile renk eşikleme yapılarak yeşil bitkilerin ön plana çıkartılır. Yeşil bitki sıraları belirgin hale gelir; ancak sıra aralarında küçük yabancı otların filtrelenmesi için OpenCV kütüphanesinden medianBlur fonksiyonu çağırılır. Medyan filtreleme sayesinde birçok gürültü giderilmiş olur; ancak çok küçük parçalar halen görüntü üzerinde yer alır. Bu istenmeyen küçük parçaları elimine etmek için öncelikle erozyon morfolojik işlemi kullanılır. Ardından morfolojik kapama işlemi ile kalan parçaların iyileştirilmesi ve son adımda bit düzeyinde veya (Bitwise Or) işlemi ile birleştirilmesi sağlanır. Tüm bu adımlar aşağıdaki bölümlerde ayrıntılı olarak açıklanmıştır.

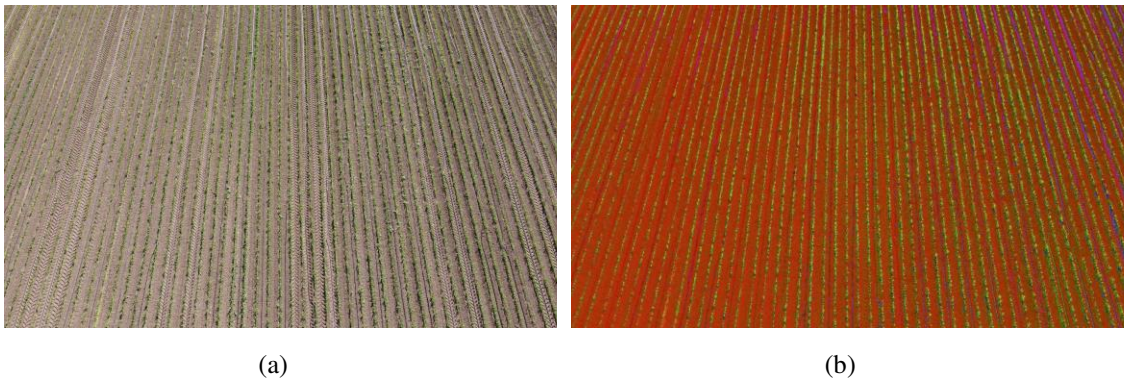


Şekil 3.3. Görüntü ön işleme akış şeması

Renk uzayı, renkleri tanımlamak için kullanılan bir koordinat sistemi veya matematiksel modeldir. Renk uzayları, belirli bir renk aralığını temsil etmek için kullanılır. RGB renk uzayı, dijital ekranlarda yaygın olarak kullanılan bir modelken, HSV renk uzayı insan görme algısına daha yakın özellikler sunar. RGB renk uzayı, kırmızı, yeşil ve mavi bileşenlerin farklı yoğunluklarda birleşmesiyle oluşturulan bir modeldir. Her bileşen 0 ile 255 arasında bir değer alır ve bu üç bileşenin kombinasyonları ile sarı, mor gibi diğer renkler elde edilir. HSV renk uzayı, parlaklık bileşenini (Value) ayrı olarak temsil ettiği için, renk bilgilerinin aydınlatma değişikliklerinden daha az etkilenmesini sağlar. Bu özellik, değişken aydınlatma koşullarında güvenilir sonuçlar elde etmeyi kolaylaştırır. HSV uzayında belirli bir renk tonunu seçmek, RGB uzayına göre daha basittir. Örneğin, belirli bir rengin tüm tonlarını tespit etmek için Hue bileşeni üzerinden eşikleme yapılabilir. RGB uzayında ise, aynı renk farklı parlaklık ve doygunluk değerlerinde birçok farklı kombinasyona sahip olabilir, bu da segmentasyon işlemlerini zorlaştırır.

Veri setinde yer alan tüm görüntüler RGB renk uzayına sahiptir. Önerilen yöntemde, görüntüler RGB renk uzayından HSV renk uzayına dönüştürülmüştür. Veri setindeki RGB görüntü ile HSV dönüştürülmüş hali sırasıyla Şekil 3.4a ve 3.4b’de verilmiştir.

Güneş ışığının geliş açısına göre bitki sıraları farklı yeşil tonlarında gözükabilmektedir. HSV görüntü üzerinde daha önce deneme yanılma yoluyla elde edilen Denklem 3.1’de verilen renk eşikleme parametreleri uygulanır.

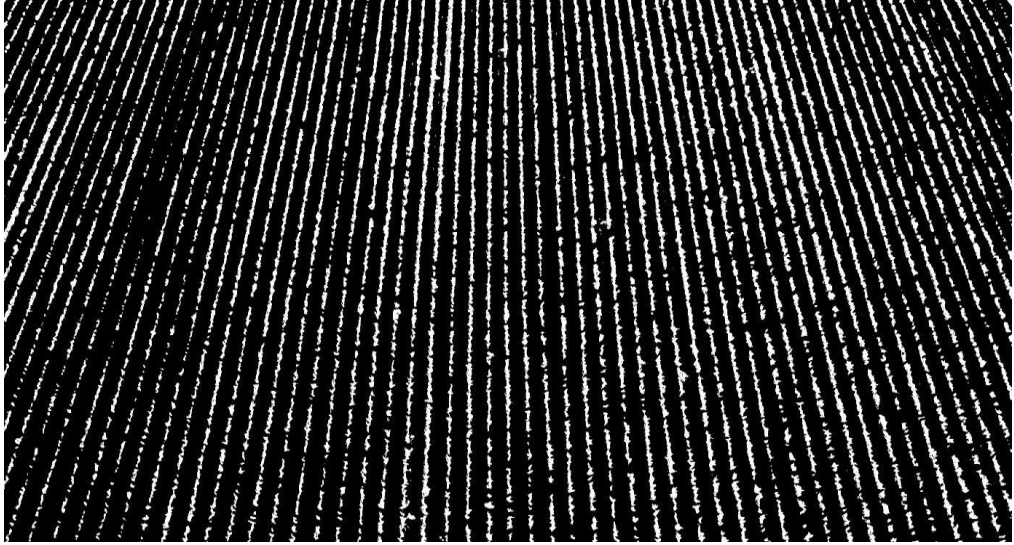


Şekil 3.4. (a) RGB renk uzayındaki orijinal görüntü, (b) HSV görüntü

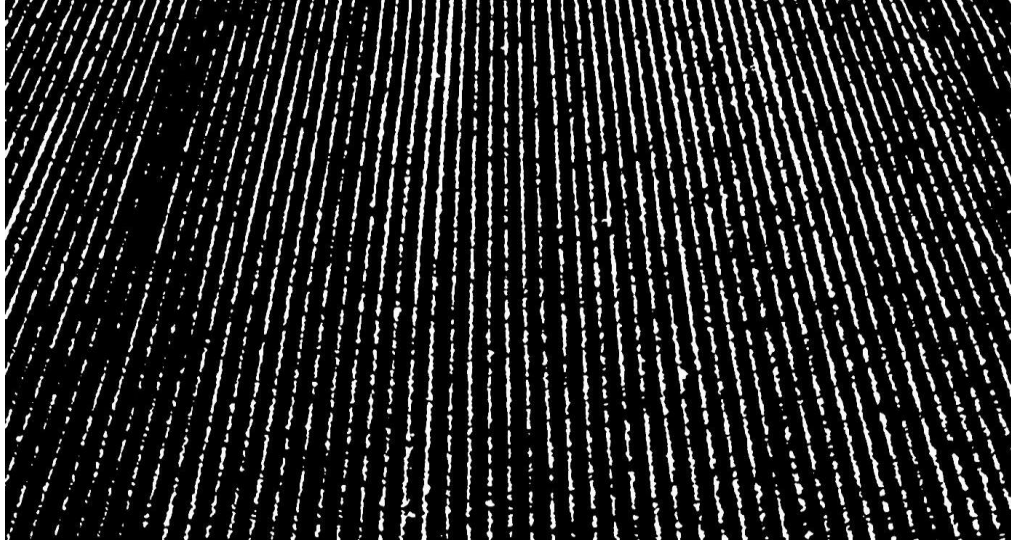
$$g_{(i,j)} = \begin{cases} 255 & \text{eğer } 28 \leq H_{(i,j)} \leq 75 \text{ ve } 0 \leq S_{(i,j)} \leq 255 \text{ ve } 46 \leq V_{(i,j)} \leq 255 \\ 0 & \text{diğer durumlarda} \end{cases} \quad (3.1)$$

Renk eşikleme sonucunda bitki sıralarının belirgin hale geldiği Şekil 3.5'te verilen ikili görüntü elde edilir.

Görüntüdeki gürültüyü azaltmak ve belirgin kenarları korumak amacıyla medyan bulanıklaştırma tekniği uygulanır. Medyan bulanıklaştırma, bir görüntünün her bir pikseli için, belirli bir komşuluk alanı içerisindeki piksellerin medyan değerini alarak, o pikseli bu medyan değer ile değiştiren bir filtredir. Şekil 3.6'da görüldüğü üzere medyan bulanıklaştırma sayesinde bitki sıraları arasındaki gürültü azalır. Bitki sıraları daha belirgin hale gelir.

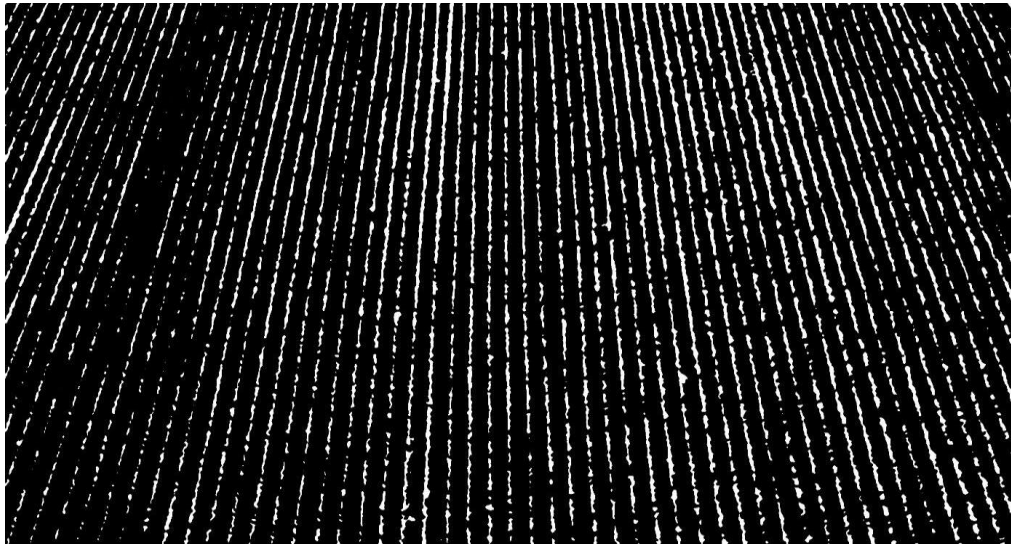


Şekil 3.5. Renk eşikleme sonucunda elde edilen ikili görüntü



Şekil 3.6. Medyan bulanıklaştırma sonucunda elde edilen ikili görüntü

Medyan bulanıklaştırma gürültünün büyük kısmını yok eder. Ancak bazı görüntülerde bitki sıraları arasında küçük parçalar kalabilir. Bu küçük parçalar morfolojik kapanma işlemi neticesinde daha büyük parçalara dönüşebilir, hatta 2 sıra arasını birleştirebilir. Bu durumda sıra bulma algoritması düzgün çalışmayabilir. Bu yüzden bu parçaları yok etmek için morfolojik erozyon uygulanır. Şekil 3.7’de morfolojik erozyon sonucu bulunmaktadır.



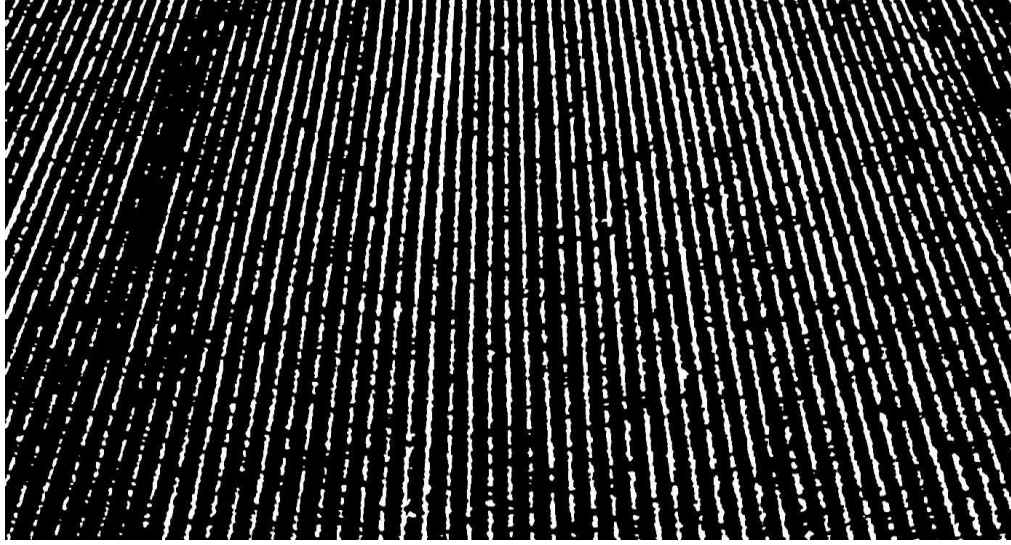
Şekil 3.7. Morfolojik erozyon işlemi sonucunda elde edilen ikili görüntü

Erozyon işleminin ardından görüntüde gürültü kalmamış olması veya çok az kalmış olması beklenir. Görüntü üzerindeki bitki sıralarının daha belirgin hale gelmesi ve bitkiler arasındaki boşlukların kapanması için morfolojik kapama uygulanır. Morfolojik kapanma, bir görüntünün belirli bir yapı elemanı (kernel) kullanılarak genişletilmesi ve ardından erozyona uğratılması işlemidir. Morfolojik kapama, görüntüdeki küçük delikleri kapatmak, gürültüyü gidermek ve nesneleri birleştirmek için etkili bir yöntemdir. Bitki sırasında bitkiler arasındaki boşlukların doldurulması için morfolojik kapama işleminde yapı elemanı (kernel) olarak dikdörtgen seçilmiştir. Şekil 3.8’de morfolojik kapama sonucunda elde edilen görüntüde beyaz renkli pikseller arasındaki boşlukların dolduğu görülmektedir.

Son olarak medyan bulanıklaştırma sonucu elde edilen görüntü ile morfolojik kapama sonucu elde edilen görüntü bit düzeyinde veya işlemi ile birleştirilir. Medyan bulanıklaştırmanın ardından yapılan erozyon işlemini sonucunda istenmeyen küçük parçalar elimine edilmiştir. Bit düzeyinde veya işlemi sonucunda bu küçük parçalar yeniden Şekil 3.9’da elde edilen görüntüye eklenmiştir. Bu daha sonraki sıra bulma algoritması için problem olmayacaktır çünkü alanları az olduğu için komponentler oluşturulurken filtrelenecektir.



Şekil 3.8. Morfolojik kapama işlemi sonucunda elde edilen ikili görüntü



Şekil 3.9. Bitki sırası bulma görüntü ön işleme sonucu elde edilen görüntü

Görüntü ön işleme sonucunda bitki sıralarının belirgin hale geldiği ve gürültüden arındığı gözlemlenir.

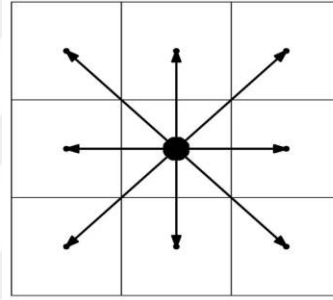
3.1.2. Komponentlerin oluşturulması

Görüntü ön işleme sonucunda bitki sıraları Şekil 3.9’da görüldüğü gibi beyaz renkli kalmış olur. Elde edilen görüntüdeki beyaz bölgelerin tespit edilmesi gerekmektedir. Bunun için OpenCV kütüphanesindeki `connectedKomponentsWithStats` fonksiyonu kullanılır. Bağlantılı Bileşenler Etiketleme (Connected Components Labeling - CCL), birçok görüntü işleme ve bilgisayarla görü işleme hattının önemli bir parçasını temsil eder. Bu alanla olan ilgisi göz önüne alındığında, en ileri teknolojiye sahip görüntü işleme kütüphanelerinin bir parçası olmuştur (Bolelli vd., 2022). `ConnectedKomponentsWithStats` görüntüdeki konturları ve kontur alanları gibi her konturla ilgili yararlı bilgileri veren bir OpenCV fonksiyonudur. `ConnectedKomponentsWithStats` fonksiyonu bir hücrenin kaç adet bağlantıya sahip olması gerektiğini kullanıcının ayarlamasına izin vermektedir. Bu değer varsayılan olarak 8 ayarlanmıştır. Bir hücrenin etrafında 8 hücreye bağlı olduğu anlamına gelir. Şekil 3.10’da merkezi hücre, hücrenin etrafında 8 hücreye bağlı olduğu gösterilir.

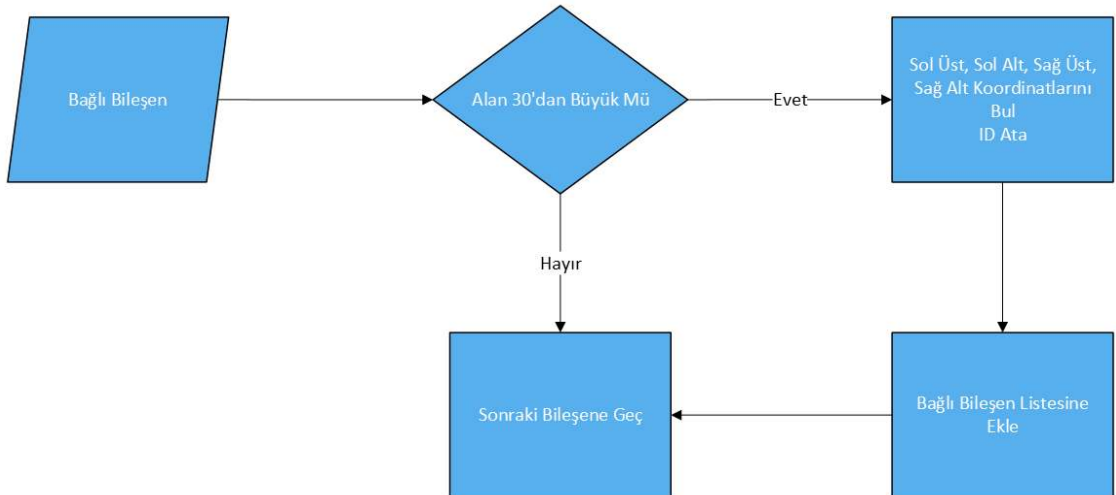
Fonksiyon çıktı olarak bağlı bileşenin 2D koordinatlarını, genişliğini, yüksekliğini ve piksel alanını döndürür (Kim vd., 2024). Gürültü veya olası görüntü bozuklukları nedeniyle algoritmanın hatalı çalışmasını engellemek için alanı belirli bir alan eşiğinin

altında olan bağılı bileşenler göz ardı edilir (Gonroudobou vd., 2022). Görüntünün ne kadar yükseklikten çekildiği yani ne kadar bitki sırasına sahip olduğu bu değeri ayarlamak için kritik öneme sahiptir. Kullanılan veri seti için bu değer 30 olarak belirlenmiştir.

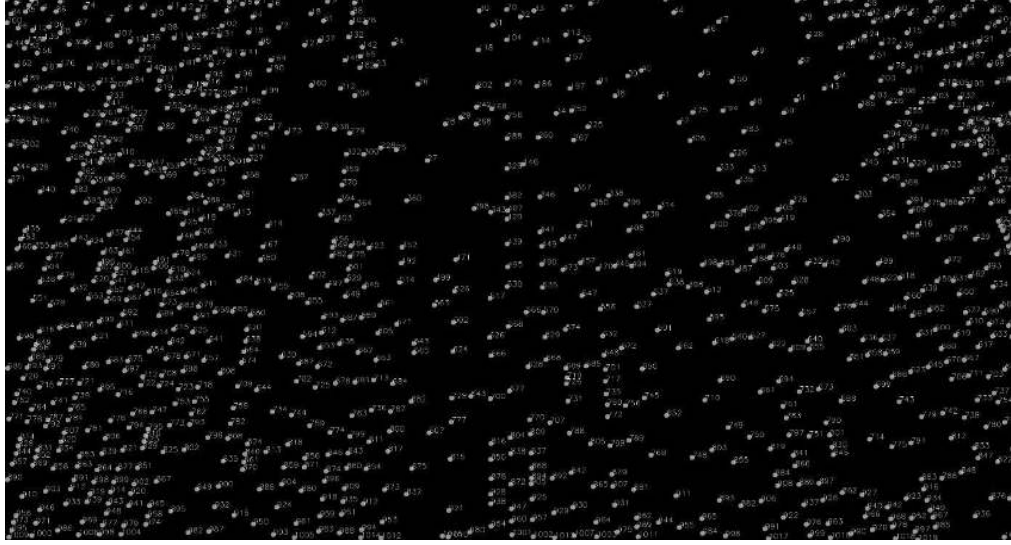
Eşik değeri olması gerektiğinden küçük bir değer olması durumunda algoritmada iterasyon sayısı artar. Görüntü ön işleme sonucunda filtrelenemeyen küçük noktalar işleme alınır ve sıraların bulunması zorlaşabilir. Filtreden geçen bağılı bileşenlerin sol üst, sol alt, sağ üst, sağ alt koordinatları hesaplanır. Merkez koordinatları kayıt altına alınır. Her bir bağılı bileşene özel ID atanır. Şekil 3.11’de bağılı bileşen (komponent) oluşturma akış şeması, Şekil 3.12’de ise elde edilen komponentler gösterilmektedir.



Şekil 3.10. Merkezi hücre ve 8 bağlantı



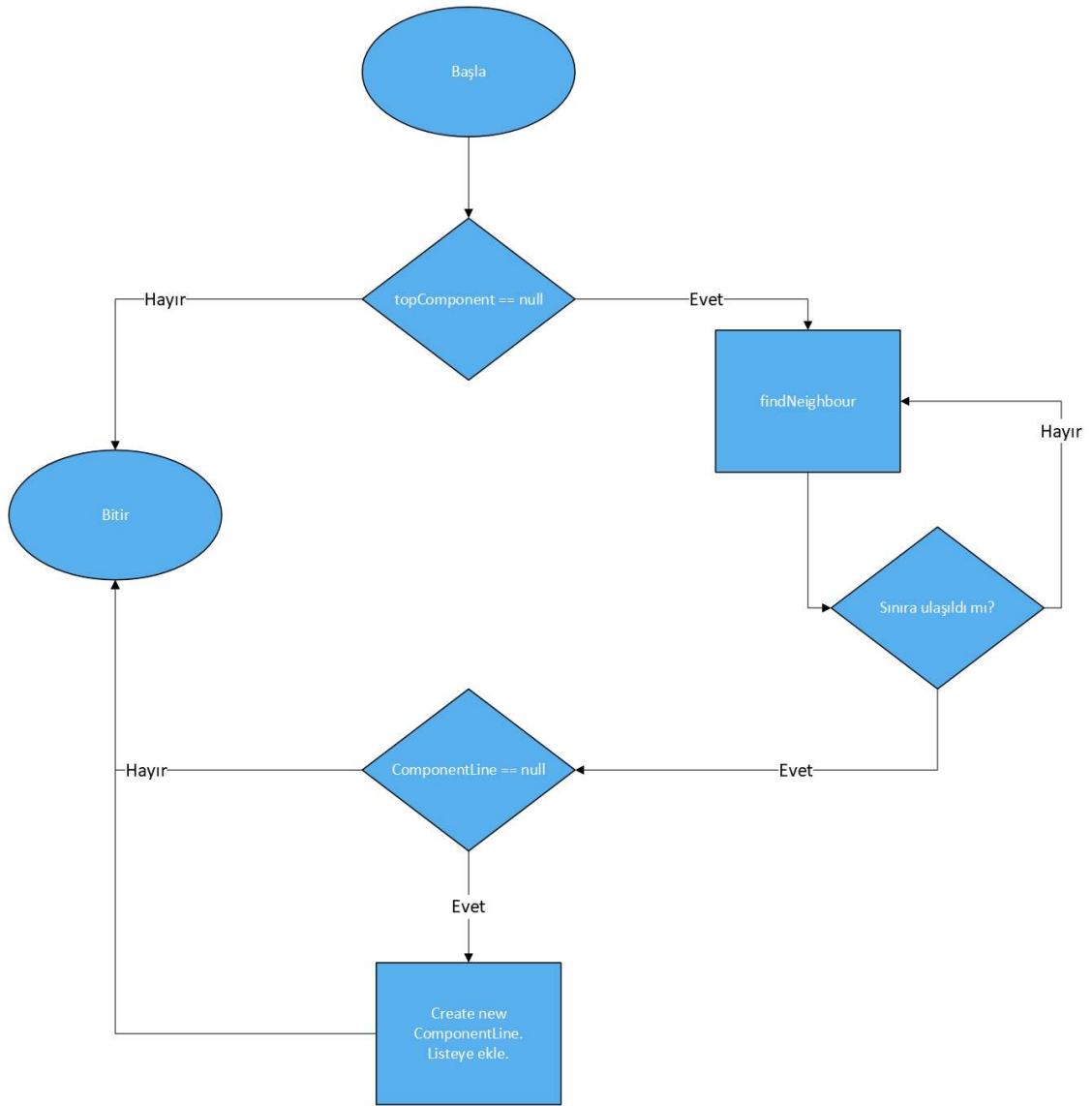
Şekil 3.11. Bağılı bileşen oluşturma şeması



Şekil 3.12. Şekil 3.9'a bağlı bileşen oluşturma adımları uygulandıktan sonra elde edilen komponentler

3.1.3. Komponent sıralarının oluşturulması

Komponentler filtrelendikten sonra kalan komponentler üzerinde belirli koşullarla en yakın komşusu bulunmaya çalışılır. Tüm komponentler bir döngü ile dolaşılır. Komponentin bağlı olduğu üst komşusu yoksa arama fonksiyonu çağrılır. Üst komşu bulunduktan sonra öz yinelemeli olarak üst komşunun üst komşusu aranır. Bu süreç son komşu bulunana kadar devam eder. Son komşu bulma koşulu komponentin görüntüde sınır bölgede yer almasına bağlıdır. Sınır bölge görüntünün en alt bölgesidir. Üst komşu ile olan mesafe görüntünün en alt noktasıyla olan mesafeden büyükse sınır bölgede olduğu tespit edilir. Şekil 3.13'te algoritmanın akış şeması yer almaktadır. Algoritma 1'de sözde kodu verilmiştir.



Şekil 3.13. Komponent sırası oluşturma şeması

Algoritma 1:

for each component in m_components

if component.topComponent is null then

findNeighbour(component)

if component.compLine is null then

newCompLine = create new Component_Line

newCompLine.m_lineIndex = size of m_componentLines

insert newCompLine into m_componentLines at position size of m_componentLines

newCompLine.append(component)

```
if component.topComponent is not null then  
    newCompLine.append(component.topComponent)
```

Komşu arama algoritması tüm komponentleri dolaşır. Komponent ile arasındaki mesafeyi ölçmeden önce aşağıda verilen 4 koşulu sağlaması beklenir:

1. Karşılaştırılan komponent kendisi değilse.
2. Karşılaştırılan komponent yasaklı komponent listesinde değilse.
3. Karşılaştırılan komponent'in en alt noktası görüntünün en alt noktasından 50 birim yukarıda yer alıyorsa.
4. Karşılaştırılan komponent'in merkez koordinatının Y noktası 50 birimden küçük değilse.

Yukarıdaki 4 koşulun sağlanması durumunda karşılaştırılan komponent ile komponent arasında konum değerlendirilmesi yapılır. Eğer karşılaştırılan komponentin sol üst köşesinin Y noktası komponentin sağ alt köşesinin Y noktasından daha büyükse yani komşu komponent daha aşağıda bulunuyorsa mesafe hesaplaması ve karşılaştırılması yapılır. Y koordinatları karşılaştırılırken küçük tolerans değeri kullanılır. Bunun sebebi komponentlerin şekli her zaman düzgün olmayabilir. Komşu komponent daha aşağıda olmasına rağmen komponentin sınırını çok küçük bir parçasıyla geçmiş olabilir. Bu durumda komşu komponenti elemek doğru bir değerlendirme olmaz.

Eğer bu koşul sağlanmıyorsa belirlenen tolerans değeri yeterli gelmemiş olabilir, bu durumda komponentlerin merkez Y koordinat değeri karşılaştırılır. Algoritma 2'de komşu bulma algoritmasının başlangıç sözde kodu yer almaktadır. Tüm komponentler dolaşarak en yakın komşu komponent aranır.

Algoritma 2:

```
function findNeighbour(comp)  
    if comp.topComponent is not null then  
        return  
    for each component in m_components  
        if component == comp then
```

```

        continue

    if comp.bannedCompList contains component then

        continue

    if comp.m_bottomLeftY > m_height - 50 then

        if comp.compLine is not null then

            comp.compLine.endComponent = comp

        continue

    if comp.m_cY < 50 then

        if comp.compLine is not null then

            comp.compLine.endComponent = comp

        continue

    if component.m_topLeftY + 17 > comp.m_bottomLeftY ☐ En yakın komşu 1. koşul durumu
        ....

    else if component.m_cY > comp.m_cY then ☐ En yakın komşu 2. koşul durumu
        ....

```

En yakın komşu bulunduktan sonra sınır bölge kontrolleri ve komponent sıra kontrolleri gerçekleştirilir.

Birinci koşul sağlandığı durumda komponent ile karşılaştırılan komponent arasındaki uzaklık aşağıda verilen Denklem 3.2'ye göre hesaplanır.

$$Uzaklık = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (3.2)$$

X ve Y koordinatları olarak komponent oluştururken hesaplanan komponentin en uç noktaları kullanılır. Komponent için sol alt koordinat, karşılaştırılan komponent için sol üst koordinat ile uzaklık ölçümü yapılır. Uzaklık ölçümünün ardından iki komponent arasındaki merkez koordinatların uzaklığı aynı formülle hesaplanır. İki komponent arasındaki açı Denklem 3.3 ile hesaplanır.

$$Açı = atan2(y_1 - y_2, x_1 - x_2) * \frac{180}{PI} + 180.0 \quad (3.3)$$

Uzaklık değeri komponentin mevcut uzaklık değerinden küçükse komşuluk atamasına başvurulur.

```
if distance < comp.m_topCompDistance then
```

```
    if assignComponent(component, neighborComponent, distance) then
```

```
        comp.m_arc = arc
```

Komşu ataması için ‘assignComponent’ fonksiyonu çağırılır. Parametre olarak ‘component’ ve komşu olarak atanmak istenilen ‘neighborComponent’ komponentlerini alır. Fonksiyon içerisinde ‘component’ için comp1, ‘neighborComponent’ için comp2 ifadeleri kullanılır. Bu fonksiyonda üst komşu olma koşulları kontrol edilir. Üst komşu olabilmek için sadece uzaklığın küçük olması yeterli değildir. Comp1 bir komponent sırasına dahilse ve bu sırada en az 2 komponent bulunuyorsa comp1 ve comp2 arasında açı kontrolü de yapılır. Comp1 hali hazırda bir üst komşuya sahipse ileride kullanmak üzere mevcut üst komşu en son üst komşu olarak comp1 içerisinde saklanır. Comp2 başka bir komponent’in üst komşusu değilse comp2’nin mevcut alt komşusu olarak comp1, comp1’in mevcut üst komşusu olarak comp2, comp1’in üst komşu uzaklığı da comp1 ile comp2’nin uzaklığı olarak atanır. Comp2 başka bir komponent’in üst komşusu ise yani başka bir komponent’e bağlı ise bağlı olduğu komponent’in doğru bir bağlantı mı olduğu kontrol edilir. Bu kontrolde comp1 ile comp2 arasındaki açı, comp2’nin mevcut alt komşu bağlantısı arasındaki açı ile karşılaştırılır. Karşılaştırma için comp2’nin bağlı olduğu komponent sırası’nın açısı ile her 2 açı değeri çıkarılır. Comp1 ile olan açı değeri comp2’nin mevcut alt komşusu ile olan açı değerinden daha küçükse comp2, comp1’in üst komşusu olarak atanır. Comp1 bağlantıdan önce bir üst komşuya sahipse bu üst komşu comp1’in yasaklı komponent listesine eklenir. Bu listeye eklemedeki amaç eğer comp1 komşu ararken aynı komponenti tekrar komşu olarak atamasını engellemektir. Comp1’a comp2 üst komşu olarak atanır.

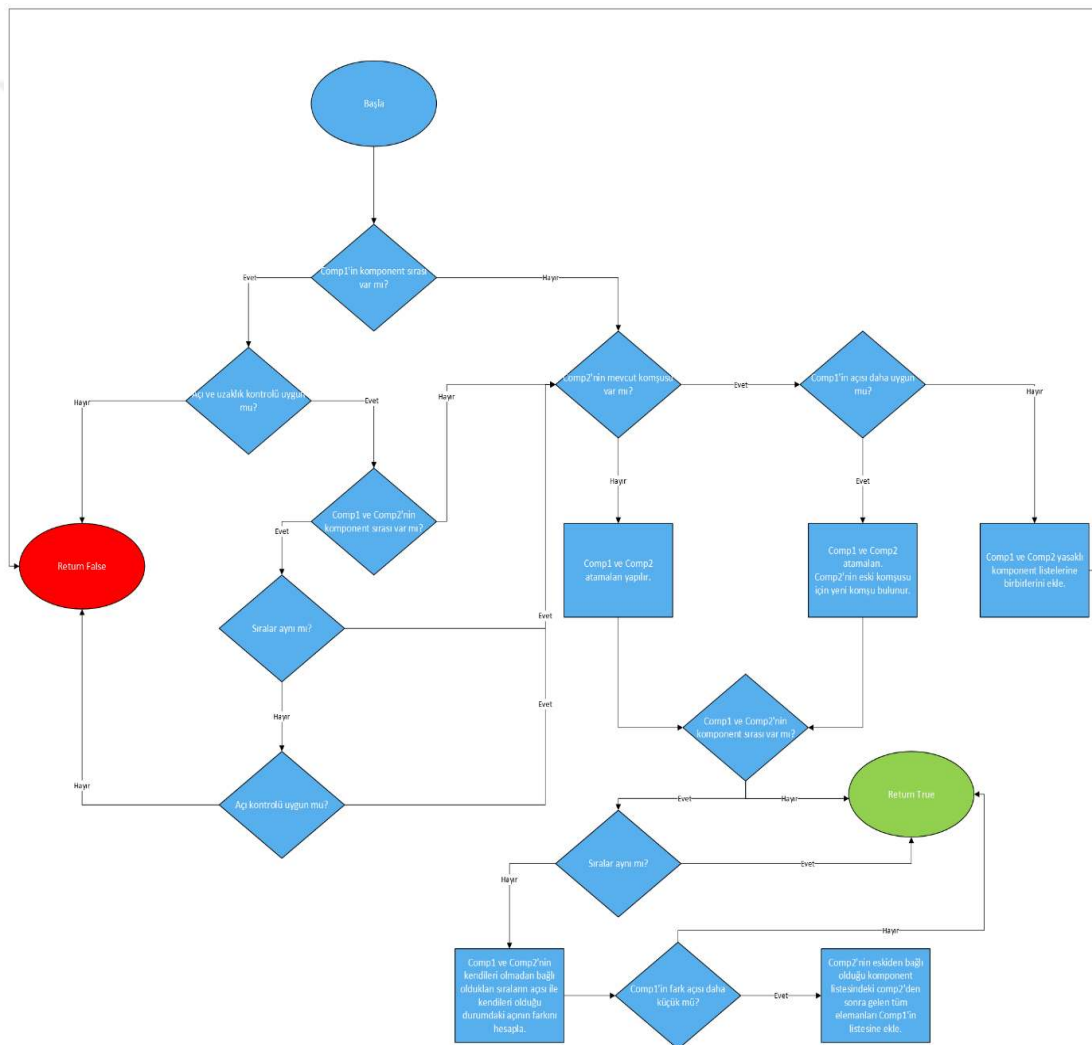
```
comp1->topComponent = comp2;
```

```
comp1->m_topCompDistance = distance;
```

```
comp2->lastBottomComponent = comp2->currentBottomComponent;
```

```
comp2->currentBottomComponent = comp1;
```

Comp2, comp1'e atandığı için comp2'nin eski bağlı olduğu komşusu açıkta kalmış oldu. Comp2'nin eski komşusunun yasaklı listesine comp2 eklenir ve comp2'nin eski komşusunun bağlı olduğu komponent sırasında sadece comp2'nin eski komşusu kaldıysa sıra silinir. Comp2'nin eski komşusu için yeni bir komşu bulma amacıyla 'findNeighbour' fonksiyonu çağırılır. Algoritmanın sadeleştirilmiş halinin akış şeması Şekil 3.14'te gösterilmektedir.



Şekil 3.14. *Komşu atama şeması*

En yakın komşu 1. Koşul durumu sağlanmadığı durumda 2. Koşul kontrol edilir. Bu kez komponentlerin merkez noktalarının Y koordinat değeri karşılaştırılır. Eğer

karşılaştırılan komşunun Y koordinat değeri daha büyükse, komponente göre daha aşağıda kalıyorsa, komponent'in bağlı olduğu komponent sırası var mı kontrol edilir. Bağlı bir komponent sırası varsa bu kez en az 2 elemana sahip olması beklenir. Bu kontrollerin sebebi merkez noktaların kullanımının hataya açık olmasıdır. Merkez noktalarını kullanabilmek için bağlı bir komponent sırasının olması beklenir çünkü komponent sırasının açısı bağlanacak komponent'in açısının %10'luk tolerans değerini aşıyor mu kontrol edilir. Açı kontrolü uygun olması durumunda uzaklık kontrolü yapılır. Uzaklık değeri mevcut değerden küçükse komşu ataması için 'assignComponent' fonksiyonu çağırılır.

Tüm komponentler dolaşıldıktan sonra döngü tamamlanır. Döngü tamamlanınca komponent komşu komponentini bulduysa en yakın üst komşu uzaklık değeri ile en yakın merkez komponentinin uzaklık değerleri karşılaştırılır. Eğer merkez uzaklık daha küçükse merkez komponent ile komşu ataması için 'assignComponent' fonksiyonu çağırılır. Merkez komponent uzaklığı ile üst komşu uzaklığı için bir kontrol daha bulunmaktadır. Bu kontrolde en yakın merkez komponent'in uzaklığının 5 katı ile komponent'in üst komşu komponentinin merkez noktasının Y koordinatı ile kendi merkez noktasının Y koordinatı arasındaki fark karşılaştırılır. Eğer en yakın merkez komponent'in uzaklığının 5 katı bu farktan azsa komşu ataması için 'assignComponent' fonksiyonu çağırılır.

Yukarıdaki işlemler tamamlandığında en yakın komşunun bulunmuş olması beklenir. Artık sınır bölge kontrolüne başlanabilir. Sınır bölge kontrolünün amacı komponentin yeni bir üst komşu bulmasına gerek olup olmadığını anlamaktır. Sınır bölgeler Şekil 3.15'te kırmızı çerçeve içerisine alınan bölgelerdir.

Sınır bölge kontrolü komponentin üst komşu uzaklığı ile sol, sağ ve alt noktalara olan uzaklığın karşılaştırılmasıyla yapılır. Sol, sağ veya alt bölgelere olan uzaklık üst komşu uzaklığından daha küçükse sınıra ulaşıldığı anlaşılır. Komponent'e üst komşusu olarak kendisi atanır. Sınıra ulaşıldığı kayıt altına alınır.



Şekil 3.15. *Sınır bölge*

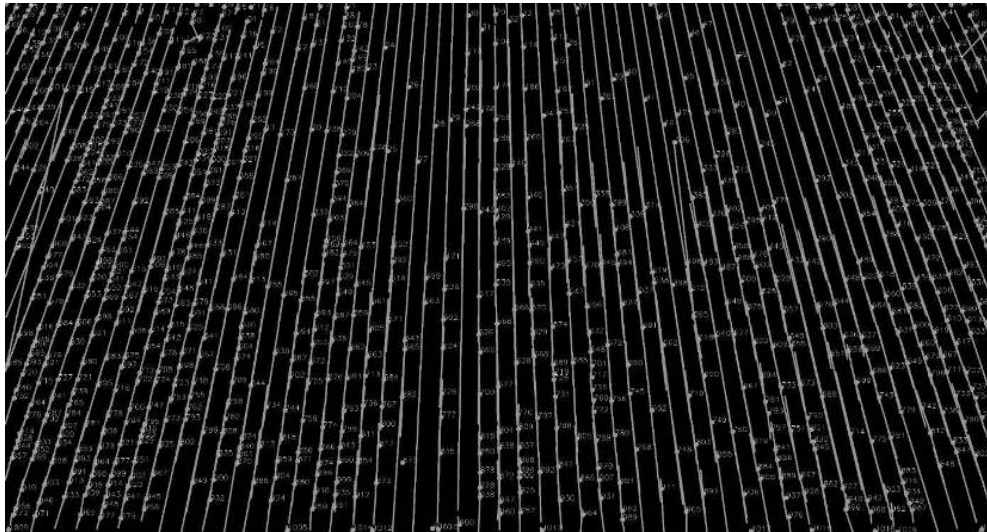
Sınır bölge kontrolünün ardından komponent sırası ayarlamasına geçilir. Komponent'in mevcut bir komponent sırası yoksa bağlı olduğu komşusunun komponent sırası kontrol edilir. Eğer bağlı olduğu komşusunun komponent sırası varsa aynı sıraya komponentte eklenir. Eğer bağlı olduğu komşusunun da komponent sırası yoksa yeni bir komponent sırası oluşturulur. Yeni oluşturulan komponent sırasına komponent ve komşusu eklenir. Komponentlerin tutulduğu haritaya yeni oluşturulan komponent sırası eklenir.

Eğer komponent bir komponent sırasına sahipse bağlı olduğu komşu komponent'in komponent sırası kontrol edilir. Komşu komponent'in bir komponent sırası yoksa komponent'in sırasına eklenir. Ancak komşu komponent farklı bir komponent sırasına bağlı ise komşu komponent olmadan sıranın açısı hesaplanır ardından komponent'in bağlı sırasına komşu komponent eklenir. Komşu komponentli sıranın açısı hesaplanır. Hesaplanan 2 açı değeri karşılaştırılır. Eğer açı farkı 1'in üzerindeyse komponent ve komşu arasında uyumsuzluk olduğu anlaşılır. Bu durumda komponent için tekrar komşu araması başlatılır. Eğer açı farkı 1'den büyük değilse komşu komponent, komponent'in sırasına eklenir.

Sınır bölge kontrolünde komponent sınır bölgede değilse ve komponent'in bir üst komşusu bulunduyorsa komşu arama fonksiyonu üst komşu için tekrar çağırılır. Bu şekilde bir sıranın en başından en sonuna kadar komşular bulunur. Sıranın son komponent'i sınır

bölge yer alacağı için bu komponentten sonra komşu arama tekrar çağırılmaz. Komponent sırası oluşturma fonksiyonundaki döngüye geri dönülür. Sonraki komponentten devam edilir. Tüm komponentler bu şekilde dolaşıldıktan sonra komponent sıraları oluşmuş olur. Oluşturulan komponent sıraları üzerinde hataların giderilmesi ve ilgilenilen bölgede bulunan komponent sıraları haricindeki sıraların silinmesi için komponent sıraları dolaşılır. Şekil 3.16'da komponent sıralarının hata düzeltmeden önceki bağlantı görüntüsü yer alır.

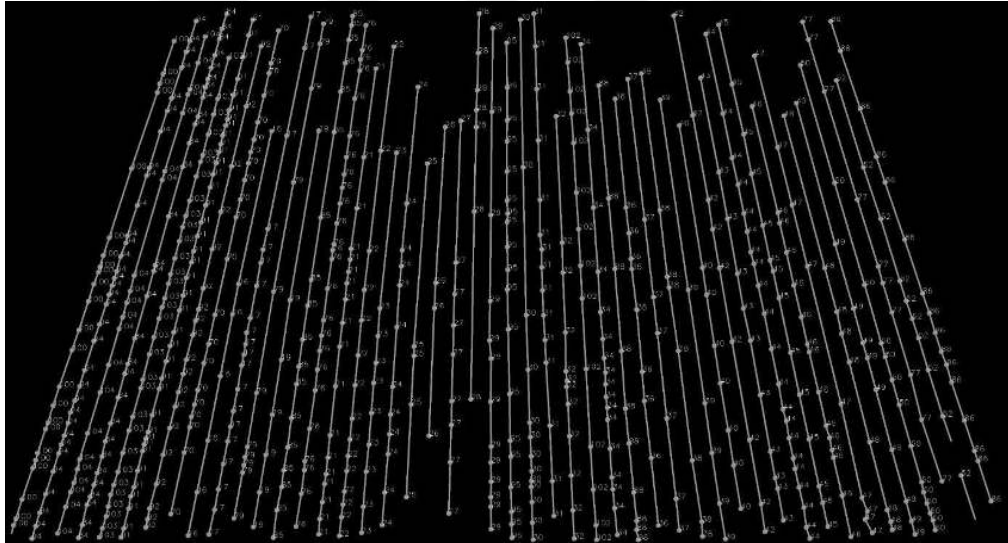
Öncelikle uzunluğu görüntü yüksekliğinin yarısından az olan komponent sıraları silinir. Komponent sırasında bulunan komponentler'in ortalama açısı hesaplanır. Komponent'in açısı ortalama açıdan %10 farklı ve uzaklığı 50'den büyük olması durumunda komponent'e yeni komşu aramak üzere findNeighbour fonksiyonu çağırılır. Yeni komşu bulunduktan sonra yeni komşu ile komponent sırasının aynı olup olmadığı kontrol edilir. Sıralar aynı değilse komponent sıra açısı ile komşu komponent ile komponent arasındaki açı karşılaştırılır. Açılar arasındaki fark %10'dan daha az ise komşu komponent sıraya eklenir. Açılar arasındaki fark %10'dan daha az değilse komponent'e yeni komşu aramak üzere findNeighbour fonksiyonu çağırılır. Komşu uzaklığı 30 üzerinde olan komponentin açısı, sıra açısının %15'den büyük veya küçük ise sıradan çıkarılır ve yeni komşu bulunması için findNeighbour fonksiyonu çağırılır. Yapılan bu kontroller sırasında komponent sırasında komponent kalmayabilir. Bu durumda boyut kontrolü yapılır ve komponenti kalmayan sıralar silinir.



Şekil 3.16. Hata düzeltmeden önce komponentlerin bağlantısı

Komponent sırasının başlangıcı görüntünün sol çeyreği içerisindeyse bitiş noktasının X koordinat başlangıç noktasının X koordinatından büyük olmamalı. Eğer bitiş noktasının X koordinatı başlangıç noktasından daha büyükse sıranın eğitimde uyumsuzluk olduğu anlaşılır. Bu durumda sıra silinir. Eğer sıranın başlangıç noktası sağ çeyrekte ise bitiş noktasının X koordinatı başlangıç noktası küçük olmamalı. Eğer bitiş noktasının X koordinatı başlangıç noktasından daha küçükse sıranın eğitimde uyumsuzluk olduğu anlaşılır. Bu durumda sıra silinir.

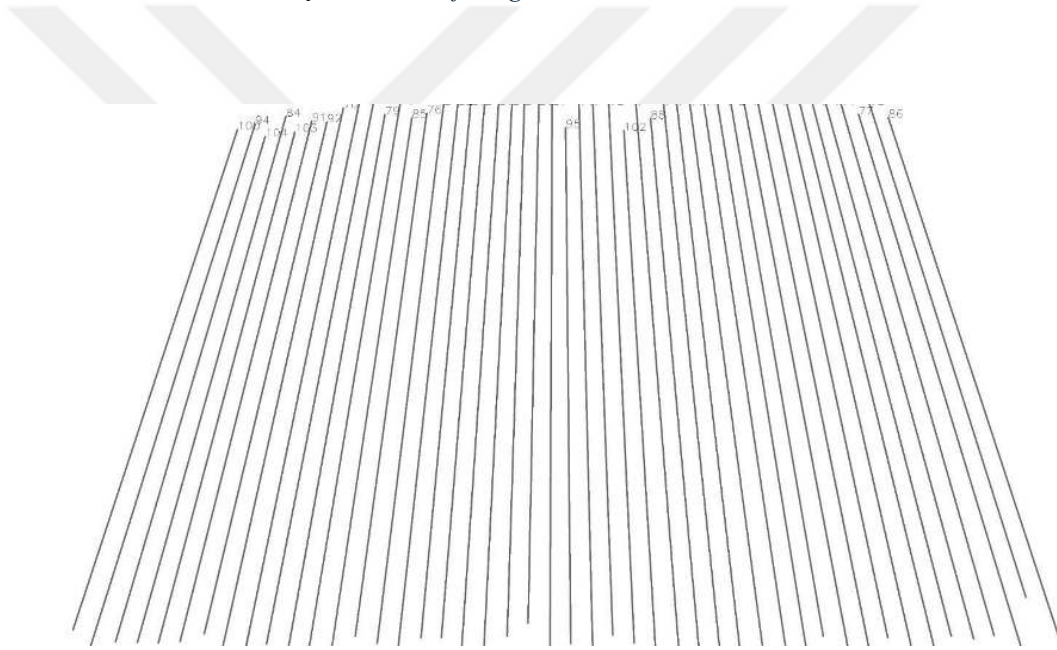
Komponent sırasının açısı değerine göre Y koordinatının en yüksek değerindeyken X koordinatının izdüşümü hesaplanır. Hesaplanan izdüşüm X koordinat değeri köşe noktaların üzerindeyse sıra silinir. Hatalar düzeltildikten sonra Şekil 3.17’de komponent sıralarının ikili görüntü formatında görüntüsü, Şekil 3.18’de ise gerçek görüntü üzerindeki görüntüsü, Şekil 3.19’da izdüşüm görüntüsü gösterilir.



Şekil 3.17. Hataları düzeltilmiş komponent sıraları



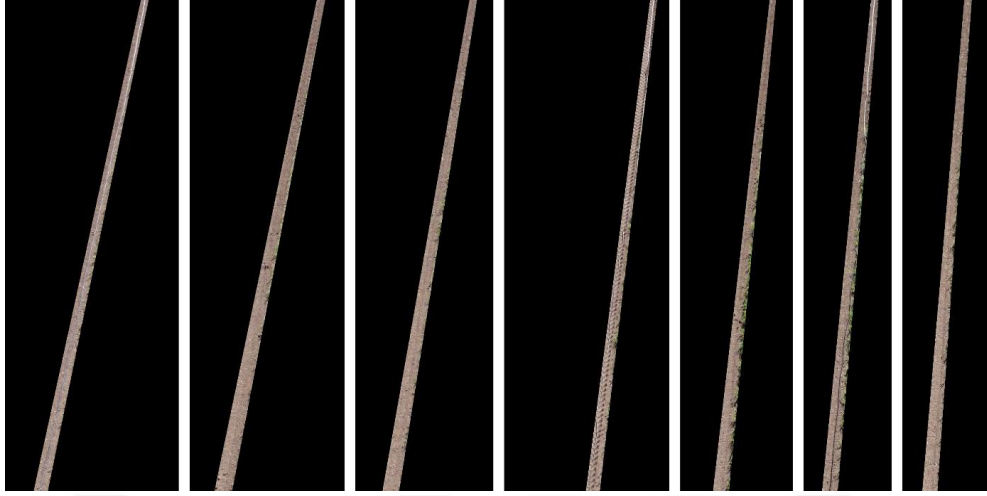
Şekil 3.18. *Orjinal görüntü üzerinde bitki sıraları*



Şekil 3.19. *Bitki sıralarının izdüşüm görüntüsü*

3.2. Damlama Borularının Tespit Edilmesi

Bitki sıralarının bulunmasıyla birlikte iki bitki sırasının arası kontrol edilerek damlama borusu tespit edilir. Öncelikle komponent sıraları kullanılarak Şekil 3.20'deki gibi iki bitki sırası arasındaki görüntüler elde edilir. Bu görüntüler üzerinde damlama borularını ön plana çıkaracak görüntü ön işleme teknikleri uygulanır. Ön işleme tekniklerinin ardından bitki sırası bulmada kullanılan komponent oluşturma ve komponent sıralarının oluşturulması algoritmaları çalıştırılır. Sonuç olarak damlama borularının bitki sıraları gibi tespit edilmesi sağlanır.



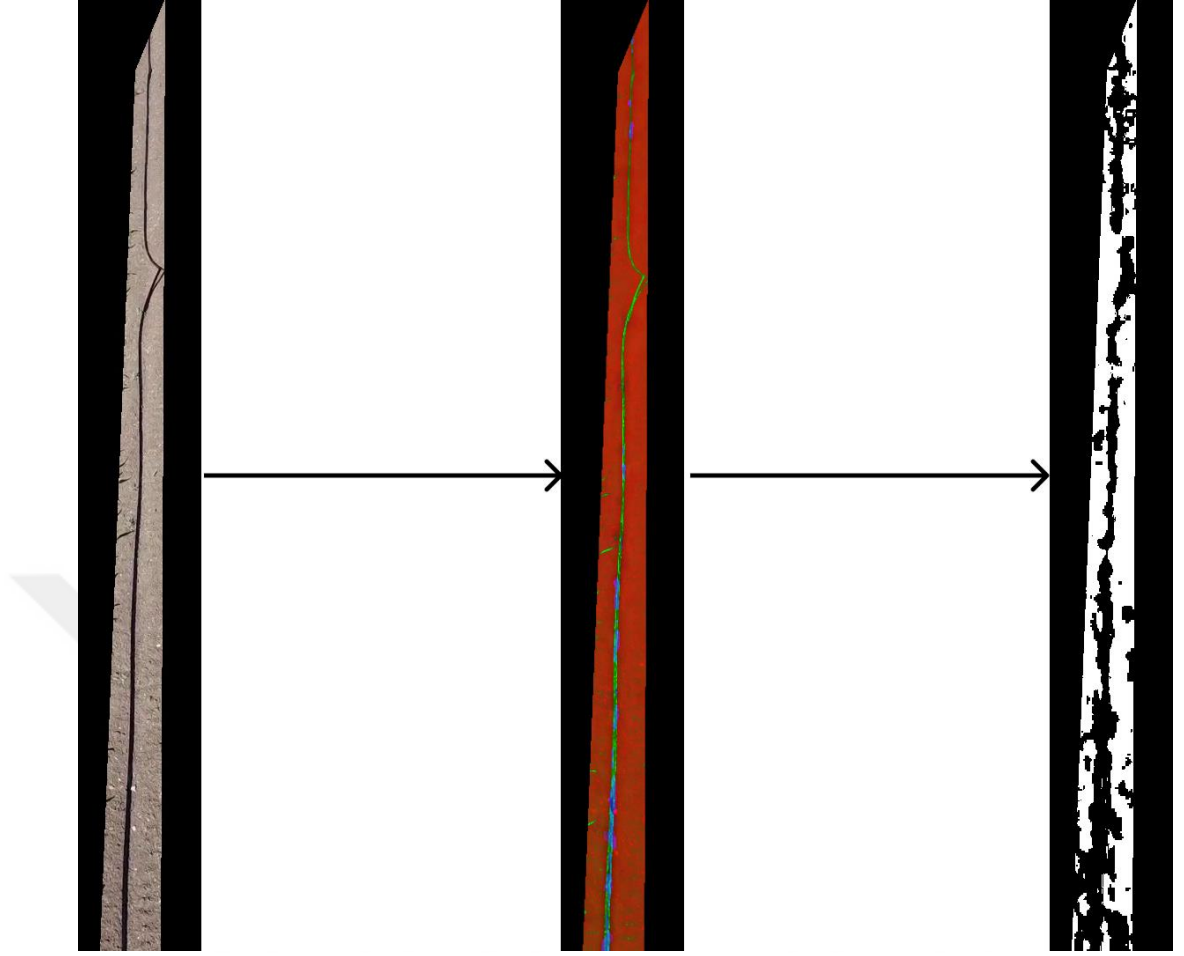
Şekil 3.20. 2'li bitki sıraları

3.2.1. Görüntü ön işleme

Damlama borusunun ön plana çıkarılması bitki sırasına göre daha zorlayıcıdır. Zorlayıcı olmasının sebepleri aşağıda listelenmiştir:

- Damlama boruları, bitki sıralarına göre güneş ışığının geliş açısına daha hassastır. Bitki sıraları yeşil renk tonlarında olduğu için HSV dönüşümüyle uygulanan renk eşikleme tekniği bitki sıralarını kolayca görünür hale getirmektedir. Ancak damlama boruları mat ve parlak renkler içerebildiği için tek bir renk eşiklemesi uygulamak damlama borusunu ön plana çıkarmak için yeterli olmamaktadır.
- Mısır bitkilerinin gölgesi damlama borularının rengine yakın olabilmektedir.
- Toprak rengi damlama borusunun rengine yakın olabilmektedir. Bu sorun daha çok traktör teker izinin olduğu bitki sıraları arasında gözlemlenir.

Damlama borusunu ön plana çıkarabilmek için ilk başta renk uzayı RGB'den HSV'ye dönüştürülür. Şekil 3.21'deki HSV uzayına çevrilmiş bitki sırası görüntüsünde alt kısımda kalan borunun mat renge, üst kısımda kalan kısmının ise parlak renge sahip olduğu Şekil 3.21'e göre daha kolay anlaşılmaktadır.



Şekil 3.21. Gerçek görüntünün HSV renk uzayına dönüşümü ve ardından renk eşiklemesinin uygulanması

Komponent sıraları arasındaki bölgelerde bitki sıraları bulunabilmektedir. Bu yüzden HSV renk uzayına dönüştürülmüş görüntü üzerinde renk eşikleme yapılarak yeşil renkli kısımların tespit edilmesi sağlanır. Şekil 3.21’de Denklem 3.4’te verilen değerlere sahip renk eşikleme formülü uygulanmış görüntü verilmiştir.

$$g_{(i,j)} = \begin{cases} 255 & \text{eğer } 11 \leq H_{(i,j)} \leq 151 \text{ ve } 0 \leq S_{(i,j)} \leq 255 \text{ ve } 0 \leq V_{(i,j)} \\ & \leq 255 \\ 0 & \text{diğer durum} \end{cases} \quad (3.4)$$

Orijinal görüntü HSV renk uzayına dönüştürüldükten sonra Şekil 3.22’de verilen gri renk uzayına dönüştürülmüş görüntü üzerinde yeşil renk eşikleme yapılmış görüntüdeki yeşil renk olan kısımlar siyah yapılır.



Şekil 3.22. Gri renk uzayı ve yeşil renk eşikleme yapılmış görüntülerin işlenmesi

Yeşil alanların filtrelediği görüntüye GABOR filtreleme uygulanarak damlama borusunun vurgulanması ve gürültünün azaltılması sağlanır. Şekil 3.23'te GABOR filtre uygulanmış görüntü yer almaktadır.

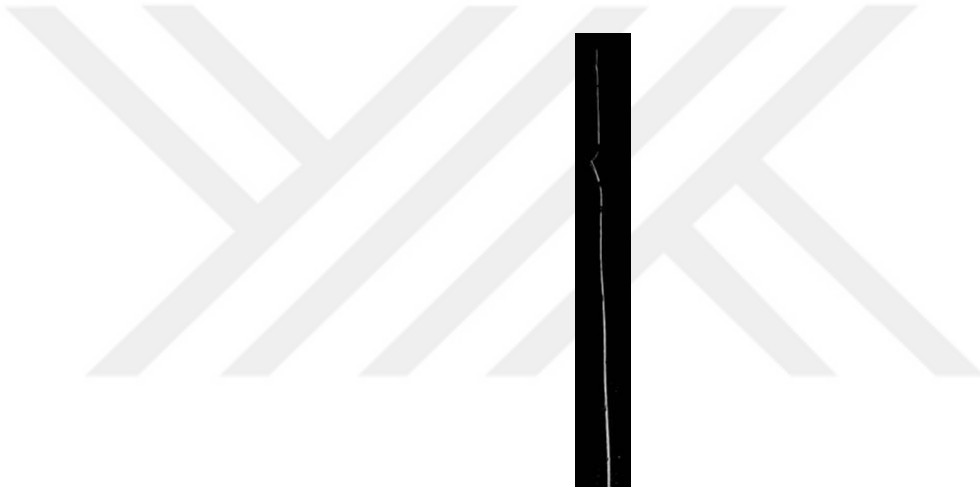


Şekil 3.23. GABOR filtresinin uygulanması ile elde edilen görüntü

Algoritmanın çalışması sırasında ilgilenilen alan olan damlama borularının beyaz olması istendiği ve bunların bulunabilmesi için Şekil 3.24'te GABOR filtresi uygulanmış görüntü üzerinde renkler tersine çevrilir.

Tersine çevrilmiş görüntüde toprak renginin damlama borusu rengi ile karışmasını azaltmak görüntü öncelikle YCrCb renk uzayına dönüştürülür. Daha sonra Denklem 3.5'teki renk eşikleme fonksiyonu uygulanır. Şekil 3.25'te Denklem 3.5 uygulanmış görüntü görülmektedir.

$$g_{(i,j)} = \begin{cases} 255 & \text{eğer } 155 \leq Y_{(i,j)} \leq 255 \text{ ve } 0 \leq Cr_{(i,j)} \leq 255 \text{ ve } 0 \leq Cb_{(i,j)} \leq 255 \\ 0 & \text{diğer durum} \end{cases} \quad (3.5)$$



Şekil 3.24. Renklerin ters çevrilmesi ile elde edilen görüntü



Şekil 3.25. Renk eşikleme sonucunda gürültüden arındırılmış görüntü

Elde edilen bu görüntü üzerinde OpenCV'nin 'connectedKomponentsWithStats' fonksiyonu çağırılır. Fonksiyon çıktısında elde edilen komponentlerin alanları toplanır. Toplam alan belirlenen eşik değerinden küçükse bu bitki sıralarının arasında damlama borusu olmadığına karar verilir ve bir sonraki görüntüye geçilir. Eğer toplam alan eşik değerinden büyükse damlama borusu olduğuna karar verilir.

3.2.2. Komponentlerin oluşturulması

Damlama borularının komponentlerini oluşturma bitki komponentlerinin oluşturulmasıyla benzer adımları içerir. Damlama borusu için yapılan görüntü ön işleme sonucu elde edilen görüntüde çok daha az komponent bulunduğu için bitki sırası komponenti oluşturmada farklı olarak komponentin alanına göre yapılan filtreleme için eşik değeri daha düşüktür.

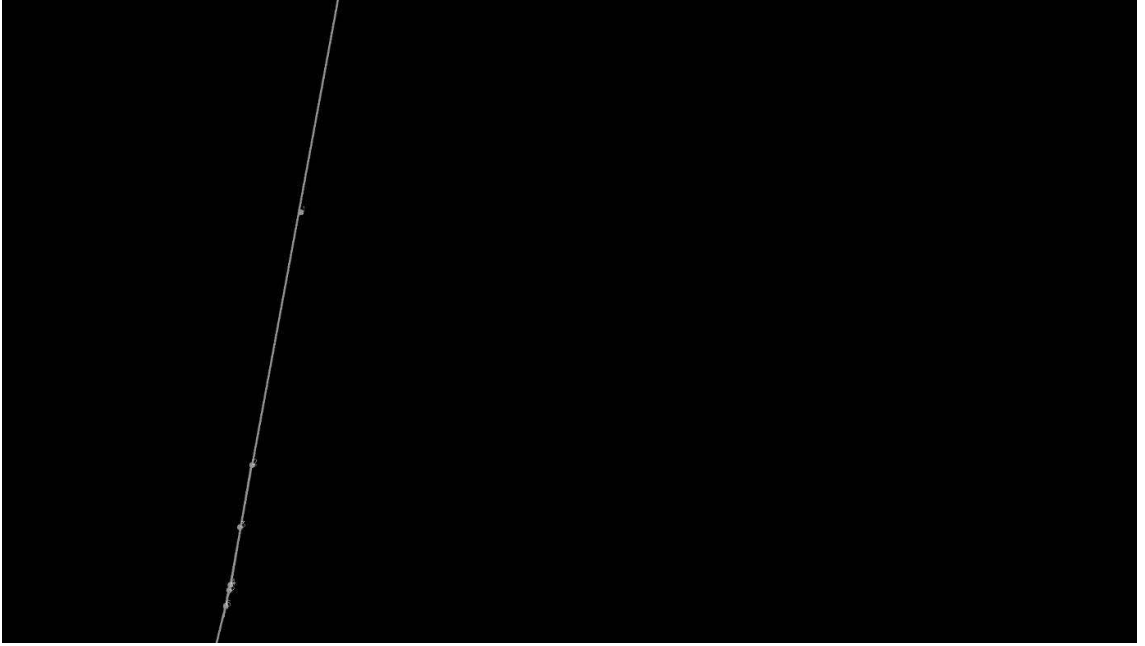
3.2.3. Komponent sıralarının oluşturulması

Damlama borularının komponent ve komponent sıralarının oluşturulması için bitki sırası oluşturulmasında kullanılan algoritma kullanılır. Bitki sıralarından farklı olarak görüntü tüm damlama borularını kapsamadığı için daha az komponent bulunmaktadır. Her bir görüntüde 1 tane damlama sırası bulunması beklendiği için komponent sırasındaki hataları düzeltmek daha kolaydır.

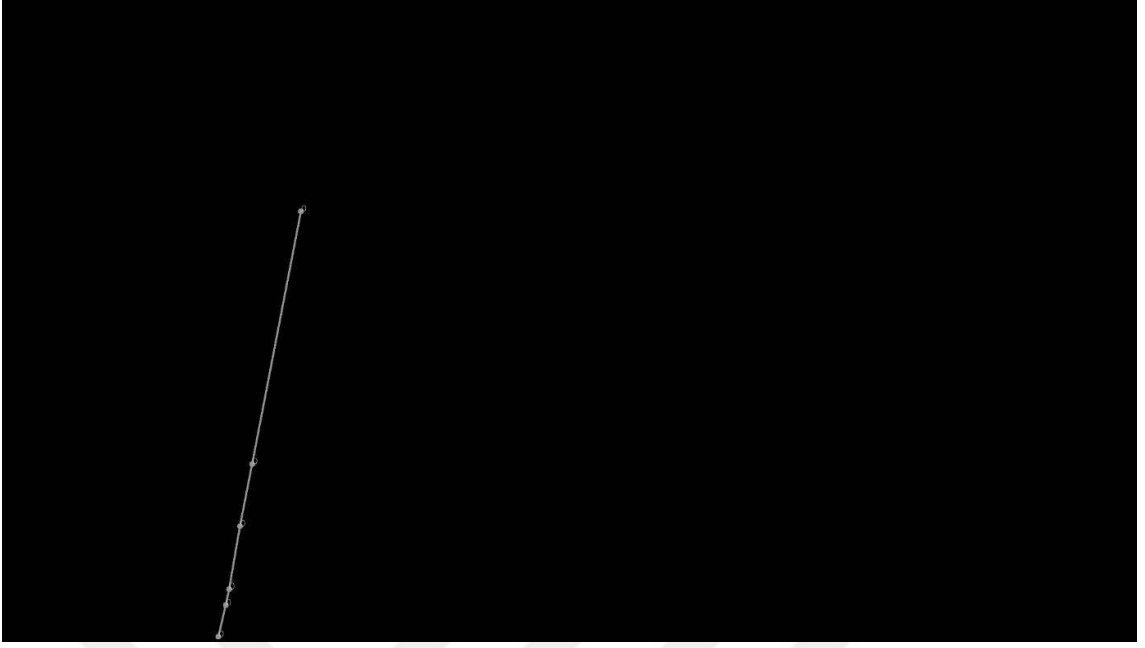
Şekil 3.34'te bitki sırası bulma işlemi sonucunda elde edilen bitki sıraları arasında kalan bir görüntü görülmektedir. Bu görüntüdeki bitki sıraları arasında damlama borusu bulunmaktadır. Şekil 3.35'te görüntü ön işlemenin ardından elde edilen komponentlerin komşuluk ilişkileri görülmektedir. Şekil 3.36'da bu komponentlerin oluşturduğu komponent sırası görülmektedir. Şekil 3.37'de komponent sırasının girdi olarak verilen görüntü üzerindeki örtüşmesi görülmektedir.



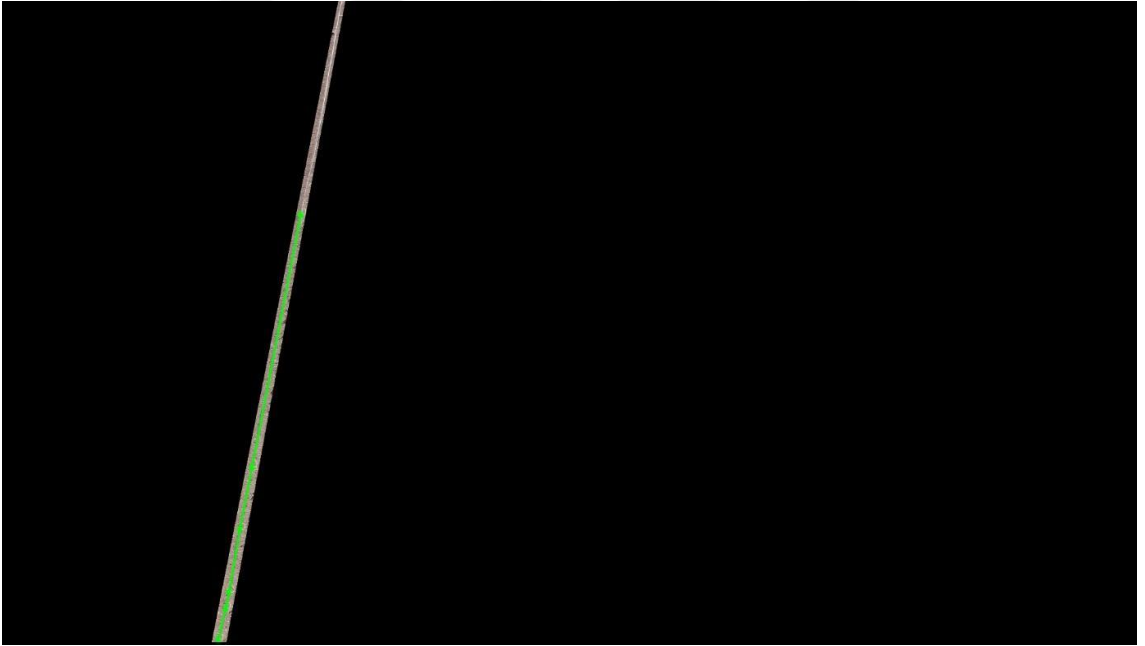
Şekil 3.26. *Bitki sıra arası görüntüsü*



Şekil 3.27. *Komponentlerin komşuluk görüntüsü*



Şekil 3.28. *Komponent sırası görüntüsü*



Şekil 3.29. *Gerçek görüntü ile komponent sırasının örtüştürülmesi*

4. DENEYSEL ÇALIŞMALAR VE TARTIŞMA

Bu bölümde geliştirilen bitki sırası ve damlama boru hatlarını bulma algoritmasının bu çalışma kapsamında oluşturulan veri seti üzerindeki çalışma sonuçları sunulmaktadır. Önceki bölümlerde algoritmanın nasıl çalıştığı özetlenmiştir ve deneysel sonuçlar aşağıda tartışılmıştır.

Önerilen yöntem, veri setindeki tüm görüntüler üzerinde tek seferde otomatik koşturulmuştur. Geliştirilen algoritma deney sonuçlarında bitki sıralarının bulunması ve damlama boru hatlarının bulunması konusunda ayrı ayrı kesinlik (ing. precision), duyarlılık (D, ing. recall) ve F1 skoru gibi standart değerlendirme ölçütleri kullanılmıştır. Test ortamında, Intel(R) Core (TM) i5-8265U, 1.60GHz işlemci ve 16 GB RAM'e sahip bilgisayar kullanılmıştır. Bu proje C++ dilinde OpenCV kütüphanesi kullanılarak geliştirilmiştir. OpenCV açık kaynak kodlu görüntü işleme kütüphanesidir. OpenCV sahip olduğu hazır fonksiyonlar sayesinde görüntü işleme algoritmalarının geliştirilmesinde kolaylık sağlamıştır.

Tablo 4.1’de önerilen yöntemin bitki sırası bulma performansı verilmiştir.

Önerilen yöntemin bitki sıralarını tespit etme sonuçları incelendiğinde farklı açılardan ve yüksekliklerden elde edilen görüntülerde çoğunlukla doğru sonuçlar ürettiği görülmüştür. Bitki sırasının yok denecek kadar az olduğu durumda bitki sırasını bulamadığı görülmüştür. Şekil 4.1’de 10. ve 11. bitki sıraları arasında bir bitki sırası olmasına rağmen bitkilerin büyük çoğunluğu olmadığı için algoritma bitki sırası tespit edememiştir. Bu görüntüden 1 sn sonra elde edilen Şekil 4.2’deki görüntüde bu bitki sırasında az miktarda bitki eklenmesiyle tespit edilemeyen bitki sırasının tespit edilebildiği görülmüştür.

Tablo 4.1. Bitki sırası bulma yöntemi karışıklık matrisi

Bitki Sırası Bulma		Tahmini Değerler	
		Pozitif	Negatif
Gerçek Değerler 4060	Pozitif	4001	59
	Negatif	41	-



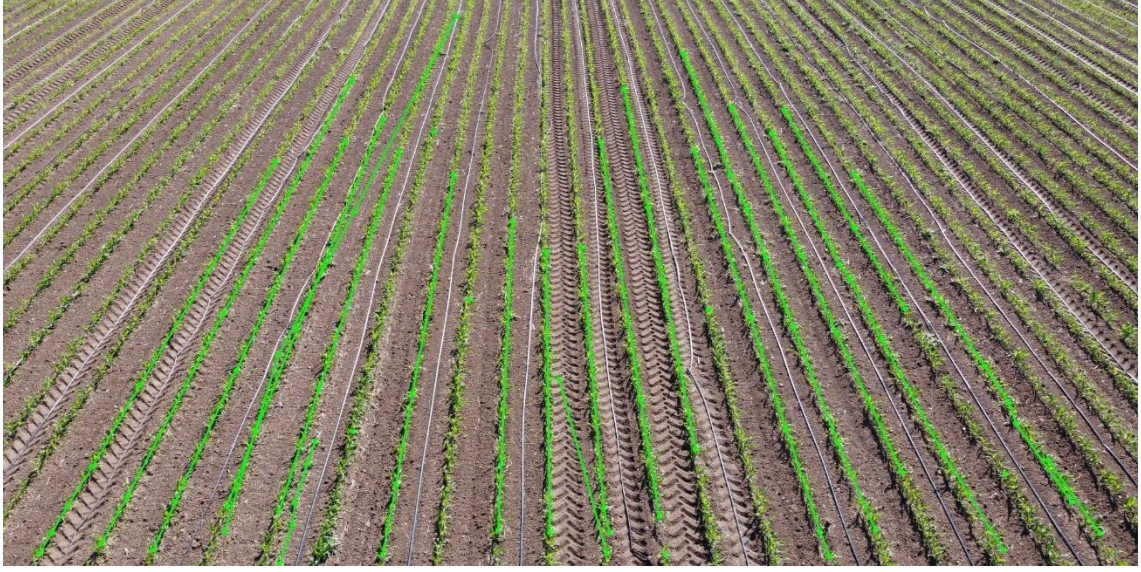
Şekil 4.1. Bitki sırası bulunamayan görüntü



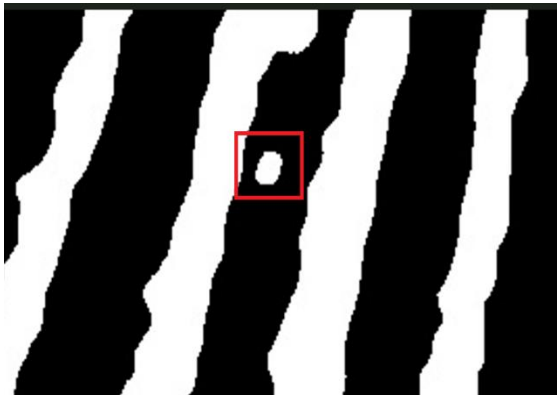
Şekil 4.2. Bitki sırasının bulunabildiği görüntü

Önerilen yöntem, kural tabanlı bir algoritma olduğu için olası tüm durumları ele almak mümkün olmayabilir. İHA'nın görüntüyü çekim açısı, çekim yüksekliği, çekim anındaki güneş ışınlarının açısı ve bitki büyüklüğü gibi çeşitli faktörler algoritmanın çalışmasını etkilemektedir. Koşum sonuçlarına bakıldığında önerilen yöntemin farklı açılarda, yüksekliklerde ve güneş ışın açısında çoğunlukla doğru çalıştığı görülmüştür. Ancak doğru sonuç veremediği görüntülerde bulunmaktadır. Şekil 4.3'te 62 numaralı

bitki sırasının hatalı bağlandığı görülmektedir. Bu hatanın sebebi Şekil 4.4a ve b’de kırmızı çerçeve içerisine alınmış component olduğu görülmüştür. Gerçek görüntü incelendiğinde bu componentin bir yabancı ot sebebiyle ayrı bir component olarak değerlendirildiği anlaşılmaktadır. Bu görüntü sayesinde önerilen kural tabanlı algoritmanın yeşil renkli oldukları için yabancı otlara karşı zafiyeti olduğu anlaşılmaktadır.



Şekil 4.3. Hatalı bitki sırası bulunan görüntü



(a)

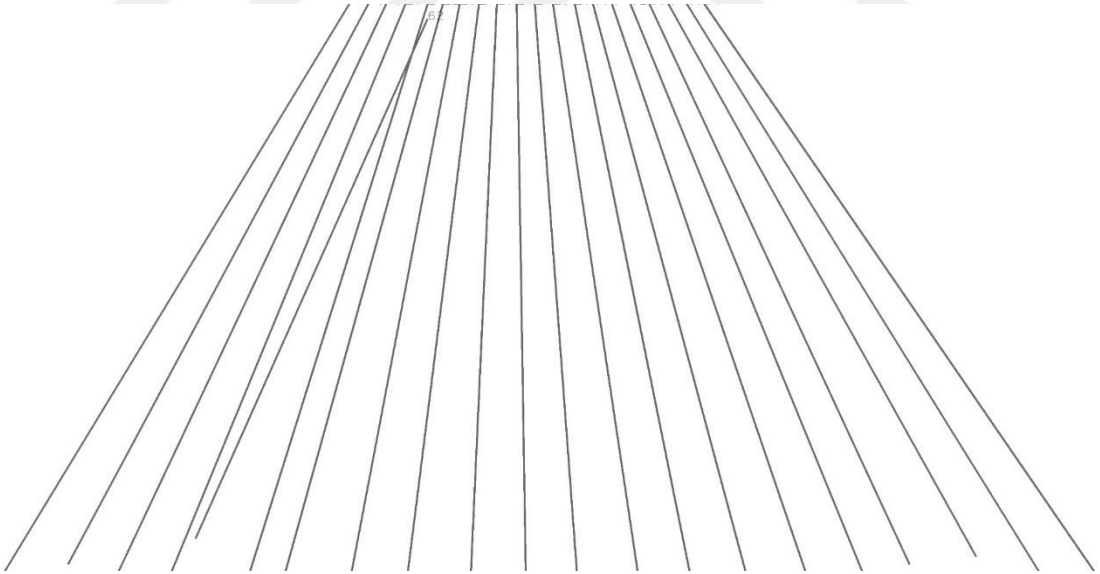


(b)

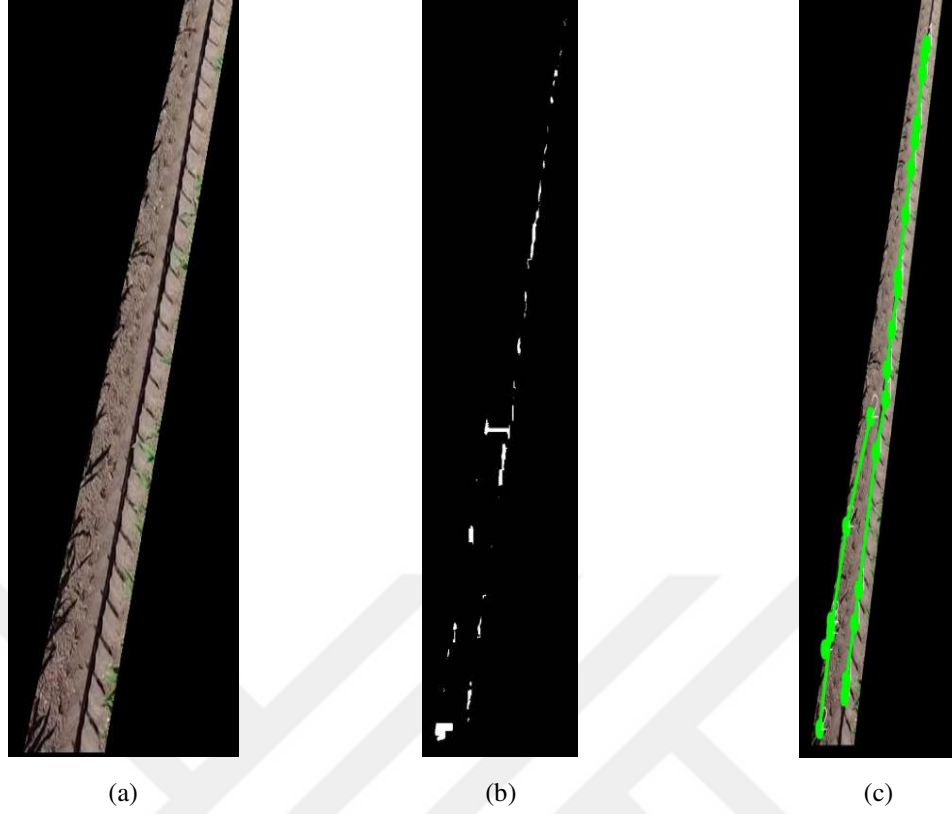
Şekil 4.4. Yabancı ot bulunan sıra arasının (a) görüntü ön işleme sonucu, (b) gerçek görüntüsü

62. sıra haricinde 27. sırada da bir farklılık olduğu görülmektedir. Algoritma geliştirilirken bir komponentin bir sıraya dahil olduğu durumda bu komponente bağlanacak diğer komponentler için açı ve mesafe uygunluk kontrolleri yapılır. Bu kontroller sayesinde sıra üzerinde olmayan bir komponentin sıraya eklenmesi durumunda Şekil 4.5'te görülebileceği üzere sıranın hizalaması değişmemektedir.

Bitki gölgelerinin ve damlama borularının siyah renkli olması algoritmanın doğru sonuç vermesini zorlaştırmaktadır. Şekil 4.6a'da 2 bitki sırası arasında bulunan damlama borusu ile bitki gölgeleri görülmektedir. Şekil 4.2b'de görüntü ön işleme sonucunda elde edilen görüntüde sağ taraftaki bitki sırası gölgesinin tıpkı damlama borusu gibi komponentlerinin oluştuğu ve Şekil 4.2c'de bu komponentlerin ayrı bir damlama borusu sırası elde edilmesine neden olduğu görülmüştür. Bu durum önerilen kural tabanlı algoritmanın bitki sırasında olduğu gibi güneş ışınlarının geliş açısına ve bitki büyüklüğüne karşı toleransının zayıf olduğunu ancak var olan damlama borularının tespit edilmesinde iyi performans sergilediğini göstermektedir.



Şekil 4.5. Hatalı sıra bulunan görüntüdeki bitki sıralarının izdüşüm görüntüsü



Şekil 4.6. Gölge içeren bitki sırasının (a) gerçek, (b) görüntü ön işleme, (c) komponent sırası görüntüleri

Tablo 4.2’de önerilen yöntemin damlama boru hattı bulma performansı verilmiştir. Tablo 4.1 ve Tablo 4.2’deki sonuçlara istinaden Tablo 4.3’te bitki sırası bulma algoritması ve damlama boru hattı bulma algoritması için ayrı ayrı duyarlılık, kesinlik ve F1 skoru hesaplanmıştır.

Tablo 4.2. Damlama boru hattı bulma yöntemi karışıklık matrisi

Damlama Boru Hattı Bulma		Tahmini Değerler	
		Pozitif	Negatif
Gerçek Değerler 759	Pozitif	656	103
	Negatif	99	-

Tablo 4.3. Önerilen yöntemin performansı (%)

Yöntem	Duyarlılık	Kesinlik	F1 Skoru
Bitki Sırası Bulma	99.01	98.55	98.78
Damlama Boru Hattı Bulma	86.54	86.88	86.71

Geliştirilen algoritmanın tüm görüntüler üzerindeki bitki sırası bulma sonuçları Ek 2’de verilmiştir. Bitki sıralarının hatasız bulunduğu görüntüler üzerinde damlama borusu tespit edilmeye çalışılmış ve sonuçları Ek 3’te aktarılmıştır.

Damlama borularının, görüntü ön işleme adımında bitki sıraları kadar arka plandan iyi ayrıştırılamadığı gözlemlenmiştir. Arka plandan iyi ayrıştırılabilen damlama borularında bitki sırası bulma algoritması benzer şekilde koşturulmuş ve damlama sırasının anomali tespiti için yeterli hale geldiği görülmüştür. Ancak iyi ayrıştırılamayan damlama boruları çok sayıda küçük komponent içerdiği için anomali tespitinde çok sayıda 1. ve 2. anomali grubu (kısa ve orta kopukluklar) oluştuğu görülmüştür. Çok sayıda oluşturulan bu anomaliler kesinlik değerini oldukça düşürmüştür. Öte yandan var olan anomalilerin büyük çoğunluğu tespit edilmiştir. Anomali tespitinin duyarlılığının yüksek olduğu görülmüştür. Tablo 4.3’ten de anlaşılabileceği üzere önerilen yöntem en iyi performansı bitki sırası bulmada göstermiştir. Veri setindeki görüntüler farklı açı, yükseklik ve ışık içermesine rağmen önerilen bitki sırası bulma algoritması ön işleme teknikleri ve komşu ilişkisine dayalı sıra oluşturma yöntemi sayesinde yüksek başarımla göstermiştir.

Literatürde damlama boru hattı bulma ve damlama sıralarındaki anomali tespiti içeren çalışma bulunmadığı için sadece bitki sırası bulma performansı karşılaştırılmıştır. Tablo 4.4’te literatürde bitki sıralarını bulan benzer çalışmalar ile karşılaştırma listesi verilmiştir.

Önerilen yöntemin bitki sırası bulma algoritması ve damlama boru hattı bulma algoritması için harcanan süreler Tablo 4.5’te çıkarılmıştır.

Tablo 4.4. Literatürde bitki sıralarını bulan benzer çalışmalar ile karşılaştırma

Referans	Veri Seti	Yöntem	Performans (%)
(Guo vd., 2024)	Özel mısır veri seti	Derin öğrenme ağı	D: 96.50
	7 günlük pirinç		D: 93.91
(Ruan vd., 2023)	14 günlük pirinç	YOLO-R	D: 95.87
	21 günlük pirinç		D: 89.87
(De Silva vd., 2023)	11 farklı özel veri seti	Üçgen tarama, U-Net	D: 82.13-90.00
(Pei vd., 2022)	Özel mısır veri seti	YOLOv4	D: 86.89
(Simon vd., 2021)	Özel mısır veri seti	KNN	D: 95.00
(Vong vd., 2021)	Özel mısır veri seti	R-CNN	D: 95.80
		Line Hough Dönüşümü	D: 80.00 – 100.00
		Dual Line Hough Dönüşümü	D: 75.00 – 100.00
(Winterhalter vd., 2018)	Özel kanola, mısır, pırasa ve şeker pancarı veri seti	Pattern Hough Dönüşümü	D: 85.00 – 100.00
		RANSAC	D: 50.00 – 100.00
(Romeo vd., 2012)	Özel mısır veri seti	Renk eşikleme, Segmentasyon	D: 97.00
Önerilen Yöntem (Bitki Sırası Bulma)	Özel mısır veri seti	Renk eşikleme, kenar bulma	D: 99.01

Tablo 4.5. Önerilen Yöntemin Çalışma Performansı

Yöntem	Ortalama Çalışma Süresi (ms)	Minimum Çalışma Süresi (ms)	Maksimum Çalışma Süresi (ms)
Bitki Sırası Bulma	1164	528	14288
Damlama Boru Hattı Bulma	683	320	6189

5. SONUÇLAR

Bu çalışmada, yeni ve etkili bir bitki sırası tespit algoritmasının geliştirilmesinin yanı sıra, literatürdeki çalışmalardan farklı olarak İHA kullanılarak elde edilen görüntülerde damlama boru hatlarını tespit eden yeni bir yöntem sunulmaktadır. Ayrıca, İHA ile mısır tarlalarından alınan görüntülerle yeni bir veri seti oluşturulmuş ve bu veri seti diğer araştırmacıların kullanımına sunulmuştur. Bu veri setindeki görüntüler üzerinde, önerilen yöntemin ilk adımı olan bitki sıralarının belirlenmesi için renk eşikleme ve çeşitli morfolojik görüntü işleme teknikleri uygulanmıştır. Ardından, elde edilen çizgi parçaları, açıları ve aralarındaki mesafelere göre birleştirilerek bitki sırasına ait bütün hat bulunmuştur. Bitki sıralarının tespit edilmesinden sonra, bu sıralar arasındaki alanlarda damlama borularının belirlenmesi için farklı görüntü ön işleme teknikleri ve çizgi parçaları arasındaki komşuluk ilişkilerinden faydalanılmıştır.

Önerilen yöntem bitki sıralarının ve damlama boru hatlarının aynı anda bulunması için herhangi bir değişiklik yapılmadan kullanılabilir. Ayrıca bitki veya damlama sıralarının kıvrımlı olmasına karşı komponentleri bağımsız değerlendirdiği için esnek çalışabilmektedir. Farklı yüksekliklerden çekilen görüntülerde herhangi bir ayar yapmadan çıktı sağlayabilmektedir. Ancak güneş ışınlarının farklı açılardan gelmesine karşı zafiyeti bulunmaktadır. Veri setindeki görüntüler genellikle öğle saatlerinde toplandığı için gölgeleme içeren görüntüler üzerinde etkin çalışamayabilir. Bu nedenle gelecek çalışmalarda görüntü ön işleme adımlarında kullanılan parametreler dinamik hale getirilmesi planlanmaktadır. Ayrıca algoritmanın geliştirme amacı gerçek zamanlı işlem yapabilmesini sağlamaktır. Bu amaca yönelik olarak yapılacak gelecek çalışmada İHA'nın sıra takibi yapabilmek için otonom sürüş kazanması ve algoritmanın bu otonom sürüşe yönelik sadece yeni gelen görüntüdeki komponentlerin işlenerek mevcut sıralara bağlanması ile sağlanabilecektir.

Son olarak bu çalışmanın önemi ve sosyal etkileri aşağıdaki şekilde sıralanabilir:

Tarım Verimliliğinin Artması: Yeni ve etkili bir bitki sırası tespiti algoritması ve damlama boru hatlarını tespit eden yöntemler, tarım süreçlerinin daha verimli yönetilmesine olanak tanır. Bu, mahsul verimliliğini artırarak çiftçilerin gelirlerini ve yaşam standartlarını yükseltebilir.

Su Kaynaklarının Verimli Kullanımı: Damlama sulama sistemlerinin daha doğru ve etkili bir şekilde tespit edilip yönetilmesi, su kullanımının optimize edilmesini sağlar. Bu da su kaynaklarının korunmasına ve daha sürdürülebilir tarım uygulamalarına katkı sağlar.

Gıda Güvenliği ve Beslenme: Artan tarım verimliliği ve daha iyi sulama yönetimi, daha bol ve kaliteli gıda üretimini mümkün kılar. Bu da gıda güvenliğinin artmasına ve toplumun genel beslenme düzeyinin iyileşmesine katkıda bulunur.

Teknolojik Eğitim ve Kırsak Kalkınma: Bu tür ileri teknoloji uygulamaları, tarım sektöründe çalışanların yeni beceriler kazanmasını ve bu alanda istihdam olanaklarının artmasını teşvik eder. Ayrıca, tarım teknolojisi üzerine eğitim programlarının ve kursların yaygınlaşmasına yol açabilir.

Bilimsel Araştırmaların Desteklenmesi: Oluşturulan veri seti, diğer araştırmacıların erişimine açılarak bilimsel çalışmalara katkıda bulunur. Bu, tarım teknolojileri ve uygulamaları alanında daha fazla araştırmanın yapılmasını teşvik eder.

Çevresel Sürdürülebilirlik: Daha verimli tarım teknikleri, toprak ve su kaynaklarının daha az tüketilmesini sağlar. Bu da çevresel sürdürülebilirliği artırır ve ekosistemlerin korunmasına yardımcı olur.

EK 1 – Bitki Sırası Başarım Tablosu

Tablo 5.1. *Bitki sırası bulma algoritmasının performansı*

Görüntü Adı	Bitki Sırası Sayısı	Doğru Tespit Edilen Bitki Sırası Sayısı	Tespit Edilemeyen Bitki Sırası Sayısı	Hatalı Tespit Edilen Bitki Sırası Sayısı
DJI_0013	48	47	1	4
DJI_0014	20	20	0	6
DJI_0016	18	18	0	4
frame0	16	15	1	0
frame1	16	16	0	0
frame2	16	16	0	2
frame3	16	16	0	3
frame4	16	16	0	0
frame5	16	16	0	1
frame6	16	16	0	2
frame7	16	16	0	0
frame8	16	16	0	0
frame9	16	16	0	0
frame10	16	16	0	0
frame11	16	15	1	1
frame12	15	15	0	1
frame13	16	16	0	3
frame14	16	16	0	2
frame15	16	16	0	1
frame16	16	16	0	1
frame17	16	15	1	1
frame18	16	16	0	0
frame19	16	16	0	2
frame20	16	16	0	2
frame21	16	16	0	1
frame22	16	16	0	0
frame23	12	12	0	5
frame24	11	10	1	4
frame25	12	12	0	1
frame26	11	11	0	8
frame27	12	11	1	3
frame28	12	12	0	1
frame29	14	14	0	0
frame30	13	13	0	2
frame31	14	14	0	0
frame32	14	14	0	1
frame33	14	14	0	2
frame34	14	13	1	1
frame35	14	14	0	1
frame36	13	13	0	4

frame37	13	13	0	2
frame38	14	14	0	1
frame39	14	14	0	2
frame40	14	14	0	3
frame41	15	15	0	4
frame42	14	14	0	4
frame43	14	12	2	0
frame44	13	13	0	1
frame45	13	12	1	3
frame46	13	12	1	3
frame47	13	12	1	0
frame48	14	14	0	3
frame49	14	14	0	0
frame50	15	14	1	2
frame51	14	13	1	5
frame52	14	14	0	0
frame53	15	11	4	1
frame54	19	19	0	0
frame55	18	18	0	0
frame56	19	19	0	1
frame57	19	19	0	0
frame58	19	19	0	0
frame59	18	18	0	0
frame60	19	19	0	1
frame61	18	18	0	2
frame62	19	19	0	0
frame63	19	19	0	0
frame64	19	19	0	0
frame65	19	17	2	0
frame66	20	20	0	3
frame67	20	20	0	1
frame68	20	20	0	1
frame69	20	20	0	2
frame70	20	20	0	0
frame71	20	20	0	1
frame72	20	20	0	1
frame73	19	19	0	0
frame74	20	20	0	0
frame75	20	19	1	1
frame76	20	20	0	1
frame77	20	19	1	0
frame78	20	20	0	0
frame79	20	20	0	1
frame80	20	19	1	1
frame81	20	20	0	3

frame82	21	21	0	0
frame83	20	20	0	1
frame84	20	20	0	2
frame85	20	20	0	7
frame86	20	20	0	1
frame87	20	20	0	1
frame88	21	21	0	2
frame89	21	21	0	0
frame90	21	21	0	1
frame91	22	21	1	0
frame92	22	22	0	1
frame93	22	22	0	1
frame94	21	20	1	2
frame95	22	22	0	1
frame96	22	22	0	0
frame97	22	21	1	0
frame98	22	22	0	1
frame99	22	22	0	0
frame100	22	22	0	1
frame101	22	22	0	3
frame102	22	22	0	0
frame103	22	22	0	1
frame104	23	23	0	0
frame105	23	23	0	0
frame106	23	23	0	5
frame107	22	22	0	0
frame108	22	22	0	0
frame109	22	21	1	2
frame110	22	22	0	3
frame111	22	21	1	0
frame112	22	21	1	
frame113	22	18	4	1
frame114	22	20	2	2
frame115	22	21	1	0
frame133	23	23	0	1
frame134	23	23	0	1
frame135	24	24	0	1
frame136	23	23	0	0
frame137	25	25	0	2
frame138	24	24	0	1
frame139	24	24	0	0
frame140	24	24	0	1
frame141	24	24	0	0
frame142	24	23	1	0
frame143	25	25	0	0

frame144	25	24	1	0
frame145	24	23	1	0
frame146	23	23	0	0
frame147	23	23	0	0
frame148	23	23	0	0
frame149	23	23	0	0
frame150	23	22	1	0
frame151	23	23	0	0
frame152	24	24	0	0
frame153	23	23	0	0
frame154	24	24	0	0
frame155	24	24	0	0
frame156	24	24	0	0
frame157	24	24	0	1
frame158	24	24	0	0
frame159	24	24	0	0
frame160	24	24	0	2
frame161	24	24	0	0
frame162	28	27	1	0
frame163	27	25	2	0
frame164	19	19	0	0
frame165	19	19	0	1
frame166	19	19	0	0
frame167	19	19	0	0
frame168	19	19	0	1
frame169	20	20	0	0
frame170	20	19	1	2
frame171	20	20	0	0
frame172	19	19	0	3
frame173	20	20	0	0
frame174	19	19	0	0
frame175	19	19	0	2
frame176	19	19	0	2
frame177	18	18	0	0
frame178	18	18	0	1
frame179	17	17	0	0
frame180	18	18	0	1
frame181	18	18	0	0
frame182	17	17	0	1
frame183	17	17	0	0
frame184	17	16	1	0
frame185	17	17	0	0
frame186	21	21	0	1
frame187	24	23	1	0
frame188	23	23	0	2

frame189	24	24	0	0
frame190	24	24	0	0
frame191	24	24	0	0
frame192	24	24	0	0
frame193	24	24	0	0
frame194	24	24	0	0
frame195	27	22	5	0
frame205	33	32	1	0
frame206	31	31	0	0
frame207	31	30	1	1
frame208	31	31	0	0
frame3884	23	23	0	0
frame3885	23	23	0	1
frame3886	23	23	0	3
frame3887	24	24	0	0
frame3888	24	24	0	0
frame3889	23	23	0	0
frame3890	23	23	0	0
frame3891	24	24	0	0
frame3892	23	23	0	1
frame3893	23	23	0	0
frame3894	23	23	0	0
frame3895	23	23	0	1
frame3896	23	23	0	1
frame3897	23	23	0	0
frame3898	23	23	0	1
frame3899	23	22	1	1
frame3900	23	18	5	1
Toplam:	4060	4001	59	41

EK 2 – Damlama Borusu Başarım Tablosu

Tablo 5.2. *Damlama boru hattı bulma algoritmasının performansı*

Görüntü Adı	Damlama Borusu Sayısı	Doğru Tespit Edilen Damlama Borusu Sayısı	Tespit Edilemeyen Damlama Borusu Sayısı	Hatalı Tespit Edilen Damlama Borusu Sayısı
frame1	9	9	0	2
frame4	7	7	0	0
frame7	8	7	1	1
frame8	7	7	0	1
frame9	8	7	1	1
frame10	9	9	0	1
frame18	7	7	0	1
frame22	7	7	0	4
frame29	7	7	0	1
frame31	7	6	1	3
frame49	7	7	0	4
frame52	7	7	0	2
frame54	7	7	0	4
frame55	9	9	0	4
frame57	9	9	0	2
frame58	9	9	0	2
frame59	8	7	1	2
frame62	9	8	1	2
frame63	10	9	1	0
frame64	9	8	1	0
frame70	9	8	1	2
frame73	9	8	1	0
frame74	10	10	0	0
frame78	10	9	0	1
frame82	10	10	0	1
frame89	9	9	0	1
frame96	10	8	2	1
frame99	9	6	3	0
frame102	9	7	2	0
frame104	11	9	2	3
frame105	11	11	0	2
frame107	11	9	2	1
frame108	9	5	4	3
frame136	11	10	1	0
frame139	11	11	0	1
frame141	11	4	7	0
frame143	11	10	1	1
frame146	11	10	1	0
frame147	11	11	0	0

frame148	10	10	1	1
frame149	11	8	3	1
frame151	9	7	2	2
frame152	11	9	2	0
frame153	12	11	1	1
frame154	9	9	0	1
frame155	11	9	2	0
frame156	11	10	1	0
frame158	13	12	1	0
frame159	11	11	0	2
frame161	11	11	0	3
frame164	9	9	0	2
frame166	9	8	1	0
frame167	9	9	0	2
frame169	10	9	1	1
frame171	10	8	2	3
frame173	9	8	1	2
frame174	8	8	0	1
frame177	8	5	3	1
frame179	8	6	2	1
frame181	8	7	1	2
frame183	8	6	2	3
frame185	7	5	2	3
frame189	11	6	5	1
frame190	11	8	3	0
frame191	10	8	2	0
frame192	10	8	2	0
frame193	10	7	3	0
frame194	10	6	4	0
frame206	13	6	7	0
frame208	14	5	9	0
frame3884	11	10	1	0
frame3887	11	10	1	0
frame3888	11	10	1	1
frame3889	11	10	1	2
frame3890	10	10	0	0
frame3891	11	10	1	2
frame3893	10	9	1	1
frame3894	11	11	0	2
frame3897	9	9	0	4
Toplam	759	656	103	99

KAYNAKÇA

- Bah, M. D., Hafiane, A., & Canals, R. (2018). Deep learning with unsupervised data labeling for weed detection in line crops in UAV images. *Remote sensing*, 10(11), 1690.
- Bah, M. D., Hafiane, A., & Canals, R. (2019). CRoWNet: Deep network for crop row detection in UAV images. *IEEE Access*, 8, 5189-5200.
- Bolelli, F., Allegretti, S., & Grana, C. (2022, May). Quest for Speed: The Epic Saga of Record-Breaking on OpenCV Connected Komponents Extraction. In *International Conference on Image Analysis and Processing* (pp. 107-118). Cham: Springer International Publishing.
- De Silva, R., Cielniak, G., Wang, G., & Gao, J. (2023). Deep learning-based crop row detection for infield navigation of agri-robots. *Journal of Field Robotics*.
- Food and Agriculture Organisation (FAO) AQUASTAT. 2013. Erişim: [<http://www.fao.org/nr/water/aquastat/main/index.stm>]
- Furioli, S., Corno, M., Cesana, P., & Savaresi, S. M. (2021, August). Automatic steering control for agricultural tractors in vineyards. In *2021 IEEE Conference on Control Technology and Applications (CCTA)* (pp. 271-276). IEEE.
- Gonroudobou, O. B. H., Silvestre, L. H., Diez, Y., Nguyen, H. T., & Caceres, M. L. L. (2022). Treetop detection in mountainous forests using UAV terrain awareness function. *Computation*, 10(6), 90.
- Guo, Z., Geng, Y., Wang, C., Xue, Y., Sun, D., Lou, Z., ... & Quan, L. (2024). InstaCropNet: An efficient Unet-Based architecture for precise crop row detection in agricultural applications. *Artificial Intelligence in Agriculture*.
- Hassanein, M., Khedr, M., & El-Sheimy, N. (2019). Crop row detection procedure using low-cost UAV imagery system. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 349-356.
- Jiang, G., Wang, Z., & Liu, H. (2015). Automatic detection of crop rows based on multi-ROIs. *Expert systems with applications*, 42(5), 2429-2441.
- Kamienski, C., Soininen, J. P., Taumberger, M., Dantas, R., Toscano, A., Salmon Cinotti, T., ... & Torre Neto, A. (2019). Smart water management platform: IoT-based precision irrigation for agriculture. *Sensors*, 19(2), 276.
- Kim, J. (2024). Autonomous Detection of Humans in Off-Limits Mountain Areas. *Sensors*, 24(3), 782
- Pérez-Ortiz, M., Peña, J. M., Gutiérrez, P. A., Torres-Sánchez, J., Hervás-Martínez, C., & López-Granados, F. (2015). A semi-supervised system for weed mapping in sunflower crops using unmanned aerial vehicles and a crop row detection method. *Applied Soft Computing*, 37, 533-544.
- Pei, H., Sun, Y., Huang, H., Zhang, W., Sheng, J., & Zhang, Z. (2022). Weed detection in maize fields by UAV images based on crop row preprocessing and improved YOLOv4. *Agriculture*, 12(7), 975.

- Romeo, J., Pajares, G., Montalvo, M., Guerrero, J. M., Guijarro, M., & Ribeiro, A. (2012). Crop row detection in maize fields inspired on the human visual perception. *The Scientific World Journal*, 2012(1), 484390.
- Ruan, Z., Chang, P., Cui, S., Luo, J., Gao, R., & Su, Z. (2023). A precise crop row detection algorithm in complex farmland for unmanned agricultural machines. *Biosystems Engineering*, 232, 1-12.
- Simon, N. A., & Min, C. H. (2021, January). Row detection using a machine learning approach for autonomous agriculture vehicles. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 0273-0276). IEEE.
- Smith, R.J., Uddin, M., Gillies, M.H., Clurey, P.M., 2016. Evaluating the performance of automated bay irrigation. *Irrig. Sci.*, 34(1): 175–185.
- Vidović, I., Cupec, R., & Hocenski, Ž. (2016). Crop row detection by global energy minimization. *Pattern Recognition*, 55, 68-86.
- Vong, C. N., Conway, L. S., Zhou, J., Kitchen, N. R., & Sudduth, K. A. (2021). Early corn stand count of different cropping systems using UAV-imagery and deep learning. *Computers and Electronics in Agriculture*, 186, 106214.
- Winterhalter, W., Fleckenstein, F. V., Dornhege, C., & Burgard, W. (2018). Crop row detection on tiny plants with the pattern hough transform. *IEEE Robotics and Automation Letters*, 3(4), 3394-3401.
- Wonprasaid, S., Xie, X., & Machikowa, T. (2023). Long-term effects of drip irrigation on water use efficiency, yield, and net profit of sugarcane production. *Sugar Tech*, 25(5), 1014-1024.
- Zhai, Z., Zhu, Z., Du, Y., Song, Z., & Mao, E. (2016). Multi-crop-row detection algorithm based on binocular vision. *Biosystems engineering*, 150, 89-103.
- Zhang, S., Liu, Y., Xiong, K., Tian, Y., Du, Y., Zhu, Z., ... & Zhai, Z. (2024). A review of vision-based crop row detection method: Focusing on field ground autonomous navigation operations. *Computers and Electronics in Agriculture*, 222, 109086.
- Zhang, X., Li, X., Zhang, B., Zhou, J., Tian, G., Xiong, Y., & Gu, B. (2018). Automated robust crop-row detection in maize fields based on position clustering algorithm and shortest path method. *Computers and electronics in agriculture*, 154, 165-175.
- Zheng, L. Y., & Xu, J. X. (2014, July). Multi-crop-row detection based on strip analysis. In *2014 International Conference on Machine Learning and Cybernetics* (Vol. 2, pp. 611-614). IEEE.

ÖZGEÇMİŞ

ORCID NO: 0009-0008-5962-5685

Ad Soyad : Alper ONRAT

Yabancı Dil : İngilizce

Eğitim ve Mesleki Geçmişi:

- 2012 - 2017, TOBB Ekonomi ve Teknoloji Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği
- 2020 - 2023, Anadolu Üniversitesi, Açıköğretim Fakültesi, Tarım Teknolojileri
- 2018 - ..., TUSAŞ Motor Sanayii A.Ş., Bilgisayar Mühendisi