

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



TRABZON



KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

HARİTA MÜHENDİSLİĞİ ANABİLİM DALI

**YOLO-V4 ALGORİTMASI İLE SOSYAL MESAFELERİN FOTOGRAMETRİK
YÖNTEMLE ÜÇ BOYUTLU GERÇEK ZAMANLI TESPİTİ**

Süheyla PİLTAN

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde
"HARİTA YÜKSEK MÜHENDİSİ"
Unvanı Verilmesi İçin Kabul Edilen Tezdir.

Tezin Enstitüye Verildiği Tarih : 10 / 05 / 2022

Tezin Savunma Tarihi : 30 / 05 / 2022

Tez Danışmanı : Prof. Dr. Fevzi KARSLI

Trabzon 2022

ÖNSÖZ

“YOLO-V4 Algoritması ile Sosyal Mesafelerin Fotogrametrik Yöntemle Üç Boyutlu Gerçek Zamanlı Tespiti” adlı bu çalışma Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Harita Mühendisliği Anabilim Dalında Yüksek Lisans Tezi olarak hazırlanmıştır.

Çalışma sürecinde bilgi ve desteklerini esirgemeyen, akademik kimliğini her daim örnek aldığım danışmanım Prof. Dr. Fevzi KARSLI’ ya sonsuz teşekkürlerimi sunarım.

Değerli fikir ve önerileriyle manevi desteklerini esirgemeyen kıymetli hocalarım Doç. Dr. Eminnur Topsakal AYHAN’ a, Dr. Öğretim Üyesi Mustafa DİHKAN’ a ve Dr. Öğretim Üyesi Hayrettin ACAR’ a teşekkürü bir borç bilirim.

Her daim yanımda olan ve desteklerini hissettiren, bugünlere gelmemin mimarları annem Ümmühan PİLTAN, babam Mehmet PİLTAN, kardeşim Oğuzhan PİLTAN ve hayat arkadaşım Sait Semih ALTAŞ’ a sonsuz teşekkürlerimi sunarım.

Süheyla PİLTAN

Trabzon 2022

TEZ ETİK BEYANNAMESİ

Yüksek Lisans Tezi olarak sunduğum “YOLO-V4 Algoritması ile Sosyal Mesafelerin Fotogrametrik Yöntemle Üç Boyutlu Gerçek Zamanlı Tespiti” başlıklı bu çalışmayı baştan sona kadar danışmanım Prof. Dr. Fevzi KARSLI’nın sorumluluğunda tamamladığımı, verileri/örnekleri kendim topladığımı, deneyleri/analizleri ilgili laboratuvarlarda yaptığımı/yaptırdığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 30/05/2022

Süheyla PİLTAN

Trabzon 2022

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET	VII
SUMMARY	VIII
ŞEKİLLER DİZİNİ	IX
TABLolar DİZİNİ.....	XI
1. GENEL BİLGİLER	1
1.1. Giriş.....	1
1.2. Problemin Tanımı	2
1.3. Çalışmanın Amacı.....	2
1.4. Metodoloji.....	3
1.5. Temel Kavramlar	5
1.5.1. Stereo Kamera Sistemleri ile Üç Boyutlu Görüş	6
1.5.1.1. Epipolar Geometri.....	8
1.5.1.2. Stereo Kamera Kalibrasyonu	10
1.5.1.3. Stereo Görüntü Rektifikasyonu.....	12
1.5.1.4. Stereo Görüntü Eşleştirilmesi	13
1.5.1.5. Üç Boyutlu Koordinat Hesabı.....	14
1.5.2. Yapay Zekâ	15
1.5.3. Makine Öğrenmesi.....	20
1.5.3.1. Denetimli Öğrenme.....	21
1.5.3.2. Denetimsiz Öğrenme	22
1.5.3.3. Yarı Denetimli Öğrenme.....	22
1.5.3.4. Pekiştirmeli Öğrenme	24
1.5.4. Derin Öğrenme.....	24
1.5.4.1. Yapay Sinir Ağları	27
1.5.4.2. Evrişimli Sinir Ağları.....	30

1.5.5.	YOLO ve YOLO – V4.....	38
1.5.5.1.	YOLO Algoritması Çalışma Prensibi	39
1.6.	Literatür Araştırması.....	52
2.	YAPILAN ÇALIŞMALAR	56
2.1.	Veri Toplama ve Kullanılan Donanım.....	56
2.2.	Stereo Kamera Kalibrasyonu	58
2.3.	Üç Boyutlu Koordinat Tespiti.....	61
2.4.	YOLO-V4 ile Kişi/Kişilerin Tespiti	63
2.5.	Gerçek Zamanlı Sosyal Mesafe Analizi	64
3.	BULGULAR VE TARTIŞMA	68
4.	SONUÇ VE ÖNERİLER.....	75
5.	KAYNAKLAR	78
6.	EKLER.....	81
ÖZGEÇMİŞ		

Yüksek Lisans Tezi

ÖZET

YOLO-V4 ALGORİTMASI İLE SOSYAL MESAFELERİN FOTOGRAMETRİK YÖNTEMLE ÜÇ BOYUTLU GERÇEK ZAMANLI TESPİTİ

Süheyla Piltan

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Harita Mühendisliği Anabilim Dalı
Danışman: Prof. Dr. Fevzi KARSLI
2022, 81 Sayfa

Koronavirüs (COVID-19), ilk kez 2019 yılının son aylarında Çin'in Hubei eyaleti başkenti Wuhan'da ortaya çıkan ve günümüzde de etkilerini sürdüren bir hastalıktır. Hastalık, solunum yoluyla bulaşması nedeniyle kısa sürede yayılmış ve 11 Mart 2020'de Dünya Sağlık Örgütü (DSÖ) tarafından hastalığın pandemi olarak ilan edilmesine neden olmuştur.

İnsanlık tarihi boyunca çeşitli salgın hastalıklar nedeni ile gündeme gelen sosyal mesafe kavramı COVID-19 ile tekrar gündeme gelmiş ve salgına karşı tedbirlerde birinci sırada yerini almıştır. Bu süreçte sosyal mesafenin kontrolü amacıyla çeşitli kısıtlamalar getirilmiş, kontrol mekanizmaları geliştirilmiş ve yardımcı sistemlere entegre çözümler ön plana çıkmıştır.

Çalışma kapsamında sosyal mesafelerin fotogrametrik yöntemle üç boyutlu koordinatlarla gerçek zamanlı tespitine yönelik bir algoritma tasarlanmıştır. İki bilgisayar kamerası ile oluşturulan stereo görüş sistemi ile elde edilen görüntülere ait sahneler değerlendirilmiş ve kişiler YOLO-V4 algoritmasıyla tespit edilmiştir. Tespit edilen kişiler arası sosyal mesafeler üç boyutlu gerçek koordinatlar bazında hesaplanmış ve gerçek zamanlı olarak analiz edilmiştir.

Sosyal mesafe tespitine yönelik var olan algoritmalar incelediğinde, mesafelerin iki boyutlu piksel koordinatları ya da sahne bağımlı üç boyutlu referans noktaları baz alınarak hesaplandığı görülmüştür. Geliştirilen algoritma, kişiler arası mesafeleri sahneden bağımsız, üç boyutlu gerçek koordinatları ile yüksek doğrulukta hesaplamakta ve bu yönüyle alanındaki diğer çalışmalardan farklı bir bakış açısı ortaya koymaktadır.

Anahtar Kelimeler: Sosyal mesafe, YOLO, Derin öğrenme, Yapay sinir ağları, Evrişimli sinir ağları, Bilgisayarlı görüş, Stereo görüş

Master Thesis

SUMMARY

THREE-DIMENSIONAL REAL-TIME DETECTION OF SOCIAL DISTANCES BY PHOTOGRAMMETRIC METHOD WITH YOLO-V4 ALGORITHM

Süheyla PİLTAN

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Geomatics Engineering Graduate Program
Supervisor: Prof. Dr. Fevzi KARSLI
2022, 81 Pages

Coronavirus (COVID-19) is a disease that first emerged in the last months of 2019 in Wuhan, the capital of China's Hubei province, and continues to be effective today. The disease spread in a short time due to airborne transmission and caused the disease to be declared as a pandemic by the World Health Organization (WHO) on March 11, 2020.

The concept of social distance, which has come to the fore due to various epidemics throughout human history, has come to the fore again with COVID-19 and has taken its place in the first place in measures against the epidemic. In this process, various restrictions have been imposed for the control of social distance, control mechanisms have been developed and solutions integrated into auxiliary systems have come to the fore.

Within the scope of the study, an algorithm was designed for real-time detection of social distances with three-dimensional coordinates using photogrammetric methods. The scenes of the images obtained with the stereo vision system created by two computer cameras were evaluated and the people were identified with the YOLO-V4 algorithm. Social distances between detected persons were calculated on the basis of real three-dimensional coordinates and analyzed in real time.

When existing algorithms for social distance detection were examined, it was seen that distances were calculated based on two-dimensional pixel coordinates or scene-dependent three-dimensional reference points. The developed algorithm calculates interpersonal distances with high accuracy, independent of the scene, with three-dimensional real coordinates, and in this respect, it presents a different perspective from other studies in the field.

Keywords: Social distance, YOLO, Deep learning, Artificial neural networks, Convolutional neural networks, Computer vision, Stereo vision

ŞEKİLLER DİZİNİ

Sayfa No

Şekil 1. Tasarlanan algoritmanın akış diyagramı	5
Şekil 2. İnsanda monoküler ve binoküler görüş alanları (URL-4,2022)	6
Şekil 3. “StereoLabs” firması tarafından tasarlanan ZED kamerası (URL-5, 2022)	7
Şekil 4. Stereo görüntü çiftlerinde epipolar geometri (Moisan vd., 2016).....	8
Şekil 5. Epipolar Düzlem	9
Şekil 6. Kameralar arası öteleme ve dönüklük bilgileri (Bradski ve Kaehler, 2008).....	11
Şekil 7. Görüntü rektifikasyon modeli (URL-6, 2022)	12
Şekil 8. Stereo düzeltme: rektifiye edilmiş görüntüler (Bradsky ve Kaehler, 2008).....	12
Şekil 9. Rektifikasyon öncesi ve sonrası (Bradsky ve Kaehler, 2008).....	13
Şekil 10. Normalize edilmiş görüntü çiftleri (Bradsky ve Kaehler, 2008).....	14
Şekil 11. Derinlik ve aykırılık ilişkisi (Bradsky ve Kaehler, 2008)	15
Şekil 12. Yapay zekâ, makine öğrenmesi ve derin öğrenme.....	16
Şekil 13. Turing testi ve oyuncuları (URL-7, 2022).....	17
Şekil 14. Bir otokodlayıcının yapısı (Canchumuni vd., 2019)	23
Şekil 15. Makine öğrenmesi ve derin öğrenme algoritmalarının işleyişi (URL-8, 2022) ...	25
Şekil 16. Derin öğrenme ağlarında süreçler (Chollet, 2021)	26
Şekil 17. Sinir ağlarının ilk modeli (McCulloch ve Pitts, 1943)	28
Şekil 18. Nöron modeli (URL-9, 2022).....	29
Şekil 19. (A) İnsan nöronu; (B) yapay nöron; (C) biyolojik sinaps; (D) yapay sinaps (Maltarollo vd., 2013).....	29
Şekil 20. Yapay sinir ağlarının yapısı (URL-10, 2022).....	30
Şekil 21. Bir evrişimli sinir ağında örüntüler arası uzamsal hiyerarşiler (Chollet, 2021) ...	31
Şekil 22. Evrişimli sinir ağında doldurma etkisi (Chollet, 2021).....	32
Şekil 23. Çeşitli aktivasyon fonksiyonları (URL-13, 2022).....	35
Şekil 24. Softmax fonksiyonunun katmanlara entegrasyonu (URL-14, 2022)	36
Şekil 25. Ortaklama yöntemleri (URL-15, 2022).....	36
Şekil 26. Ortaklama katmanından çıkan verinin düzleştirilmesi (Suriya vd., 2019).....	37
Şekil 27. LeNet-5 mimarı yapısı (LeCun vd., 1998).....	38
Şekil 28. YOLO-V4 ve diğer nesne dedektörlerinin kıyaslanması(Bochkovskiy vd,2020)	39

Şekil 29. YOLO ağı vektör veri yapısı	40
Şekil 30. YOLO ağı için “kedi” sınıfı eğitim verilerinin hazırlanması	41
Şekil 31. YOLO ağı “kedi” sınıfı eğitim verisi vektör yapısı	41
Şekil 32. YOLO ağı için “insan” sınıfı eğitim verilerinin hazırlanması	43
Şekil 33. YOLO ağı “insan” sınıfı eğitim verisi vektör yapısı	43
Şekil 34. YOLO ağı eğitim verisi vektör yapısı – nesne tespit edilememe durumu	44
Şekil 35. YOLO ağında eğitim süreci	45
Şekil 36. Eğitilmiş YOLO ağında nesne tespiti ve çıktı vektör yapısı	46
Şekil 37. Eğitilmiş bir YOLO ağında nesne tespiti	47
Şekil 38. Eğitilmiş bir YOLO ağında nesne tespiti-numaralandırılmış ızgaralar.....	47
Şekil 39. YOLO ağında nesne tespit edilmeme durumunun vektör ifadesi	48
Şekil 40. İki farklı nesnenin YOLO çıktı katmandaki vektör ifadeleri	48
Şekil 41. YOLO - farklı sınıflar için çoklu sınırlayıcı kutu saptanma durumu	49
Şekil 42. YOLO - aynı sınıflar için çoklu sınırlayıcı kutu saptanma durumu.....	50
Şekil 43. İki sınırlayıcı kutu için kesişen alan ve toplam alan	51
Şekil 44. Bir vektör veride birden fazla objenin temsili.....	52
Şekil 45. Uygulama kapsamında oluşturulan stereo kamera sistemi.....	56
Şekil 46. MATLAB “Stereo Camera Calibrator” uygulaması arayüzü	58
Şekil 47. Çalışmada kullanılan kalibrasyon deseni	59
Şekil 48. MATLAB “Stereo Camera Calibrator” uygulaması ile kalibrasyon	59
Şekil 49. Kalibrasyon sonucu elde edilen yeniden yansıtma hataları.....	60
Şekil 50. Kamera kalibrasyonu çıktıları	60
Şekil 51. Video görüntüye ait bir sahnenin aykırılık haritası	62
Şekil 52. Video görüntüye ait rektifiye edilmiş bir sahnenin anaglif görüntüsü.....	62
Şekil 53. Algoritma tarafından analiz süresince ilk sahne için oluşturulmuş nokta bulutu	63
Şekil 54. Algoritma tarafından oluşturulan “riskanalizi” matrisi.....	64
Şekil 55. 1 numaralı kişinin diğer kişilere olan mesafeleri analizi.....	65
Şekil 56. 2 ve 3 numaralı kişilerin diğer kişilere olan mesafeleri analizi.....	66
Şekil 57. 4 ve 5 numaralı kişilerin diğer kişilere olan mesafeleri analizi.....	66
Şekil 58. Oluşturulan “riskanalizi” matrisinin sosyal mesafe ihlallerine göre temsili	67
Şekil 59. 1 metre sosyal mesafe uzunluğu baz alınarak gerçekleştirilen analiz sonucu çıktı video görüntünün 25. saniye anındaki sahnesi	68
Şekil 60. Değişen sosyal mesafe eşik değerleri için gerçekleştirilen analiz sonuçları	70

TABLÖLER DİZİNİ

	<u>Sayfa No</u>
Tablo 1. Yeni bir paradigma: makine öğrenmesi (Chollet, 2021).....	20
Tablo 2. Uygulama kapsamında kullanılan donanım bilgileri ve kamera özellikleri.....	57
Tablo 3. Geliştirilen algoritmanın belirli sahneler için tespit ettiği kişiler arası mesafeler.	69
Tablo 4. Değişen sosyal mesafe eşikleri için kişilerin temsili.....	71
Tablo 5. Geliştirilen algoritmanın benzer çalışmalar ile kıyaslanması	73



1. GENEL BİLGİLER

1.1. Giriş

Salgın, bulaşıcı bir hastalığın belirli popülasyondaki çok sayıda insana kısa sürede yayılmasıdır (URL-1, 2022). Artan dünya nüfusu ve ortak yaşam alanları, salgınların önceki yüzyıllardan daha hızlı yayılmasına neden olmaktadır. Dünya tarihi boyunca insanoğlu pek çok salgınla mücadele etmiş ve ciddi kayıplar vermiştir. Öyle ki bu salgınlar yalnızca belirli kitlelerle sınırlı kalmamış, küresel salgın olarak da bilinen pandemiye neden olmuş ve tüm dünyayı etkilemişlerdir.

Koronavirüs hastalığı olarak da bilinen COVID-19 (Koronavirüs Disease 2019 – Koronavirüs Hastalığı 2019), Çin'in Hubei eyaleti Wuhan şehrinde Kasım 2019 tarihinde ortaya çıkmış bir hastalıktır (URL-2, 2022). Hastalık, solunum yoluyla bulaşması nedeni ile kısa sürede tüm dünyaya yayılmış ve büyük yankı uyandırmıştır. DSÖ (Dünya Sağlık Örgütü)'nün hastalığı pandemi ilan etmesi ile ülkeler sosyal yaşamda ciddi kısıtlamalara gitmiş, kontrol mekanizmaları geliştirmiş ve teknoloji odaklı çözümler arayışına girmişlerdir.

COVID-19 ile gündeme gelen “sosyal mesafe”, hastalık etmenini taşıma riski bulunan canlılar arasındaki fiziksel mesafedir. Sosyal mesafe, DSÖ tarafından en az 1 metre olarak belirtilmiştir (URL-3, 2022). Günümüzde sosyal mesafenin 1,5 ve 2 metre olarak kabul edildiği uygulamalar da mevcuttur.

Geçtiğimiz yıllarda dünyayı etkileyen İspanyol gribi ve Ebola gibi salgınlarda sosyal mesafeyi korumanın ve teması azaltmanın salgından ciddi oranda koruma sağladığı tespit edilmiştir. COVID-19 pandemisinin ortaya çıkması ile tekrar gündeme gelen sosyal mesafe, hastalığa karşı alınan en önemli tedbirlerden biridir. Bu amaçla ülkeler toplu yaşam alanlarında ciddi kısıtlamalara gitmiş, sosyal mesafe kurallarını katı biçimde uygulamışlardır.

1.2. Problemin Tanımı

İlk ortaya çıktığı 2019 yılından bu yana etkilerini devam ettiren COVID-19 pandemisi gün geçtikçe etkilerini azaltsa da dünya tarihinde gerçekleşen salgın hastalıklar, mesafe kavramının önemini gözler önüne sermektedir. İnsanoğlu varoluşundan bu yana çeşitli salgın hastalıklar ile karşılaşmıştır ve önümüzdeki yüzyıllarda artan nüfus ile bu hastalıkların çeşitleneceği öngörülmektedir.

Yüksek donanım ve yazılımlarla geliştirilmiş bilgisayarlar artık hayatın her alanında çeşitli problemlerin çözümünde rol almaktadırlar. Günümüzde bilgisayarlardan yalnızca insanlara yardımcı sistemler değil, insan yeteneklerine sahip karar verici sistemler de olması beklenmektedir. İnsan gibi düşünen ve hareket eden sistemler tasarlamak, insanoğlunun yüzyıllardır ilgisini çeken konulardandır. Bilgisayarların insanlar tarafından yapılabilen işlemleri otomatikleştirmesi ve insan yeteneklerine kavuşması isteği yapay zekâ, makine öğrenmesi ve derin öğrenme algoritmalarının gündeme gelmesine neden olmuştur. Bu algoritmalar ile desteklenmiş sistemlerin salgın hastalıklarda çeşitli karar ve tespit mekanizmalarında kullanımı günümüzde oldukça yaygındır. Maske kullanım analizi, sosyal mesafe tespiti gibi uygulamalarda derin öğrenme algoritmalarından sıkça yararlanılmaktadır.

Bu çalışmada hastalık bulaş riskini minimize etmek amacıyla kişiler arası sosyal mesafelerin bir stereo kamera sistemi ile üç boyutlu gerçek koordinatlar cinsinden gerçek zamanlı tespiti problemi ele alınmıştır. Çalışma kapsamında iki bilgisayar kamerası ile bir stereo görüş sistemi oluşturulmuş, bu sistemin sağladığı derinlik algısı sayesinde kişilerin üç boyutlu gerçek koordinatları hesaplanmış ve sahneden bağımsız bir geometri ile sosyal mesafeler gerçek zamanlı olarak hesaplanmıştır. Geliştirilen algoritmanın test edilmesi amacıyla 1, 1.5 ve 2 metrelik sosyal mesafe eşik değerleri ile 3 ayrı analiz gerçekleştirilmiştir. Algoritmanın değişen sosyal mesafe eşik değerleri karşısında da doğru sonuçları ürettiği tespit edilmiş, mesafeleri üç boyutlu gerçek koordinatlar ile yüksek doğrulukta analiz ettiği gözlemlenmiştir.

1.3. Çalışmanın Amacı

Sosyal mesafenin tespiti uygulamalarında temel olarak 3 yöntem gözlemlenmektedir.

Birinci yöntem, bir kamera ile sahnenin görüntülenmesi sonucu kişiler arası mesafelerin piksel cinsinden hesaplanmasına dayalı bir yöntemdir. Bu yöntem derinlik algısı içermediğinden obje koordinatlarına, dolayısı ile gerçek sosyal mesafe değerlerine erişim mümkün olmayacaktır. İlave olarak, görüntü düzlemindeki iki piksel arası mesafenin üç boyutlu gerçek düzlemdeki karşılığı görüntünün her yerinde aynı olmayacağından; piksel bazlı mesafe analizleri iyi sonuç vermeyecektir.

İkinci yöntem, bir kamera ile referans noktalarını barındıran bir sahnenin görüntülenmesi sonucu kişiler arası sosyal mesafe tespitinin gerçekleştirilmesine dayanmaktadır. Bu yöntem doğruluk açısından birinci yönteme göre nispeten yüksek doğruluk sunsa da algoritmanın sahneden bağımsız olmayışı ve referans noktalarına olan gereksinim bir sorun olarak ortaya çıkmaktadır. Ayrıca referans noktalarla sağlanan derinlik algısı nispi bir algıdır ve istenen hassasiyette çözümler üretmeyecektir.

Üçüncü yöntem ise kameranın sabit bir kamera sistemi olduğu varsayılarak sahne geometrisinin saptanması ve mesafelerin bu geometriye göre hesaplanmasını içermektedir. Bu yöntem de diğer yöntemler gibi derinlik algısında yetersiz kalacağından istenen hassasiyette sonuçlar üretmeyecektir. Ayrıca sahne değiştikçe geometri de değişeceğinden, her seferinde geometrinin güncellenmesi gerekecektir. Bu da fazladan iş yükü anlamına gelmektedir.

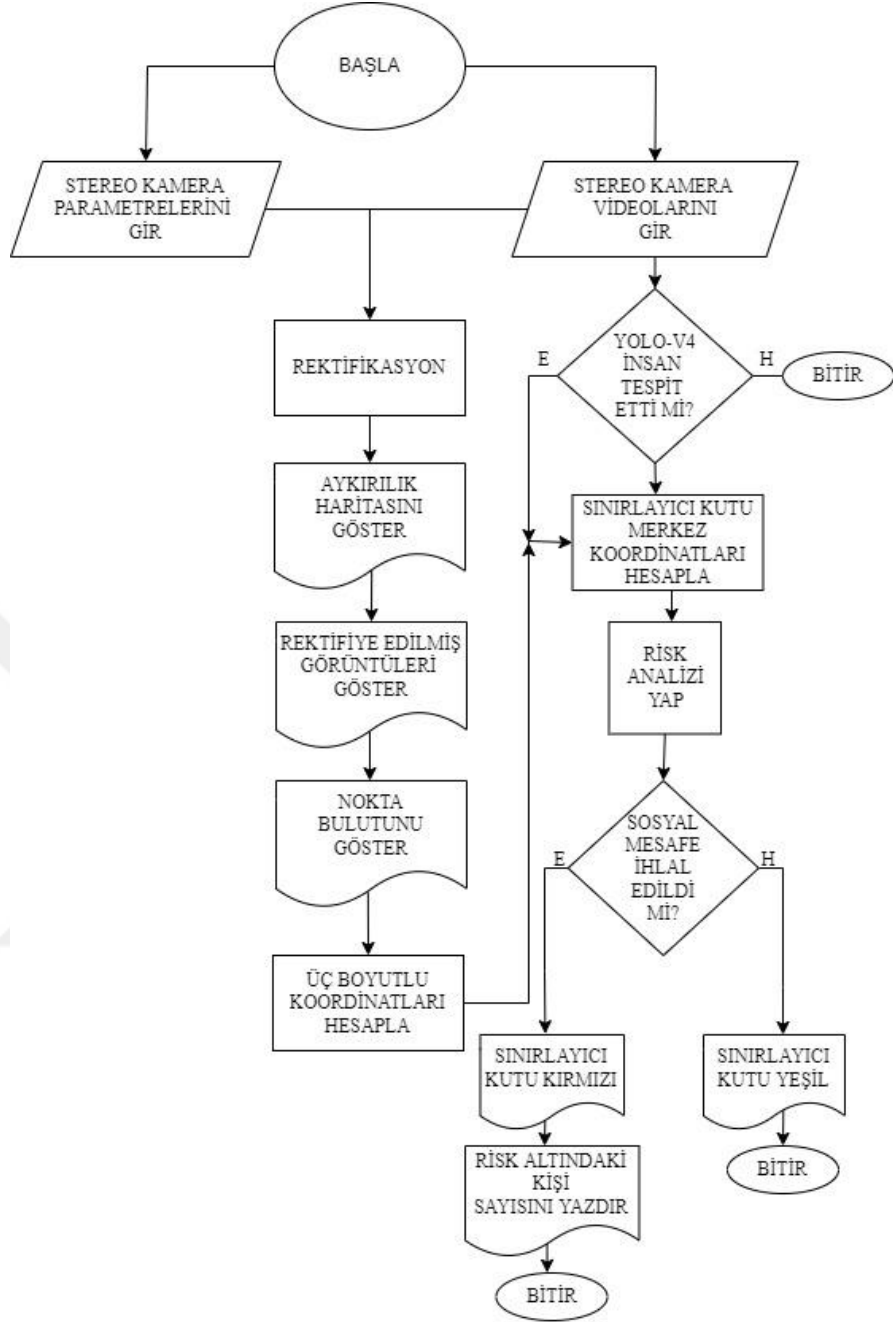
Bu çalışmanın amacı, kişilerin tespitini sağlayarak belirlenecek sosyal mesafeleri üç boyutlu obje koordinatları cinsinden hesaplayacak, sahneden bağımsız bir algoritma oluşturmaktır. Bu sayede sosyal mesafeler hassas şekilde ölçülerek hesaplanabilecek ve kural ihlalleri yüksek doğrulukla saptanabilecektir. Bu amaçla çalışma kapsamında MATLAB ortamında bir algoritma tasarlanmış, oluşturulan stereo görüş sistemi ile sunulan üç boyutlu görüş algısı sayesinde sosyal mesafe kural ihlallerinin gerçek obje koordinatları cinsinden yüksek doğrulukta saptanması gerçekleştirilmiştir.

1.4. Metodoloji

Geliştirilen algoritmanın çalışma mantığı parametreleri ile aşağıda sıralanmıştır. Ayrıca, EK. 1' de algoritmaya ait sözde kod ifade edilmiştir.

- Stereo kamera parametreleri ve stereo kamera sistemi ile elde edilen görüntü verileri, algoritmaya girdi olarak sunulur.
- Girdi veriler kullanılarak görüntüler rektifiye edilir.

- Rektifiye edilmiş görüntüler ve hesaplanan aykırılık haritaları görselleştirilerek kullanıcıya sunulur.
 - Nokta bulutu oluşturulur ve üç boyutlu koordinatlar hesaplanır.
 - YOLO-V4 algoritması girdi veri olarak sağlanan video görüntüde herhangi bir kişi tespit edemezse algoritma sonlanır. Kişi tespit edildiğinde ise YOLO-V4, bu kişilerin her biri için birer sınırlayıcı kutu oluşturur ve kutunun merkez koordinatları, geliştirilen algoritma tarafından üç boyutlu koordinatlar cinsinden hesaplanır.
 - Algoritma, merkezler arası mesafeleri öklid uzunluğu cinsinden hesaplayarak sosyal mesafe kural ihlallerini saptamak üzere risk analizini gerçekleştirir.
 - Sosyal mesafeyi ihlal ettiği tespit edilen kişiler algoritma tarafından kırmızı renkli sınırlayıcı kutu ile görselleştirilerek kullanıcıya sunulur. Sunulan bilgi ayrıca risk altındaki toplam kişi sayısını da içermektedir.
 - Algoritma tarafından herhangi bir kural ihlali yapmadığı saptanan kişiler ise yeşil renkli sınırlayıcı kutu ile görselleştirilerek ekrana aktarılır ve analiz sonlandırılır.
- Çalışma kapsamında tasarlanan algoritmanın akış diyagramı Şekil 1’de verilmiştir:



Şekil 1. Tasarlanan algoritmanın akış diyagramı

1.5. Temel Kavramlar

Çalışma kapsamında geliştirilen algoritmanın parametreleri, parametrelere ilişkin temeller ve temellerin dayandığı kavramlar bu bölümde açıklanmıştır.

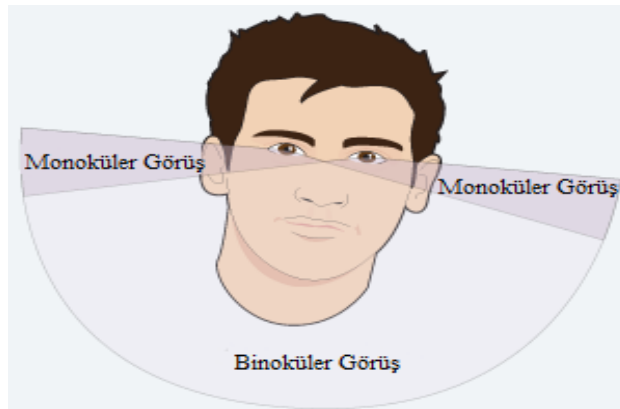
1.5.1. Stereo Kamera Sistemleri ile Üç Boyutlu Görüş

Doğal zekâyı taklit eden bir yapay zekânın mümkün olup olamayacağı tartışmaları makinelere insan yeteneklerini entegre eden çalışmaların hız kazanmasına neden olmuştur. Bu çalışmaların amacı makinelerin tıpkı birer insan gibi dünyayı algılama, yorumlama ve karar verme becerisine sahip olmalarını sağlamaktır.

Bilgisayarlı görü bilimi, üç boyutlu görüşe sahip canlıların üç boyutlu görme yeteneklerinin makinelere aktarılması düşüncesi üzerine yapılan çalışmaları kapsamaktadır. Günümüzde medikal görüntüleme, yüz tanıma, modelleme ve tasarım, güvenlik, otonom sistemler gibi pek çok alanda bilgisayarlı görü bilimi yer almaktadır.

Bilgisayarlı görü alanındaki çalışmalar 1900'li yıllara gelindiğinde hız kazanmıştır. 1963 yılında bilgisayarlı görü biliminin babası Lawrence Roberts iki boyutlu perspektif görüntülerin üç boyutlu algılanması üzerine yaptığı çalışma ile büyük yankı uyandırmıştır (Roberts, 1963).

İnsan gözü üç boyutlu bir nesneyi algıarken sol ve sağ gözden gelen görüntüyü ters algılar ve bu görüntülerin her biri aslında iki boyutlu görüntülerdir. Gözler aynı sahneye iki farklı perspektifle bakarak ortak alanda üç boyutlu görüşü gerçekleştirirler. Bir nesneye çok yakından sol ve sağ gözlerle ayrı ayrı bakıldığında, nesnenin yerinin değiştiğini fark etmek mümkündür. Ancak iki gözle aynı anda görme işlemi gerçekleştirilirken bu ayrım yapılamaz. Bunun nedeni beynimizin iki ayrı görüntüyü birlikte değerlendirerek bize üç boyutlu bir görüş ve derinlik algısı sunmasıdır. İki göz ile birden gerçekleştirilen görme sürecine binoküler görüş denmektedir. Bu eylem tek gözle gerçekleştiriliyor ise monoküler görüş adını alır (Şekil 2).



Şekil 2. İnsanda monoküler ve binoküler görüş alanları (URL-4,2022)

Bir gözü görme yetisini sonradan kaybetmiş kişiler bir süre sonra derinlik algısını da kaybetmektedir. Algı kaybı, beynin görü ile ilgili önceden öğrendiklerini yavaş yavaş unutmasından dolayı hemen gerçekleşmez. Ancak tek gözle görmenin devam ettiği süreç boyunca birey derinlik algısını kaybetmeye başlayacaktır. Doğuştan bir gözü görmeyen bireylerde ise derinlik algısı yaşamsal döngüleri boyunca hiç oluşmamıştır.

Bir makinenin üç boyutlu algısı kameralarla oluşturulabileceği gibi, farklı aygıtlarla da bunu sağlamak mümkündür. Ancak kamera sistemleri basit yapıları ve kolay ulaşılabilir olmaları nedeniyle birçok çalışmada tercih edilmektedirler. Kamera sistemlerinin üç boyutlu görüş ve derinlik algısını sunması isteniyor ise, iki gözü temsilen iki kameraya ihtiyaç duyulacaktır. Bu amaçla çeşitli firmalarca tasarlanmış iki lensli sistemler mevcuttur (Şekil 3). Bu sistemler temel olarak insan gözünün yapay bir formunu oluşturmayı hedeflemektedirler.



Şekil 3. “StereoLabs” firması tarafından tasarlanan ZED kamerası (URL-5, 2022)

Stereo kamera sistemleri ile üç boyutlu görüşün sağlanması için şu dört temel adımın uygulanması gerekmektedir (Bradski ve Kaehler, 2008):

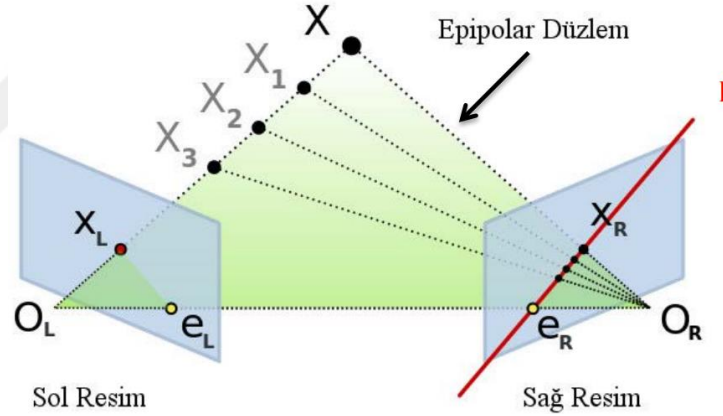
- Mercek bozulmalarının giderilmesi: Radyal ve teğetsel mercek bozulmalarının matematiksel olarak ortadan kaldırılması gerekmektedir. Bu adımın çıktıları bozulmamış görüntülerdir.
- Stereo düzeltme işleminin gerçekleştirilmesi: Kameralar arasındaki açı ve mesafelerin ayarlanması işlemidir. Bu adımın çıktıları, satır hizalı ve düzeltilmiş görüntülerdir.
- Stereo eşleştirme: Sol ve sağ kameradaki aynı özelliklerin saptanmasını içeren adımdır. Bu adımın çıktısı aykırılık haritalarıdır. Aykırılıklar, saptanmış olan özelliklerin sol ve sağ görüntü düzlemlerindeki x koordinatları arasındaki farklardır.

- Yeniden yansıtma işlemi: Kameraların geometrik ilişkileri saptandığında aykırılık haritaları yardımıyla üçgenleme işlemi yapılır ve uzaklıklar hesaplanır. Bu adımın sonucunda derinlik haritaları elde edilmektedir.

1.5.1.1. Epipolar Geometri

Epipolar geometri, iki kameranın ayrı noktalardan aynı sahneyi görüntülemeleri ile oluşan geometrik ilişkileri temsil etmektedir. Epipolar geometri kamera parametrelerine ve kameraların bağıl konumlarına bağlı iken sahneden bağımsızdır. Görüntülerin birbirleri ile ilişkileri temel matrisler ile ortaya çıkarılarak görüntüler normalize edilir ve böylece görüntüler görüntü eşleştirme işlemine hazır hale getirilmektedirler.

Stereo görüntü analizleri epipolar geometri ile açıklanmaktadır (Şekil 4).

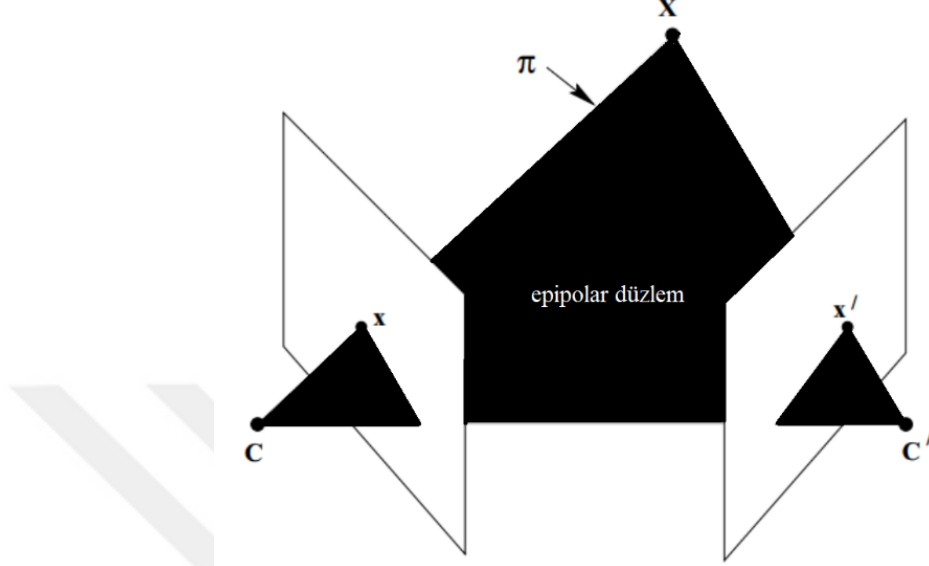


Şekil 4. Stereo görüntü çiftlerinde epipolar geometri (Moisan vd., 2016)

O_L ve O_R , sol ve sağ kamera merkezlerini temsil etmektedir. Bu merkezleri birleştiren doğru parçası taban çizgisi olarak adlandırılır ve çizginin görüntü düzlemlerini kestiği noktalara (e_L , e_R) epipol nokta adı verilir. X obje noktasıdır. Obje noktasının sol ve sağ görüntü düzlemindeki karşılıkları sırası ile X_L ve X_R görüntü noktalarıdır. Obje noktası, sol veya sağ görüntü düzlemindeki karşılığı ve kamera merkezi aynı doğru üzerinde yer almaktadır.

X , X_L , X_R , O_L ve O_R noktaları aynı düzlemdeyler ve bu düzlem epipolar düzlem (π) adı verilir. Bu düzlemin sol ve sağ görüntü düzlemleri ile kesişimi olan epipolar çizgi (l'),

obje noktasının görüntü düzlemlerinde yüksek korelasyonlu karşılıklarının aranacağı çizgidir.



Şekil 5. Epipolar Düzlem

Şekil 5'te görüldüğü üzere bir X obje noktasının sol görüntü düzlemindeki karşılığı x olsun. Bu noktanın sağ görüntü düzlemindeki karşılığının bulunmasında epipolar çizgi (l') referans alınarak arama başlatılır. x noktası ile çizgi üzerinde en büyük korelasyonu sağlayan nokta, X obje noktasının sağ resimdeki karşılığı olan x' noktası olacaktır.

x ve x' noktaları homojen vektörlerle temsil edilmektedirler:

$$x = (x, y, 1)^T \quad (1)$$

$$x' = (x', y', 1)^T \quad (2)$$

Bu noktalar arası ilişkiler ise temel matris ile hesaplanmaktadır.

Temel matris, görüntü çiftleri arası geometrik ilişkilerin matematiksel ifadesidir. Bu matrisler sol görüntü noktası (x), epipolar çizgi (l') ve sağ görüntü noktası (x') arası ilişkileri kurmaktadır. Her iki kameranın yönelme bilinmeyenlerinin yanı sıra görüntüler arası dönüklük ve öteleme bilgilerini de içermektedir.

$$x'^T F x = 0 \quad (3)$$

$$F = K'^T R K^T [K R^T t]_x \quad (4)$$

Eşitliklerde x ve x' noktaları sırası ile X obje noktasının sol ve sağ görüntü düzleminde karşılık gelen noktalarıdır. K ve K' değerleri kameralara ait iç yöneltme parametrelerini, R değeri sağ kameranın bağıl dönüklük bilgisini ve T vektörü öteleme miktarını ifade etmektedir. Stereo kalibrasyon ile öteleme (T) vektörü ve dönüklük (R) matrisi hesaplanacak, bu durum bir sonraki bölümde açıklanacaktır.

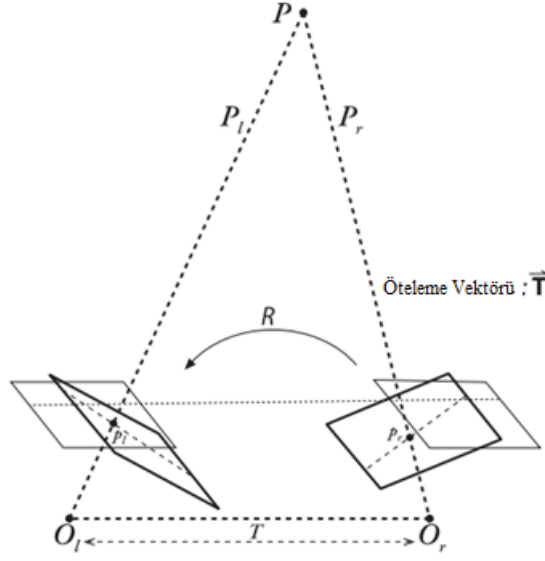
Temel matrisin hesabı ile sol görüntüde belirlenen bir x noktasının sağ görüntüde karşılığının aranacağı epipolar çizgi (I') oluşturulmuş olacak, bu çizgi üzerinde en yüksek korelasyonlu nokta x' olarak tespit edilecek ve karşılıklı noktalar saptanmış olacaktır.

1.5.1.2. Stereo Kamera Kalibrasyonu

Stereo görüş için kullanılacak ikili kamera sistemlerinde kalibrasyon işlemi, derinliğin doğru algılanması ve noktaların uzaydaki geometrik ilişkilerinin tespiti açısından önemlidir. Bu işlem için 1, 2 veya 3 boyutlu kalibrasyon nesnelerinden yararlanılabilmektedir. Kalibrasyon nesnesini içeren sahnenin aynı anda her iki kamera tarafından farklı perspektiflerde görüntülerinin alınması ve değerlendirmelerinin gerçekleştirilmesi ile kamera kalibrasyon işlemi tamamlanmış olacaktır. Çeşitli yazılımlarda bu süreci otomatikleştiren çözümler sunulmaktadır. Kalibrasyon sonucu iki kameranın birbirlerine göre konumları ve nesne uzayındaki yerleri geometrik olarak ifade edilecektir.

Bunun için öncelikle sol ve sağ kameralardan aynı sahnenin farklı perspektiflerden görüntüleri alınır.

Stereo kamera kalibrasyonu için sol ve sağ kameranın birbirlerine göre dönüklük matrisi (R) ve öteleme vektörü (T) bulunmalıdır (Şekil 6).



Şekil 6. Kameralar arası öteleme ve dönüklük bilgileri (Bradski ve Kaehler, 2008)

İki kamera için iki öteleme ve iki dönüklük olmak üzere toplam dört matris vardır ve bunlar sol kamera için R_l , T_l ; sağ kamera için ise R_r , T_r şeklinde tanımlanmaktadır.

$$P_l = R_l P + T_l \quad (5)$$

$$P_r = R_r P + T_r \quad (6)$$

eşitlikleri ile P obje noktasının sol ve sağ görüntü düzlemindeki karşılıkları hesaplanır. Bu işlemin ardından kameralar için dönüklük ve öteleme bilgileri elde edilir.

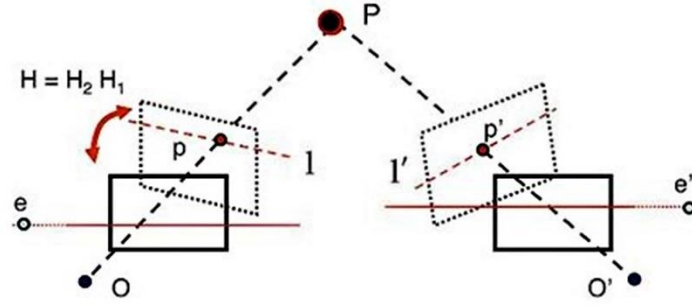
$$P_l = R^T (P_r - T) \quad (7)$$

$$R = R_r (R_l)^T \quad (8)$$

$$T = T_r - R T_l \quad (9)$$

1.5.1.3. Stereo Görüntü Rektifikasyonu

Stereo rektifikasyon işleminde amaç, farklı perspektiflerden aynı sahne için oluşturulmuş iki stereo görüntü çiftinin ortak bir düzlemde ifade edilmesinin sağlanmasıdır. Böylece görüntülerin çekildiği kameraların epipolar çizgileri birbirleri ile çakışmış olacaktır (Şekil 7). Bu işlem görüntü normalizasyonu olarak da adlandırılmaktadır.



Şekil 7. Görüntü rektifikasyon modeli (URL-6, 2022)

Rektifikasyon işlemi sonucu iki görüntüdeki ortak noktalar aynı doğru üzerinde hizalanmış olacak ve bu da bir sonraki adım olan görüntü eşlemeye görüntüleri hazırlamış olacaktır (Şekil 8).

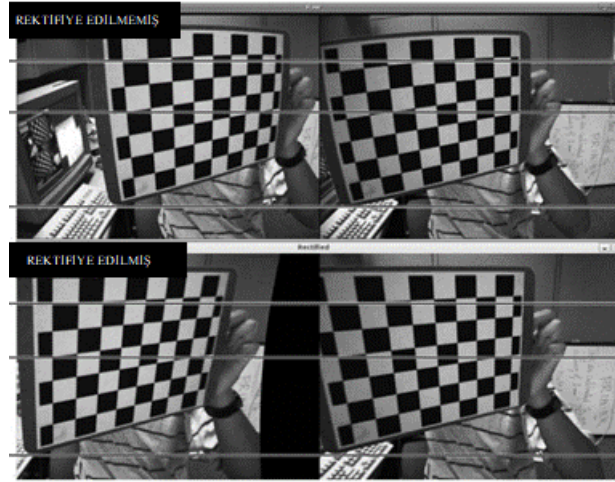


Şekil 8. Stereo düzeltme: rektifiye edilmiş görüntüler (Bradsky ve Kaehler, 2008)

1.5.1.4. Stereo Görüntü Eşleştirilmesi

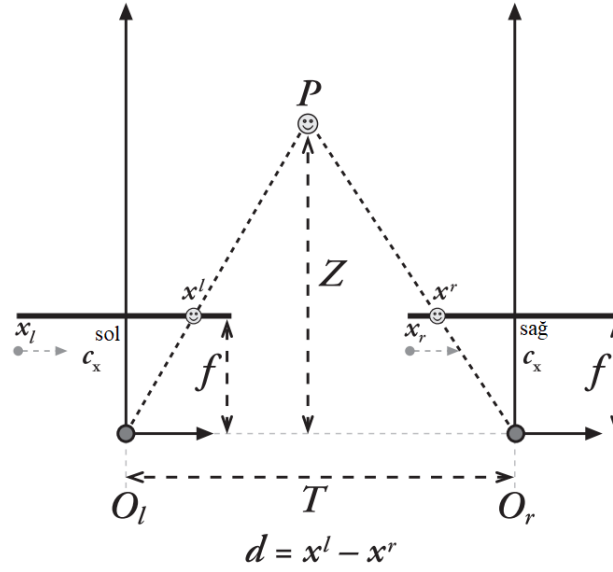
Bu adımda rektifikasyonu yapılmış görüntülerin eşlenik noktalarının bulunması işlemi gerçekleştirilmektedir. Bir görüntü eşleme işleminde iki temel ölçüt vardır. Birinci ölçüt, eşleşme birimi seçimidir. Eşleşme birimi, birbirleri ile ilişkilendirilecek detaylar grubudur. Bir diğer ölçüt ise benzerlik ölçütüdür ve detayların birbirleri ile benzerlik oranlarının bir ifadesidir.

Bir görüntü eşleştirme probleminde görüntüleri piksel bazlı değerlendirerek eşleştirme işlemini uygulamak, birtakım kısıtları da beraberinde getirecektir. Görüntü içerisinde analizi yapılan pikseller aynı gri değere sahip olsalar dahi aynı detayı temsil etmeyebilirler. Bu durum, anlam karmaşasına neden olacaktır. Ayrıca görüntü gürültüleri içerebilir ve gürültülerin sahip olduğu gri değerlerin görüntünün pikselleri ile benzerlik gösterdiği durumlarda hangisinin detay hangisinin gürültü noktası olduğu algoritma tarafından saptanamayarak karmaşaya yol açacaktır. Tüm bu dezavantajlarının yanı sıra piksel tabanlı eşleştirmeler fazla zaman alması nedeni ile de tercih edilmemektedir. Bu sebeplerden dolayı özellik tabanlı, alan tabanlı ve ilişkisel algoritmalar ile görüntüler eşleştirilir. Eşleştirme yalnızca her iki kamera tarafından da gözlemlenebilen ortak alanlardaki detaylar için yapılabilmektedir (Şekil 9).



Şekil 9. Rektifikasyon öncesi ve sonrası (Bradsky ve Kaehler, 2008)

1.5.1.5. Üç Boyutlu Koordinat Hesabı



Şekil 10. Normalize edilmiş görüntü çiftleri (Bradsy ve Kaehler, 2008)

Stereo görüntülerin eşleştirilmesi ile karşılıklı eşlenik noktalar tespit edilmiş olacaktır (Şekil 10). Üç boyutlu koordinatların hesaplanabilmesi için eş piksel çiftleri arası derinlik bilgisi çıkarılmalıdır. Bunun için öncelikle aykırılık bilgisi hesaplanır.

Aykırılık bilgisinin hesabında eşlenik noktaların konumları arası yatay mesafeler kullanılmaktadır. x^l ve x^r nesne uzayındaki P noktasının sol ve sağ görüntü düzlemindeki karşılıkları olmak üzere d aykırılık bilgisi:

$$d = x^l - x^r \quad (10)$$

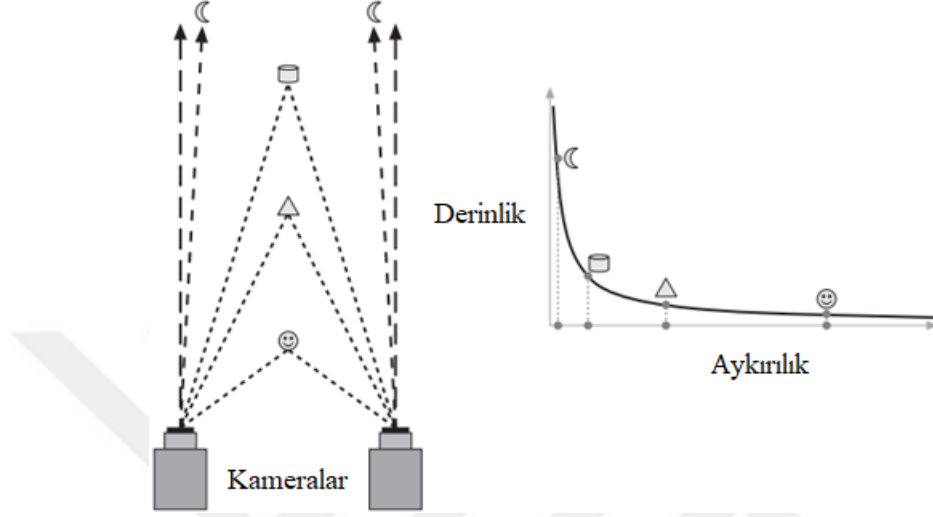
eşitliği ile hesaplanmaktadır.

Eşlenik noktalar arası aykırılık değerleri görüntüdeki tüm çiftler arası hesaplanır ise, görüntünün tamamının aykırılık değerini temsil eden bir aykırılık haritası oluşturulmuş olacaktır.

Derinlik değeri (Z) ise şu eşitlikle hesaplanmaktadır:

$$Z = f (B/d) \quad (11)$$

İlgili eşitlikte f kamera odak uzaklığını, B iki kamera merkezi arası mesafe ya da baz değerini ifade etmektedir. Formülden de anlaşılacağı üzere derinlik (Z) ve aykırılık (d) birbirleri ile ters orantılıdır (Şekil 11).



Şekil 11. Derinlik ve aykırılık ilişkisi (Bradsky ve Kaehler, 2008)

Bir obje kameradan uzaklaştıkça, kameranın nesne ile ilgili derinlik algısı azalacaktır. Benzer durum göz için de geçerlidir. Uzak mesafedeki nesnelerin derinlik ayırdı, yakın nesnelere göre daha zor yapılmaktadır. Buna derinlik çözünürlüğü denir ve şu eşitlik ile hesaplanır:

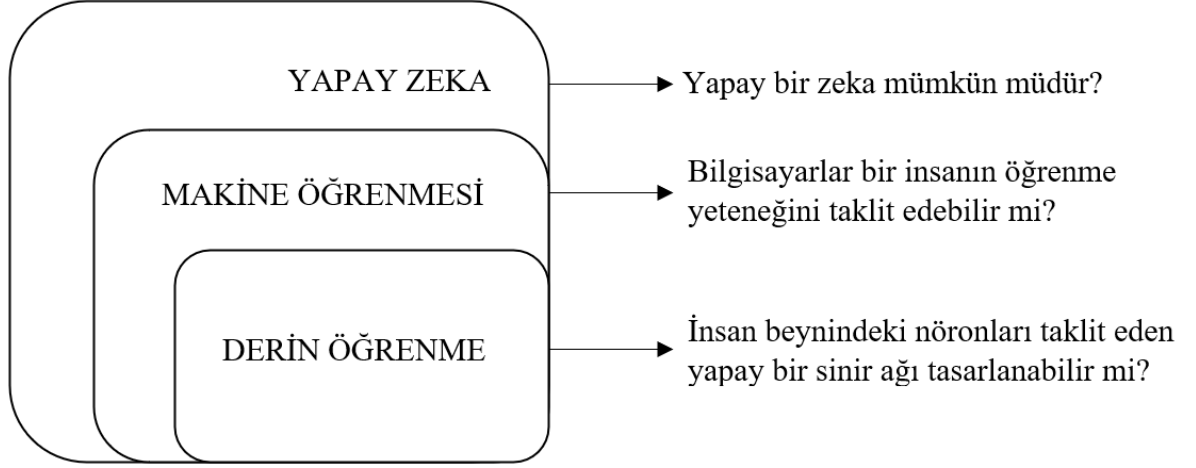
$$\Delta Z = (Z^2 \Delta d) / (fT) \quad (12)$$

Üç boyutlu koordinatların hesaplanması için ise aykırılık haritalarından ve projeksiyon matrisinden yararlanılmakta ve sonuç koordinatlar tespit edilmektedir.

1.5.2. Yapay Zekâ

Geçtiğimiz birkaç yıl içerisinde teknolojik gelişmelere paralel olarak yapay zekâ, her kesimden araştırmacının dikkatini çeken ve birçok disiplindeki problemlerin çözümüne eşlik eden bir konu haline gelmiştir. Tarihsel süreçte farklı alanlarda aynı arayışlar devam etmiştir

(Şekil 12). Yapay zekânın tarihi daha eskilere dayansa da özellikle 1990’lı yıllarda bilişim alanındaki gelişmeler, günümüzde yapay zekâ alanına yapılan yatırımların ve beklentilerin artmasına neden olmuştur.

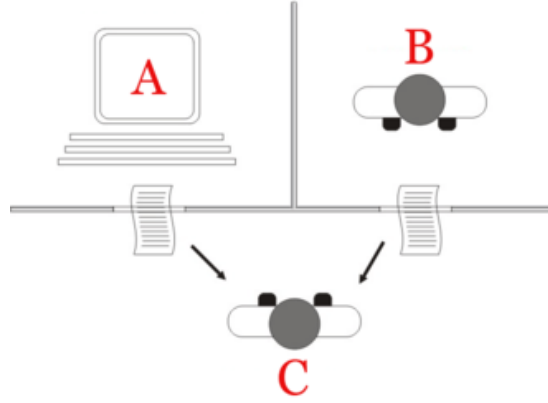


Şekil 12. Yapay zekâ, makine öğrenmesi ve derin öğrenme

İnsan gibi düşünen ve hareket eden sistemler tasarlamak, insanoğlunun yüzyıllardır ilgisini çeken konulardandır. Terminolojik olarak net bir isimle anılmasa da bu çaba, yapay zekâ kavramının ortaya çıktığı 1950’li yılların çok daha öncesine dayanmaktadır.

Yunan mitolojisinde Daedalus yetenekli ve zanaatkar bir karakterdir. Usta bir heykeltıraş olmasının yanı sıra mekanik sistemlere merakı ile bilinmektedir. Yaptığı heykeller, insanlar tarafından gerçeğe yakınlıkları ile adeta bir “yapay insan”a benzetilmektedir. “Yapay zekâ” her ne kadar o dönemlerde kullanılan bir terim değilse de farklı adlandırmalarla aynı arayış yüzyıllar boyu devam etmiştir.

Yapay bir zekânın var olup olamayacağı sorularına öncü olan ve makine zekâsının sınırlarının ilk kez tartışılmasında büyük rol oynayan Alan Mathison Turing, kendi adıyla anılan ve bugün hala pek çok çalışmaya temel olan “Turing Testi” kavramını ortaya atmıştır. 1950 yılında bir felsefe dergisi olan Mind dergisinde “Computing Machinery and Intelligence” başlıklı makalesinde açıkladığı bu test ile makinelerin düşünebilir olmalarına mantıksal açıklamalar getirmeye çalışmıştır (Turing, 2009). Test, temelde bir makinenin insan zekâsını taklit etme yeteneğini ölçmektedir.



Şekil 13. Turing testi ve oyuncular (URL-7, 2022)

Turing testinde değerlendirici (C) ve ikna edici (A ve B) pozisyonunda bulunan oyuncular yer almaktadır (Şekil 13). Burada A ve B oyuncularından biri insan (B), diğeri ise insan zekâsını taklit eden bir makinedir (A). Değerlendirici (C) sorduğu sorulara aldığı yazılı cevaplar neticesinde hangi oyuncunun makine olduğunu tespit edemez ise, makine değerlendiriciyi insan olduğuna ikna etmiş ve testi başarıyla geçmiş olacaktır.

“Makineler insan zekâsını taklit edebilir ve düşünebilir.” tezi o dönemlerde sıkça tartışılan, ancak bazı kesimlerce kabul edilebilirliği düşük olan bir tezdi. Makine ve insan arasındaki temel beceri farklılıklarının ve insan doğasının karmaşıklığının, makinelerin insan gibi düşünebilmesine engel olacağı görüşü hakimdi. Turing, test ile insan ve makine arasındaki bu farkı görünmez kılarak tezin doğruluğunu ispat etmeyi hedeflemişti.

“İnsansı makineler”i tartışmaya açan Alan Turing, “yapay zekânın babası” olarak bilinmektedir ve savları günümüzde hala tartışılmakta, çalışmaları yapay zekâyâ temel oluşturmaktadır.

Yapay zekâ kavramının terminolojik kimliğine bürünmesi ilk kez 1956 yılında New Hampshire, Dartmouth College’de düzenlenen ve döneminin başarılı araştırmacılarının ağırlandığı bir konferansta bilgisayar bilimci John McCarthy tarafından kullanılması ile gerçekleşmiştir. Günümüze kadar ulaşan “yapay zekâ” kavramı bir terim olarak ilk kez bu konferans ile ortaya atılmıştır.

İnsan zekâsının yapay bir formunun oluşturulabileceğini savunan John McCarthy, 1979 yılında yayınladığı bir çalışmada (McCarthy, 1979) makinelerin insana özgü birtakım zihinsel niteliklere ve yeteneklere sahip olabileceğini savunmuş, bu özelliklerin de onlara problem çözme yetisi kazandıracağını belirtmiştir.

Yine bu yıllarda “Bilim Kuramının Babası” olarak bilinen Claude Elwood Shannon bilgisayarların satranç oynayabileceklerini ileri sürmüştür ve taşların olası tüm hareketlerini programlamayı hedeflemiştir (Shannon, 1950). Bunun gibi, yapay bir zekâ ile insanın karşılıklı oynayabileceği oyunlar üzerine çalışmalar uzun yıllardan beri süregelmektedir.

1950’li yıllarda yapay zekâ alanındaki bu hızlı gelişmeler, makinelerden beklentilerin de artmasına neden olmuştur.

Programlanabilen bilgisayarların ortaya çıkmasıyla makinelerin yeteneklerinin sınırları hakkında pek çok varsayım ortaya atılmıştır. Süregelen “Bilgisayarlar bir problemi kendi kendine çözebilir mi?” sorusu “Bilgisayarlar öğrenebilir, düşünebilir ve bir problemin çözümünde doğru kararları verebilir mi?” şekline dönüşmüştür. Bu durum yapay zekâdan beklentilerin her geçen gün arttığını gözler önüne sermektedir.

Yapay zekâ sayesinde ilk zamanlarda insanların çözmekte güçlük çektiği, bilgisayarlar için ise sıradan olan basit matematiksel hesaplara ve temellere dayanan problemler hızla çözülmüştür. Ancak ilerleyen zamanlarda esas problemin, insanlar için sıradan ancak matematiksel olarak tanımlanması zor olan problemlerin ifadesi olduğu görülmüştür. Çünkü bu problemlerin tanımlanması için insan dilinin makine dilindeki karşılığı bulunmalı ve bu karşılık, bilimsel temellere ve biçimsel kurallara dayanmalıydı.

Uzun yıllar yapay zekâyı insan zekâ ve yetenekleri seviyesine çıkaracak kurallar dizininin elle kodlanmasının mümkün olabileceği görüşü hakimdi. Bu yaklaşım, 1950-1980 yılları arasında oldukça uzun bir süre hakim olan sembolik yapay zekâ devrinin temel taşıydı.

İnsanoğlu tarih boyunca makinelerin insan gibi öğrenme, düşünme ve karar verme yetilerine sahip olma ihtimalini tartışmışlardır. Bir kesim tamamen insan gibi düşünebilen ve hareket edebilen sistemlerin var olabileceğinden, hatta bu sistemlerin dünyayı ele geçirebileceğinden bahsederken; diğer bir kesim bu sistemlerin yalnızca manuel olarak verilen bilgilere göre hareket edebileceğini, herhangi bir öğrenme ve kendi kendine karar verme yetisinin mümkün olamayacağını savunmuştur. Hem bu ikilik hem de dönemin imkanlarını aşan yüksek beklentiler, 1970’li yıllarda yapay zekâyı ayıran bütçelerin azalmasına sebep olmuştur ve yapay zekânın ilk kışı olarak da adlandırılan sürece girilmiştir. 1980 yılına gelindiğinde yapay zekânın öneminin ivme kazandığı yıllar yeniden başlamıştır. Burada bahsi geçen yapay zekâ kavramı, yine bir “sembolik yapay zekâ”yı temsil etmektedir. “Uzman sistemler” olarak da anılan bu dönem teknolojik gelişmelere paralel olarak yapay zekâ alanına yapılan yatırımların artmasına neden olmuştur. Bu yıllarda yapay

zekânın ivme kazanmasının bir diğer nedeni ise yapay zekâ uygulamalarının daha somut ve güncel hayata uygulanabilir hedeflerle tasarlanmaya çalışılmasıdır. El yazısı ve şekil tanımlama sistemleri, kelime tahmininde bulunan sistemler gibi günlük hayata entegrasi kolay çalışmalar ilgi çektiğinden bu alana yatırım da fazla olmuştur.

1990'lı yıllarda yapay zekânın ikinci kışı dönemine girilmiştir. Artan maliyetler, karşılanamayan ütöpik beklentiler, uygulamada yaşanan zorluklar ve sistemsel gereksinimler tekrar duraklama dönemine girilmesine neden olmuştur. Duraklama döneminde yatırımlar ve projeler yer yer devam etse de istenilen düzeyde ilerlemenin sağlanamadığı bir gerçektir.

2000'lerden günümüze kadar olan süreçte yapay zekâ alanında gerçek hayatı ve hayatın işleyişini kolaylaştırmaya yönelik somut adımlar göze çarpmaktadır. Özellikle bulunduğumuz dönemde yapay zekâ için iyimser yaklaşımların ve yüksek beklentilerin söz konusu olduğu bir çağı yaşamaktayız. Yapay zekâ bilim dalı, zamanla değişen ihtiyaç ve gelişen algoritmalar ile makine öğrenmesi ve derin öğrenme bilim dallarının oluşmasına yol açmıştır. Farklı adlarla anılan ve birbirine paralel gelişim gösteren bu süreçler günümüzde hala değişime uğramakta ve gelişmeye devam etmektedir.

Günümüzde yüksek donanımlı bilgisayarlar, veriye ulaşmadaki kısmi kolaylıklar, amaçlara uygun tasarlanmış altyapılar ve bilişim ağının getirdiği sınırsız bilgi paylaşımı; yapay zekâ, makine öğrenmesi ve derin öğrenme kavramlarıyla yeni tanışan girişimciler için dahi kolay bir çalışma ortamı sunmaktadır.

Varoluşu 1950'li yıllara dayanan sembolik yapay zekâ, iyi tanımlanmış ve matematiksel temsillere dayanan problemlere iyi çözümler sunmaktaydı. Ancak problemler karmaşık ve soyut bir hale geldikçe, istenen hassasiyette çözümler üretilemediği fark edildi. Örneğin, matematiksel kurallarla tanımlanmış bir oyunda algoritma iyi çözümler sunarken; aynı başarıyı ses tanıma, metin tanıma, nesne tespiti gibi daha spesifik ve temsili zor problemlerde gösteremiyordu. Çünkü “öğrenme” işleminde yetersizdi.

Belirli ve kesin tanımlanmış kurallara dayalı bir oyunun, kuralları ile bilgisayara tanımlanması sonucu bilgisayar “öğrenme” işlemini gerçekleştirmiş olmayacaktır. Çünkü kullanıcı tüm kuralları manuel olarak tek tek algoritmaya tanımlamıştır ve bu süreç makinenin herhangi bir bilgi çıkarımını içermediğinden öğrenmeyi de kapsamamaktadır.

Zamanla makinelerin çözümleyebileceği problem çeşidinin artması için, tıpkı bir “insan” gibi kendi bilgilerini elde etmesi yani “öğrenmesi” gerekliliği ortaya atıldı. Bu durum “makine öğrenmesi” kavramının ortaya çıkmasına neden olmuştur.

1.5.3. Makine Öğrenmesi

Yapay zekâ algoritmaları öğrenme süreci içeren ve içermeyen pek çok uygulamayı kapsar ancak makine öğrenmesi algoritmaları öğrenme tabanlı uygulamalardır.

Makine öğrenmesinde temel amaç, bilgisayarın problemlere yönelik doğru cevapları bulabilmesi için kuralları kendi kendine belirlemesini sağlamaktır. Dolayısıyla bu süreç bir programlama değil, öğrenme sürecidir. Makine öğrenme algoritmaları “Bilgisayarlar kendi kendine öğrenme işlemini gerçekleştirebilir mi?” sorusunu cevaplandırmak üzere tasarlanmış algoritmalardır.

Klasik yaklaşımda kurallar ve veriler geliştirici tarafından sağlanırken, makine öğrenmesinde veri ve cevaplar girdidir (Tablo 1) ve kuralları algoritmanın belirlemesi beklenmektedir (Chollet, 2021). Dolayısı ile makineye düşen asıl görev, bir girdiyi istenilen sonuca ulaştıracak kurallar bütününi inşa etmektir. Kuralların girdiyi ne ölçüde beklenen çıktıya ulaştırdığı devamlı ölçülmektedir ve algoritma, kendi başarısını devamlı denetler. Süreç, verinin en iyi temsili bulunana dek sürer. Bu sürece “öğrenme süreci” denir.

Tablo 1. Yeni bir paradigma: makine öğrenmesi (Chollet, 2021)

Algoritma	Girdiler	Çıktılar
Klasik Yaklaşım	Veri + Kurallar	Cevaplar
Makine Öğrenmesi	Veri + Cevaplar	Kurallar

Burada “eğitim” kavramı öne çıkmaktadır. Çünkü makine öğrenmesinde temel amaç programlama değil eğitim ve bunun sonucu olarak da öğrenmedir. Eğitim sonucu belirlenen kurallara göre problemlere çözüm üretebilen makineler, değişen veri kümeleri için de doğru sonuçlar üreteceklerdir.

Bir makine öğrenmesi algoritmasının inşası için üç temel basamak vardır:

- Girdi veri
- Beklenen çıktı
- Başarı ölçütü

Algoritmanın amacı girdi verileri beklenen çıktılara ulaştıracak kurallar dizinini inşa etmek ve geri bildirimler ile bu süreci eniyilemektir. Algoritma bu işlemi, veriden farklı gösterimler bularak gerçekleştirmektedir. Girdileri beklenen çıktılara ulaştıracak ideal veri gösterimleri algoritma tarafından tespit edilir ve öğrenme sürecine paralel olarak daha iyi gösterimler elde edilir.

Teknolojik gelişmelere paralel olarak “veri” kavramının tanımı değişmiş ve gelişmiştir. Özellikle oyun sektörüne yapılan yatırımlar sonucu gelişen donanımlar büyük verilerin işlenmesini mümkün kılmış, karmaşık veri setleri ile yapılan çalışmalar bu alandan beklentileri arttırmıştır. Günümüzde çok büyük boyutta veriler yüksek donanımlı bilgisayarlarda kolaylıkla analiz edilebilmektedir. Ayrıca bu bilgisayarların yaygınlaşması ve bu alana ait projelere yatırımların çoğalmasıyla farklı amaçlara hizmet eden çok yönlü makine öğrenme algoritmaları geliştirilmiştir. Bu algoritmalar, makine öğrenmesinin çok daha karmaşık alt alanlarını temsil etse de temelde dört başlık altında toplamak mümkündür:

1.5.3.1. Denetimli Öğrenme

Denetimli öğrenmede temel amaç, algoritmanın eğitim örneklerini kullanarak girdileri hedef çıktılara dönüştüren kuralları öğrenmesidir. Kullanıcı, elindeki verilerle ilgili ön bilgiye sahip ise verileri etiketler ve modelini eğitir. Veri etiketleme sayesinde veriler algoritmaların daha kolay algılayabileceği temsillere dönüşmektedir. Günümüzde gerçekleştirilen çalışmaların çoğu denetimli öğrenmeye dayanmaktadır.

Denetimli öğrenme projelerinde verilerin hazırlanması ve etiketlenmesi aşamaları, modelin eğitimi süresince dikkat edilmesi gereken temel aşamalardandır. Eğitim verisi ne kadar doğru etiketlenmiş ve tüm olasılıkları kapsayacak miktarda fazla ise model de o kadar doğru eğitilmiş olacaktır.

Örnek olarak bir nesne tespit problemi ele alınsın. Çeşitli nesnelerin bulunduğu bir ortamda insanların sayısını hesaplayan bir ağın tasarımı problemi eğer denetimli öğrenme algoritmaları ile çözümlenecek ise, etiketlenmiş veri setine gereksinim duyulacaktır. İnsanlar farklı ten renklerinde, boylarda, yapılarda olabilmektedirler. Etiketlenmiş veri setimiz tüm bu olasılıkları içerdiği ve verinin değişik gösterimlerini de kapsayabildiği derecede ağıımızı doğru eğitecektir.

Veri etiketleme işi eğer alan uzmanlığı gerektiren bir konuda ise uzmanlar tarafından, değilse herhangi bir kullanıcı tarafından da gerçekleştirilebilmektedir. Günümüzde açık

verinin yaygınlaşması ile önceden etiketlenmiş veri setleri pek çok projede kullanılmakta ve iş yükünü hafifletmektedir.

1.5.3.2. Denetimsiz Öğrenme

Denetimsiz öğrenme algoritmaları, denetimli öğrenmenin aksine etiketlenmiş veri ile çalışmazlar. Bu algoritmalar kullanıcıların veri hakkında herhangi bir ön bilgiye sahip olmadığı durumlarda kullanılmaktadır. Algoritma, verileri benzerliklerine göre gruplara ayırarak “kümeleme” işlemini gerçekleştirmektedir. Bu durum bazı dezavantajlara neden olmaktadır. Örneğin, algoritmanın sonuç çıktısı hedeflenen sınıflardan bazılarını içermeyebilir veya bir nesne herhangi bir grupta temsil edilememiş olabilir. Ayrıca algoritma, çıktı katmanında o verinin “ne olduğunu” söyleyemeyecek, ancak “nelerden farklı olduğu ve nelere benzediği” yorumunu yapabilecektir.

Denetimsiz öğrenme algoritmalarının dezavantajlarının yanı sıra avantajları da oldukça fazladır. Kimi zaman denetimli öğrenmenin ön aşamasında veri analizi için kullanılmakta ve veride kullanıcı tarafından gözlemlenemeyen öznel özelliklerin açığa çıkarılmasını sağlamaktadırlar. Denetimsiz öğrenme algoritmalarının kullanılış amaçlarından bazıları şunlardır: veriyi sıkıştırmak, görselleştirmek, kümelemek, gürültü azaltmak, boyut indirgemek, bir veri kümesi içinde korelasyonlu verileri bir kümede toplayacak dönüşümleri bulmak...

1.5.3.3. Yarı Denetimli Öğrenme

Yarı denetimli öğrenme, denetimli ve denetimsiz öğrenmenin arasında bir yere konumlandırılabilir. Hem denetimli hem de denetimsiz öğrenmenin birbirine göre üstün ve üstün olmayan yönleri bir arada düşünüldüğünde, yarı denetimli öğrenme algoritmaları iyi bir seçenek olacaktır. Denetimli öğrenmede etiketli verilerin kullanıldığına değinilmişti. Yarı denetimli öğrenmede ise bu etiketli verilere gereksinim duyulmamaktadır ve bu yönüyle denetimli öğrenmeden ayrılmaktadır. Ancak bu durum, yarı denetimli öğrenme sürecinin hiçbir şekilde denetime tabii tutulmayacağı anlamına gelmemektedir. Etiketler vardır ancak bu etiketler girdilerden sezgisel olarak oluşturulmaktadır (Chollet, 2021).

Otokodlayıcılar girdi verilerin yeniden yapılandırılmasını sağlayan bir tür derin öğrenme ağıdır ve yarı denetimli öğrenmenin yaygın olarak kullanılan bir yöntemidir.

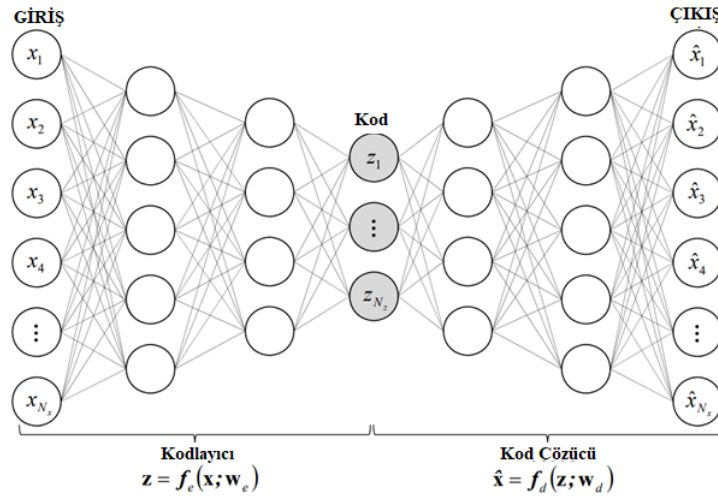
Otokodlayıcı, bir problemin çözümü için gerekli bilgileri veriden otomatik olarak çıkarmak üzere tasarlanmış bir çeşit sinir ağıdır. Bir otokodlayıcı, girdi vektörüne sınıf etiketi vermek yerine onu yeniden oluşturmak için eğitilmektedir (Ravi vd., 2017).

Bir derin öğrenme algoritmasında otokodlayıcıların kullanılma amacı, bilgiyi otomatik olarak çıkarmanın yanı sıra boyut indirgeme işlemini yapmaktır (Hinton ve Salakhutdinov, 2006).

Bir otokodlayıcı 2 bölümden oluşmaktadır:

- Encoder (Kodlayıcı): Gizli katmana giren ağırlıklar
- Decoder (Kod Çözücü): Gizli katmandan çıkan ağırlıklar

Kodlayıcı kısmında girdi veri giderek boyutların daha da azaltıldığı katmanlarca işlenir. Kod çözücü kısımda ise boyutu azaltılmış veri giderek boyutların artırıldığı katmanlarca işlenir ve sonuç olarak kodlayıcıya giren verinin boyutu ile kod çözücüden çıkan verinin boyutları birbirine eşittir (Şekil 14).



Şekil 14. Bir otokodlayıcının yapısı (Canchumuni vd., 2019)

Otokodlayıcıların çalışma prensibi yapay sinir ağlarına benzemektedir. Yapay sinir ağında girdi katmanı, gizli katman ve çıktı katman olmak üzere 3 katman vardır. Otokodlayıcılar daha karmaşık veriler için bu yapıları kullanarak analizler yaparlar. Yapay sinir ağlarında çıktı katman sayısı sınıf sayısı baz alınarak belirlenirken, otokodlayıcılarda girdi ve çıktı katmanların eşit nöron sayısına sahip olması gözetilmektedir. Böylece girdi

katmanındaki veri çıktı katmanında yeniden oluşturulmuş olur ve girdi verisinin boyutları azaltılmış gösterimlerinden özniteliklerin öğrenilmesi gerçekleştirilir. Çıktı katmanında beklenen gösterime ulaşıncaya dek ağırlıklar geri yayılım ile güncellenir.

Otokodlayıcıların başlıca kullanım alanları arasında görüntü tanımlama, gürültü azaltma, boyut indirgeme ve kelime tahmin motorları örnek gösterilebilir.

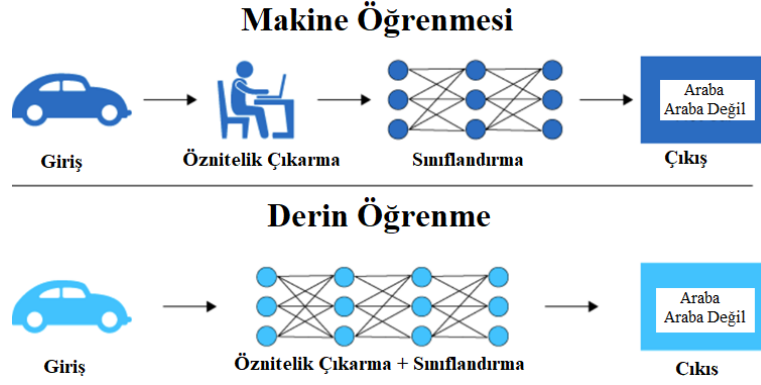
1.5.3.4. Pekiştirmeli Öğrenme

Pekiştirmeli öğrenme diğer makine öğrenme algoritmalarına göre daha az ön planda olsa da gelişime açık ve sınırları tartışılan bir alandır. Amaç doğrultusunda çalışan bu algoritmalar, amaca ulaşmak için yapılması gereken görevleri öğrenir ve uygularlar.

Bu algoritmaların temeli “ajan” ve “keşif” kavramlarına dayanmaktadır. Ajan sürekli bir keşif halindedir, çevre ile iletişime girer ve her bir adımını en iyiye ulaşacak şekilde tasarlar. Verdiği tepkilere göre bir ödül puanı alan ajan, bu puanı maksimuma çıkarana kadar keşfe devam eder. Bir tür deneme yanılma olan bu yöntemde ajan önceden denediği ve başarısını arttıran eylemleri seçerek işlemlere aşamalı olarak devam etmekte ve öğrenmeyi gerçekleştirmektedir.

1.5.4. Derin Öğrenme

Değişen veri yapıları ve artan veri boyutları derin öğrenme kavramını ortaya çıkarmıştır. Derin öğrenme algoritmalarında amaç aslında makine öğrenme algoritmaları ile aynıdır: veriden daha iyi gösterimler elde eden dönüşümler kuralını bulmak. Ancak makine öğrenme algoritmalarında birkaç katman söz konusu iken bir derin öğrenme mimarisi pek çok katman içermektedir (Şekil 15).



Şekil 15. Makine öğrenmesi ve derin öğrenme algoritmalarının işleyişi (URL-8, 2022)

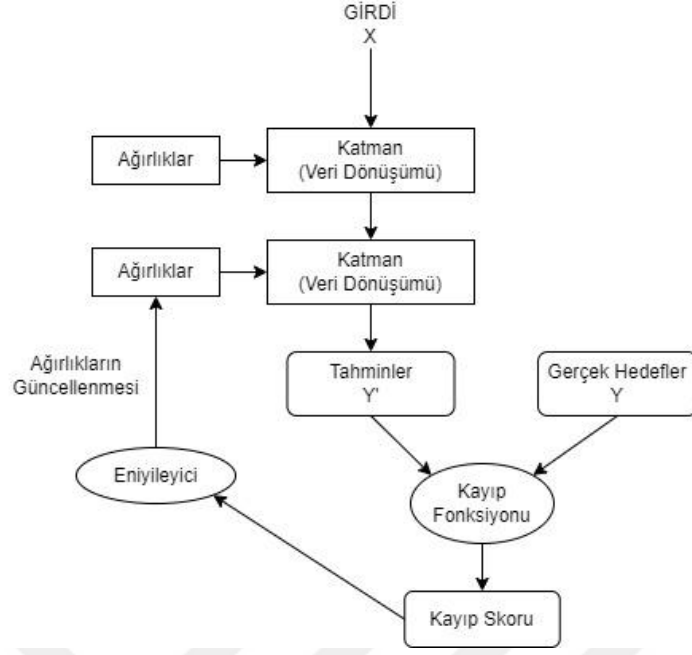
Bu katmanlar birbirini takip eden birer gösterim katmanıdır. Katman sayısı, model derinliği ile ifade edilmektedir.

Makine öğrenme algoritmaları küçük ve orta boyutlardaki veriler ile çalışırken, derin öğrenme algoritmaları büyük boyutlu verileri dahi kolaylıkla işleyebilmektedir. Eğitim süreleri de veri boyutları arttıkça artacaktır. Dolayısı ile, makine öğrenmesi ve derin öğrenme algoritmaları için ihtiyaç duyulan sistem gereksinimleri farklıdır. Derin öğrenme algoritmaları genellikle büyük verilerde karşılaşılan sorunların çözümü için tasarlandığından, daha güçlü sistemlere ve gelişmiş donanımlara gereksinim duymaktadırlar. Son yıllarda özellikle oyun sektörüne yapılan yatırımlar güçlü donanımları da beraberinde getirmiş, bu durum derin öğrenme algoritmalarının gelişiminin hızlanmasına zemin hazırlamıştır.

Makine öğrenme algoritmalarında veriler için öznitelik analizi gereklidir. Problemin çözümüne yönelik öznitelikler tespit edilerek uygun olanları seçilir. Öznitelik mühendisliği olarak da adlandırılan bu süreç, derin öğrenme algoritmaları için gerekli değildir.

Makine öğrenme algoritmaları daha çok yapısal verilerin analizleri için tasarlanmışlardır. Yapısal veriler matematiksel ifadeleri kolay olan, vektör formunda ifade edilen verilerdir. Yapısal olmayan veriler ise tanımlanması insanlar için kolay ancak bir bilgisayarlar için zor olan verilerdir. Ses tanıma, metin analizi, yüz tanıma gibi problemler yapısal olmayan problemlerdir.

Bir derin öğrenme algoritmasının çalışma prensibi basit olarak şu şekilde ifade edilebilir (Şekil 16):



Şekil 16. Derin öğrenme ağlarında süreçler (Chollet, 2021)

İlk katman görünür katmandır ve girdi veri burada yer almaktadır. Veri, katmanlarda sırayla çeşitli işlemlerden geçirilir. Her bir katmanın veriye uyguladığı bu işlem, katmanların ağırlıkları baz alınarak gerçekleştirilmektedir. İşlemlerin sonucunda bir tahmin elde edilir ve bu tahmin ile gerçek çıktı arasındaki fark hesaplanır. İdeal bir ağda bu farkın olabildiğince az olması beklenmektedir. Bu fark kayıp fonksiyonudur ve kayıp fonksiyonu, ağ başarısının bir ölçütüdür. Kayıp fonksiyonunun sonucu ağ tarafından devamlı olarak kontrol edilir ve ağ gerçek çıktıya en yakın sonucu verene yani kayıp fonksiyonunu minimuma indirgeyene kadar katmanları tekrar tekrar ağırlıklandırır. Derin öğrenme adı da buradan gelmektedir: katman sayısınca derinliğe sahip model, girdiyi beklenen çıktıya ulaştıran ağdaki tüm katmanların ağırlıklarını tekrarlı olarak bulur ve öğrenme işlemini gerçekleştirir. Öğrenmenin ilk aşamalarındaki kayıp puanı, öğrenmenin sonunda olabileceği minimum değerine ulaşmış olacaktır. Kayıp puanını 0'a yakınsayan derin öğrenme ağı, başarılı bir ağ sayılmaktadır (Chollet, 2021).

Yapay zekâ, makine öğrenmesi ve derin öğrenme algoritmaları birbirleri ile kıyaslandığında yapay zekânın makine öğrenmesi ve derin öğrenmeye göre; makine öğrenmesinin ise derin öğrenmeye göre daha matematiksel ağırlıklı olduğunu söylemek mümkündür. Ancak buradan derin öğrenme algoritmalarının matematiğe daha az ihtiyaç duyduğunu söylemek yanlış olacaktır. Buradaki kıyas, algoritmaların birbirlerine göreceli durumları baz alınarak yapılmıştır.

Özellikle 2010’lu yıllarda derin öğrenme, bilgisayarlı görü tabanlı çalışmalarda çok iyi sonuçlar elde etmiştir. Derin öğrenmenin bu başarısının diğer alanlarda da dikkat çekmeye başlaması ile bu alana bir yatırım dalgası başlamıştır. Girişimciler projeler için çok büyük bütçeler ayırmış; dünya devi şirketler yapay zekâ, makine öğrenmesi ve derin öğrenmeyi neredeyse her alanda kullanmaya çalışmışlar, bu alanlarda adeta birbirleri ile rekabet eder hale gelmişlerdir.

“Bilgi Çağı” olarak adlandırılan bu yüzyıl, bilgiye erişimin en kolay ve hızlı olduğu yüzyıldır. Geçmiş yıllarda derin öğrenme üzerinde çalışmalar yapabilmek için sınırlı ara yüzler mevcuttu ve alanda uzmanlık gerekmektedir. Bugün sınırsız kütüphaneler ve herkes tarafından kolaylıkla kullanılabilen basit programlama dilleri, toplumun ayırt etmeksizin her kesiminden araştırmacının derin öğrenme üzerinde yoğunlaşmasına neden olmuştur.

Günümüzde yapay zekâ yerini kısmen derin öğrenmeye bıraksa da aslında derin öğrenmenin ipuçları yapay zekâdan gelmektedir. Bu üç bilim alanı terminolojide farklı anlamlar içerse de tarihsel süreçler takip edildiğinde birbirlerini besledikleri görülmektedir.

İnsanoğlu makinelerin bilgi öğrenme kabiliyetlerinin sınırlarını hep merak etmiştir. Yüzyıllar süren bu süreçte kimi zaman insanoğlunun yapay zekâdan beklentisi fazla olmuş, kimi zaman donanımsal imkanlar çalışmaları sınırlandırmış, kimi zaman da bu çalışmalara ayrılan bütçelerle ilgili sorunlarla karşı karşıya kalınmıştır. Çeşitli düşüncelerin tartışıldığı dönemlerde yapay zekânın kışı denilen dönemler başlamış, teknolojik gelişmeler ve iyimser beklentilerle bu dönemler sonlanmıştır. Tüm bu dalgalı süreçler göz önüne alındığında, şu anda yapay zekâ için ayrılan bütçelerin fazla olduğu ve yüksek donanımlı bilgisayarların yaygınlaştığı bir çağda bulunmaktayız. Alana yapılan yatırımların özellikle son yıllarda hiç olmadığı kadar fazla olduğu aşıkardır. Ancak bu yatırımların çok az bir kısmı günlük hayata entegre edilmiş ürünlere dönüştürülebilmıştır. Yapay zekânın yeniden kışı gelir mi bilinmez ancak dönemin ihtiyaçlarını karşılayan gerçekçi projeler ve makul beklentiler ile yapay zekânın gelişimine ivme kazanarak devam edeceği görüşü hakimdir.

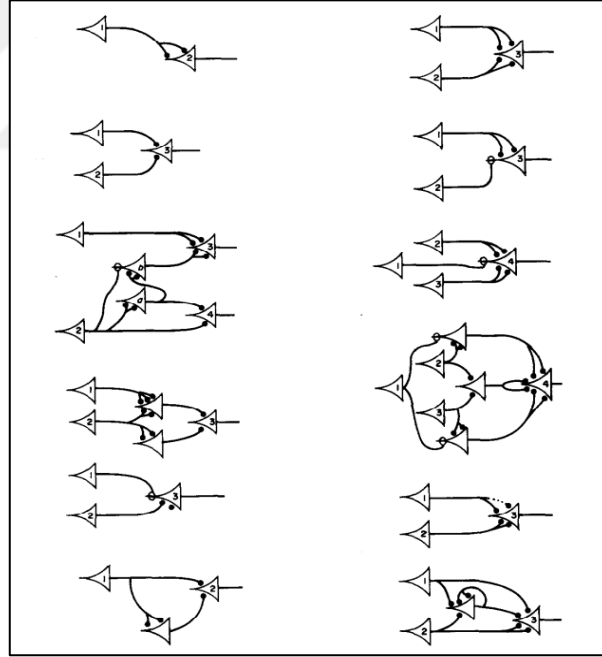
1.5.4.1. Yapay Sinir Ağları

Yapay zekânın “Makineler de insanlar gibi düşünebilir mi?” sorusu “Makineler insanlar gibi nasıl düşünebilir, öğrenebilir ve karar verebilir?” şekline evrilmiş ve yapay sinir ağlarının hayatımıza girmesine neden olmuştur.

Yapay zekâ ile anlam kazanan sibernetik ya da güdüm bilimi, insan zekâsını ve öğrenme süreçlerini matematiksel modellerle somutlaştırmayı ve insansı makineler üretmeyi hedefleyen bir bilim dalıdır.

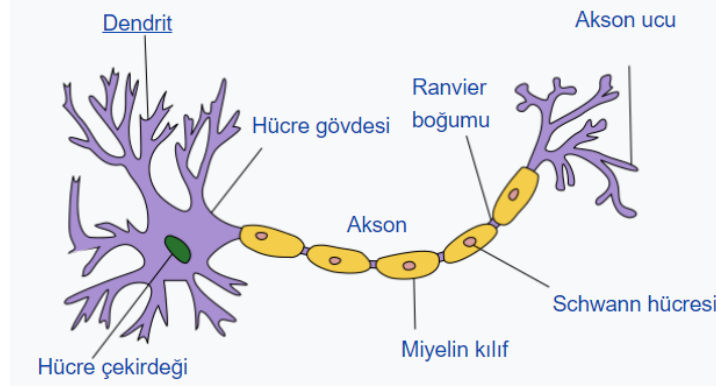
İnsan beyninin ve öğrenme süreçlerinin modellenmesi konuları yapay zekâ alanına duyulan ilgiyle beraber artmıştır.

1943 yılına gelindiğinde Walter Pitts ve Warren McCulloch adlı bilim insanları, sinir ağlarının ilk modelini ortaya atmışlardır (McCulloch ve Pitts, 1943). Araştırmacılar insan beyninden esinlenerek bir elektronik beyin tasarlamışlardır. Bu elektronik beyin, insan düşüncelerini taklit eden ve matematiksel temellere dayandırılan bir modeldir. Araştırmacılar, insan beynindeki bu süreçlerin modellenebilir olduğunu savunmuşlar ve insan beyninin öğrenme ve analiz etme yeteneğini elektrik devreleri kullanarak görselleştirmişlerdir (Şekil 17). Çalışma yapay sinir ağları alanında atılan ilk bilimsel adımlardan olması nedeni ile bugün dahi araştırmacıların ilgisini çekmektedir.



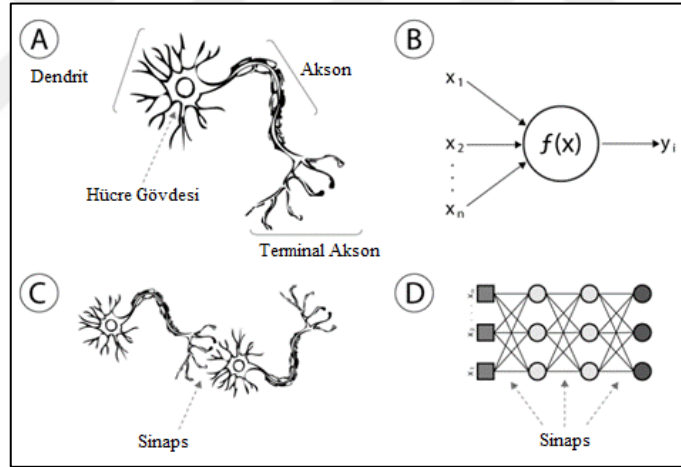
Şekil 17. Sinir ağlarının ilk modeli (McCulloch ve Pitts, 1943)

Yapay sinir ağları ile insan beyninde bilgi transferlerini gerçekleştirmekle yükümlü nöronların (Şekil 18) temsili ve modellenmesini hedeflemektedir.



Şekil 18. Nöron modeli (URL-9, 2022)

Maltarollo 2013 yılında yayınladığı makalesinde (Maltarollo vd., 2013) insan beynindeki nöronlar ile bir yapay sinir ağındaki elemanları kıyaslamış ve görselleştirmiştir (Şekil 19).



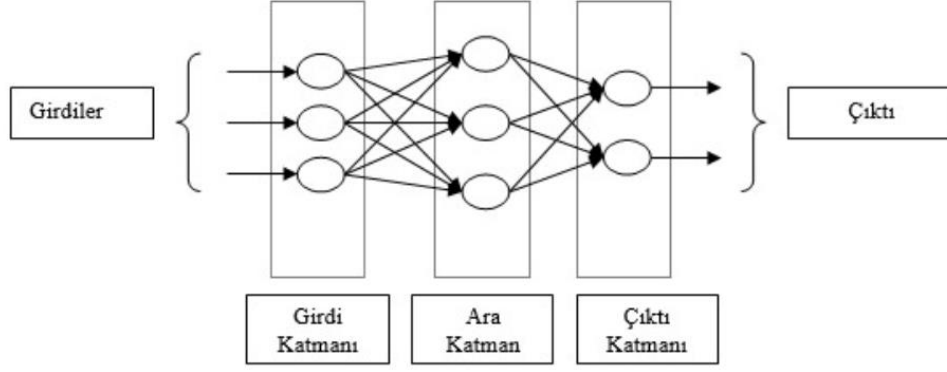
Şekil 19. (A) İnsan nöronu; (B) yapay nöron; (C) biyolojik sinaps; (D) yapay sinaps (Maltarollo vd., 2013)

Sinir ağları alanındaki ilk çalışmalar 1950'lere dayansa da gelişen teknoloji ve yüksek beklentilere bağlı olarak çalışmaların son yıllarda ivme kazandığı bilinmektedir. Bilgisayarların yapay bir zekâya sahip olmasında en önemli adım olan öğrenme adımının nasıl gerçekleştirileceği uzun yıllar tartışma konusu olmuş, “geri yayılım” algoritmalarının ağlara uyarlanması ile eğitim kavramı yeni bir boyut kazanmıştır.

Bir yapay sinir ağı şu etmenleri içermektedir:

- Katmanlar

- Girdi verisi ve beklenen çıktılar
- Kayıp fonksiyonu
- Eniyileme algoritması



Şekil 20. Yapay sinir ağlarının yapısı (URL-10, 2022)

Bir yapay sinir ağında birden çok katman vardır (Şekil 20). Katmanlar sinir ağlarını oluşturan, verinin işlendiği kısımlardır ve tensörü tensöre eşleyen modüllerdir. Matematikte tensör, çok boyutlu verinin simgelenbildiği geometrik bir nesnedir (URL-11, 2022). Katmanlarda veri dönüşümleri yapılmaktadır ve bu işlem katmana tanımlı belirli ağırlıklarla gerçekleştirilmektedir. Girdi veri katmana girer ve ağırlıklara göre dönüşüme uğrar. Katmanlar boyunca devam eden bu işlem sonucu tahminler oluşturulur. Tahminler ve beklenen çıktılar arasındaki fark kayıp skoru olarak hesaplanır. Kayıp skoru ile kayıp değeri hesaplanmış olur ve bu değer kullanılarak eniyileme algoritmaları ile kayıp değeri minimum değerine ulaşmaya kadar ağırlıklar devamlı güncellenir. Eniyileme algoritması, ağırlıkların ne ölçüde güncelleneceği bilgisini içermektedir. Amaç fonksiyonu olarak da adlandırılan kayıp fonksiyonu eğitim sırasında minimize edilecek, bu sayede ağın veri için daha kullanışlı gösterimlere ulaşması sağlanacaktır.

1.5.4.2. Evrişimli Sinir Ağları

Evrişimli sinir ağları, yapay sinir ağlarını esas alan mimarilerdir. Evrişimli sinir ağları evrişim katmanı içerir ve katman sayısı bir yapay sinir ağı mimarisine göre daha fazladır.

Bu fazlalık, eğitim süresini uzatacağından güçlü donanımlara olan gereksinimi de arttıracaktır.

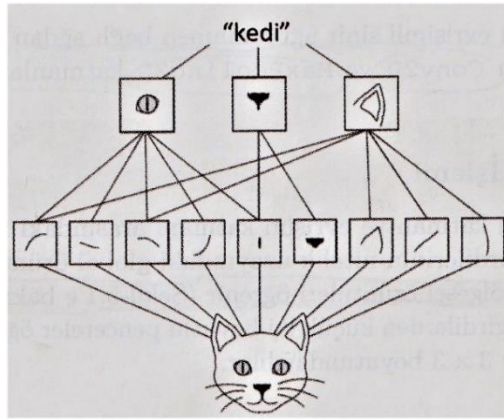
Bir evrişimli sinir ağı:

- Girdi
- Evrişim Katmanı (CONV)
- Tam Bağlantı (FC-Fully Connected)

basamaklarından oluşmaktadır. Evrişimli sinir ağları genellikle görüntü işleme problemlerinin çözümünde kullanılmaktadırlar.

Evrişimli sinir ağı, sinir ağının evrişim adı verilen bir matematik işlemini kullanmaktadır. Konvolüsyon (evrişim) özel bir doğrusal (lineer) işlem türüdür. Evrişimli sinir ağları, katmanlarından en az birinde genel matris çarpımı yerine evrişimi kullanan basit sinir ağlarıdır (Goodfellow vd., 2016).

Evrişimli sinir ağları, öğrendikleri örüntüleri bir görüntünün her yerinde tekrar tekrar tanıyabilirler. Dolayısı ile farklı bir bölgede aynı tür örüntü ile karşılaşıldığında yeniden öğrenme yapmasına gerek kalmayacaktır. Ayrıca her evrişimli katman, bir önceki evrişimli katmanın öğrendiği örüntünün üzerine katarak öğrenme işlemine devam eder (Şekil 21). Bu sayede giderek karmaşıklaşan öğrenme süreci özelden genele tümevarım ile basitleştirilmiş olacaktır (Chollet, 2021).

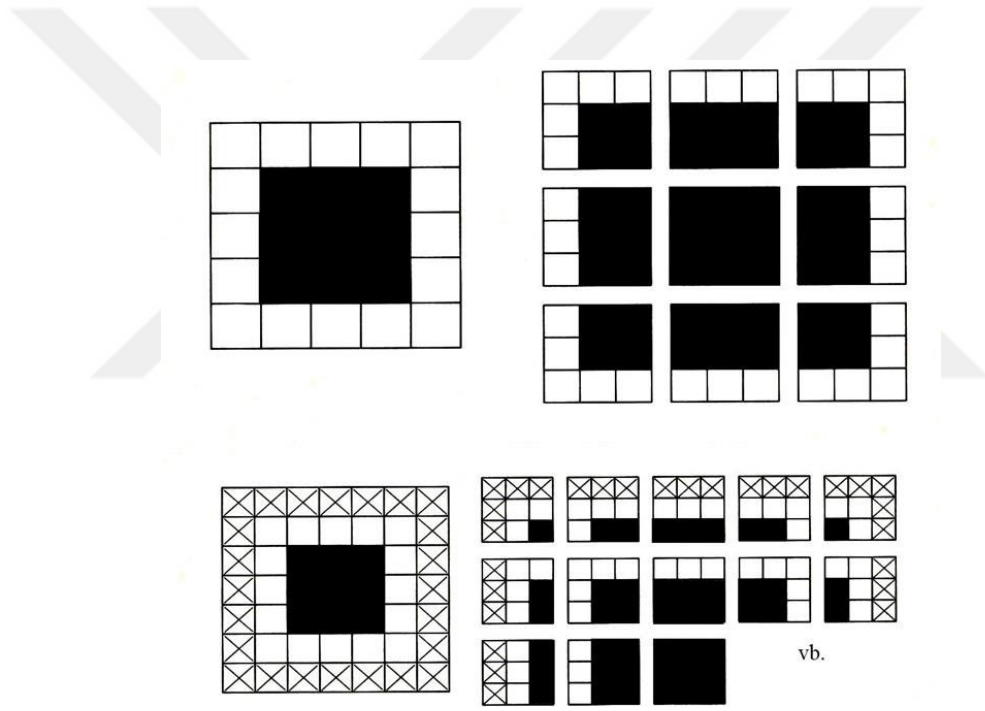


Şekil 21. Bir evrişimli sinir ağında örüntüler arası uzamsal hiyerarşiler (Chollet, 2021)

Evrişim aşamasında konvolüsyon (convolution), aktivasyon fonksiyonu ve ortaklama/havuzlama (pooling) basamakları yer almaktadır.

Konvolüsyon adımımda girdi görüntünün birtakım özellikleri çıkarılır ve öznitelik haritaları elde edilir. Aslında buradaki işlem, görüntüye farklı filtreler uygulayarak görüntünün çeşitli bölgelerine ait özelliklerin tespitini sağlamaktır. Filtreler ağırlık matrisleridir ve algoritma tarafından geri yayılım ile ağırlıklar devamlı güncellenir. Nöronlar için öğrenme gerçekleştirilir. Evrişim işlemi devam ettikçe tespiti yapılacak nesne ile ilgili daha fazla öznitelik algılanır ve bu durum algoritmanın veri ile ilgili en iyi gösterime ulaşmasını sağlar.

Evrişim katmanına giren ve bu katmandan çıkan verinin boyutları farklı olabilir. Bu durum kenar etkisi ve adım aralığından kaynaklanmaktadır. Evrişim katmanında girdi nitelik haritası üzerine uygulanacak filtreler sonucu çıktı nitelik haritasının boyutları azalacaktır.



Şekil 22. Evrişimli sinir ağında doldurma etkisi (Chollet, 2021)

Görsel incelendiğinde, 5×5 'lik veriye 3×3 'lük bir pencere uygulandığı görülmektedir (Şekil 22). Pencerenin veriye uygulanması sonucu yeni nitelik haritası 3×3 'lük olacaktır. Burada verinin boyutunun küçüldüğü gözlemlenebilir. Eğer girdi ile çıktı nitelik haritalarının aynı boyutta olması isteniyor ise veriye doldurma işlemi (padding) uygulanmalıdır. Bu adım, girdi katmanının boyutlarının genişletilerek yeniden yapılandırılmasını sağlamaktadır.

Çıktı büyüklüğü yalnızca kenar etkisi faktörüne değil, adım aralığına da bağlıdır. Adım aralığı, işlemler boyunca pencerenin kaç piksel ilerlediği ile ilgili bir kavramdır. Adım aralığı 1 veya 1'den çok olabilir. Adım aralığının 3 olarak alınması, nitelik haritasının üç katı oranında küçülmesi anlamına gelecektir. Ayrıca kenar etkisi ile bu boyut daha da azalabilir. Genellikle uygulamalarda kabul gören adım aralığı 1'dir ancak farklı adım aralıklarının uygulandığı teknikler de mevcuttur.

Bir evrişimli sinir ağının tasarlanma amacı, doğal zekânın yapay bir formunun oluşturulmasıdır. Yapay zekânın matematiksel ifadelere ve biçimsel kurallara dayalı problemleri çözmede yeterli olduğu ancak insanlar için kolay tanımlanabilen, makineler için ise tanımlanması güç olan ses tanıma, metin analizi gibi problemlerde aynı yeterliliği gösteremediği önceki bölümlerde açıklanmıştı. “İnsansı makineler” tasarlanmak isteniyor ise öncelikle dünya gerçekliğinin makinelere aktarılması gerekmektedir.

Aktivasyon fonksiyonları, bir yapay sinir ağında giriş sinyali çıkış sinyaline dönüştüren fonksiyonlardır ve dönüşüme uğrayan veri bir sonraki katmana girdi olarak verilir. Yapay bir sinir ağında girdi ürünler ve bunlara karşılık gelen ağırlıklar hesaplanarak bir çıktı elde edilir. Elde edilen çıktı, bir sonraki katmana girdi olarak sağlanmak üzere aktivasyon fonksiyonu tarafından işlenir ve diğer katmana sunulur (Sharma vd., 2017).

Aktivasyon fonksiyonu adımı amaç algoritmanın gerçek dünyayı tıpkı bir doğal zekâ gibi algılamasını sağlamaktır. Konvolüsyon adımı ile ağırlık matrisleri tarafından işlenen veri, bu adımda aktivasyon fonksiyonu ile sadece doğrusal değil, doğrusal olmayan durumları da öğrenecektir. Aktivasyon fonksiyonu katmanın son aşamasındaki verilere uygulanmaktadır.

Çeşitli aktivasyon fonksiyonları vardır. Doğrusal aktivasyon fonksiyonları tek derecelidirler. Bir yapay sinir ağı tasarımında yalnızca doğrusal fonksiyonlara yer vermek, ağın dünya gerçekliğini tam olarak algılayamaması ve veri analizini doğru yapamaması ile sonuçlanacaktır. Evrişimli sinir ağları sayısız katmandan oluşmaktadır ve tüm katmanlarda doğrusal fonksiyonların kullanımı, giriş katmanı ile çıkış katmanı arasında hep aynı doğrusal sonucu verecektir. Bu durum ara katmanları işlevsiz hale getirecek, katmanların tamamının birbiri ile entegre çalışmasını mümkün kılmayacaktır.

Çok dereceli fonksiyonlar ise doğrusal olmayan fonksiyonlardır ve bu fonksiyonlar sayesinde ağlar daha fazla öğrenme yetisine sahip olmaktadır.

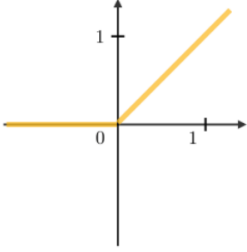
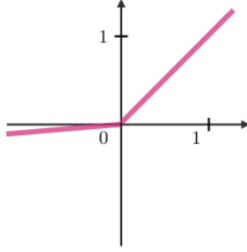
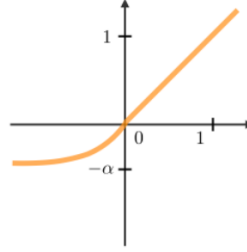
Önceki bölümlerde ilk yapay zekâ uygulamalarının matematiksel olarak iyi tanımlanmış ve biçimsel problemlere iyi çözümler sunduğu ancak ses tanıma, nesne tespiti

gibi matematiksel ifadelerle dayanmayan problemlerde pek de başarılı olamadığına değinilmişti. Aktivasyon fonksiyonu sayesinde artık sinir ağları yüz tanıma, ses tanıma, metin tanıma gibi pek çok gerçek dünyayı temsil eden karmaşık ve zor problemleri de anlamakta, tanımakta ve işlemektedir.

İyi çözümler sunan bir aktivasyon fonksiyonunda şu özellikler olmalıdır (URL-12, 2022):

- Doğrusal Olmama: Bir evrişimli sinir ağında kullanılacak olan aktivasyon fonksiyonunun öncelikle doğrusal olmaması gerekmektedir. Doğrusal olmayan fonksiyonlar baz alınarak oluşturulmuş bir ağ, dünya gerçekliğine daha iyi yaklaşacak ve öğrenme işlemini daha iyi yapacaktır.
- Türevlenebilirlik: Aktivasyon fonksiyonu aynı zamanda türevlenebilir bir fonksiyon olmalıdır. Doğrusal fonksiyonların türevleri sabittir ve bu durumda algoritma, veriden daha iyi gösterimler elde edeceği öğrenme işlemini gerçekleştiremez. Sabit sonuçlu türev, algoritmanın nöronlar arası ilişki kurulmasını engelleyecek ve çok katmanda bütünleşik çalışma gerçekleştiremeyecektir.
- Aralık: Daha etkili bir model eğitimi için fonksiyonun belli aralıklarda olması istenir.
- Monotonluk: Aktivasyon fonksiyonlarının maksimum ve minimum noktalarının olması beklenmektedir.
- Orijine Göre Yakınsaklık: Aktivasyon fonksiyonları orijine yakınsama yaptıklarında modelin başlangıç ağırlığı da minimum değerler olarak atanacaktır.

Çeşitli aktivasyon fonksiyonları vardır: sigmoid, basamak, hiperbolik tanjant, Softmax ve ReLU bunlardan bazılarıdır (Şekil 23). Her bir fonksiyonun birbirine göre üstün yönleri vardır ve çeşitli ağ tasarımlarında belirli amaçlara yönelik kullanılmaktadırlar.

ReLU	Sızıntı ReLU	ELU
$g(z) = \max(0, z)$	$g(z) = \max(\epsilon z, z)$ ile $\epsilon \ll 1$	$g(z) = \max(\alpha(e^z - 1), z)$ ile $\alpha \ll 1$
		
• Doğrusal olmama karmaşıklığı biyolojik olarak yorumlanabilir	• Negatif değerler için ölen ReLU sorununu giderir	• Her yerde türevlenebilir

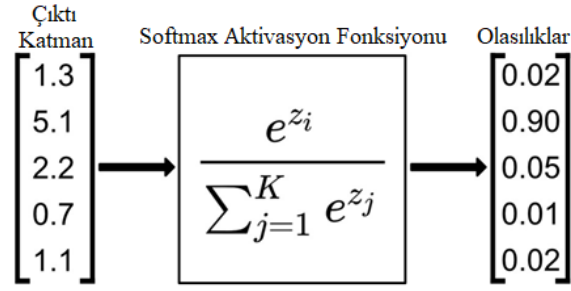
Şekil 23. Çeşitli aktivasyon fonksiyonları (URL-13, 2022)

Özellikle evrişimli sinir ağlarında ReLU aktivasyon fonksiyonu sıkça kullanılmaktadır (Agarap, 2018). Gizli katmanlarda en çok tercih edilen fonksiyonlardan biri olan ReLU fonksiyonu, doğrusal olmayan bir fonksiyondur. Fonksiyonun değeri 0'dan başlar, sonsuza uzanır. Bu da girdi değerinin negatif bir değer ise 0'a; pozitif bir değer ise de kendine eşitleneceği anlamına gelmektedir. Bu durum ağın çalışma performansını artırır ve böylece ağ daha hızlı çalışır. Dolayısıyla hesaplama yükü de azalacaktır. Çok katmanlı ağlar açısından düşünüldüğünde hesaplama yükünün azalması, ReLU fonksiyonunun aktivasyon fonksiyonu olarak seçilmesinde büyük rol oynamaktadır.

ReLU fonksiyonu negatif değerleri sıfıra indirirken aslında bu bölgenin türevlerini de sıfıra indirgemiş olacaktır. Dolayısı ile öğrenme bu bölge için gerçekleşmeyecektir. Zamanla ReLU fonksiyonu geliştirilerek Sızıntı ReLU fonksiyonu geliştirilmiştir. Bu fonksiyon negatif bölgede sıfır yerine sıfıra çok yakın değerler atar ve başlangıcı eksi sonsuzdur. Bu sayede öğrenme negatif bölgede de gerçekleşmiş olacaktır.

Sofmax fonksiyonu ise derin öğrenme ağlarında genellikle çıkış katmanında tercih edilen, girdinin hangi sınıfa ait olduğu olasılığını veren bir fonksiyondur (Hu vd., 2018). Olasılık problemlerinde sıklıkla tercih edilmektedir.

Bir evrişimli sinir ağında belli bir sınıf için nesnelerin o sınıfa ait olma olasılıkları tespit edilmek isteniyorsa, çıktının negatif olması beklenmeyen bir durum olacaktır. Çünkü olasılık 0 ile 1 arasındadır. Bu sorun ağın ilgili katmanına Softmax fonksiyonunun entegrasyonu ile aşılabilmektedir (Şekil 24).



Şekil 24. Softmax fonksiyonunun katmanlara entegrasyonu (URL-14, 2022)

Ortaklama katmanı herhangi bir öğrenme süreci içermeyen, verinin kolay işlenebilirliğini sağlayan bir süreçtir. Bu süreç ile nitelik haritalarının boyutlarının küçültülmesi amaçlanmaktadır. Günümüzde sınırsız artan veri boyutları ve sınırlı donanımına sahip bilgisayarlar, sinir ağlarında verinin özelliklerini saklı tutarak verinin boyutlarının azaltılması gereksinimine neden olmuştur. Böylece milyonlarca parametre oluşmayacak ve ağda aşırı uydurma gerçekleşmeyecektir.

Maksimum ortaklama ve ortalama ortaklama, sıkça kullanılan ortaklama türlerindendir (Şekil 25).

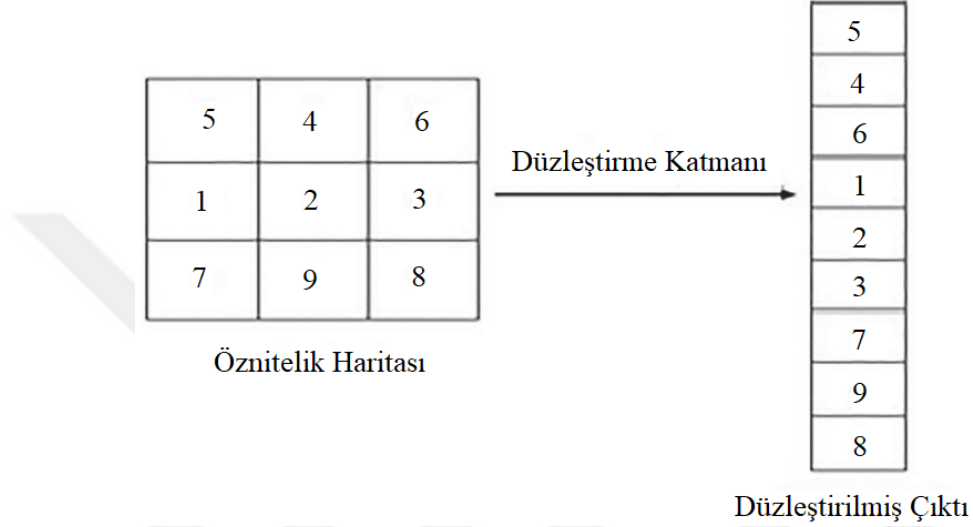
Maksimum ortaklamada matris değerleri içinden maksimum olanı, ortalama ortaklamada ise matris değerlerinin ortalaması esas alınmaktadır. Bu iki yöntem arasından maksimum ortalama daha çok tercih edilmektedir çünkü girdi matriste var olan değerlerden biri alınır ve veriye ait bilgi değiştirilmeden korunmuş olur.

Tip	Maksimum ortaklama	Ortalama ortaklama
Amaç	Her ortaklama işlemi, geçerli matrisin maksimum değerini seçer	Her ortaklama işlemi, geçerli matrisin değerlerinin ortalaması alır
Görsel Açıklama		
Açıklama	- Algılanan özellikleri korur - En çok kullanılan	- Boyut azaltarak örneklenmiş özellik haritası - LeNet'te kullanılmış

Şekil 25. Ortaklama yöntemleri (URL-15, 2022)

Evrişimli sinir ağında tam bağlantı katmanı, bu katmana kadar çıkarılmış bilgilerin değerlendirildiği kısımdır. Bu bilgiler kullanılarak sınıflandırma işlemi yapılmaktadır. Tam bağlantı katmanları ağın son kısımlarındadır.

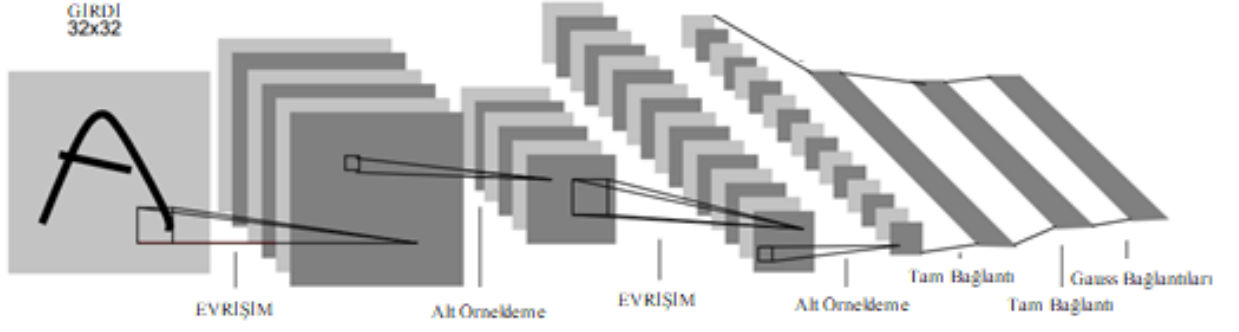
Bu katmana gelen veriler matrisler halindedir. Düzleştirme (flattening) işlemi ile matrisler tam bağlantı katmanına giriş için hazırlanırlar (Şekil 26).



Şekil 26. Ortaklama katmanından çıkan verinin düzleştirilmesi (Suriya vd., 2019)

Çıkış katmanındaki düğüm (node) sayısı sınıf sayısı kadar olmalıdır. Örneğin, rakamların tespiti için bir ağ analizi tasarlanmak isteniyorsa çıkış katmanındaki düğüm sayısı 10 olmalıdır.

Evrişimli sinir ağları ile ilgili çalışmalar 1998 yılında Yann LeCun ve arkadaşları tarafından geliştirilen ve başarılı sonuca erişen bir evrişimli sinir ağı olan LeNet-5 ile hız kazanmıştır (LeCun vd., 1998). Metin tanıma amaçlı tasarlanan ağ, ortaklama katmanında ortalama ortaklama işlemini; aktivasyon fonksiyonu adımıyla ise sigmoid ve hiperbolik tanjant işlemini kullanmaktadır (Şekil 27).



Şekil 27. LeNet-5 mimarı yapısı (LeCun vd., 1998)

2000’li yıllara gelindiğinde bilgisayar sistemlerinin gelişmesi, durağan yapay zekâ döneminin tekrar hareketlenmesine neden olmuştur. ImageNet yarışması milyonlarca farklı görüntü ile yüzlerce farklı nesne kategorisi kullanılarak eğitilen ağların yer aldığı bir yarışmadır. Yarışmada evrişimli sinir ağlarının yüksek başarı göstermesi, bu ağlara olan ilginin ve çalışmaların artmasına neden olmuştur. Özellikle 2012 yılında AlexNet adlı evrişimli sinir ağı modeli, sağladığı yüksek başarı performansı ile dikkatleri üzerine çekmiştir.

AlexNet Alex Krizhevsky, Geoffrey Hinton ve Ilya Sutskever tarafından geliştirilmiştir. Katmanlarının sıralanışı bakımından LeNet’e çok benzemektedir ancak aktivasyon fonksiyonu olarak ReLU fonksiyonunu, ortaklama katmanında ortalama ortaklama değil maksimum ortaklama işlemini kullanmaktadır.

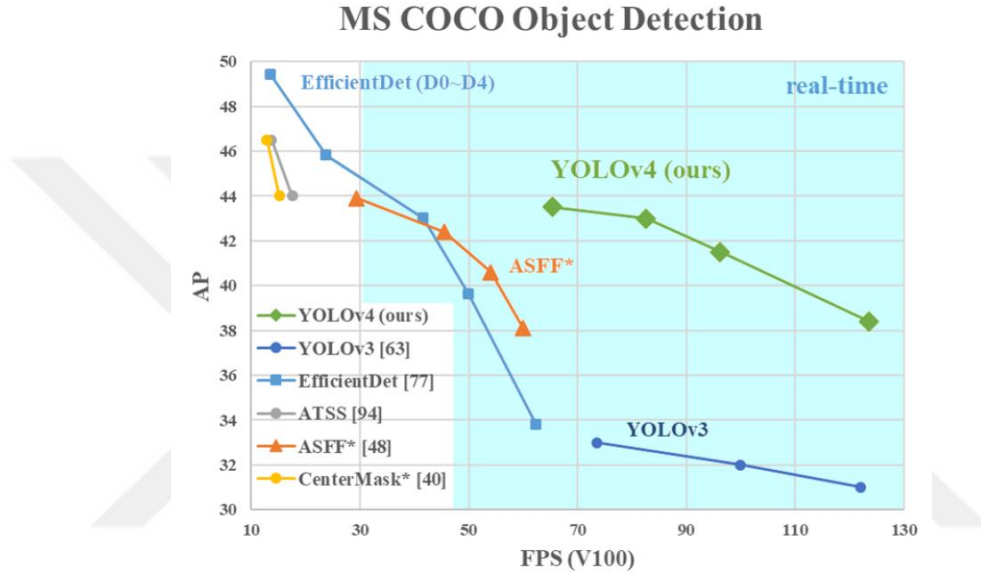
1.5.5. YOLO ve YOLO – V4

YOLO ilk kez 2015 yılında öne sürülmüş bir algoritmadır. Zaman içinde çeşitli versiyonları geliştirilmiş ve sunulmuştur. YOLO algoritması evrişimli sinir ağları temelli bir algoritmadır. Açılımı “You Only Look Once” (Sadece Bir Kez Bak) olan algoritma, nesne tespitini tek seferde ve yüksek hızda gerçekleştirmektedir. Girdi verinin tamamı ağıdan bir seferde geçirilir ve bu da ağı yüksek hızda analiz yapmasına olanak tanır. Bu nedenle YOLO gerçek zamanlı nesne tespitinde ve sürücüsüz araba gibi hızlı analiz ve karar gerektiren uygulamalarda yaygın olarak kullanılmaktadır.

Hızlı bir algoritma olması nedeni ile özellikle bilgisayarlı görü alanındaki problemlerin çözümünde sıkça kullanılmaktadır. YOLO’dan önce RCNN, Fast RCNN ve

Faster RCNN gibi algoritmalar ortaya atılmış olsa da YOLO, hızlı sonuçlar ve yüksek doğruluklar sağlaması nedeni ile ön planda olmuştur.

Çalışma kapsamında tasarlanan algoritmada insan tespiti için YOLO algoritmasının dördüncü versiyonu olan YOLO-V4 (Bochkovskiy vd., 2020) kullanılmıştır. YOLO-V4 algoritması 2020 yılında geliştirilmiş olup kendinden önceki nesne tespit algoritmalarına göre daha hızlı ve yüksek doğrulukta sonuçlar elde ettiği tespit edilmiştir (Şekil 28).



Şekil 28. YOLO-V4 ve diğer nesne dedektörlerinin kıyaslanması (Bochkovskiy vd., 2020)

1.5.5.1. YOLO Algoritması Çalışma Prensibi

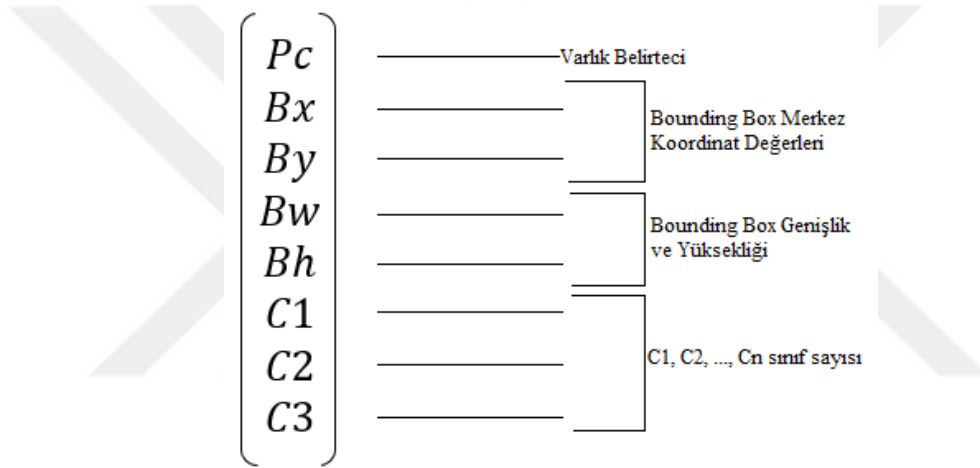
YOLO sadece nesneleri kategorize etmez, aynı zamanda tespitini yaptığı nesnenin konumunu da belirler. YOLO algoritmasına giren bir görüntü, çıkış katmanında artık sınıflandırılmıştır ve nesne konumunu ifade eden “bounding box” adı verilen sınırlayıcı kutu ile temsil edilmiştir.

Sınırlayıcı kutu şu parametrelerden oluşmaktadır:

- Sınırlayıcı kutunun merkez koordinatları
- Kutunun genişliği ve yüksekliği
- Kutu içinde kalan nesnenin sınıfı

Bir YOLO ağı öncelikle eğitime tabii tutulmalıdır. Çeşitli veri setlerinin kullanılabileceği bu süreçte YOLO ağına sınıflandırma problemine konu nesneler tanıtılmaktadır. Bunun için ilgili nesnelerin bulunduğu görüntülerdeki objeler sınıflarına göre sınırlayıcı kutular ile temsil edilerek ağı tanıtılır ve ağ eğitimi tamamlanmış olur. Makinenin girdi görüntüyü anlaması, analiz etmesi ve yorumlaması için verinin vektör dönüşümünün sağlanması gerekmektedir. Çünkü YOLO ağı görüntüleri ancak vektör biçimleri ile algılayabilmektedir.

YOLO ağına giren eğitim verileri ve eğitilmiş bir YOLO ağından çıkan sonuçlar vektörler halindedirler ve bu vektörlerdeki değerlerin temsili Şekil 29’da ifade edilmiştir.



Şekil 29. YOLO ağı vektör veri yapısı

Ağın tespiti yapılmak istenen nesneleri içeren görüntüler ile eğitilme süreci, eğitim aşaması olarak adlandırılmaktadır. Örneğin, “kedi” ve “insan” nesnelerinin sınıflandırılmasına yönelik bir problemde ağ bu iki nesnenin bulunduğu görüntülerle öncelikle bir eğitime tabii tutulur. Eğitim sonunda eğitilmiş ağ artık nesne sınıflandırma problemlerinde kullanıma hazırdır ancak “insan” ve “kedi” dışında herhangi bir nesne sınıfını tanımlayamayacaktır.

Eğitim aşamasında YOLO, görüntüleri vektör formlarında algılayabileceği için bu aşamadan önce eğitim verileri birtakım işlemlerden geçirilmelidir. Bunun için sınıflandırılması istenen nesne türlerine ilişkin çok sayıda görüntü gerekmektedir. Görüntülerdeki nesneler sınırlayıcı kutularla çevrilir ve bu işlem tüm eğitim görüntüleri için ilgili nesnelere uygulanır.

İnsan ve kedilerin sınıflandırılmasına yönelik bir problemde bir kediye, bir insanı ve bu ikisini bir arada içeren 3 farklı görüntü eğitim amaçlı kullanılmak istendiğinde objelerin vektör temsillerinin oluşturulması gerekecektir. Örnek olarak bu 3 eğitim görüntüsünün eğitime nasıl hazırlanacağı gösterilmiştir ancak gerçek bir eğitim sürecinin binlerce hatta on binlerce eğitim verisi gerektirdiği unutulmamalıdır.



Şekil 30. YOLO ağı için “kedi” sınıfı eğitim verilerinin hazırlanması

P_c	_____	1
B_x	_____	110
B_y	_____	95
B_w	_____	120
B_h	_____	100
C_1	_____	1
C_2	_____	0

Şekil 31. YOLO ağı “kedi” sınıfı eğitim verisi vektör yapısı

Şekil 31’de ifade edildiği üzere P_c değeri varlık belirticidir. Eğer sınırlayıcı kutuda herhangi bir nesne tespit edilmiş ise P_c değeri 1, edilememiş ise 0 olacaktır. B_x ve B_y değerleri sınırlayıcı kutunun merkez noktasının (x, y) değerleridir. B_w ve B_h değerleri ise sırası ile sınırlayıcı kutunun genişliği ve yüksekliğidir. Bu değerler piksel cinsindedir. Sınıf belirteci C1 ve C2 olmak üzere iki adettir çünkü eğitim iki sınıf için gerçekleştirilmiştir ve bu değerler sırası ile kedi ve insan sınıflarını temsil etmektedirler. Eğitim görüntüsü üzerinde sınırlayıcı kutuda kedi nesnesi tespit edildiğinden (Şekil 30), kedi sınıfını temsil eden C1 değeri 1 olarak kaydedilmiştir. Diğer nesne sınıfı olan insan objesi ise ilgili görüntüde saptanamadığından C2 değeri matrise 0 olarak girilmiştir.

YOLO ağına eğitim verisi olarak bu sefer yalnızca insan nesnesini içeren bir görüntü tanıtılır ise (Şekil 32) hazırlık sonucu oluşacak vektör veri ve temsili gösterimi Şekil 33’te belirtildiği gibi olacaktır:



Şekil 32. YOLO ağı için “insan” sınıfı eğitim verilerinin hazırlanması

$$\begin{pmatrix} Pc \\ Bx \\ By \\ Bw \\ Bh \\ C1 \\ C2 \end{pmatrix} \begin{matrix} \text{————} 1 \\ \text{————} 100 \\ \text{————} 100 \\ \text{————} 90 \\ \text{————} 180 \\ \text{————} 0 \\ \text{————} 1 \end{matrix}$$

Şekil 33. YOLO ağı “insan” sınıfı eğitim verisi vektör yapısı

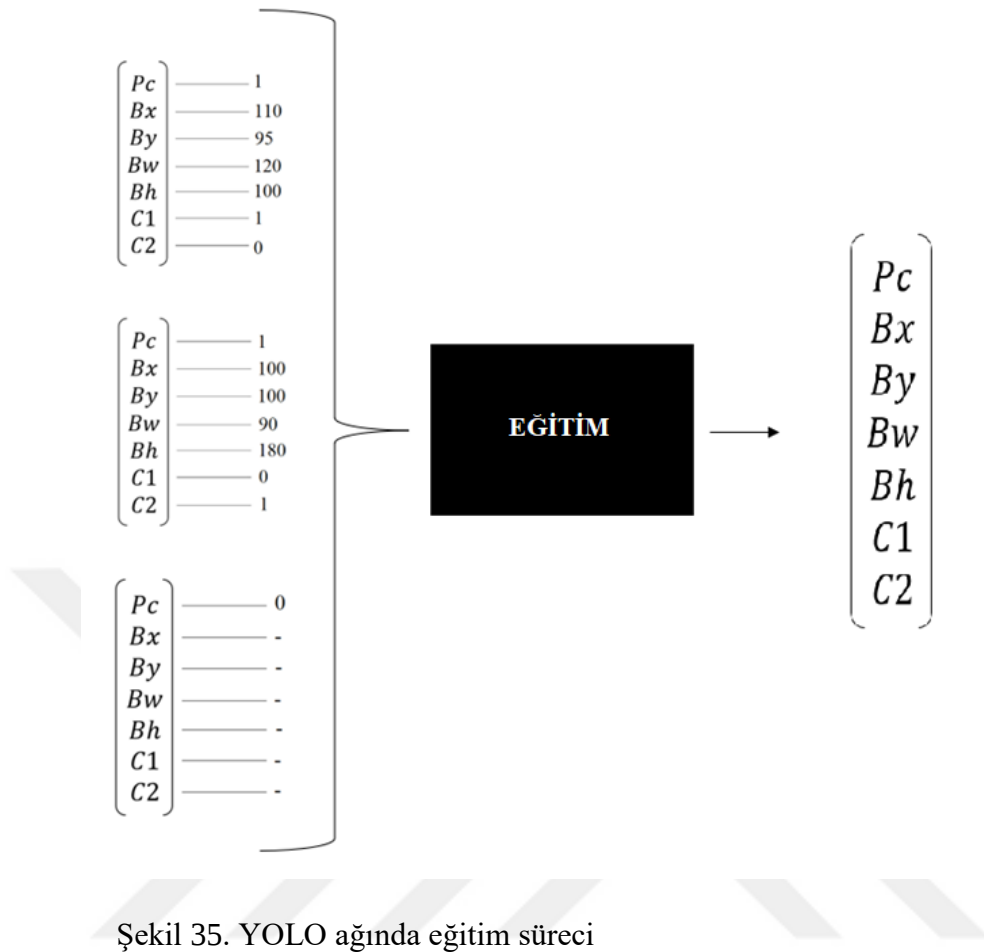
Bir önceki vektörde C1 değeri 1 iken C2 değeri 0'dı. Bu görüntüde ise kedi nesnesi mevcut olmadığından C1 değeri 0 ve insan objesi saptandığından C2 değeri 1 olarak kaydedilecektir (Şekil 33).

Eğer görüntüde hiçbir nesne yok ise sınırlayıcı kutu oluşmayacak, P_c değeri 0 alınarak durumun vektör ifadesi tanımlanmış olacaktır (Şekil 34).

$$\begin{pmatrix} P_c \\ B_x \\ B_y \\ B_w \\ B_h \\ C_1 \\ C_2 \end{pmatrix} \begin{matrix} \text{_____} 0 \\ \text{_____} - \\ \text{_____} - \\ \text{_____} - \\ \text{_____} - \\ \text{_____} - \\ \text{_____} - \end{matrix}$$

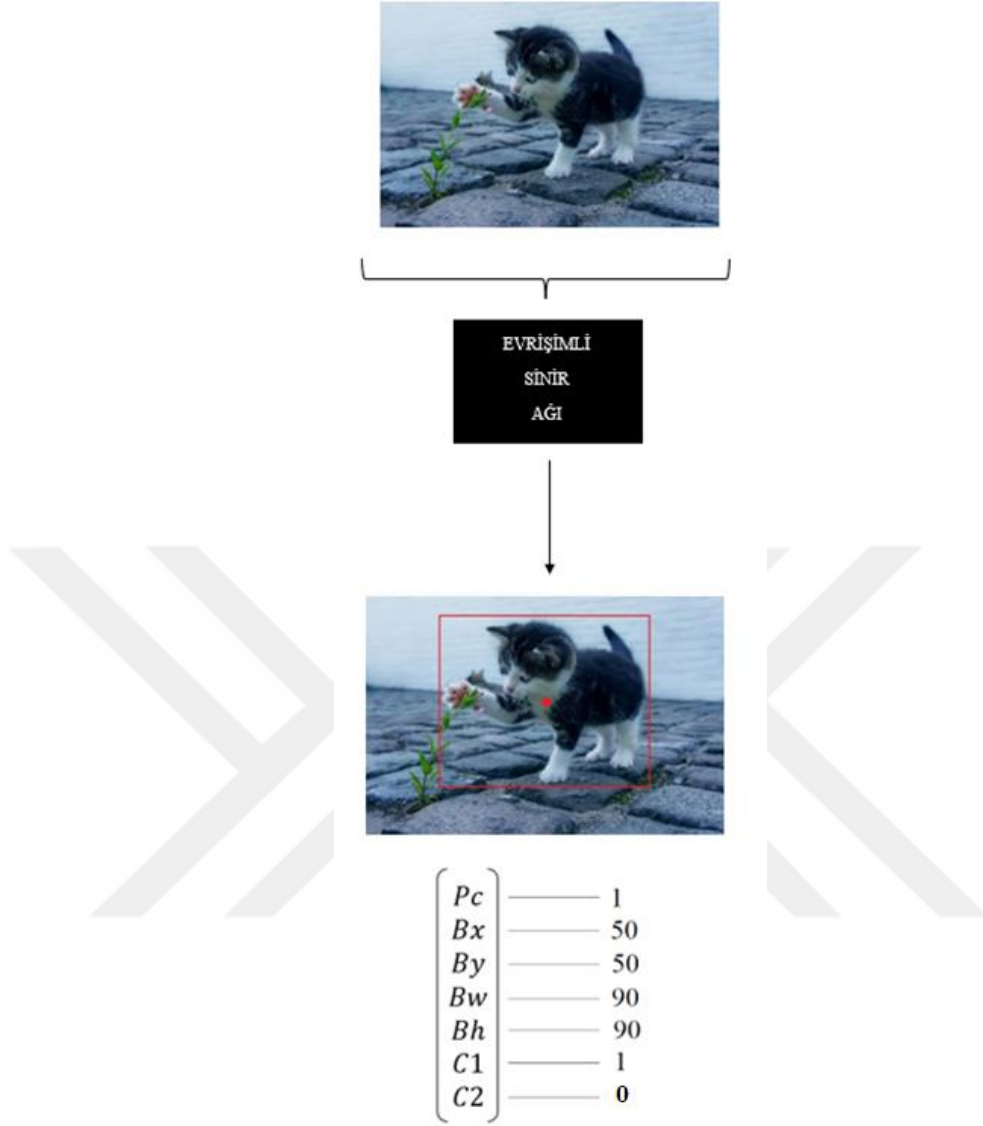
Şekil 34. YOLO ağı eğitim verisi vektör yapısı – nesne tespit edilmeme durumu

Vektör formunda ifade edilen eğitim verileri ile ağı eğitilir (Şekil 35):



Şekil 35. YOLO ağında eğitim süreci

Eğitimi tamamlanmış ağ artık herhangi bir görüntüde eğitimde kendisine tanımlanmış olan sınıfları tanıyacak, sınırlayıcı kutusunu oluşturacak ve nesnenin hangi sınıfa ait olduğu bilgisini kullanıcıya sunacaktır (Şekil 36).

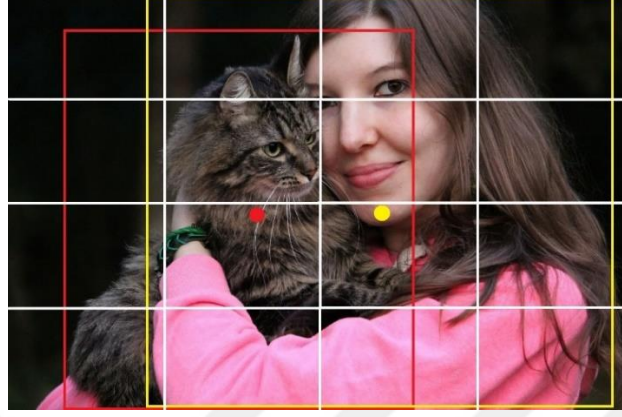


Şekil 36. Eğitilmiş YOLO ağında nesne tespiti ve çıktı vektör yapısı

Örneklerde olduğu gibi bir görüntü her zaman tek bir nesne sınıfına ait objeyi içermez. Bir görüntüde birden fazla nesne sınıfına ait objeler de bulunabilmektedir. Bu durumda eğitilmiş YOLO ağı görüntüdeki objeleri tespit etmek için obje sayısı kadar sınırlayıcı kutu oluşturmalı ve her bir kutunun, dolayısı ile her bir objenin merkez koordinatları ve vektör formu için gerekli bilgileri çıktı katmanında sunmalıdır.

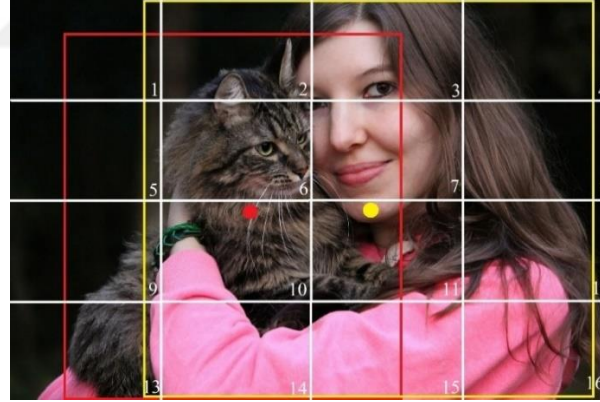
Bu nedenle eğitilmiş bir YOLO algoritması, nesne tespitinin yapılması istenen ve girdi katmanına verilen görüntüyü öncelikle ızgaralara böler. Bu ızgaralar 3'e 3, 4'e 4 veya değişken formlarda olabilirler. Izgaraların sık döşenmiş olması, aynı ızgarada birden fazla nesne için merkez atamasının önüne geçeceğinden tercih edilebilmektedir.

Örnekte insan ve kedi nesnesini içeren görüntü, 4×4'lük ızgaralara bölünerek işleme başlanır (Şekil 37).



Şekil 37. Eğitilmiş bir YOLO ağında nesne tespiti

İfade kolaylığı bakımından ızgaralar manuel numaralandırılmıştır (Şekil 38):



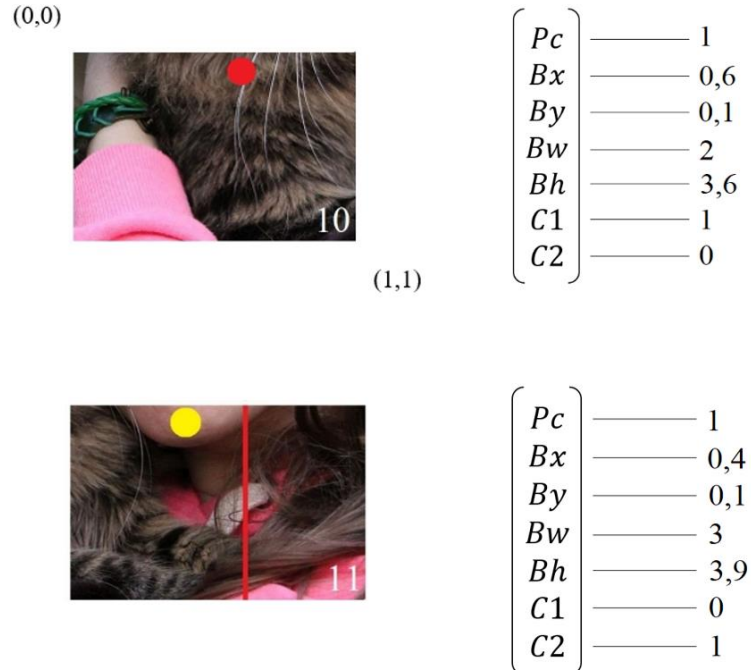
Şekil 38. Eğitilmiş bir YOLO ağında nesne tespiti-numaralandırılmış ızgaralar

5 numaralı ızgarada hiçbir nesne bulunmamaktadır. Bu ızgara için çıktı vektör şu şekilde olacaktır (Şekil 39):

$$\begin{pmatrix} P_c \\ B_x \\ B_y \\ B_w \\ B_h \\ C_1 \\ C_2 \end{pmatrix} \begin{matrix} \text{-----} 0 \\ \text{-----} - \\ \text{-----} - \\ \text{-----} - \\ \text{-----} - \\ \text{-----} - \\ \text{-----} - \end{matrix}$$

Şekil 39. YOLO ağında nesne tespit edilmeme durumunun vektör ifadesi

Kedi nesnesi ise birden fazla ızgarada yer almaktadır. Tespit edilen kedi nesnesinin sınırlayıcı kutusu genişlik ve yükseklik değerleri ile saptanarak merkez koordinatların bulunması amaçlanır. Objenin merkezi hangi ızgarada ise nesne o ızgaraya ait kabul edilir. 10 numaralı ızgaraya bakıldığında kedi nesnesinin merkezinin bu ızgarada yer aldığı, dolayısı ile kedi nesnesinin 10 numaralı ızgaraya ait olduğu söylenilebilir. İnsan olduğu tespit edilen nesnenin merkezi ise 11 numaralı ızgarada yer almaktadır. Dolayısı ile bu nesne 11 numaralı ızgaraya aittir. Bu ızgaralar için vektör ifadeler şu şekilde oluşturulmaktadır (Şekil 40):

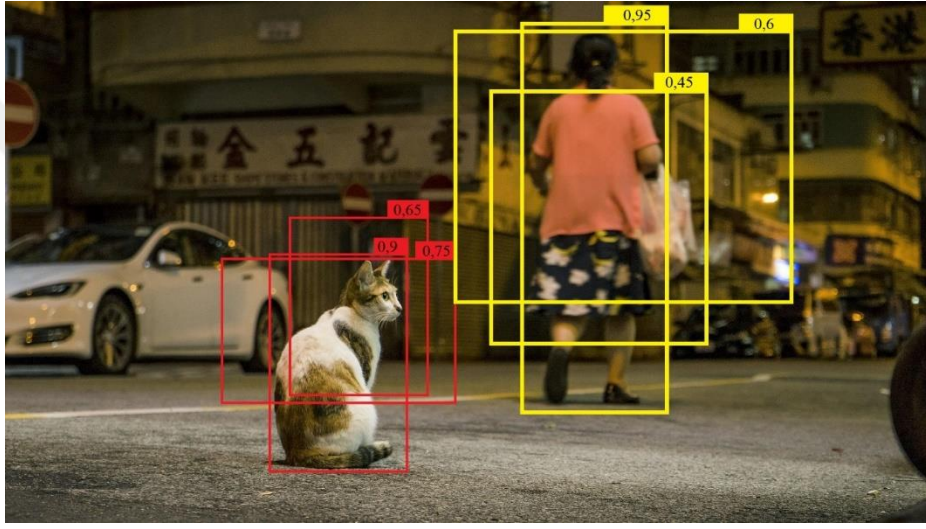


Şekil 40. İki farklı nesnenin YOLO çıktı katmandaki vektör ifadeleri

Gerçekte bir YOLO ağının eğitimi için binlerce hatta on binlerce görüntü gerekmektedir. Eğitilen YOLO ağı artık eğitildiği sınıf türü kadar çeşitli objeleri tespit edebilecek, sınırlayıcı kutularının çizebilecek ve kutuların merkezlerini hesaplayabilecektir. Nesnenin görüntüdeki konumunun yanı sıra ait olduğu sınıfı da çıktı vektörde temsil edebilecektir.

Bir YOLO ağında karşılaşılması muhtemel 2 sorun vardır:

1) Algoritma bir nesne için birden fazla sınırlayıcı kutu oluşturmuş olabilir (Şekil 41).

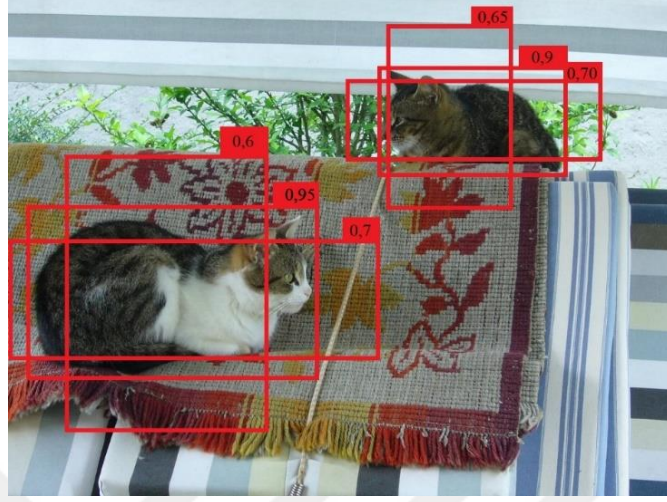


Şekil 41. YOLO - farklı sınıflar için çoklu sınırlayıcı kutu saptanma durumu

Şekil 41’de görüldüğü gibi YOLO, ayrı ayrı nesnelerin her biri için birden fazla sınırlayıcı kutu saptamış olabilir. Algoritma, nesnelerin sınıflarına göre farklı renklerle temsil edildiği sınırlayıcı kutuları görselleştirir. Her bir sınırlayıcı kutunun ilgili nesneyi içeriyor olma olasılığı algoritma tarafından saptanır ve olasılık değeri ile ifade edilir.

Aynı nesne için var olan sınırlayıcı kutular içinde maksimum olasılığa sahip olanı geçerli sınırlayıcı kutu olarak seçmek ve diğerlerini elemek mantıklı gibi görülse de aynı görüntüde o sınıfa ait birden fazla nesne olduğunda bu yöntem işe yaramayacaktır (Şekil 42). Çünkü algoritma görüntünün tamamını analiz edecek ve aynı sınıfa ait tüm sınırlayıcı kutular içinde maksimum olasılığa sahip olan bir kutuyu seçecektir. Bu durumda bir sınırlayıcı kutu oluşacak, diğer nesne için herhangi bir sınırlayıcı kutu tanımlanamamış olacaktır. Şekil 42’ye bakıldığında algoritmanın 0,95 olasılığa sahip kutuyu seçmesi

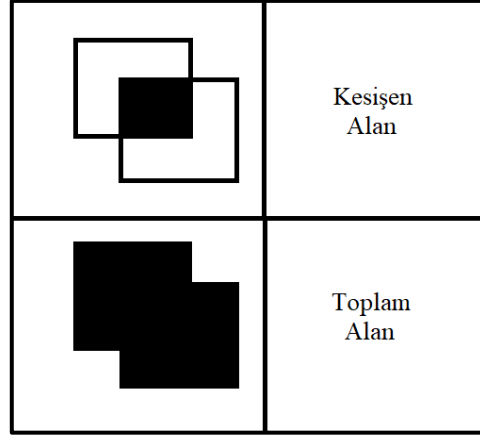
beklenir. Bu da görüntünün sağ kısmında bulunan kedi için herhangi bir sınırlayıcı kutu oluşmayacağı, dolayısı ile tespitin sağlanamayacağı anlamına gelmektedir.



Şekil 42. YOLO - aynı sınıflar için çoklu sınırlayıcı kutu saptanma durumu

Burada sadece aynı nesneyi temsil eden sınırlayıcı kutular tespit edilir ve maksimum olasılığa sahip kutu yalnızca bunların içinden seçilir ise her bir objenin ayrı ayrı tespiti mümkün olacaktır. Bunun için oluşturulan sınırlayıcı kutuların kesişimlerine bakılır (Şekil 43). Aynı nesneye ait sınırlayıcı kutular kesişeceklerdir. Kesişen bu kutuların içinde maksimum olasılığa sahip olan seçilir ise herhangi bir kayıp olmadan görüntüdeki tüm nesnelerin tespiti sağlanmış olacaktır.

$$\text{Kesişen Alan} / \text{Toplam Alan} = \text{IoU (Intersection Over Union)} \quad (13)$$

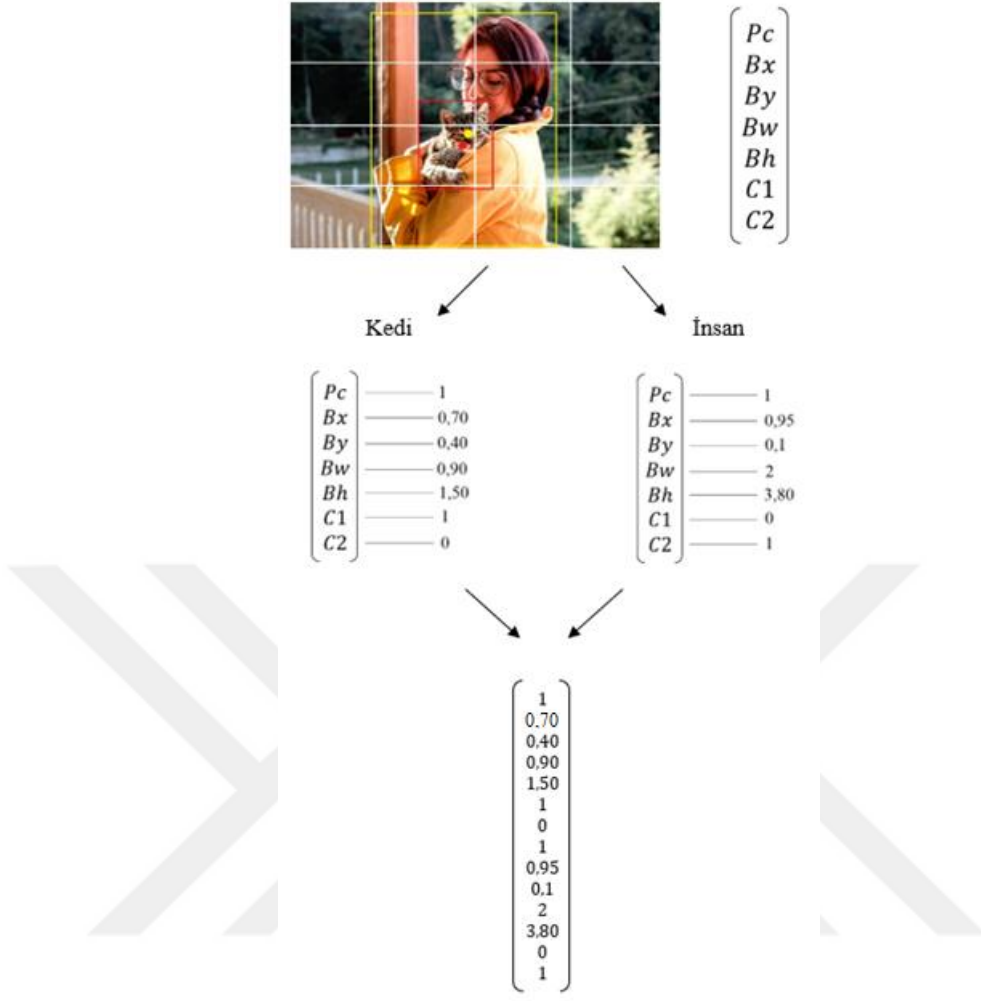


Şekil 43. İki sınırlayıcı kutu için kesişen alan ve toplam alan

“IoU” değeri için bir eşik belirlenir ve bu eşik aşıyor ise kutular örtüşüyor demektir. Örtüşen kutular aynı varlığı temsil eden kutulardır ve artık bunların içinden maksimum olasılığa sahip olan kutu, sınırlayıcı kutu olarak tercih edilebilir. Bu tekniğe “Maksimum Olmayanı Bastırma” (Non-Max Suppression) adı verilmiştir.

2) Bir ızgarada birden fazla nesneye ait merkez olabilir.

Bu problem, görüntüye uygulanacak ızgaranın sıklığı artırılarak aşılabilir ancak böyle bir teknik her zaman yeterli olmayacaktır. Bu durumda ilgili ızgaraya ait vektör verinin boyutları artırılır ve böylece bir ızgaraya ait vektör veri, her iki objeyi de temsil edebilecek hale gelir (Şekil 44).



Şekil 44. Bir vektör veride birden fazla objenin temsili

1.6. Literatür Araştırması

Değişen teknoloji ve ihtiyaçlar, yapay zekâ alanından beklentilerin artmasına neden olmuştur. Yapay zekâ artık hayatın her alanında problemlerin çözümünde karar verici sistemler olarak bizlere destek vermektedirler. COVID-19 pandemisinin hayatımıza girmesi ile yapay zekâyâ entegre uygulamalar geniş bir çalışma alanı oluşturmuştur.

Tez çalışması kapsamında geliştirilen algoritma ve bu çalışmaya paralel geçmiş çalışmalar, literatür araştırması kapsamında irdelenmiştir. Söz konusu inceleme, pandeminin ortaya çıktığı 2019 yılından günümüze kadar olan araştırma sonuçlarını kapsamaktadır. Bu süreçte gerçekleştirilen araştırmalar içinde ön plana çıkan ve klasik yöntemden farklı teknikler ortaya koyan çalışmalar değerlendirilmiş, sonuçları ile ifade edilmiştir.

Bhambani ve diğerleri (2020), YOLO-V3 algoritması tarafından kişilerin gerçek zamanlı tespitini ve sosyal mesafe ihlalleri ile maske kullanım durumunu saptayan bir algoritma geliştirmişlerdir. Tek bir kameradan çekilen görüntüler algoritmaya girdi olarak sunulmaktadır. Bundan dolayı hesaplamalar iki boyutlu düzlemde gerçekleştirilmiş, kişilerin x ve y koordinatları piksel cinsinden elde edilmiştir. İki boyutta olan bu görüşü tek kamera ile üç boyuta çıkarmak için x ve y piksel değerlerinin her biri için bir d değeri hesaplamış ve bir noktaya ait üç boyutlu bilgiyi dolaylı yoldan elde etmişlerdir. Burada d değeri, derinlik bilgisidir ve derinlik hesabında odak uzaklığı, obje yüksekliği ve sensör genişliği ile arasında üçgen benzerliği kullanılarak hesaplanmış bir değerdir. Objeye yüksekliği ya da bir diğer ifade ile kişilerin boyları, yaklaşık olarak 1,60 metre alınarak bu işlem gerçekleştirilmiştir. Kişilerin derinliklerinin saptanmasının ardından kişiler arası sosyal mesafelerin tespiti işlemi gerçekleştirilmiştir. Tasarladıkları algoritmada sosyal mesafeler sezgisel olarak oluşturulmuş, gerçek koordinatlar cinsinden ifadesi gerçekleştirilmemiştir.

Ramadas ve diğerleri (2020), tasarladıkları algoritma ile kişilerin tespitini gerçekleştirerek maske kullanma ve sosyal mesafeyi ihlal durumlarını analiz etmişlerdir. Tasarladıkları sistem bir insansız hava aracı ve bu araca entegre bir kamera sisteminden oluşmaktadır. Araç aynı zamanda sosyal mesafeleri ihlal eden kişileri uyaran, maske dağıtan ve durumu ilgili polis noktasına bildiren yetkinliklere sahiptir. Kişilerin tespitinde YOLO-V3 analizini kullanan algoritma, sosyal mesafeler için üç boyutlu gerçek koordinatları kullanmamaktadır.

Ahmed ve diğerleri (2021), sosyal mesafelerin izlenmesi için bir sistem geliştirmişlerdir. Geliştirilen sistem kişilerin tespiti için YOLO-V3 algoritmasını kullanmakta, transfer öğrenme gerçekleştirmesi nedeni ile model performansını arttırmaktadır. Derin öğrenme ağlarının tasarımında karşılaşılan problemlerin başında, soruna özgü veri setlerinin temini gelmektedir. Günümüzde büyük veri setleri mevcuttur ve içerdikleri binlerce sınıf için örnekler içermektedirler. Ancak bazı spesifik problemlerin çözümü için belirli sınıflara ait örnekleri içermeyebilirler. Transfer öğrenme ile yüksek doğrulukta veri kümelerinden bilgi transferi sağlanır ve model doğruluğu artar.

Araştırmacılar, geliştirdikleri algoritmaya YOLO-V3 ve transfer öğrenme yaklaşımlarını birlikte entegre etmişlerdir. Transfer öğrenme yaklaşımı ile modelde hız yakalayan araştırmacılar, bir başka tez daha ortaya atmışlardır. Sosyal mesafenin yüksek doğrulukla elde edilmesi için kameranın ilgili sahneyi kuşbakışı perspektifle izlemesi gerektiğini öne sürmüşlerdir. Sahnenin yan veya önden gözlemlenmesi durumunda kişiler

kameraya göre birbirlerinin önüne geçerek diğer kişiyi kameranın gözlemlemesini engelleyebilir. Sahnenin ön veya yandan alınan görüntüsü yerine kuşbakışı tepeden alınan görüntüsü ile bu sorunun çözümlenmesi hedeflenmiştir. Ancak yine bu çalışmada da koordinatlar görüntüler üzerinden piksel birimli iki boyutlu ölçülmüş, kişiler arası mesafeler yaklaşık bir şekilde hesaplanmıştır. Ayrıca kalibrasyonu yapılmamış kameralarda bozulma değerleri çerçeve için gözetilemeyeceğinden, ilgili sahnede iki piksel arası gerçek mesafe, görüntünün her yeri için eşit olmayacaktır.

Punn ve diğerleri (2020), geliştirdikleri uygulama ile sosyal mesafeleri gerçek zamanlı ölçmüşler, YOLO-V3'e Deep SORT'u entegre ederek yüksek FPS değeri elde etmişlerdir. Klasik bir evrişimli sinir ağında vektörler son katmanda bir sınıfla ilişkilendirilerek o sınıfa atanırlar. Deep SORT algoritmalarında ise bir nesnenin vektör ifadesi diğer nesnenin vektör ifadesi ile ilişkilendirilir. Algoritma genelde tespiti sağlanan hareketli nesnelerin sonraki hareketini tahmin etmek için Kalman filtrelerini kullanır. Böylece nesneleri birbirleri ile ilişkilendirmede ayırt edici özellikleri saptanır ve bu durum da yüksek hızda analizi sağlamaktadır. Yüksek hızda analiz ve yüksek FPS değeri, özellikle gerçek zamanlı uygulamalarda önem arz etmektedir ve araştırmacılar algoritmaları ile yüksek hızda analizi gerçekleştirmeyi başarmışlardır. Sosyal mesafenin tespitinde ise ikili vektörler kullanılmış; üç boyutlu gerçek koordinatlar saptanmamıştır.

Rahim ve diğerleri (2021), sosyal mesafelerin tespiti için RGB kameralar yerine ToF ("time of flight" – uçuş süresi) kameraları kullanarak analizleri gerçekleştirmişlerdir. Bu kameralar nesnelere enerji yollar ve nesnelere çarpan enerji geri yansıyarak kaynağına ulaşır. Böylece nesnelerin kameraya göre konumları belirlenmiş olacaktır. Yöntem, düşük ışıktaki sosyal mesafelerin izlenebilirliğini sağlaması açısından diğer çalışmalardan ayrılmaktadır ancak iki kısıt vardır. Birincisi, bu yöntemin yalnızca sabit kameralarda işe yarayacağıdır. Gezici kameralarda böyle bir sistem, eş zamanlı olarak koordinat hesaplayamayacaktır. İkincisi, kamera sisteminin yerleştirileceği ortama 2 geçici hedef yerleştirilmelidir. Bu hedefler koordinatı bilinen noktalardır ve görüntüye ait piksel birimi cinsinden ifade edilen uzunlukların gerçek koordinatlar cinsinden ifadesi bu noktalarla yapılmaktadır.

Dandıl ve diğerleri (2019), stereo kamera sistemi ile yaptıkları çalışmada sahnedeki kişilerin yüzleri arası mesafeleri hesaplayan bir algoritma geliştirmişlerdir. Stereo eşleştirme işlemi sonucu aykırılık haritası saptanan sahnenin üç boyutlu koordinat bilgisini hesaplamışlardır. Kişilerin yüz tespitini MATLAB ortamında "facedetector" uygulaması ile tespit etmişlerdir. Algoritma bire bir sosyal mesafelerin tespiti amacıyla oluşturulmamıştır

ancak bu amaçla da kullanımı sağlanabilmektedir. Ancak bu durumda kişilerin tespitleri yüzlerin saptanması ile gerçekleştiğinden, kameraya yüzü dönük olmayan kişileri algoritma algılayamayacak ve bu kişileri sosyal mesafe ihlali hesabına dahil etmeyecektir.

Pandit ve diğerleri (2020) tarafından geliştirilen algoritma, sahneye ait YOLO-V3 ile tespit edilen kişiler arası sosyal mesafeleri analiz etmektedir. Bu yöntem için kamera sistemi sabit pozisyonda konumlandırılmalıdır. Çünkü algoritma, sahnede gerçek koordinatlarının bilindiği 8 referans noktasını içermekte ve herhangi iki kişi arası sosyal mesafeleri buna göre hesaplamaktadır.

Tez çalışması kapsamında geliştirilen algoritma, literatürde var olan diğer algoritma ve uygulamalardan; sosyal mesafeleri üç boyutlu gerçek koordinatları kullanarak elde etmesi ve yüksek doğrulukta mesafe hesabı yapması nedeni ile ayrılmaktadır. Ayrıca geliştirilen algoritma, sahneden bağımsızdır ve bu yönüyle herhangi bir referans noktasına ihtiyaç duymamaktadır. Bunun yanı sıra, geliştirilen algoritma için birtakım kısıtlar da mevcuttur. Geliştirilen algoritma ile literatürde örneği verilen çalışmaların kapsamlı karşılaştırılması, “3. BULGULAR VE TARTIŞMA” başlığı altında açıklanmıştır.

2. YAPILAN ÇALIŞMALAR

Yapılan çalışma kapsamında MATLAB ortamında geliştirilen algoritma, stereo kamera sistemiyle görüntülenen bir sahnede YOLO-V4 algoritması tarafından tespiti sağlanan insanlar arası sosyal mesafelerin gerçek zamanlı ve üç boyutlu obje koordinatları ile tespitini sağlamaktadır. Çalışma kapsamında iki adet aynı özellikli bilgisayar kamerası ile bir stereo kamera sistemi oluşturulmuş, bu sistemin getirdiği derinlik algısı sayesinde algoritma mesafeleri obje koordinatları cinsinden hesaplamıştır.

Geliştirilen algoritmanın tasarım aşamaları, çalışma prensipleri ve parametreleri; başlıklar altında süreçler halinde sıralı bölümler ile açıklanmıştır.

2.1. Veri Toplama ve Kullanılan Donanım

Algoritmanın tasarımı aşamasında öncelikle algoritmaya girdi veriyi sağlayacak bir veri toplama sistemine ihtiyaç duyulmaktadır. Söz konusu veri toplama sistemi, iki bilgisayar kamerası ile oluşturulmuş bir stereo kamera sistemidir. Stereo kameralar bilgisayarlı görü uygulamalarında sıklıkla kullanılmakta, sağladıkları üç boyutlu görüş ve gerçek koordinat değerleri ile yüksek doğrulukta analiz imkânı sunmaktadırlar.

Bir stereo kamera sistemi iki ayrı mercek sisteminden oluşmaktadır. Bu amaçla çeşitli firmalarca tasarlanmış ikili kamera sistemleri mevcuttur. Çalışmada kullanılan kamera sistemi ise aynı özellikli iki bilgisayar kamerasının entegresi ile oluşturulmuş bir sistemdir (Şekil 45).



Şekil 45. Uygulama kapsamında oluşturulan stereo kamera sistemi

Derin öğrenme tabanlı uygulamalarda kullanılan donanım çalışma performansını etkileyen en önemli etmendir. Günümüzde büyük veri yapıları yüksek donanımlı bilgisayarlarda kolaylıkla işlenebilmekte ve analiz edilmektedir.

Çalışma kapsamına kullanılan donanım bilgileri ve kamera özellikleri Tablo 2’de verilmiştir:

Tablo 2. Uygulama kapsamında kullanılan donanım bilgileri ve kamera özellikleri

Donanım Bilgileri	
Marka-Model	MSI GP73 Leopard 8RE-601TR
İşlemci	Intel Core i7-8750H CPU 2,20 – 4,10 GHz
İşletim Sistemi	Windows 10 Home
Grafik	6GB GDDR5 ile NVIDIA GeForce® GTX 1060
RAM	16,0 GB
Sistem Türü	64 bit işletim sistemi, x64 tabanlı işlemci
Kamera Bilgileri	
Marka-Model	LOGITECH C270
Ürün Tipi	Web Kamerası
Video Çözünürlüğü	1280 × 960

Tasarlanan stereo kamera sistemi, kamera parametrelerinin hesaplanması ve obje noktalarının uzaydaki geometrik ilişkilerinin belirlenmesi amacıyla kalibre edilmiş; kalibrasyonda yine bu kamera sistemi ile çekilen 1280×960 çözünürlüğündeki 17 çift kalibrasyon görüntüsünden yararlanılmıştır.

Algoritmaya girdi olarak sunulmak üzere, sol ve sağ kameralar ile 1280×960 çözünürlüğünde, 37 saniye uzunluğunda video kaydı gerçekleştirilmiştir. Söz konusu videoda sosyal bir alanda yayaların hareketlerini içeren kesitler mevcuttur. Algoritmadan kişileri tespit etmesi, aralarındaki mesafeleri üç boyutlu gerçek koordinatlar cinsinden hesaplaması ve sosyal mesafe kuralını ihlal edenleri görsel olarak kullanıcıya sunması beklenmektedir.

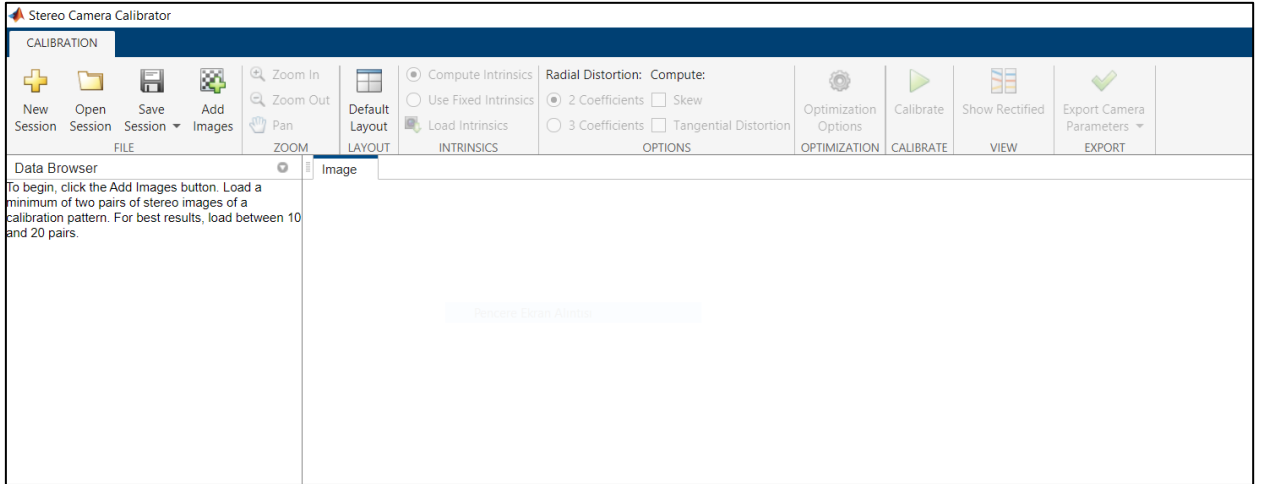
Algoritmanın çalışma performansı ve doğruluğunun denetlenmesi için kayıt altına alınan video sırası ile 1, 1.5 ve 2 metrelik sosyal mesafe eşik değerleri için analiz edilmiş ve bu analizler videonun 20, 22, 25 ve 27. saniyelerindeki sahneler için, üç boyutlu gerçek mesafeler ile gerçekleştirilmiştir.

2.2. Stereo Kamera Kalibrasyonu

Stereo görüş için kullanılan ve binoküler görüşün yapay bir formunu sunan ikili kameralarda kalibrasyon işlemi, derinliğin doğru algılanması ve uzaydaki obje noktaları arası geometrik ilişkilerin tespiti açısından önemlidir. Kalibrasyon işlemi için kalibrasyon nesnesini içeren sahnenin her iki kameradan aynı anda farklı perspektiflerden görüntüleri alınır ve bu görüntüler arası bağıntılar hesaplanarak kamera parametreleri tespit edilir, kalibrasyon işlemi tamamlanır.

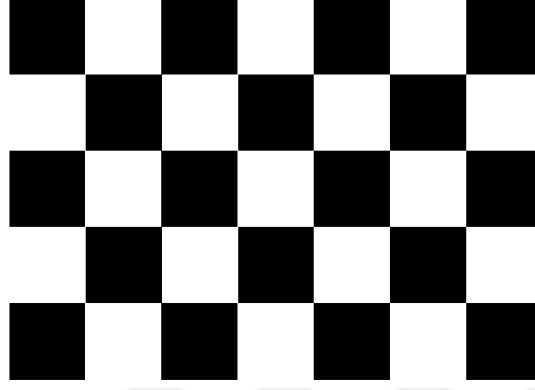
Çeşitli yazılımlarda bu süreci otomatikleştiren çözümler sunulmaktadır. Kalibrasyon sonucu iki kameranın birbirlerine göre konumları ve nesne uzayındaki yerleri geometrik olarak ifade edilecektir.

MATLAB uygulamalarından biri olan “Stereo Camera Calibrator” uygulaması, stereo kameraları kalibre etmek için tasarlanmış bir uygulamadır (Şekil 46).



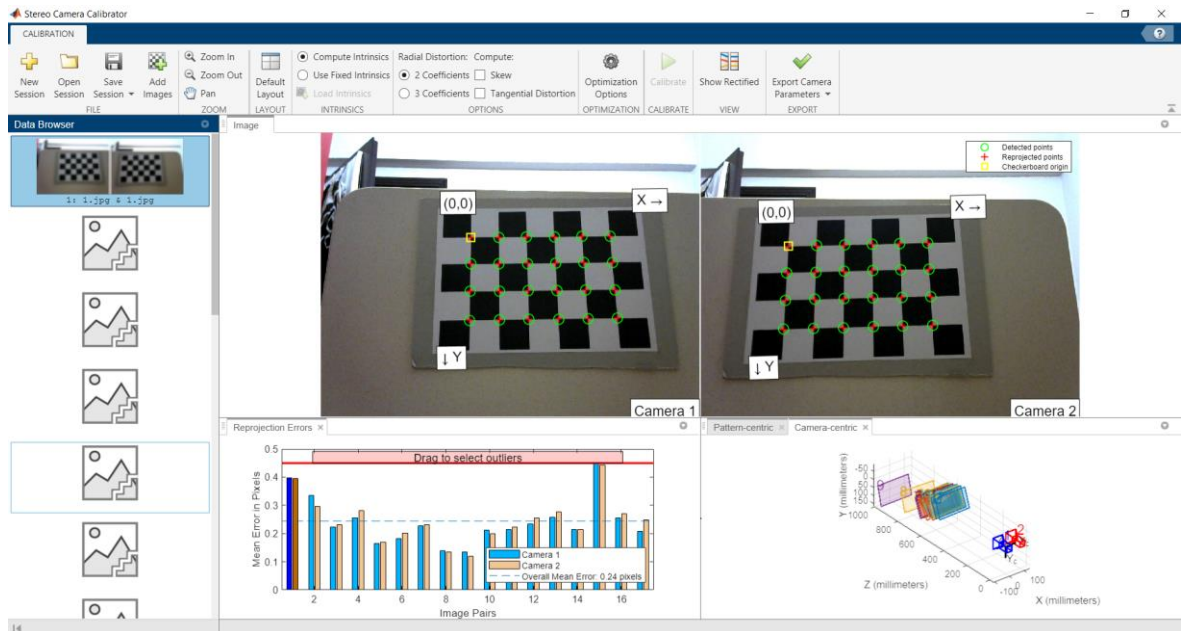
Şekil 46. MATLAB “Stereo Camera Calibrator” uygulaması arayüzü

Kalibrasyon işlemi için bir kalibrasyon desenine ihtiyaç duyulmaktadır. Çalışmada kalibrasyon deseni olarak iki boyutlu ve siyah-beyaz 4 cm kenar uzunluğuna sahip karelerden oluşan bir dama tahtası kullanılmıştır (Şekil 47).



Şekil 47. Çalışmada kullanılan kalibrasyon deseni

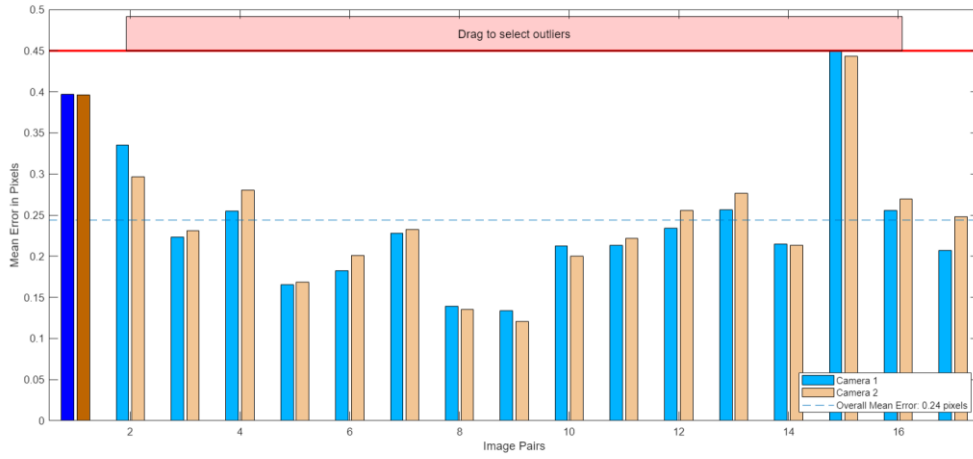
Kalibrasyon deseni tespitinin ardından desene ait görüntü çiftleri sol ve sağ kameralar ile elde edilmiştir. Uygulama için en az 3 çift görüntü gerekmektedir ancak iyi bir kalibrasyon doğruluğu 10-20 çift görüntü ile sağlanmaktadır. Bu kapsamda çalışmada 17 çift görüntü kullanılmıştır ve kalibrasyon uygulamasına girdi olarak eklenmiştir (Şekil 48).



Şekil 48. MATLAB “Stereo Camera Calibrator” uygulaması ile kalibrasyon

Kalibrasyon sonucunda stereo kamera parametreleri matrisi ve yeniden yansıtma hatalarını içeren grafik elde edilmiştir. Grafik üzerinden genel ortalama hata değeri, her bir görüntünün hata miktarı ve hata miktarı eşik değeri aşan görüntü çiftleri tespit edilebilmektedir.

Kalibrasyon işlemi sonucu genel ortalama hata miktarı 0,24 piksel olarak saptanmıştır. Görüntü çiftlerinin her biri için hata eşik değeri 0,50 pikseldir (Şekil 49). Herhangi bir kalibrasyon nesnesi görüntüsü bu eşik değeri aşmadığından rapor yüksek doğrulukta kabul edilmiş ve stereo kamera parametreleri geliştirilen algoritmaya girdi olarak eklenmiştir.



Şekil 49. Kalibrasyon sonucu elde edilen yeniden yansıtma hataları

Kalibrasyon işlemi ile oluşturulan ve algoritmaya eklenen stereo kamera parametreleri ile ilgili bilgiler Şekil 50’de ifade edilmiştir:

stereoParams	
1x1 stereoParameters	
Property	Value
CameraParameters1	1x1 cameraParameters
CameraParameters2	1x1 cameraParameters
RotationOfCamera2	[0.9998,-0.0137,-0.0139;0.0134,0.99...
TranslationOfCamera2	[-90.8035,1.0203,-1.2546]
FundamentalMatrix	[-1.5896e-08,6.1926e-07,4.4719e-0...
EssentialMatrix	[-0.0314,1.2304,1.0489;-2.5129,-2.1...
MeanReprojectionError	0.2440
NumPatterns	17
WorldPoints	24x2 double
WorldUnits	'millimeters'

Şekil 50. Kamera kalibrasyonu çıktıları

Bu çıktıların ifade ettiği bilgiler şunlardır:

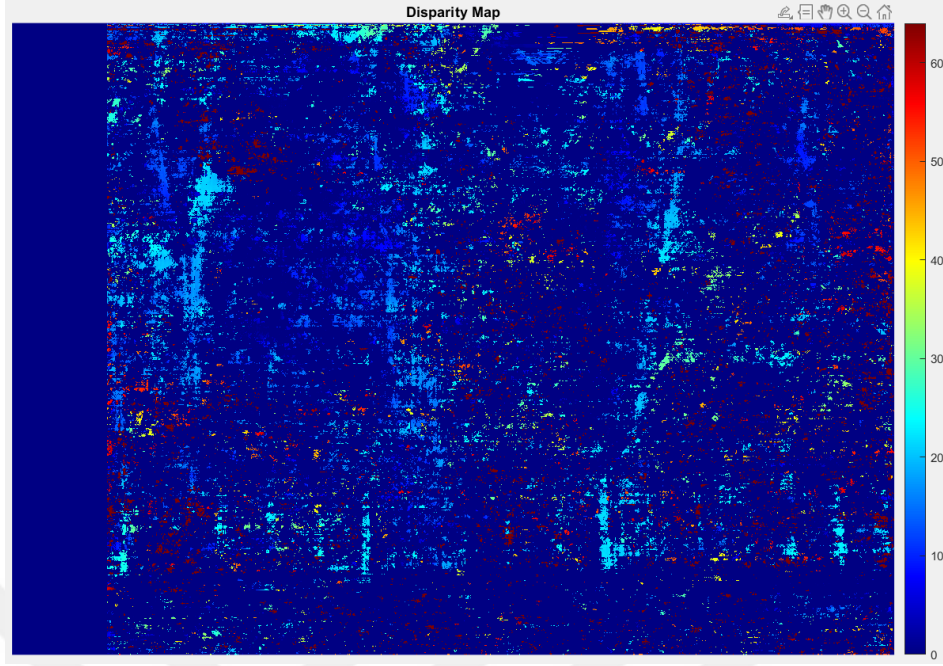
- cameraParameters1, cameraParameters2: Kameraların iç ve dış yöneltme parametreleri ile lens distorsiyon bilgileri
- rotationOfCamera2, translationOfCamera2: Kamera 2'nin Kamera 1'e göre dönüklük ve ötelemeleri
- FundamentalMatrix, EssentialMatrix: Temel ve esas matrisler
- meanReprojectionError: Algılanan noktalar ve yeniden yansıtılan noktalar arası hata miktarı. Bu miktar, mesafelerin öklid cinsinden bir ifadesidir.
- NumPatterns: Stereo kameraların kalibrasyonu için kullanılan desenlerin sayısı
- WorldPoints: Kalibrasyon deseni üzerinde tespiti sağlanan anahtar noktaların koordinatları
- WorldUnits: Koordinat ölçü birimleri

2.3. Üç Boyutlu Koordinat Tespiti

Tasarlanan algoritmaya girdi olarak verilen iki veri vardır. Birinci veri, stereo kamera parametrelerini içeren veridir. İkinci veri ise kalibre edilmiş sol ve sağ kameralar tarafından sahnenin görüntülenmesi ile elde edilen iki video görüntü verisidir. Algoritma, bu girdi verileri kullanarak sosyal mesafe ihlallerini tespit etmektedir ve bu kapsamda öncelikle mesafe değerlerini üç boyutlu gerçek koordinatlar cinsinden hesaplama işlemini gerçekleştirmektedir.

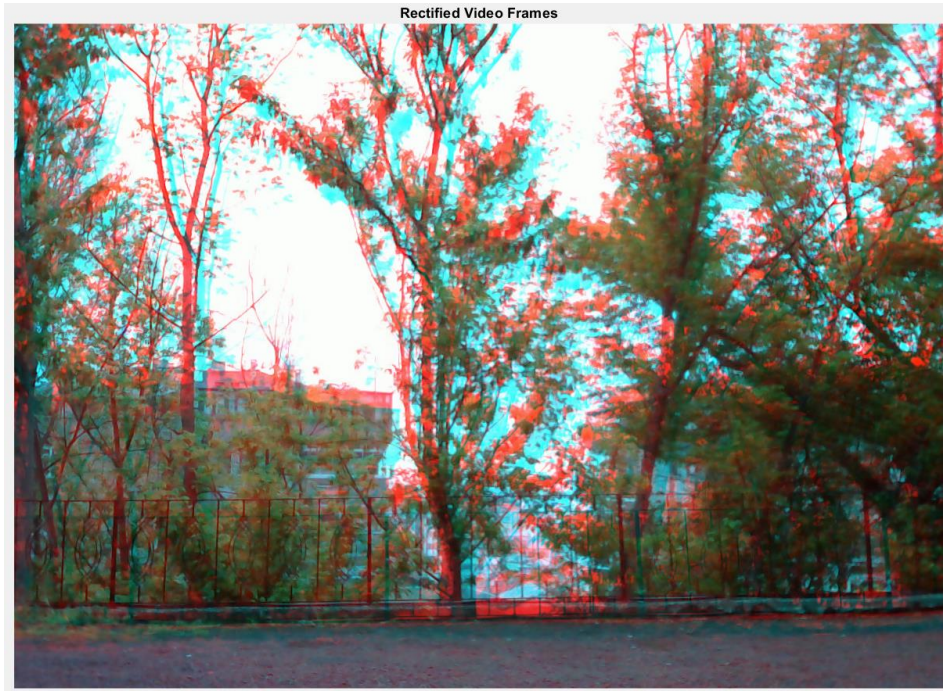
İlk olarak sol ve sağ kameradan gelen görüntüler rektifiye edilmiştir. Böylece farklı perspektiflerden ayrı kameralarla aynı sahneler için oluşturulmuş görüntülerin ortak bir düzlemde ifadesi sağlanmıştır. Rektifikasyon işlemi sonucu iki görüntüdeki ortak noktalar hizalanmış olacak, stereo görüntüler eşleştirilerek aykırılık haritalarının tespiti sağlanacaktır. Aykırılık haritası, yarı global eşleme yöntemi ile oluşturulmuştur. Bu yöntemde amaç, tek boyuttaki işlemlerin art arda gerçekleştirilerek iki boyutlu optimizasyonun sağlanmasıdır.

Aykırılık haritası, sol ve sağ görüntülerdeki eşlenik noktalar arası mesafeler hesaplanarak oluşturulmuştur (Şekil 51).



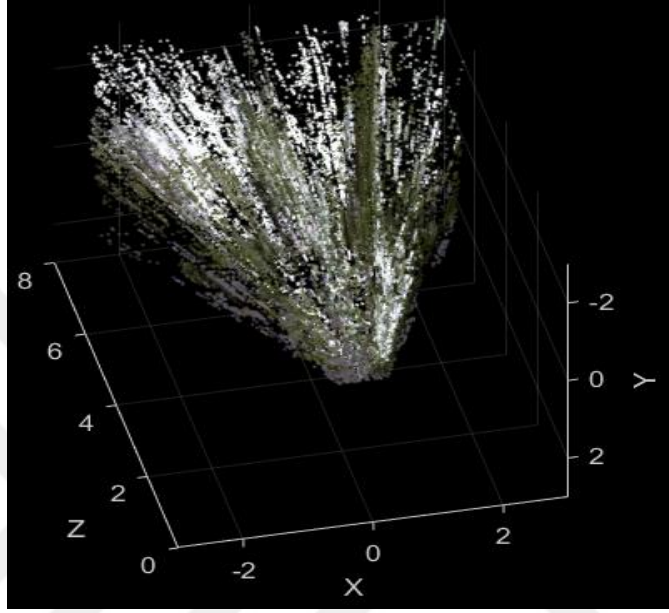
Şekil 51. Video görüntüye ait bir sahnenin aykırılık haritası

Algoritma, rektifikasyon işlemini gerçekleştirir ve rektifiye edilmiş bir sahnenin stereo görüntü çiftine ait kırmızı-camgöbeği anaglif görüntüsünü çıktı olarak sunar. Bu görüntü, bir stereo gözlük ile üç boyutlu olarak izlenebilmektedir (Şekil 52).



Şekil 52. Video görüntüye ait rektifiye edilmiş bir sahnenin anaglif görüntüsü

Algoritma aykırılık haritaları ve projeksiyon matrisi ile piksellerin üç boyutlu koordinatlarını hesaplar ve nokta bulutunu oluşturur (Şekil 53). Böylece video görüntüdeki her bir pikselin üç boyutlu gerçek koordinatları saptanmış olacak, bu koordinatlar ile hesaplanan öklid uzaklıkları sosyal mesafe ihlallerinin gözlemlenmesinde kullanılacaktır.



Şekil 53. Algoritma tarafından analiz süresince ilk sahne için oluşturulmuş nokta bulutu

2.4. YOLO-V4 ile Kişi/Kişilerin Tespiti

Sosyal mesafelerin tespiti için öncelikle video görüntüdeki kişilerin tespiti sağlanmalıdır. Bu amaçla algoritmaya bir dedektör tanımlanması gerekmektedir. Algoritmanın bu aşamasında dedektör olarak YOLO-V4 tercih edilmiştir. Bunun için MATLAB ortamında bulunan “*yolov4ObjectDetector*” adlı fonksiyondan yararlanılmıştır. Fonksiyon ile önceden eğitilmiş bir veri seti üzerinde çalışılabilmekte veya önceden eğitilmiş bu veri setinde birtakım değişiklikler yapılarak tespiti istenen spesifik sınıflar analiz edilebilmektedir. Fonksiyon, COCO veri kümesinde eğitilmiş verileri içermektedir ve çalışma kapsamında bu veri seti ile önceden eğitilmiş veriler kullanılmıştır.

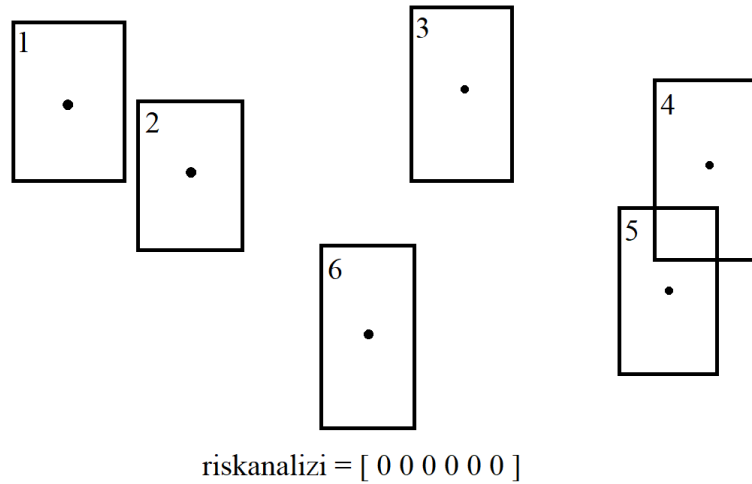
Girdi video görüntüler üzerinden eş zamanlı olarak YOLO-V4 tarafından insan sınıfına ait objeler ve bu objelere ait sınırlayıcı kutular gerçek zamanlı tespit edilmekte ve

görselleştirilmektedir. Algoritma sınırlayıcı kutuların merkezlerini bulmakta ve önceden hesaplanan üç boyutlu koordinatlar arasından bu kutuların merkezlerine denk gelen koordinatları belirleyerek ilgili kutuya atamasını yapmaktadır.

2.5. Gerçek Zamanlı Sosyal Mesafe Analizi

Algoritma tarafından her bir merkezin koordinatları arası gerçek mesafeler öklid uzaklığı cinsinden hesaplanır ve sosyal mesafe için DSÖ tarafından minimum değer olarak kabul edilen 1 metre mesafe eşiğini ihlal eden kişiler tespit edilerek “tehlikede” sınıfına atanırlar, kırmızı sınırlayıcı kutu ile temsil edilirler. Herhangi bir mesafe ihlali saptanmamış kişilerin temsili ise “güvende” sınıfına atanarak yeşil sınırlayıcı kutular ile sağlanmaktadır. Tespit edilen kişilerin herhangi bir sosyal mesafe ihlali yapıp yapmadığının analizi için “riskanalizi” adlı bir matris oluşturulmuş ve algoritmaya eklenmiştir. “riskanalizi” matrisinin oluşturulmasında izlenen yol ve kurallar bütünü şu şekilde izah edilebilir:

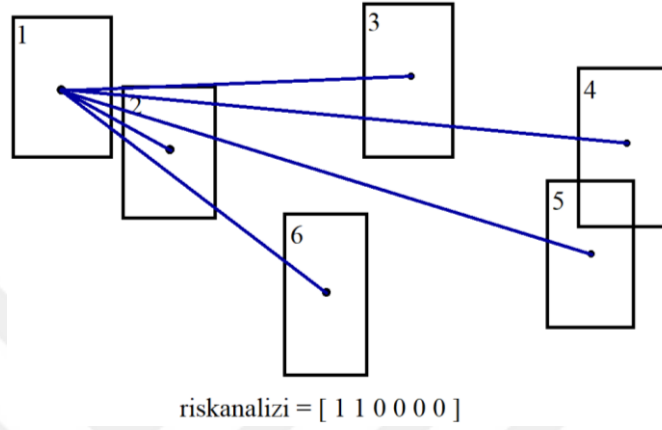
“riskanalizi” matrisi $1 \times n$ boyutunda bir matristir. n , sınırlayıcı kutuların sayısı ya da diğer bir ifade ile YOLO-V4 tarafından ilgili sahne için tespit edilmiş kişi sayısıdır. Algoritma bu boyutlarda boş bir matris tanımlayarak işleme başlar (Şekil 54).



Şekil 54. Algoritma tarafından oluşturulan “riskanalizi” matrisi

Şekil 54’te görüldüğü gibi algoritma ilgili sahne için 6 kişi tespit etmiş ve sınırlayıcı kutularını oluşturmuş, 1×6 boyutunda boş bir “riskanalizi” matrisini tanımlamıştır.

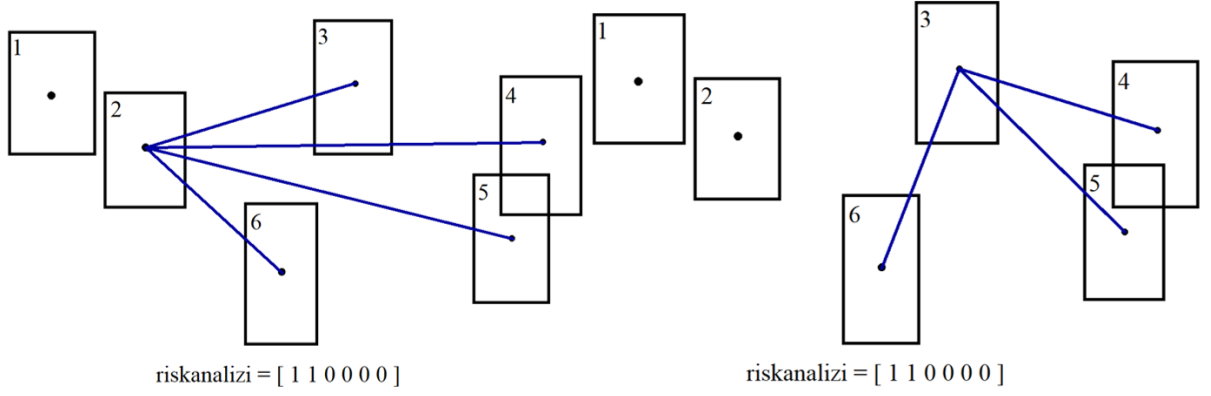
Algoritma bir sonraki adımda her bir kişinin diğer kişilere olan mesafelerini saptar ve bu mesafe 1 metre olan sosyal mesafeyi aşar ise “riskanalizi” matrisinde ilgili kişilerin tanımlı olduğu sütunlara 1 puan atar. Mesafeler, kutu merkezlerinin üç boyutlu koordinatları arası öklid uzaklığı ile hesaplanmıştır.



Şekil 55. 1 numaralı kişinin diğer kişilere olan mesafeleri analizi

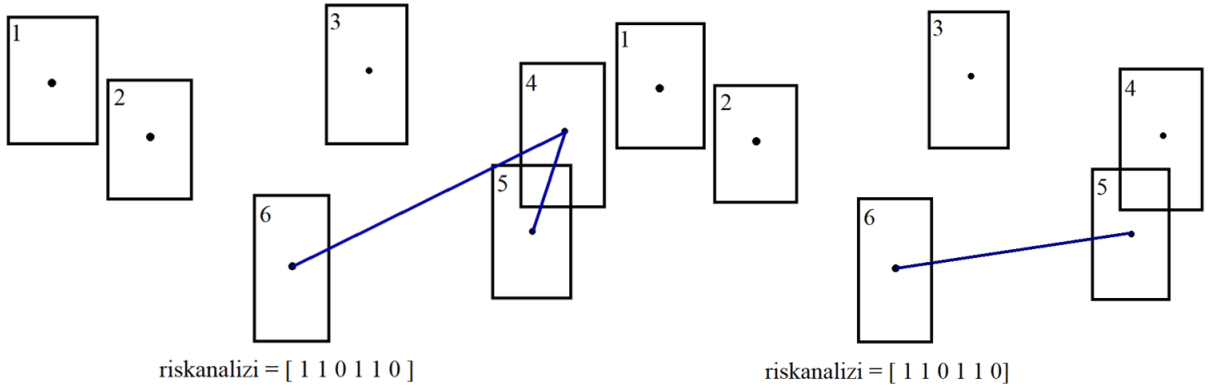
Şekil 55’te görüldüğü gibi algoritma, 1 numaralı kişinin diğer kişilere olan mesafelerini hesaplamış ve 1 ile 2 numaralı kişilerin sosyal mesafeyi ihlal ettiğini tespit ettiğinden matriste ilgili sütunlara 1’er puan atmıştır.

Aynı işlem bu sefer 2 numaralı kişi için gerçekleştirilecek ve sırası ile 3, 4, 5 ve 6 numaralı kişilerle olan mesafeler değerlendirilecektir. Algoritma bir önceki adımda 1 ve 2 numaralı kişiler için mesafe tanımlamasını yapmış olduğundan tekrar bu kişileri için herhangi bir analiz gerçekleştirmeyecektir. Bu sayede işlem yükünün hafifletilmesi hedeflenmiştir. 3 numaralı kişi için de benzer işlem adımları uygulandığında herhangi bir ihlal tespit edilemediğinden matris değişmeden kalacaktır (Şekil 56).



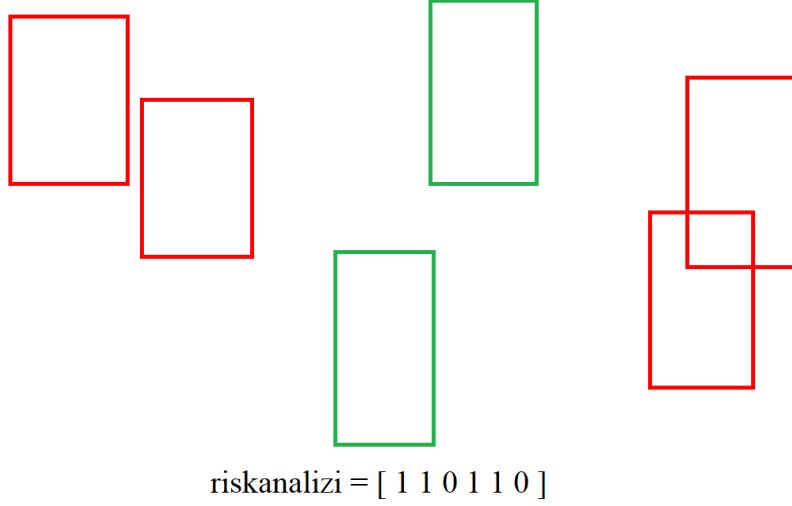
Şekil 56. 2 ve 3 numaralı kişilerin diğer kişilere olan mesafeleri analizi

Algoritma 4 numaralı kişi için analiz yaptığında 5 numaralı kişi ile sosyal mesafenin ihlal edildiğini tespit edecek ve 4 ile 5 numaralı kişiler için matristeki ilgili sütunlara 1'er puan atayacaktır (Şekil 57). Analiz 5 ve 6 numaralı kişiler için de gerçekleştirilerek “riskanalizi” matrisi tamamlanır ve sonuçların görselleştirilmesi aşamasına geçilir.



Şekil 57. 4 ve 5 numaralı kişilerin diğer kişilere olan mesafeleri analizi

Oluşturulan “riskanalizi” matrisinin değerlendirilmesi sonucu sosyal mesafe ihlali yapmayan kişiler “güvende” sınıfına atanarak yeşil, yapanlar ise “tehlikede” sınıfına atanarak kırmızı renk ile temsil edilerek durum görselleştirilir ve kullanıcı tarafından kolay yorumlanabilir hale getirilir (Şekil 58).



Şekil 58. Oluşturulan “riskanalizi” matrisinin sosyal mesafe ihlallerine göre temsili

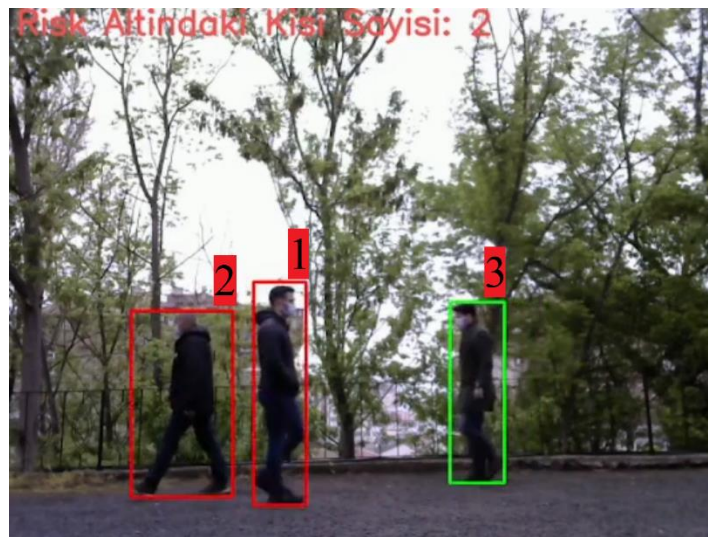
İlgili örnek, algoritma için tasarlanmış “riskanalizi” matrisinin çalışma mantığını daha anlaşılır kılmak için görselleştirilmiş bir temsildir. Video boyunca “riskanalizi” matrisi devamlı hesaplanarak güncellenir ve bu hesap videodaki her bir kare için tek tek yapılır. Örneğin 60 FPS bir kamera ile çekilmiş video için bu analiz saniyede 60 kez gerçekleştirilecek, her saniye 60 ayrı “riskanalizi” matrisi hesaplanacak ve mesafeler anlık olarak üç boyutlu gerçek koordinatlar cinsinden tespit edilecektir. Örneği verilen temsil yalnızca bir sahnedeki analizi temsil etmekte ve bu analiz video boyunca defalarca kez tekrarlanmaktadır.

3. BULGULAR VE TARTIŞMA

Çalışma kapsamında kişiler arası sosyal mesafeleri üç boyutlu gerçek koordinatlar ile hesaplayarak sunan bir algoritma tasarlanmıştır. Algoritma, stereo kamera sisteminin sağladığı derinlik algısı sayesinde sosyal mesafe ihlallerini yüksek doğrulukla ve gerçek zamanlı olarak üç boyutlu tahmin edebilmektedir.

Geliştirilen algoritmanın test edilmesi amacıyla stereo kamera sistemini oluşturan sol ve sağ kameralar ile aynı anda 37 saniye uzunluğunda ve 1280×960 çözünürlüğünde iki video görüntü kaydedilmiştir. Algoritma, analizlerin sonucunu yine bir video görüntü olarak kullanıcıya sunmaktadır. Söz konusu çıktı video görüntü, girdi video görüntüde tespiti yapılan kişilerin sınırlayıcı kutularla temsil edildiği bir görseldir. Çıktı videodaki kişiler, video süresince herhangi bir anda sosyal mesafe kuralını ihlal ederler ise kırmızı, bir ihlalde bulunmazlar ise yeşil sınırlayıcı kutularla temsil edilirler. Ayrıca algoritma, video süresince risk altındaki kişi sayısını kullanıcıyla görsel olarak paylaşmaktadır.

Algoritma girdi video görüntüde toplam 3 kişi tespit etmiş ve bu kişileri ihlal durumlarına göre farklı renkler ile temsil ettiği sınırlayıcı kutularla tespit ederek çıktı video görüntüde kullanıcıya sunmuştur. Çıktı video görüntü 37 saniye uzunluğunda, 1280×960 çözünürlüğünde bir görüntüdür (Şekil 59).



Şekil 59. 1 metre sosyal mesafe uzunluğu baz alınarak gerçekleştirilen analiz sonucu çıktı video görüntünün 25. saniye anındaki sahnesi

Sonuç görüntüsünde algoritma tarafından tespit edilen 1, 2 ve 3 numaralı kişiler arası mesafelerin 20, 22, 25 ve 27. saniye anlarındaki değerleri Tablo3'te belirtilmiştir.

Tablo 3. Geliştirilen algoritmanın belirli sahneler için tespit ettiği kişiler arası mesafeler

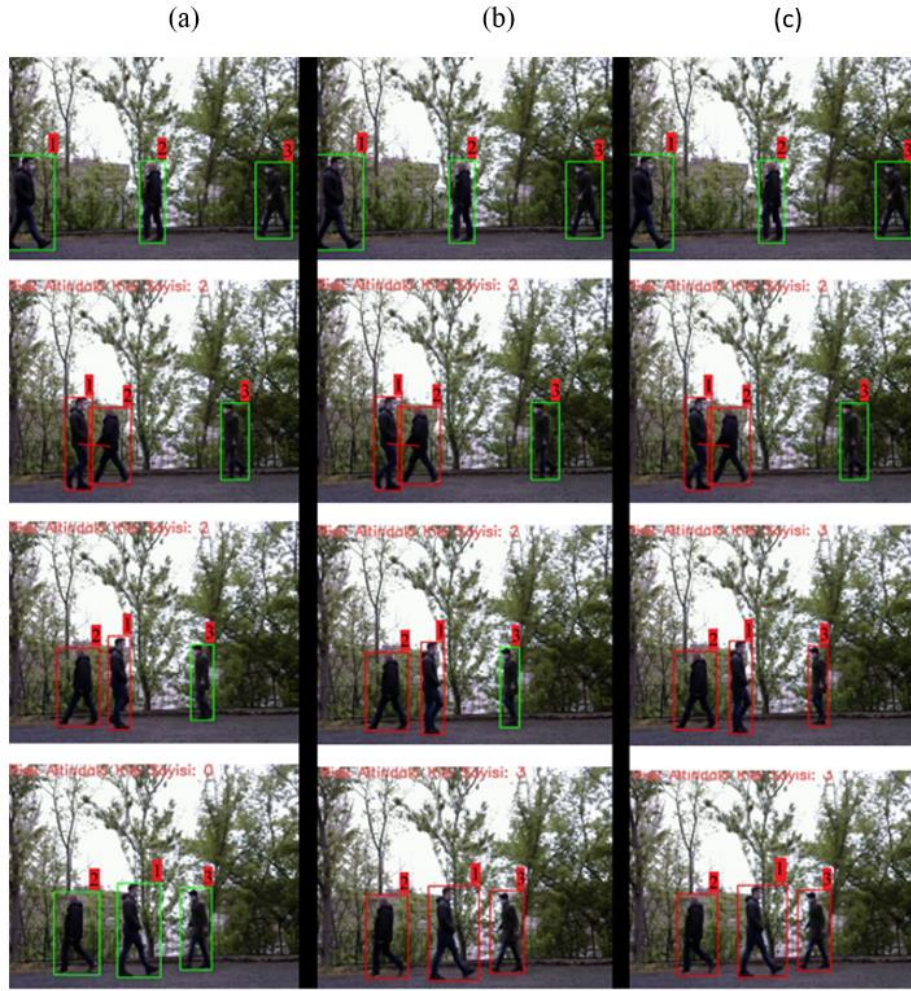
Video Sahne Anı (saniye)	Mesafesi Ölçülen Kişiler	Mesafeler (metre) (3B)
20	1 — 2	2,53
	1 — 3	5,05
	2 — 3	2,22
22	1 — 2	0,81
	1 — 3	4,02
	2 — 3	2,12
25	1 — 2	0,93
	1 — 3	1,95
	2 — 3	2,20
27	1 — 2	1,47
	1 — 3	1,42
	2 — 3	2,15

Örneği verilen analiz, 1 metrelik sosyal mesafe değeri baz alınarak gerçekleştirilen bir analizdir. Algoritmanın değişen sosyal mesafe değerleri karşısında farklı sonuçlar üretmesi beklenmektedir. Örneğin, algoritma tarafından tespit edilen A ve B kişileri arası mesafenin 1,60 metre olduğu varsayılın. Algoritma, sosyal mesafe değerini 1 metre olarak baz aldığı bir analizde bu kişileri yeşil; sosyal mesafe değerini 2 metre olarak baz aldığı bir analizde ise kırmızı sınırlayıcı kutular ile temsil etmelidir.

Algoritmanın test edilmesi amacıyla sol ve sağ kamera ile çekilen görüntüler geliştirilen algoritmaya girdi olarak sunulmuş ve algoritmanın sırası ile 1, 1.5 ve 2 metre sosyal mesafe uzunluklarını baz alarak 3 ayrı analiz gerçekleştirmesi sağlanmıştır.

Üç ayrı sosyal mesafe uzunluğunun baz alındığı üç ayrı analiz sonucu algoritma, üç çıktı video görüntüyü kullanıcıya sunmuştur. Çıktı video görüntülerin her biri 37 saniye uzunluğundadır ve analizlerin değerlendirilmesi amacıyla her bir videonun 20, 22, 25 ve 27. saniye anlarındaki sahneler baz alınarak sonuçlar değerlendirilmiştir.

Algoritma tarafından tespit edilen kişiler arası üç boyutlu gerçek uzunluklar Tablo 3’te belirtilmiştir. Bu anlarda algoritma tarafından çıktı video olarak sunulan sosyal mesafe ihlallerinin temsil edildiği görsel Şekil 60’ta sunulmuştur ve (a), 1 metre sosyal mesafe uzunluğu baz alınarak gerçekleştirilen analiz sonucu çıktının yukarıdan aşağı sırası ile 20, 22, 25 ve 27. saniyelerine ait ekran görüntüsü; (b), 1,5 metre sosyal mesafe uzunluğu baz alınarak gerçekleştirilen analiz sonucu çıktının yukarıdan aşağı sırası ile 20, 22, 25 ve 27. saniyelerine ait ekran görüntüsü; (c), 2 metre sosyal mesafe uzunluğu baz alınarak gerçekleştirilen analiz sonucu çıktının yukarıdan aşağı sırası ile 20, 22, 25 ve 27. saniyelerine ait ekran görüntüsü verilerini içermektedir.



Şekil 60. Değişen sosyal mesafe eşik değerleri için gerçekleştirilen analiz sonuçları

Değişen sosyal mesafe eşik değerlerinin değerlendirildiği 3 farklı analiz sonucu; 1, 2 ve 3 numaralı kişilerin 20, 22, 25 ve 27. saniyeler anlarında temsil edildikleri sınırlayıcı kutuların eşik değerlere göre renkleri, Tablo 4’te verilmiştir.

Tablo 4. Değişen sosyal mesafe eşikleri için kişilerin temsili

Video Sahne Anı (saniye)	Sosyal Mesafe Eşiği (metre)	Kişiler	Sınırlayıcı Kutu Rengi
20	1	1	
		2	
		3	
	1,5	1	
		2	
		3	
	2	1	
		2	
		3	
22	1	1	
		2	
		3	
	1,5	1	
		2	
		3	
	2	1	
		2	
		3	
25	1	1	
		2	
		3	
	1,5	1	
		2	
		3	
	2	1	
		2	
		3	
27	1	1	
		2	
		3	
	1,5	1	
		2	
		3	
	2	1	
		2	
		3	

Geliştirilen algoritmanın değişen sosyal mesafe eşik değerleri karşısında doğru sonuçlar ürettiği tespit edilmiştir. Örneğin algoritma, çıktı görüntünün 27. saniyesinde 1 ve 2 numaralı kişiler arası mesafeyi 1,47 metre; 1 ve 3 numaralı kişiler arası mesafeyi ise 1,42 metre olarak hesaplamıştır (Tablo 3). Şekil 60'ta görüldüğü üzere 3 ayrı analiz ile test edilen algoritmanın 1 metrelik sosyal mesafeyi baz aldığı analizde 27. saniyede bu kişileri yeşil, 1,5 ve 2 metrelik sosyal mesafeleri baz aldığı analizlerde ise bu kişileri kırmızı sınırlayıcı kutu ile temsil ettiği gözlemlenmiştir.

Geliştirilen algoritma, kişiler arası mesafeleri üç boyutlu gerçek koordinatlar ile hesaplamaktadır. Bu yönüyle mesafeleri yüksek doğrulukta hesaplamakta, sosyal mesafeleri gerçek değerleri ile ortaya koymaktadır. Ayrıca algoritmanın başarısı değişen sosyal mesafe eşik değerlerine karşı ürettiği sonuçlar ile değerlendirilmiş ve algoritmanın eşik alınan tüm uzunluklarda doğru sonuçlar ürettiği tespit edilmiştir.

Geliştirilen algoritmanın literatüre katkılarının irdelenmesi bakımından var olan diğer çalışmalar incelenmiş, çalışmaların birbirlerine üstün olan ve olmayan yönleri irdelenmiştir. “1.2. Literatür Araştırmaları” başlığında detayları sunulan çalışmalar ile tez kapsamında tasarlanan algoritmanın kıyaslanması Tablo 5'te temsil edilmiştir.

Tablo 5. Geliştirilen algoritmanın benzer çalışmalar ile kıyaslanması

	Bhambani vd. (2020)	Ramadas vd. (2020)	Ahmed vd. (2021)	Punn vd. (2020)	Rahim vd. (2021)	Dandil vd. (2019)	Pandit vd. (2020)	Geliştirilen Algoritma
Kamera Sistemi	Tek Kamera	Drone + Tek Kamera	Tek Kamera	Tek Kamera	ToF	Stereo Kamera	Tek Kamera	Stereo Kamera
Mesafe Dedektörü	Evet	Evet	Evet	Evet	Evet	Hayır	Evet	Evet
Maske Dedektörü	Evet	Evet	Hayır	Hayır	Hayır	Hayır	Hayır	Hayır
Üç Boyutlu Koordinatlar ile Ölçüm	Hayır	Hayır	Hayır	Hayır	Hayır	Evet	Hayır	Evet
YOLO Versiyon	V3	V3	V3	V3	V4	YOLO desteklenmiyor	V3	V4
Düşük Işıktaki Algılama	Hayır	Hayır	Hayır	Hayır	Evet	Hayır	Hayır	Hayır
Referans Noktasına Gereksinim Duymama ve Sahneden Bağımsızlık	Evet	Evet	Evet	Evet	Hayır	Evet	Hayır	Evet

MATLAB ortamında geliştirilen algoritma, herhangi bir sahnedeki kişilerin tespiti ile sosyal mesafe ihlal durumlarının gerçek zamanlı ve yüksek doğruluklu analizi amacıyla tasarlanmış bir algoritmadır. Algoritma literatürdeki diğer çalışmalardan, mesafeleri üç boyutlu gerçek koordinatları ile sunması bakımından ayrılmaktadır.

Literatürdeki çalışmalar incelendiğinde, sosyal mesafe ihlallerinin üç farklı yöntem ile irdelendiği görülmüştür.

Birinci yöntem, tek kamera ile çekilmiş bir sahnedeki kişilerin tespiti sonrası sosyal mesafelerin piksel cinsinden ifadesinin sağlandığı bir yöntemdir. Bu yöntemde sosyal mesafeler yalnızca iki boyutta hesaplanacak, derinlik algısı göz ardı edilerek piksel bazlı mesafeler sosyal mesafe olarak kabul edilecektir. Ayrıca bir görüntüdeki tüm ardışık pikseller arası mesafelerin görüntünün tamamı için aynı olacağını söylemek doğru bir yaklaşım olmayacaktır. Çünkü her kamera, bozulmaları da içermektedir ve gerçek uzunluklara ulaşmak isteniyor ise bu bozulmalar dikkate alınarak analizler yapılmalıdır.

İkinci yöntem, yine tek kamera ile çekilmiş ve referans noktalarını içeren bir sahnenin görüntülenmesi sonucu kişiler arası sosyal mesafelerin tespitine dayalı bir yöntemdir. Bu yöntem ile mesafeler ilk yönteme kıyasla daha hassas ölçülmüş olsa da referans noktalarının konumlandırılmasına bağlı geometri; üç boyutlu gerçek koordinatların sağladığı doğruluğu sağlayamayacaktır. Ayrıca referans noktalarına olan bağlılık, algoritmanın herhangi bir başka sahne için performans gösteremeyeceğini anlamına gelmektedir.

Üçüncü yöntem ise, kamera sisteminin sabit olduğu varsayımıyla sahne geometrisinin saptanması ve bu geometrinin kurallarına dayalı olarak sosyal mesafelerin bulunmasını sağlayan bir yöntemdir. Bu yöntem, sahneden bağımsız olmayışı ile ikinci yönteme benzemektedir ve mesafe ihlalleri yüksek doğrulukla gözlemlenemeyecektir.

Geliştirilen yöntem, kişiler arası mesafeleri üç boyutlu gerçek koordinatlar cinsinden hesaplaması ile diğer çalışmalardan üstün konumdadır. Bu sayede sosyal mesafeler gerçek değerleri ile ortaya koyulmuştur. Ayrıca, nesne tespitinde kullanılan YOLO algoritmasının dördüncü versiyonu, diğer versiyonlarına ve nesne tespit algoritmalarına göre daha hızlı ve daha yüksek doğruluklu sonuçlar üretmesi bakımından geliştirilen algoritmada tercih edilmiş; bu da çalışmayı literatürdeki diğer çalışmalardan farklı kılmıştır.

4. SONUÇLAR VE ÖNERİLER

Yeryüzünün varoluşundan bu yana insanoğlu pek çok salgınla mücadele etmiştir. Dünya tarihine yön veren İspanyol gribi, SARS hastalığı gibi hastalıklar hava yoluyla bulaşmaları nedeni ile kısa sürede geniş kitlelere yayılmış ve küresel çapta etkiye sebep olmuşlardır. Tarihsel süreçte salgın hastalıklarla gündeme gelen sosyal mesafe kavramı 2019 yılının son aylarında hayatımıza giren COVID-19 ile tekrar gündem olmuştur. İlerleyen dönemlerde artan nüfus ile salgınların devam edeceği ve çok geniş kitleleri etkileyeceği öngörülmektedir.

Hava yolu ile bulaşan salgın hastalıklarda sosyal mesafe, salgına karşı alınan tedbirlerde birinci sırada yerini almıştır. Geçmiş dönemlerde ve günümüzde olduğu gibi gelecekte de sosyal mesafe kavramı salgın hastalıklarla beraber tekrar tekrar gündem olacaktır.

Gelişen teknoloji ile hayatımıza giren yüksek donanımlı bilgisayarlar; yapay zekâ, makine öğrenmesi ve derin öğrenme algoritmalarının gündelik yaşama entegre uygulamalarının artmasına neden olmuştur. Artık günümüzde hemen hemen her problemin çözümü için yapay zekâ tabanlı uygulamalar geliştirilmeye çalışılmakta ve bu projelere çok geniş bütçeler ayrılmaktadır.

Son yıllarda gündemde olan ve kısa sürede küresel salgın haline gelen COVID-19 ile tüm ülkeler sıkı tedbirler almış ve bu tedbirlerin denetimi için kontrol mekanizmaları geliştirmişlerdir. Özellikle salgına karşı en önemli tedbir olan sosyal mesafe kuralının uygulanabilirliği için toplu alanlarda ciddi kısıtlamalara gidilmiş, çeşitli mekanizmalar ile süreçler denetim altına alınmıştır.

Sosyal mesafe analizlerinde kişiler arası mesafelerin yüksek doğrulukta ortaya konması büyük önem arz etmektedir. Mesafelerin gerçek değerleri ile sunulması analizin doğruluğunu arttıracak, sosyal mesafe kural ihlali yapan kişilerin doğru bir şekilde belirlenmesini ve önlemlerin alınmasını sağlayacaktır.

Bu tez çalışması kapsamında sosyal bir alanda yer alan kişilerin sosyal mesafe kural ihlali yapıp yapmadıklarının denetlenmesi için kontrol mekanizması işlevi üstlenecek bir algoritma geliştirilmiştir. Geliştirilen algoritma, kişiler arası sosyal mesafeleri gerçek zamanlı olarak üç boyutlu gerçek koordinatlar cinsinden ifade etmekte, risk analizlerini

gerçekleştirerek kural ihlali yapan kişileri sayıları ile görsel biçimde kullanıcıya sunmaktadır.

Sosyal mesafelerin denetimine yönelik tasarlanan pek çok algoritma mevcuttur. Bu tip analizlerde kişiler arası mesafelerin yüksek doğrulukta ortaya konması büyük önem arz etmektedir. Mesafelerin gerçek değerleri ile sunulması analizin doğruluğunu arttıracak, sosyal mesafe kural ihlali yapan kişilerin doğru bir şekilde belirlenmesini sağlayacaktır.

Geliştirilen algoritmanın literatüre en büyük katkısı, oluşturulan stereo sistemin sağladığı derinlik algısı sayesinde kişiler arası mesafeleri üç boyutlu gerçek koordinatlar cinsinden hesaplamasıdır. Ayrıca algoritma, sahneden bağımsız olması ve herhangi bir referans noktasına gereksinim duymaması ile ön plana çıkmaktadır. Sağlanan derinlik algısı, tespiti sağlanan kişilerin birbirlerine veya kamera perspektiflerine göre konumları fark etmeksizin analizin doğru sonuçlanmasını sağlayacaktır. İlave olarak geliştirilen algoritmada kişilerin tespiti aşamasında YOLO-V4 kullanılmıştır. YOLO-V4, diğer versiyonlarından ve nesne tespit algoritmalarından daha hızlı ve yüksek doğruluklu sonuçlar üretmesi bakımından tercih edilmiş ve algoritmaya eklenmiştir.

Geliştirilen algoritmada birtakım kısıtlar da mevcuttur. Kamera sistemi, iki adet bilgisayar kamerasından oluşturulan bir stereo kamera sistemidir. Dolayısı ile sosyal mesafeleri düşük ışıktaki veya karanlıkta izleyememektedir. İlerleyen süreçlerde bu kamera sistemine entegre bir sistem ile gün boyu mesafelerin tespit edilebilmesine yönelik bir çalışma gerçekleştirilecektir.

Ayrıca, kişilerin tespiti için kullanılan eğitilmiş YOLO modelinin hız kazanması ve yüksek FPS değerlerine ulaşması için kişi tespit mekanizmasının geliştirilmesi çalışmaları yapılacaktır. Bu sayede algoritmanın gerçek zamanlı uygulamalara uyumu arttırılacaktır.

Ramadas ve diğerleri (2020), geliştirdikleri sisteme bir sesli uyarı ve ilgili birimlere bildiri mekanizması eklemişlerdir. Kişiler sosyal mesafe kural ihlali yaptıklarında sesli bir mesaj sistemi ile uyarılmakta ve durum en yakın merkeze bildirilmektedir. Kontrol mekanizmasının yönetilebilirliğini sağlayan böyle bir sistemin tez çalışması kapsamında geliştirilen algoritmaya entegrasyonu yine bir sonraki çalışma için hedef olarak belirlenmiştir.

Kişiler arası sosyal mesafelerin tespitine yönelik literatürdeki araştırmalar; geliştirilen algoritmanın mesafe analizine bakış açısı bakımından farklı bir perspektif sunduğunu gözler önüne sermiştir. Literatürde sosyal mesafeleri YOLO-V4 destekli üç boyutlu gerçek koordinatlar ile hesaplayan başka bir uygulamaya rastlanılmamıştır. Sosyal mesafe analizlerinde mesafelerin gerçek değerleri ile ortaya koyulması analizin doğruluğu açısından

büyük önem arz etmektedir. Algoritmanın bu farklılıklarının yanı sıra bahsedilen birtakım ilave iyileştirmeler ile geliştirilmesi hedeflenmektedir.



5. KAYNAKLAR

- Agarap, A. F., 2018. Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375.
- Ahmed, I., Ahmad, M., Rodrigues, J. J., Jeon, G., ve Din, S. 2021. A deep learning-based social distance monitoring framework for COVID-19. Sustainable Cities and Society, 65, 102571.
- Bhambani, K., Jain, T., ve Sultanpure, K. A. 2020. October. Real-time face mask and social distancing violation detection system using yolo. In 2020 IEEE Bangalore Humanitarian Technology Conference (B-HTC), 1-6.
- Bochkovski, A., Wang, C. Y., ve Liao, H. Y. M., 2020. YoloV4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.
- Bradski, G., Kaehler, A., 2008. Learning OpenCV, O'Reilly Media, United States of America.
- Canchumuni, S. W., Emerick, A. A., ve Pacheco, M. A. C., 2019. Towards a robust parameterization for conditioning facies models using deep variational autoencoders and ensemble smoother, Computers & Geosciences, 128, 87-102.
- Chollet, F. ,2021. Python ile Derin Öğrenme, İkinci Baskı, Buzdağı Yayınevi, Ankara.
- Dandıl, E., & Çevik, K. K. 2019, October. Computer vision based distance measurement system using stereo camera view. In 2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT) (pp. 1-4). IEEE.
- Goodfellow, I., Bengio, Y. ve Courville, A., 2016. Deep Learning. MIT Press.
- Hinton, G. E. ve Salakhutdinov, R. R. 2006. Reducing the dimensionality of data with neural networks. Science, 313, 5786, 504-507.
- Hu, R., Tian, B., Yin, S., ve Wei, S., 2018. Efficient hardware architecture of softmax layer.in deep neural network. In 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), 1-5.
- LeCun, Y., Bottou, L., Bengio, Y., ve Haffner, P.,1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86, 11, 2278-2324.
- Maltarollo, V. G., Honório, K. M., ve da Silva, A. B. F. ,2013. Applications of artificial neural networks in chemical problems. Artificial neural networks-architectures and applications, 203-223.

- McCarthy, J. 1979. Ascribing mental qualities to machines. In: Philosophical perspectives in artificial intelligence, ed. M. Ringle. Atlantic Highlands, N.J.: Humanities Press.
- McCulloch, W. S., ve Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5, 4, 115-133.
- Moisan, L., Moulon, P. ve Monasse, P., 2016. Fundamental matrix of a stereo pair, with a contrario elimination of outliers, Image Processing On Line, 6, 89–113
- Pandit, D., Chougule, S., Fatepurwala, H., Kulkarni, A., Kakade, N., ve Sundge, A. 2020, December. Generalized method to validate social distancing using median angle proximity methodology. In 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), 279-284.
- Punn, N. S., Sonbhadra, S. K., Agarwal, S., ve Rai, G. 2020. Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques. arXiv preprint arXiv:2005.01385.
- Rahim, A., Maqbool, A., ve Rana, T. 2021. Monitoring social distancing under various low light conditions with deep learning and a single motionless time of flight camera. Plos One, 16, 2, e0247440.
- Ramadass, L., Arunachalam, S., ve Sagayasree, Z. 2020. Applying deep learning algorithm to maintain social distance in public place through drone technology. International Journal of Pervasive Computing and Communications, 16, 3, 223-234.
- Ravi, D., Wong, C., Deligianni, F., Berthelot, M., Andreu-Perez, J., Lo, B., ve Yang, G. Z. 2016. Deep learning for health informatics. IEEE journal of biomedical and health informatics, 21, 1, 4-21.
- Roberts, L., 1963. Machine perception of three-dimensional solids (Doctoral dissertation, Massachusetts Institute of Technology)
- Shannon, C. E. 1950. A chess-playing machine. Scientific American, 182, 2, 48-51.
- Sharma, S., Sharma, S., ve Athaiya, A. 2017. Activation functions in neural networks. towards data science, 6, 12, 310-316.
- Suriya, M., Chandran, V., ve Sumithra, M. G., 2019. Enhanced deep convolutional neural network for malarial parasite classification. International Journal of Computers and Applications, 1-10.
- Turing, A. M., 2009. Computing machinery and intelligence, Parsing the Turing Test, 23-65, Springer, Dordrecht.
- URL-1, https://tr.wikipedia.org/wiki/Salginlar_listesi, 10 Mart 2022

URL-2, <https://tr.wikipedia.org/wiki/COVID-19>, 5 Mart 2022

URL-3, <https://www.who.int/westernpacific/emergencies/covid-19/information/physical-distancing>, 1 Mayıs 2022

URL-4, <https://ocularpro.com/living-with-monocular-vision>, 1 Mayıs 2022

URL-5, <https://www.stereolabs.com/zed>, 2 Mayıs 2022

URL-6, https://en.wikipedia.org/wiki/Image_rectification, 25 Nisan 2022

URL-7, https://en.wikipedia.org/wiki/Turing_test, 5 Nisan 2022

URL-8, <https://www.quora.com/What-is-the-difference-between-deep-learning-and-usual-machine-learning>, 1 Mayıs 2022

URL-9, https://tr.wikipedia.org/wiki/Sinir_hücre, 26 Nisan 2022

URL-10, <https://www.hostingdergi.com.tr/yapay-zekâ-ve-sinir-aglari>, 22 Mart 2022

URL-11, <https://tr.wikipedia.org/wiki/Tensör>, 3 Mayıs 2022

URL-12, <https://devreyakan.com/aktivasyon-fonksiyonlari/>, 25 Nisan 2022

URL-13, <https://stanford.edu/~shervine/l/tr/teaching/cs-230/cheatsheet-convolutional-neural-networks>, 15 Nisan 2022

URL-14, <https://towardsdatascience.com/softmax-activation-function-explained-a7e1bc3ad60>, 5 Nisan 2022

URL-15, <https://stanford.edu/~shervine/l/tr/teaching/cs-230/cheatsheet-convolutional-neural-networks>, 1 Mayıs 2022

6. EKLER

Ek 1. Geliştirilen algoritmanın sözde (pseudo) kod şeması

<u>Girdi</u>	stereoParams	stereo kamera parametreleri
	videoLeft	sol kamera için elde edilen görüntü
	videoRight	sağ kamera için elde edilen görüntü
<u>Çıktı</u>	Pc	varlık belirteci
	Bx, By	sınırlayıcı kutu merkez koordinatları
	Bw, Bh	sınırlayıcı kutu genişlik ve yüksekliği
	C1, C2, ..., Cn	sınıflar
	M	sınırlayıcı kutu merkez koordinatları (3B)
	S (kırmızı), S (yeşil)	sınırlayıcı kutu (kırmızı/yeşil)

Görüntüleri rektifiye et
Aykırılıkları hesapla
Nokta bulutunu oluştur
Üç boyutlu koordinatları hesapla [X, Y, Z]

başla

x: tespit edilen kişi sayısı / oluşturulan sınırlayıcı kutu sayısı
oku (x)
Pc, Bx, By, Bw, Bh, C1, ..., Cn: sınırlayıcı kutu parametreleri
S: sınırlayıcı kutu
M: sınırlayıcı kutunun merkez koordinatları (3B)
if (x > 0)
 S ← Pc, Bx, By, Bw, Bh, C1, ..., Cn
 M ← [X, Y, Z]
 i: diğer kişiler ile mesafesi hesaplanacak kişi
 j: diğer kişiler
 k: sınırlayıcı kutu sayısı
 D: sınırlayıcı kutu merkezleri arası mesafe (3B)
 for i:= 1 to (k-1)
 for j:= (i+1) to (k-1)
 D ← [X, Y, Z]
 if (D < 1 metre)
 riskanalizi (1, i) ← 1
 göster S (kırmızı)
 else riskanalizi (1, i) ← 0
 göster S (yeşil)
 end
 end
 end
end
else BİTİR
end

ÖZGEÇMİŞ

İlköğrenimini Trabzon, İskenderpaşa İlköğretim Okulu'nda tamamladı. Lise öğrenimini Ünye Mehmet Refik Güven Anadolu Öğretmen Lisesi'nde tamamladı. 2018 yılında Yıldız Teknik Üniversitesi, İnşaat Fakültesi, Harita Mühendisliği Bölümü'nden mezun oldu. Aynı yıl Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Harita Mühendisliği Anabilim Dalı, Fotogrametri Bilim dalında Yüksek Lisans eğitimine başladı. Halen, Karadeniz Teknik Üniversitesi, Harita Mühendisliği Bölümü'nde Araştırma Görevlisi olarak çalışmaktadır. İngilizce bilmektedir.

