

**T.C.
SÜLEYMAN DEMİREL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**ZARARLI VİDEO İÇERİKLERİNİN DERİN ÖĞRENME TEKNİKLERİ
İLE TESPİTİ VE FİLTRELENMESİ İÇİN BİR YAZILIM ARACI
GELİŞTİRİLMESİ**

Fatma Gülşah TAN

**Danışman
Doç. Dr. Asım Sinan YÜKSEL**

**DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
ISPARTA- 2023**



© 2023 [Fatma Gülşah TAN]

İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER.....	i
ÖZET	iii
ABSTRACT	iv
TEŞEKKÜR.....	v
ŞEKİLLER DİZİNİ	vi
ÇİZELGELER DİZİNİ	viii
SİMGELER VE KISALTMALAR DİZİNİ.....	ix
1. GİRİŞ.....	1
1.1. Hedef ve Katkılar	2
1.2. Tez Organizasyonu.....	3
2. KAYNAK ARAŞTIRMASI.....	4
2.1. Derin Öğrenme Alanında Yapılan İlgili Çalışmalar	4
2.2. Sanal Ortamlarda Bulunan Risklere Yönelik Yapılan Çalışmalar.....	7
2.3. Çoklu Nesne Tespiti Alanında Yapılan Çalışmalar	8
3. METARYAL VE YÖNTEM.....	15
3.1. Nesne Tespitinde Kullanılan Yöntemler	15
3.1.1. Derin öğrenme.....	15
3.1.2. Derin öğrenmenin uygulama alanları.....	16
3.1.3. Derin öğrenme mimari yapısı.....	17
3.1.3.1. Konvolüsyonel sinir ağları.....	18
3.1.3.2. Tekrarlayan/yineleyen sinir ağları	18
3.1.3.3. Sınırlı boltzmann makineleri.....	19
3.1.3.4. Uzun kısa vadeli hafıza ağları	19
3.1.3.5. Derin oto-kodlayıcılar	20
3.1.3.6. Derin inanç ağları.....	20
3.1.4. Derin öğrenme alanında kullanılan diller/kütüphaneler	21
3.1.5. Derin öğrenme ile nesne tespiti ve nesne takibi	23
3.1.5.1. Ön işleme	23
3.1.5.2. Nesne tespiti.....	23
3.1.5.3. Nesne sınıflandırma	25
3.1.5.4. Nesne takibi.....	26
3.1.5.5. Nesne tespiti ve takibi veri setleri.....	27
3.2. Zararlı İçerik Tespiti Sisteminin Uygulanması.....	28
3.2.1. Anket sorularının hazırlanması ve deney sonuçlarına göre kategorilerin belirlenmesi	28
3.2.2. Veri setlerinin oluşturulması.....	29
3.2.3. Algoritmaların testi ve performanslarının karşılaştırılması	31
3.2.4. Derin öğrenme tabanlı video oynatıcı yazılımının geliştirilmesi.....	50
3.2.4.1. Tek aşamalı sansür	53
3.2.4.2. İki aşamalı sansür	55
3.2.4.3. Web uygulamasının oluşturulması ve test edilmesi.....	58
3.2.4.4. Uygulamaya ait ekran görüntüleri.....	59
4. ARAŞTIRMA BULGULARI VE TARTIŞMA.....	65
5. SONUÇ VE ÖNERİLER.....	67
KAYNAKLAR	68
EKLER.....	77

EK A. Anket Soruları.....	78
ÖZGEÇMİŞ.....	83



ÖZET

Doktora Tezi

ZARARLI VIDEO İÇERİKLERİNİN DERİN ÖĞRENME TEKNİKLERİ İLE TESPİTİ VE FİLTRELENMESİ İÇİN BİR YAZILIM ARACI GELİŞTİRİLMESİ

Fatma Gülşah TAN

Süleyman Demirel Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Asım Sinan YÜKSEL

21. yüzyılın başından beri internet, yaşamın hemen hemen her alanında kullanılmaktadır. Bu, dijital bağımlılığı ortaya çıkarmakta ve bu durumdan en çok etkilenen grup genç bireyler olmaktadır. Ayrıca pandemi kısıtlamaları nedeniyle gençlerin sosyal ağlarda ve video yayın platformlarında daha fazla zaman geçirirken silah, bıçak gibi tehlikeli nesnelerin sansürsüz bir şekilde gösterilmesine maruz kalmaları ruh sağlıklarını olumsuz etkileyebilecek önemli bir sorun haline gelmiştir. Dijital ortamlarda bulunan zararlı içerikleri sansürleyen politikalar sınırlı veya hiç yoktur. Ayrıca çocukların olumsuz etkilenmesine neden olabilecek açık içerikler, sigara, şiddet unsurları ve alkol içeren videolar da bulunmaktadır. Bu nedenle, zararlı içeriklerin gerçek zamanlı, otomatik olarak tespit edilmesini ve yayın akışı boyunca sansürlenmesini sağlayan sistemlere ihtiyaç duyulmaktadır. Bu sorunlara çözüm bulmak için, dijital platformlara kolayca entegre edilebilen özgün bir platform geliştirilmiştir. Zararlı içeriklerin tespiti için de anket çalışması düzenlenmiştir. Anket sonuçlarımıza göre katılımcılar sigara, alkol, şiddet unsurları ve açık içeriklerin videolar üzerinde sansürlenmesi gerektiğini belirtmişlerdir. Oluşturulan eğitim modellerinin hiperparametre değerleri optimize edilmiş ve en yüksek başarı oranı sağlayan modeller anket sonucunda belirlenen tüm kategoriler için kullanılmıştır. Sonuç olarak yüksek doğrulukla zararlı içerik tespiti yapan bir sistem elde edilmiştir. Boru hattı mimarisinin nesne tespit modellerine entegre edilmesi ile de nesne algılama süresinin iyileştirilmesi sağlanmıştır. Sistem kişiye özgü, dinamik çalışabilmesi ve gecikmesiz yayın akışı sağlaması ile seçilen kategorilere göre gerçek zamanlı olarak zararlı içerik tespiti ve sansürleme yapabilmektedir. Bu çalışmada gerçek zamanlı performans elde etmek için farklı platformlara entegre edilebilen iki farklı boru hattı tabanlı yaklaşım önerilmiştir: Tek aşamalı sansür ve iki aşamalı sansür. Tek aşamalı sansür yöntemi görüntüler ve web kameralarında gerçek zamanlı sansür yapabilmektedir. İki aşamalı sansür ise büyük video dosyaları üzerinde gerçek zamanlı sansürleme ve yayın akışı sağlamaktadır. Geliştirdiğimiz YOLO tabanlı derin öğrenme yöntemlerimiz, diğer nesne tespit algoritmalarına göre yaklaşık %58.6 daha hızlı çalışmaktadır. İçerik bazında doğruluk oranları ise açık içerikler için %98.4, alkol için %97.1, sigara için %98.3 ve şiddet unsurları için %97.1 olarak elde edilmiştir.

Anahtar Kelimeler: Derin öğrenme, nesne tespiti, nesne takibi, karar destek sistemi, zararlı içerik tespiti, gerçek zamanlı sansür.

2023, 83 sayfa

ABSTRACT

Ph. D. Thesis

DEVELOPMENT OF A SOFTWARE TOOL FOR DETECTING AND FILTERING HARMFUL VIDEO CONTENT WITH DEEP LEARNING TECHNIQUES

Fatma Gülşah TAN

**Süleyman Demirel University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering**

Supervisor: Assoc. Prof. Dr. Asım Sinan YÜKSEL

Since the beginning of the 21st century, the internet has been utilized in almost every aspect of life. This widespread usage has brought forth digital addiction, primarily affecting young individuals. Additionally, due to pandemic restrictions, young people have been spending more time on social networks and video streaming platforms, exposing them to uncensored display of dangerous objects such as guns and knives, which can have significant negative effects on their mental well-being. Policies censoring harmful content in digital environments are limited or non-existent. Moreover, there are explicit content, videos containing cigarette, violence and alcohol that can potentially have adverse effects on children. Therefore, there is a need for systems that can automatically and in real-time detect harmful content and censor it during the streaming process. To address these issues, an innovative platform has been developed that can be easily integrated into digital platforms. An online survey was conducted to identify harmful content and according to the survey results, participants indicated the necessity of censoring cigarette, alcohol, violence and explicit content in videos. The hyperparameter values of the generated training models were optimized and the models that achieved the highest success rate were utilized for all categories determined in the survey. As a result, a system capable of accurately detecting harmful content was obtained. The integration of the pipeline architecture into object detection models improved the object detection time. The system can perform real-time content detection and censorship based on selected categories, offering personalized and dynamic functionality, as well as seamless streaming. In this study, two different pipeline-based approaches that can be integrated into various platforms were proposed to achieve real-time performance: single-stage censorship and two-stage censorship. The single-stage censorship method enables real-time censorship on images and web cameras. The two-stage censorship method allows real-time censorship and streaming on large video files. Our YOLO-based deep learning methods perform approximately 58.6% faster than other object detection algorithms. The accuracy rates for specific content categories are 98.4% for explicit content, 97.1% for alcohol, 98.3% for cigarette and 97.1% for violence.

Keywords: Deep learning, object detection, object tracking, decision support system, harmful content detection, real time censorship.

2023, 83 pages

TEŞEKKÜR

Bu araştırmada bana rehberlik etmek, bilgeliği ve deneyimiyle beni yönlendirmek için her zaman yanımda olduğunu hissettiğim sabrı, özverisi ve motivasyonu sayesinde tezimde hedeflerime ulaşmamda yardımcı olan değerli danışman hocam Doç. Dr. Asım Sinan YÜKSEL'e teşekkürlerimi sunarım.

Tezimin her aşamasında teşvik edici sözleri, anlayışları ve destekleri ile bu tez çalışmasının anlamını ve değerini arttıran, haklarını asla ödeyemeyeceğim canım AİLEME sonsuz sevgi ve saygılarımı sunarım.

Ayrıca, bölüm hocalarım, arkadaşlarım, sevdiklerimden oluşan ve tez çalışmasının gerçekleşmesinde katkısı olan herkese teşekkür ve minnetimi özellikle belirtmek istiyorum.

Fatma Gülşah TAN
ISPARTA, 2023

ŞEKİLLER DİZİNİ

	Sayfa
Şekil 3.1. Derin öğrenme ağ mimarisi.....	16
Şekil 3.2. Konvolüsyonel sinir ağı mimarisi	18
Şekil 3.3. Tekrarlayan sinir ağı mimarisi.....	18
Şekil 3.4. Sınırlı boltzmann makineleri sinir ağı mimarisi.....	19
Şekil 3.5. Uzun kısa vadeli hafıza ağı mimarisi.....	20
Şekil 3.6. Derin oto-kodlayıcı sinir ağı mimarisi.....	20
Şekil 3.7. Derin inanç ağları mimarisi	21
Şekil 3.8. Derin öğrenme mimarilerinin zamansal gelişimi	21
Şekil 3.9. LabelImg ile örnek nesne etiketleme görüntüsü.....	33
Şekil 3.10. Sigara nesnesine ait örnek YAML dosyası.....	34
Şekil 3.11. Sigara nesnesine ait örnek ptxt dosyası	34
Şekil 3.12. Sigara nesnesine ait eğitim modelinin oluşturulması için kullanılan kod	34
Şekil 3.13. Sigara nesnesine ait kayıp ve performans ölçümleri	35
Şekil 3.14. Sigara nesnesine ait etiketlenmiş veriler ve modelin tahmin sonuçları	39
Şekil 3.15. Sigara nesnesine ait Faster R-CNN modelinin tahmin sonuçları.....	40
Şekil 3.16. Sigara nesnesine ait SSD modelinin tahmin sonuçları	41
Şekil 3.17. Şiddet unsurları nesnesine ait kayıp ve performans ölçümleri.....	42
Şekil 3.18. Şiddet unsurları nesnesine ait son kayıp ve performans ölçümleri ..	43
Şekil 3.19. Eğitim modelinin test edilmesi.....	44
Şekil 3.20. Sentetik verilerde şiddet unsurlarının tespiti	44
Şekil 3.21. Şiddet unsurlarına ait etiketlenmiş veriler ve modelin tahmin sonuçları	45
Şekil 3.22. Alkol unsurları nesnesine ait kayıp ve performans ölçümleri	46
Şekil 3.23. Alkol unsurlarına ait etiketlenmiş veriler ve modelin tahmin sonuçları	47
Şekil 3.24. Alkol unsurları test verilerinde yanlış-pozitif çıkan örnek görüntü ..	48
Şekil 3.25. Açık içerikli nesnelere ait kayıp ve performans ölçümleri	49
Şekil 3.26. Eğitim modelinin farklı uzantılı eğitim modellerine dönüştürülmesini sağlayan kod.....	50
Şekil 3.27. Eğitim modelleri ve dosya bilgileri	51
Şekil 3.28. Klasik nesne tespiti için kullanılan mimari.....	52
Şekil 3.29. Önerilen yöntem ile nesne tespiti için kullanılan mimari.....	52
Şekil 3.30. Web kamerası kaynağından gelen video girişinden nesne algılama işlemi sırasında çekilen fotoğraflar	53
Şekil 3.31. İki aşamalı sansür akış diyagramı	56
Şekil 3.32. Proje yapısına ait ekran görüntüsü.....	57
Şekil 3.33. Yolov5Net.Scorer sınıfları.....	57
Şekil 3.34. Sigara nesnesine ait oluşturulan ONNX modeli	58
Şekil 3.35. Uygulamada kullanılan kategorilere ait modeller.....	59
Şekil 3.36. Giriş paneli.....	60
Şekil 3.37. Uygulamada kullanılan modüller	61
Şekil 3.38. Uygulamada kullanılan veri seti indirme ekranı	61
Şekil 3.39. Uygulamada görüntüler üzerinde nesne tespiti yapan modül	63
Şekil 3.40. Uygulamada videolar üzerinde nesne tespiti yapan modül	64

Şekil A.1. Cinsiyetiniz	78
Şekil A.2. Yaş aralığınız	78
Şekil A.3. Mesleğiniz	78
Şekil A.4. Medeni durumunuz	79
Şekil A.5. Çocuğunuz var mı?	79
Şekil A.6. Kötü unsurları görmekten rahatsız olma durumu	79
Şekil A.7. Kötü unsurların filtrelendiğini ya da filtrelendiğini gösteren şekil	80
Şekil A.8. Kötü unsurların oranları	80
Şekil A.9. Şiddet içeriklerinin filtrelendiğini isteyen kullanıcı yorumları	81
Şekil A.10. Şiddet içeriklerinin filtrelendiğini isteyen kullanıcı yorumları	82
Şekil A.11. Şiddet içeriklerinin filtrelendiğini isteyen kullanıcı yorumları	82



ÇİZELGELER DİZİNİ

	Sayfa
Çizelge 3.1. Derin öğrenmede kullanılan diller/kütüphaneler	22
Çizelge 3.2. Sınıflandırma için kullanılan özniteliklerin kıyaslanması	25
Çizelge 3.3. Nesne takibi yöntemleri	26
Çizelge 3.4. Şiddet unsurlarına ait eğitim ve test veri sayısı	30
Çizelge 3.5. Sentetik tabanca eğitim ve test veri sayıları	30
Çizelge 3.6. Sigara nesnesine ait eğitim ve test verileri sayısı	30
Çizelge 3.7. Alkol nesnesine ait eğitim ve test verileri sayısı	31
Çizelge 3.8. Sigara nesnesine ait eğitim modellerinin başarı oranları	38
Çizelge 3.9. Benzer nesneleri içeren veri türlerine göre eğitim ve test verileri sayısı	42
Çizelge 3.10. Kategorilere göre başarı oranları	49
Çizelge 3.11. Kullanılan yöntemlerin karşılaştırılması	55

SİMGELER VE KISALTMALAR DİZİNİ

AlexNet	Alex ağı
AP	Ortalama hassasiyet
API	Uygulama programlama arayüzü
BM	Boltzman makinesi
BT	Bilgisayarlı tomografi
CNN	Evrişimli sinir ağları
COCO	Bağlamdaki ortak nesneler
COSMOS	Bulutla geliştirilmiş açık yazılım tanımlı test aracı
CPU	Merkezi işlem birimi
Cuda	Birleşik hesaplama mimarisi
DaCoLT	Öğrenme ve test aşamalarında koyulaştırma ve kontrast
DAE	Derin otomatik kodlayıcılar
DBN	Derin inanç ağı
DCN	Derin evrişimsel ağ
Deep	Derin eşleştirme
DeepSort	Derin basit çevrimiçi ve gerçek zamanlı takip
ESP	Genişletilmiş SP
Fast R-CNN	Hızlı bölgesel tabanlı evrişimli sinir ağları
Faster R-CNN	Daha hızlı bölgesel tabanlı evrişimli sinir ağları
GoogleNet	Google ağı
GPU	Grafik işlemci birimi
IMFDB	İnternet film silahları veritabanı
KITTI	Karlsruhe Teknoloji Enstitüsü ve Toyota Teknolojik Enstitüsü'nün ortak nesneleri
LSTM	Uzun kısa vadeli hafıza ağları
mAP	Ortalama kesinlik değerlerinin ortalaması
Mask R-CNN	Maskelemeli bölgesel tabanlı evrişimli sinir ağları
MOTA	Çoklu nesne takip doğruluğu
MS-COCO	Microsoft ortak bağlamdaki nesneler
MSRC	Microsoft Research Cambridge
Multicut	Çoklu kesim
ONNX	Açık yapay sinir ağı değişimi
PASCAL	Desen analizi, istatistiksel modelleme ve hesaplamalı öğrenme
PASCAL-VOC	PASCAL görsel nesne sınıfları
RAM	Rastgele erişim belleği
RBM	Kısıtlı boltzmann makineleri
R-CNN	Bölgesel tabanlı evrişimli sinir ağları
RNN	Tekrarlayan sinir ağları
RPN	Alan öneri ağı
SqueezeNet	Sıkıştırılmış ağ
SSD	Tek atış çok kutu dedektörü
SVM	Destek vektör makinesi
Txt	Metin dosyası
VGG	Görsel geometri grubu
Xml	Genişletilebilir işaretleme dili
YAML	YAML işaretleme dili değildir
YOLO	Sadece bir kere bak

1. GİRİŞ

İnternet, hemen hemen her alanda kullanılmaya başlanmasıyla birlikte önemli bir ihtiyaç haline gelmiştir. Bu durumun ilerlemesi ile en çok etkilenecek kitle genç bireyler olmaktadır. Gerekli önlemlerin alınmadığı durumlarda bu bireylerin sanal ortamlardaki risklerle kolayca karşılaşması kaçınılmazdır. Özellikle pandemi döneminde bu tarz sosyal medya araçlarına olan ilgi daha da artmıştır. İnsanlar pandeminin etkilerinden biraz olsun uzaklaşabilmek için boş zamanlarını dolduracak Netflix gibi medya içerik üretici platformlara yönelmişlerdir (Gracová, 2022). Popüler bir video platformu olan Youtube'da da paylaşılan videoların içeriğini incelemek zor bir durum haline gelmiştir. Uygun başlıklar ve video görselleri şeklinde paylaşılan ve içeriğinde uygunsuz görüntüler olan birçok video hala bu platformlarda yayınlanmaktadır. İş ve eğitim hayatında da kötü unsurlar içeren bu videolarla karşılaşmak kolay ve sıkıntılı bir durum haline gelmiştir. Zoom gibi canlı yayın platformlarında da olası olumsuz durumlarla karşılaşma oranı oldukça yüksektir. Tüm bu durumlardan dolayı video paylaşım platformlarının kendi politikalarına ek olarak internet tabanlı tüm videolarda belirlenen kötü unsurları anlık olarak tespit eden ve filtreleyen bir sisteme ihtiyaç duyulmaktadır. Derin öğrenme tabanlı nesne tespiti yöntemlerinin algılama doğruluğunun ve güvenilirliğinin yükselmesiyle birlikte birçok akıllı video uygulaması, nesne algılama teknolojilerini aktif olarak kullanmaya başlamıştır.

Tez çalışmasında, kullanıcılara uygulanan anket ile internet üzerinde izlemekten rahatsız oldukları videolar için kategoriler belirlenmiştir. Bu kategoriler; açık içerikli görüntüler, alkol, sigara ve şiddet unsurlarıdır. Bu unsurların erken tespiti güvenlik ve birçok olumsuz etkinin çözülmesinde büyük önem taşımaktadır. Videodaki bu nesneleri tanımak için eğitilmiş modeller kullanılarak özgün bir sistem geliştirilmiştir. Bu model, Netflix, Youtube gibi video izleme ve Zoom gibi canlı yayın platformlarından esinlenerek iki farklı platformda çalışabilecek şekilde tasarlanmıştır. Belirlenen kategoriler kullanıcı bazlı olarak çalışmaktadır. Eğer herhangi bir kategori seçilirse, nesneler bu unsurlara göre tespit edilip ekranda gösterilmekte, seçilmezse orijinal haliyle

gösterilmeye devam edilmektedir. Sonuç olarak, video akışının olduğu platformlarda meydana gelebilecek zararlı içeriklerin engellenmesi amaçlanmaktadır. İlk olarak sigara kategorisi üzerinde nesne tespit modelleri denenmiş ve en iyi sonucun YOLO (You Only Look Once, Sadece Bir Kere Bak) ile alındığı belirlenmiştir. YOLO'nun başarı oranı ne kadar yüksek olsa da özellikle kameralar üzerinde gerçek zamanlı nesne tespitinde hız sorunu önemli bir konu haline gelmiştir.

1.1. Hedef ve Katkıları

Tez çalışmasında, kullanıcılara uygulanan anket ile internette izlemekten rahatsız oldukları kategoriler baz alınmıştır. Bu unsurların erken tespiti, güvenlik ve birçok olumsuz etkinin giderilmesi açısından büyük önem taşımaktadır. Bundan dolayı probleme uygun algoritma ve mimarinin seçilmesi gerekmektedir. Tez çalışmasında, derin öğrenme, nesne tespiti ve takibinde yapılan araştırmalar incelenerek kullanılan mimariler, ağlar, algoritmalar, veri setleri, kütüphaneler hakkında bilgiler verilmiştir.

Videodaki zararlı unsurları tanımak için eğitilmiş modeller kullanılarak örnek bir sistem geliştirilmiştir. Geliştirilen sistem sadece çocukların güvenliği için değil iş veya eğitim hayatında da rahatlıkla kullanılabilir. Belirlenen kategoriler kullanıcı bazında çalışmaktadır. Önceden eğitilmiş modeller sayesinde, seçilen kategorilere göre istenilen videolar ve görseller üzerinde nesne tespiti yapılabilmektedir.

Tez çalışmasında geliştirilen sistem ile literatüre sağlanan katkılar şu şekilde özetlenebilir:

- Güvenilir nesne algılama ile zararlı içeriklerin tespiti ve sansürlenmesi,
- Seç izle sistemlerine entegre olabilecek özgün bir platform geliştirilmesi,
- Kişiye özgü, dinamik sansürleme yapabilme ve gecikmesiz yayın akışı sağlama,

- Canlı yayınlar ve videolar için gerçek zamanlı sansürleme ve yayın akışı yapabilecek yeni algoritmalar geliştirilmesi,
- Boru hattı mimarisinin nesne tespit modellerine entegre edilmesi ve nesne algılama süresinin iyileştirilmesi.

Geliştirilen uygulamada kullanılan tüm veri setleri paylaşılmış ve bu alanda çalışacak olan araştırmacılara destek olmak amaçlanmıştır. Bu veri setleri ile farklı eğitim modellerinin oluşturulabilmesi ya da eğitilen modeller kullanılarak nesne tespiti gerçekleştirilmiş görsellere ve videolara ulaşılabilmesi sağlanmıştır. Tez çalışmasında uygulanan yöntemlerin, kullanılan modellerin başka platformlarda da kullanılabilmesi ve geliştirilebilmesi açısından uygulamaya ait kaynak kodlar Github (Tan, 2022) üzerinde paylaşılmıştır.

1.2. Tez Organizasyonu

Tez çalışmasının geri kalanı aşağıdaki gibi organize edilmiştir. Bölüm 2'de literatürdeki benzer çalışmalar incelenmiştir. Bölüm 3'te derin öğrenme hakkında detaylı bilgiler verilmiştir ve geliştirilen sistem ile videolardaki zararlı içeriğin gerçek zamanlı tespiti ile ilgili metodoloji sunulmuştur. 4. bölümde sonuçlar paylaşılmıştır ve gelecekteki çalışmalar için değerlendirmelere yer verilmiştir.

2. KAYNAK ARAŞTIRMASI

Günümüzde şiddet içeren durumların ve unsurların erken tespiti, güvenlik açısından büyük önem taşımaktadır. Bu tarz durumların önlenmesinin bir yolu da kamera görüntülerinde ve videolarda tabanca ve bıçak gibi tehlikeli nesnelerin varlığını tespit etmektir. Yangın gibi olayların erken tespitinde ise sigara veya duman tespiti önemli bir yere sahiptir. Sigara, alkol ve açık içerikli videolara özellikle çocukların maruz kalması olumsuz etkiler bırakabilir. Ancak, yapılan anket çalışmasına göre kullanıcıların bu tür zararlı içeriklerden etkilenme oranları farklılık göstermektedir. Bazıları alkolden olumsuz etkilenmezken, bazı kullanıcılar bu tür içerikleri görmek istemediklerini belirtmişlerdir. Bu yüzden, kullanıcı bazlı çalışan ve gerçek zamanlı dinamik nesne tespiti yapan uygulamalara ihtiyaç duyulmaktadır. Bu bölümde, bu unsurların olumsuz etkileri ve tehlikeli unsurlar, sigara, alkol ve açık içerikli unsurların tespiti ile ilgili yapılmış çalışmalar hakkında bilgiler verilmiştir.

2.1. Derin Öğrenme Alanında Yapılan İlgili Çalışmalar

Bewley vd. (2016), Macar algoritması ve Kalman filtresi kullanarak gerçek zamanlı nesne takibi için uygulamalı bir yöntem önermiştir. Faster R-CNN (Faster Region-Based Convolutional Neural Network, Daha Hızlı Bölge-Tabanlı Evrişimli Sinir Ağı) algoritması ile eğitilen modelin nesne takip performansında %18.9 oranında artış olduğunu belirtmişlerdir.

Liu vd. (2017), çalışmalarında son yıllarda popüler olan derin öğrenme mimarileri ve uygulamaları hakkında son gelişmeleri özetlemektedir. DAE (Denoising Autoencoder, Derin Otomatik Kodlayıcılar), CNN (Convolutional Neural Networks, Evrişimli Sinir Ağları), DBN (Deep Belief Network, Derin İnanç Ağı) ve RBM (Restricted Boltzmann Machine, Kısıtlı Boltzmann Makineleri) olmak üzere dört farklı derin öğrenme mimarisine genel bir bakış sunulmuştur. CNN'lerin genellikle görüntü işleme problemlerinde kullanıldığını ve özellik çıkarma, konumsal invariyan ve sınıflandırma gibi görevler için etkili olduğunu belirtmişlerdir. Ayrıca derin sinir ağı mimarilerinin farklı uygulama alanlarına

yönelik kullanımlarını araştırmışlardır. Bu alanlar arasında görüntü tanıma, doğal dil işleme, konuşma tanıma, oyun oynama, robotik ve tıp gibi birçok uygulama bulunmaktadır. Araştırma sonucunda, derin öğrenme teknikleri ve uygulamaları hakkında zamanında incelemeler ve tanıtımlar sağlayarak okuyuculara farklı derin öğrenme mimarileri ve alandaki son gelişmeler sunmayı amaçlamışlardır. Lu vd. (2017), sıralı çerçeveler arasındaki nesne ilişkilerini modelleme sorununu tartışmışlardır. SSD (Single Shot MultiBox Detector, Tek Atış Çok Kutu Dedektörü) ağı ve Macar algoritmasını kullanarak önerdikleri bu modelde, nesne öznitelikleri arasındaki eşleştirme hatalarını en aza indirdiklerini belirtmişlerdir.

Brunetti vd. (2018), sınıflandırma stratejilerine ve insan-bilgisayar etkileşimine dayalı bir video gözetimi sunmuşlardır. Daha yakın zamanlarda, yaya algılama ve izleme için CNN dahil olmak üzere farklı derin öğrenme yöntemleri üzerine analitik araştırmaları dahil etmişlerdir. Video gözetimi, tek veya birden çok nesneyi bulmak ve izlemek için birden çok sensör kullanan karmaşık bir süreçtir. Bu çalışmada yaya algılama ve takip sistemleri ile ilgili araştırmalar sunulmuştur. Yapılan çalışmaların otomatik yaya algılamada başarılı sonuçlar elde ettiğini belirtmişlerdir.

Chaudhary vd. (2018), suç oranlarının artması nedeniyle kalabalık sosyal ortamlarda gözetleme sistemlerindeki videolarda olağan dışı olayları otomatik olarak tespit etmeyi hedeflemişlerdir. Önerilen yöntemin, tespit doğruluğuna kadar birçok anormal aktiviteyi başarılı bir şekilde tespit edebildiğini göstermişler ve %90 algılama oranı elde etmişlerdir. Kim vd. (2018), çalışmalarında önceden eğitilmiş YOLOv2 kullanarak çevrimiçi yaya takibi gerçekleştirmişlerdir. Önceden eğitilen bu modelin öğrenme süresini kısaltmışlar ve geleneksel yöntemlere göre daha iyi performans sağladıklarını iddia etmişlerdir.

Toğaçar ve Ergen (2019), biyomedikal alanındaki araştırmalara ilişkin derin öğrenme konusunda araştırma yapmışlar ve bilgiler sunmuşlardır. Kullanılan yöntemleri geleneksel yöntemlerle karşılaştırarak sonuçlarını göstermişlerdir.

Toğaçar vd. (2019), erişilebilir göğüs röntgen filmlerinde CNN modeli kullanarak pnömoninin erken tespitini sağlamışlardır. Modelin başarı oranını diğer sınıflandırma algoritmaları ile karşılaştırarak test etmişlerdir. Çalışma sonuçlarında en yüksek başarı oranını (%95.8) SVM (Support Vector Machines, Destek Vektör Makineleri) ile elde ettiklerini belirtmişlerdir.

Zhang vd. (2019), bazı algılama uygulamalarında ışık, bozulma ve tıkanıklık gibi sorunları çözerek, manuel etiketleme işlemi olmadan domuzların gerçek zamanlı algılanmasını sağlayan bir model önermişlerdir. CNN yöntemine dayanan ve Macar algoritması kullanan bu model ile ticari bir çiftlikteki veri kümeleriyle yapılan deneylerde başarılı bir şekilde domuz tespit edip izleyebildiklerini belirtmişlerdir.

Kanna vd. (2020), tanıklar tarafından sağlanan tanımlayıcı belirteçlere dayalı olarak kamera görüntülerinde suçluları tespit etmeyi amaçlayan bir yöntem önermişlerdir. Derin öğrenme tabanlı bu yöntem, suçluları tespit etmek için girdi olarak gömlek stili, cinsiyet ve gözlük parametrelerini kullanmaktadır. Önerilen yaklaşımın, video görüntülerinde suçlu insanları tanımlamada %87 doğruluk sağladığını belirtilmiştir.

Yang vd. (2020), akıllı şehir kullanım örneklerini desteklemek ve engelli yayalara yardımcı olmak için kavşaklarda gerçek zamanlı nesne algılama ve izlemeyi ele almışlardır. Çalışmalarında, veriler COSMOS (Cloud-Enhanced Open Software Defined Testing Tool, Bulutla Geliştirilmiş Açık Yazılım Tanımlı Test Aracı) kullanılarak incelenmiştir. Bu çalışmada, algılama algoritmasının (YOLO, Mask R-CNN (Mask Region-Based Convolutional Neural Network, Maskelemeli Bölge-Tabanlı Evrişimli Sinir Ağı)) ve çoklu nesne izleme algoritmasının (DeepSort (Deep Simple Online and Realtime Tracking, Derin Basit Çevrimiçi ve Gerçek Zamanlı Takip), Multicut (Çoklu Kesim) ve Deep (Derin Eşleştirme)) performanslarını karşılaştırarak sonuçları sunmuşlardır. MOTA (Multiple Object Tracking Accuracy, Çoklu Nesne Takip Doğruluğu) doğruluk oranının, arabalar için %73.2, yayalar için yaklaşık %2.8 olduğunu belirtmişlerdir.

2.2. Sanal Ortamlarda Bulunan Risklere Yönelik Yapılan Çalışmalar

Arslankara ve Usta (2020), çalışmalarında lise öğrencilerine yönelik sanal ortamlarda bulunan risklere karşı algı düzeylerini belirlemeyi ve ölçmeyi amaçlamışlardır. Dijital bağımlılığın da sigara, uyuşturucu, alkol gibi bir bağımlılık türü olarak düşünülmesi gerektiğini belirtmişlerdir. Çalışma sonucunda Ağrı ilindeki lise öğrencilerinin, internet kullanımıyla birlikte sanal risklerdeki algı değişikliğinin oranının %38 olduğunu belirtmişlerdir. Bu nedenle tüm eğitim paydaşlarının, teknolojinin ve avantajlarının oluşturduğu tehdit ve tehlikelerin farkında olunması ve önemlerin alınması gerektiğini belirtmişlerdir.

Kalaman vd. (2019), çalışmalarında internete yönelik bireysel inançlarının ne düzeyde olduğunu tespit etmek için Akdeniz Bölgesi'nde bulunan ortaokul öğrencilerine yönelik bir anket çalışması yapmışlardır. İnternet kullanım saati, internet kullanım amacı gibi özelliklere göre inanç düzeylerini belirlemişlerdir. Katılımcıların internete olan inançları ve güvenilirliği 100 puan üzerinden 64.12 olarak bulmuşlardır.

Alghowinem (2018), çalışmasında çocuk dostu programlara ilişkin var olan tehditleri önleyebilmek amacıyla Youtube Kids üzerinde bir filtreleme modeli önermiştir. Bu bilgi çağında çocukların internetteki her türlü bilgiye ulaşması kaçınılmazdır. YouTube Kids, videoları filtrelemek için meta verileri ve topluluk geri bildirimlerini kullanan algoritmalara dayanmaktadır. Ancak bu yöntem, çocukları filtreleme mekanizmasını geçebilecek zararlı içerikli videolardan korumak için yeterli değildir. Bu çalışmada, çocukların güvenliği için ekstra bir katman olarak otomatikleştirilmiş video ve ses analizini kullanan gelişmiş bir gerçek zamanlı içerik filtreleme yaklaşımı önerilmiştir. Filtreleme mekanizmasından geçen videolar, dilimleme ve görüntü, ses ve transkripsiyon analizi dahil olmak üzere uygunluk analizine yönelik birkaç adımdan geçmektedir. Deneme sonuçlarının umut verici olduğu ve farklı dillerde de denenebileceği belirtmişlerdir.

Papadamou vd. (2019), çalışmalarında Youtube platformunda çocukları hedefleyen zararlı içerikli videoları tespit etmeyi amaçlamışlardır. Youtube çocuklar için Youtube Kids platformunu oluşturmuş olsa bile burada da birçok zararlı içerikli video yayınlanmaktadır. Bu çalışmada, Youtube Kids, API (Application Programming Interface, Uygulama Programlama Arayüzü) sağlamadığından dolayı veri setini elde etmek için videoların meta verilerini sağlayan Youtube API'sini kullanmışlardır. Video verileri iki farklı şekilde toplanmıştır. İlk olarak, rahatsız edici videolar sorunuyla ilgili farkındalığı artırmak için oluşturulmuş bir alt dizin olan ElsaGate'deki bilgileri kullanmışlardır. İkinci olarak, YouTube'da bulunan çizgi film videolarını listelemeye yarayan bir alt dizin olan fullcartoonsonyoutube'daki bilgileri kullanmışlardır. Kesin olarak etiketlenmiş bir veri kümesinden, %84.3 doğruluk sağlayan bir derin öğrenme sınıflandırıcı geliştirmişlerdir. Bu şekilde yeni yürümeye başlayan çocukları baz olarak Youtube üzerinde uygun videoları izlemeye başlayan ve sonrasında öneri dahilinde uygun olmayan videoların tavsiye edilmesinde %3.5'lik bir şans olduğunu belirtmişlerdir.

Giannakopoulos vd. (2010), çalışmalarında video paylaşım sitelerindeki şiddet içeriklerini tespit etmek için bir model önermişlerdir. Önerilen yaklaşımın ilk aşaması ses, hareketli görüntü ve metin verileri üzerinde çalışmaktadır. İkinci aşaması kullanıcı yorumlarından toplanmıştır. Sorun, dokuz boyutlu bir özellik uzayında ikili bir sınıflandırma (şiddet içeren ve şiddet içermeyen içerik) olarak ele alınmış ve burada 9 özellikten 7'si ses akışından çıkarılmıştır. Önerilen yöntem, 210 YouTube videosunda değerlendirilmiş ve %82 doğruluk oranı tespit etmişlerdir.

2.3. Çoklu Nesne Tespiti Alanında Yapılan Çalışmalar

Olmos vd. (2018), çalışmalarında insan müdahalesi ve gözetimi beklemeden videolarda otomatik tabanca algılama sistemi sunmuşlardır. Kayan pencere ve bölge önerisi yaklaşımını denemişler ve en iyi sonucu Faster R-CNN modeli ile elde etmişlerdir. Sonuçlarına bakıldığında 30 sahne arasından 27 sahnede 0,2 saniyeden daha kısa bir zaman aralığında art arda beş gerçek pozitifin ardından

alarmı başarıyla etkinleştirmişlerdir. Modelin düşük kaliteli videolarda bile yüksek performans gösterdiğini belirtmişlerdir. Modelin doğruluk oranını ise %84.21 olarak tespit etmişlerdir.

Verma ve Dhillon (2017), çalışmalarında hırsızlık, tecavüz, çatışma gibi olaylarda en çok kullanılan suç aletinin tabanca olduğunu ve otomatik tabanca tespitinin birincil bir gereklilik olduğunu ifade etmişlerdir. Otomatik silah tespitinde en gelişmiş yöntemlerden olan Faster R-CNN modeli olan DCN (Deep Convolutional Network, Derin Evrişimli Ağ) yöntemini kullanmışlardır. El tabancası tespitinde sonuçların uygun performansı elde ettiğini göstermişlerdir. Sistem, özellik çıkarıcı olarak CNN tabanlı VGG-16 (Visual Geometry Group, Görsel Geometri Grubu) mimarisini ve çeşitli sınıflandırıcıları içermektedir. Gerçek zamanlı olarak çalışan sistem ile %93 doğruluk oranı elde etmişlerdir.

Castillo vd. (2019), çalışmalarında videolarda bir veya daha fazla kişi tarafından kullanılan soğuk çelik silahların otomatik olarak algılanması için CNN tabanlı bir model önermişlerdir. Nesne tespitinde orta ile yüksek aydınlatma koşulları altında yüzey yansımalarının tespitlerini imkânsız hale getirdiğini belirtmişler ve bu durumu çözmek için DaCoLT (Darkening and Contrast at Learning and Test Stages, Öğrenme ve Test Aşamalarında Koyulaştırma ve Kontrast) adı verilen bir ön işleme prosedürü önererek, yöntemin ışık koşullarına dayanıklılığını güçlendirmişlerdir. Çalışma sonuçlarında silah dedektörü tespitinde en iyi sonucu %89.87 doğruluk oranı ile elde etmişlerdir.

Olmos vd. (2019), çalışmalarında tabanca tespitinde yanlış pozitiflerin sayılarını azaltmak için binoküler görüntü füzyon yaklaşımı önermişlerdir. Videolarda tabanca tespitinin değerlendirmelerinde kullanılan modellerin %73 doğruluk oranında iyi ortalama hassasiyetlerine ulaştığını ve bu modellerin gözetim videoları gibi düşük kaliteli videolarda çok sayıda yanlış pozitif ürettiğini belirtmişlerdir. Önerdikleri yaklaşımın sonuçlarına bakıldığında, yanlış pozitiflerin sayısının azaldığı, aynı zamanda algılama modelinin genel performansını da iyileştirerek gözetim videolarında nesne tespiti için kullanımının uygun olduğunu göstermişlerdir.

Pérez-Hernández vd. (2020), çalışmalarında küçük nesnelerin algılanmasının sağlamlığını, doğruluğunu ve güvenilirliğini arttıran teknikler önermişlerdir. İkili sınıflandırıcılarla nesne algılama adı verilen derin öğrenmeye dayalı iki seviyeli bir metodoloji kullanarak videolarda küçük nesnelerin tespitini iyileştirmeyi amaçlamışlardır. Birinci seviyede, giriş çerçevesinden aday bölgeler seçilmiş ve ikinci seviyede bire karşı hepsi veya bire karşı bir yöntemi ile bir CNN sınıflandırıcısına dayalı ikileme tekniği uygulanmıştır. Çalışmalarında kullanmak üzere tabanca, bıçak, akıllı telefon, fatura, çanta ve kart olmak üzere altı nesneyi göz önünde bulundurarak bir veri tabanı oluşturmuşlardır. Önerilen metodolojinin sonuçlarına bakıldığında başlangıç tespitine bağlı olarak yanlış pozitiflerin sayısını %56.50'ye kadar azalttığını belirtmişlerdir.

Justin ve Sydney (2017), çalışmalarında videolarda gerçek zamanlı olarak tabanca tespiti için CNN tabanlı bir uygulama gerçekleştirmişlerdir. Görüntülerdeki silahları tespit etmek ve sınıflandırmak için entegre bir ağ olarak Overfeat ağının Tensorflow tabanlı bir uygulamasını kullanmışlardır. Bu algoritma, hem nesne tespiti hem de sınıflandırma problemlerini çözmek için CNN kullanır. En iyi performansı, %93 eğitim doğruluğu ve %89 test doğruluğu ile Overfeat-3'te elde etmişlerdir.

Fernandez-Carrobles vd. (2019), çalışmalarında Faster R-CNN metodolojisine dayalı bir silah ve bıçak algılama sistemi sunmuşlardır. CNN tabanlı GoogleNet (Google Network, Google Ağı) ve SqueezeNet (Squeeze Network, Sıkıştırılmış Ağ) mimarisini karşılaştırmışlardır. Silah ve bıçak görüntüleri için COCO (Common Objects in Context, Bağlamdaki Ortak Nesneler) veri kümesini kullanmışlardır. Tabanca tespiti için en iyi sonucu, SqueezeNet mimarisi ile %85.44 doğruluk oranıyla, bıçak tespiti için ise GoogleNet yaklaşımı ile %46.68 doğruluk oranıyla elde etmişlerdir.

Egiazarov vd. (2019), çalışmalarında dünya çapında artan terör saldırılarını örnek vererek güvenlik kameralarının suç önleme ve tespitinde yetersiz kaldığını belirtmişlerdir. Bu sorunu çözmek için bir silahın ayrı ayrı bileşen

parçalarının tespitinde semantik CNN grubuna dayalı bir silah algılama sistemi sunmuşlardır. Sentetik veriler ve gerçek dünya verilerindeki sonuçları karşılaştırarak umut verici sonuçlar elde ettiklerini ve yaklaşımlarının tek bir derin konvolüsyonel ağı dayanan monolitik yaklaşıma kıyasla avantajlara sahip olabileceğini öne sürmüşlerdir.

De Azevedo Kanehisa ve De Almeida Neto (2019), CNN'lerin görüntü ve videolardaki nesne tespitinde insanlardan daha tutarlı ve doğru sonuçlar ürettiğini belirterek ateşli silah algılama sistemi oluşturmuşlardır. IMFDB (Internet Movie Firearms Database, İnternet Film Silahları Veritabanı) adını verdikleri internet tabanlı bir veri seti oluşturmuşlar ve YOLO algoritmasını kullanarak bu veri seti üzerinde çalışmalar yapmışlardır. 669 test görüntüsünden 644'ünü doğru şekilde sınıflandırarak %95.73 duyarlılık, %97.30 özgüllük, %96.26 doğruluk oranı ve %70 mAP (Mean Average Precision, Ortalama Hassaslık Oranı) elde etmişlerdir.

Yin vd. (2017), çalışmalarında el yapımı özelliklere dayanan duman algılama sistemlerini otomatik tespit etmek amacıyla 14 katmandan oluşan derin normalizasyon ve CNN önermişlerdir. Başarı oranını arttırmak için geleneksel veri setlerini çeşitlendirerek duman algılama modeli oluşturmuşlardır. Çalışma sonuçlarına bakıldığında %96.7 başarı oranı elde etmişlerdir.

Tao vd. (2016), çalışmalarında yangın önlemede duman tespitinin öneminden ve dumanın şekil, renk, doku olarak tespitinin zor olduğunu belirtmişlerdir. Bu nedenle duman algılama doğruluğunu iyileştirmek için, ham piksel değerlerinden sınıflandırıcı çıktılarına kadar uçtan uca eğitilebilen ve görüntülerden bu özellikleri otomatik olarak çıkaran CNN'e dayalı yeni bir yaklaşım önermişlerdir. Çalışma sonuçlarında %99.4 algılama oranı elde etmişlerdir.

Han vd. (2019), çalışmalarında sigara tespit zamanını azaltmayı amaçlamışlardır. Sigara veri setlerindeki ışık ve görüntü kalitesinden dolayı tespit edilmesi zor bir nesnedir. Faster R-CNN kullanarak renk

segmentasyonuna dayalı bir model geliştirmişler ve bu strateji ile karmaşıklığı azaltarak sigara tespiti için yanlış tespit oranını önemli ölçüde azalttıklarını belirtmişlerdir.

Dhanwal vd. (2020), çalışmalarında videolarda sigara içimine maruz kalmanın özellikle çocuklar için olumsuz etkiler bıraktığını belirtmişlerdir. Bu yüzden videolardaki sigaraların sansürlenmesi için derin öğrenme kullanarak Inception V3 modeli ile bir sinir ağı eğitmişlerdir. Çalışmalarında %89.99 başarı oranı elde etmişlerdir.

Açık içerik olarak kabul edilebilecek bir tür zararlı içerik, çıplaklığı tespit etmek ve uygun eşik değerlerini kullanarak içeriği filtrelemeyi içerir. Perez vd. (2017), hareket verilerinin çıplaklığa ek olarak dahil edilmesinin otomatik açık içerik tespiti alanında önemli bir başarı olduğunu belirtmişlerdir. Optik akış ve MPEG hareket vektörleri kullanarak statik (resim) ve dinamik (hareket) bilgilerin birleştirilmesi için yöntemler önermişlerdir. Deneylerinde Pornography-800 ve Pornography-2k adlı iki veri kümesini kullanmışlardır. Oluşturdukları yeni veri kümesi, altı saniyeden 33 dakikaya kadar süren yaklaşık 140 saatlik 1.000 pornografik ve 1.000 pornografik olmayan videodan oluşmaktadır. Önerilen yöntem, %97.9 sınıflandırma doğruluğu elde etmiştir. Bununla birlikte, pratik bir uygulama geliştirilmemiş ve açık içeriklerin gerçek zamanlı olarak filtrelenmesi gelecekteki çalışmalara bırakılmıştır.

Samal vd. (2023), açık içeriğe odaklanarak müstehcen görüntü ve videoların tespiti için derin öğrenme modeli önermişlerdir. Model, MobYOLov3 mimarisi üzerine inşa edilmiştir. Geliştirilen modül, giriş verileri içindeki önemli özellikleri seçici olarak artırırken, daha az önemli olanları bastırmıştır. Bu yaklaşım, modelin müstehcen içeriği tanımlamak için en uygun bilgilere odaklanmasını sağlamıştır. Yazarlar, müstehcen görüntü ve videolardan oluşan bir veri kümesi kullanarak modelin performansını değerlendirmiş ve %98 üzerinde doğruluk elde etmişlerdir. Ayrıca, modeli diğer güncel yaklaşımlarla karşılaştırmış ve hem doğruluk hem de verimlilik açısından üstünlüğünü göstermişlerdir.

Park vd. (2022), çalışmalarında seç izle sistemlerinde cinsel içerikler, şiddet unsurları ve uyuşturucu sahnelerini sansürlemeye yönelik bir çalışma yapmışlardır. Sansür içeriklerini kullanıcılara uyguladıkları anket sonuçlarına göre belirlemişlerdir. 16 kullanıcı ile yapılan çalışmalarında videolar üzerinde bu tür nesnelerin sansürlenmesi gerektiğini belirtmişler ve ne tür bir sansürleme tekniği kullanılması gerektiğini araştırmışlardır. Anket sonuçlarında insanların izlemek istemediği sahnenin içeriği ve istenen sansür yöntemi, sahnenin türüne göre farklılık göstermektedir. Bu yüzden videolar üzerinde bu nesneleri kullanıcı bazlı olarak gerçek zamanlı sansürleyen bir uygulamaya ihtiyaç duyduklarını belirtmişlerdir. Alkol tespiti ile ilgili çalışmalar incelendiğinde genel olarak bu çalışmaların deneklerden alınan kan değerleri, kalp atış hızları (P. Bernstein vd., 2017) ya da içeceklerden alınan örneklere göre alkol oranlarının tespiti ile ilgili olduğunu görmüşlerdir.

Yapılan çalışmalar incelendiğinde alkol ve sigara nesnesi tespitine ait yeterli çalışma bulunmamaktadır. Yapılan çalışmalar sürücülerin alkollü olup olmaması (Pain vd., 2023) ve ortamlarda sigara içildiğini tespit etmeye yarayan sensörlerle (Bahhar vd., 2023) alakalıdır. Resimlerde ve videolarda alkol, tabanca tespitini gerçekleştiren ve bunu API olarak sunan (Image Moderation Api Demo, 2021), (Realtime Image & Video Moderation API, 2021) uygulamalar mevcuttur. Yapılan çalışmalar genel olarak tek bir nesnenin tespitinde kullanılan yöntemlerin incelenmesi ve karşılaştırılması şeklindedir. Uygulamalar ise videodan zararlı içerik tespit edilip o sahnelerin atlanması şeklindedir (Feller ve Ventimiglia, 2020). Bu yaklaşım videoda anlamsal bozulmalara yol açabilir.

Tez çalışmasında, derin öğrenme teknikleri kullanılarak nesne tespiti alanında önemli bir sorun ele alınmıştır. İlk olarak mevcut eğitim modellerinin başarı oranını artırmaya yönelik bir öneri sunulmuştur. Önerilen yöntem, eğitim modellerinin yanlış tespit ettiği nesneleri veri setine dahil ederek yanlış-pozitif sayısını azaltmayı hedeflemektedir. Bu şekilde, eğitim modelinin başarı oranını arttırarak daha etkili bir tespit işlemi gerçekleştirmek amaçlanmıştır.

Ayrıca, klasik nesne tespit modellerindeki hız sorununu çözmek için boru hattı mimarisi nesne tespit mimarisine entegre edilmiştir. Bu yöntem sayesinde, nesne tespit adımlarının eş zamanlı olarak çalışması sağlanarak işlem hızı önemli ölçüde arttırılmıştır ve orijinal videoya yakın izleme süreleri elde edilmesi sağlanmıştır. Boru hattı mimarisini web kameraları üzerinde kullanarak gerçek zamanlı bir nesne tespit işlemi gerçekleştirmek de mümkün hale gelmiştir. Tek aşamalı sansür adını verdiğimiz bu yenilikçi yöntemde web kameralarından gelen görüntüler üzerinde herhangi bir donma veya ses kaybı olmadığı gözlemlenmiştir. Ancak, tek aşamalı sansür yöntemi videolar üzerinde titremelere ve donmalara neden olmuştur. Bu nedenle, daha gelişmiş bir yöntem olan iki aşamalı sansür yöntemi geliştirilmiştir. Bu yöntemde, ilk aşamada videolarda tüm kategorilere ait nesneler tespit edilmekte ve bu bilgiler meta dosyasına kaydedilmektedir. İkinci aşamada ise seçilen kategoriye ait bilgiler meta dosyasından okunarak videoda sansürleme işlemi gerçek zamanlı olarak gerçekleştirilmektedir. Boru hattı mimarisi ile geliştirilen iki yöntem görüntüler, web kameraları ve videolar üzerinde gerçek zamanlı nesne tespiti yapmayı mümkün kılmaktadır. Tez çalışmasında geliştirilen bu yöntemler, farklı kategorilerde ve platformlarda model bağımsız olarak yüksek doğrulukla gerçek zamanlı sansür yapabilme yetenekleriyle akademik alandaki boşlukları doldurarak açık ve net bir şekilde katkılarını sunmaktadır.

3. METARYAL VE YÖNTEM

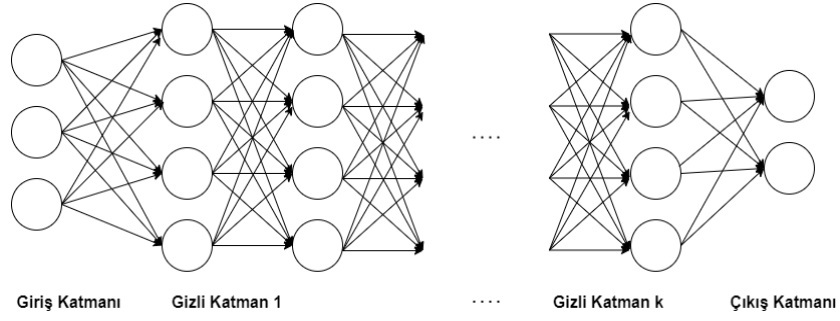
Son yıllarda teknolojinin ilerlemesi ile görüntü ve video paylaşımının yaygınlaşması anlamsız verilerin artışına neden olmuştur. Bu sorunu çözmek için verilerin analiz edilmesinde görüntü işleme algoritmaları kullanılır. Bu bölümde derin öğrenme hakkında detaylı bilgiler verilmiştir ve geliştirilen sistem ile videolardaki zararlı içeriğin gerçek zamanlı tespiti ile ilgili metodoloji sunulmuştur.

3.1. Nesne Tespitinde Kullanılan Yöntemler

Videolar ve görüntüler üzerinde plaka tespiti, faydalı bilgi seçimi, yüz tanıma, nesne tespiti gibi konularda görüntü işleme algoritmaları kullanılmaktadır (Luo vd., 2014). Veri miktarındaki bu artış ve karmaşıklaşmayla birlikte derin öğrenme yöntemleri de nesne tespitinde sıklıkla kullanılmaya başlanmıştır.

3.1.1. Derin öğrenme

Derin öğrenme yöntemleri, görüntüleri başarıyla işleyebilir ve büyük miktarda veriyi başarıyla yorumlayabilir. Statik olmayan görüntülerden nesne tespiti ve takibi yapmak kolay bir işlem değildir. Derin öğrenme, yüksek performansla nesne algılama-izleme gerçekleştirmek için çeşitli yöntemler içerir. Tanımlanmış bir nesnenin konumu, şekli, hızı, rengi gibi özellikleri kullanılarak, görüntü ve videolarda ilgilenilen nesne tespit edilip izlenebilir (Wang, 1998) ve nesne takibi yapılabilir. Derin öğrenme; sesten metne dönüştürme, görüntü analizi, görüntü sınıflandırma, doğrusal olmayan katmanlar kullanarak veri işleme vb. işlemler için kullanılmaktadır. Şekil 3.1'de gösterildiği gibi derin öğrenme giriş katmanı, gizli katmanlar ve çıkış katmanından oluşur. Her katmanın çıktısı bir sonraki katmanın girdisini oluşturur (Deng ve Yu, 2013). Derin öğrenme ağ mimarisi Şekil 3.1'de gösterilmektedir.



Şekil 3.1. Derin öğrenme ağ mimarisi (Amidi ve Amidi, 2020)

Microsoft, Google, Apple ve Facebook gibi büyük yazılım ve teknoloji şirketleri, ürünlerinde derin öğrenmeyi kullanmakta (CBINSIGHTS, 2019) ve kullandıkları teknikleri paylaşarak alanında ilerlemek isteyen kullanıcılara olanaklar sağlanmaktadır.

3.1.2. Derin öğrenmenin uygulama alanları

Derin öğrenme, verinin olduğu doğal dil işleme, görüntü işleme, biyoinformatik, sinyal işleme, sanat, metin tanıma, bilgisayar oyunları, robotlar, video işleme, sürücüsüz arabalar, ses, duygu analizi, dijital pazarlama, nesne algılama gibi birçok popüler alanda uygulanmaktadır.

Video, görüntü ve sinyal işleme alanında Sun ve ark. (2017), görüntü işlemeli BT (Bilgisayarlı Tomografi) görüntülemesinden akciğer nodüllerini tespit etmek için otomatik olarak oluşturulan modelin algılama performansını, manuel olarak tanımlanan özellikleri kullanan geleneksel yöntemlerle karşılaştırmışlardır. Akciğer nodüllerinin tespitinde istenilen performansı elde ettiklerini iddia etmişlerdir.

Sorin vd. (2020), çalışmalarında radyoloji alanında doğal dil işleme çalışmalarını gözden geçirmişlerdir. Kırık ve pulmoner emboli gibi etiketleme tanımlarının doğal dil işleme ile görüntü protokollerinin otomatik seçiminde iyi performans elde ettiklerini ifade etmişlerdir. Havaei ve ark. (2017), sağlık alanında derin öğrenme yöntemlerini inceleyerek beyin tümörlerini

segmentlere ayırma yöntemi önermişlerdir ve CNN tabanlı uygulamaların iyi sonuçlar verdiğini belirtmişlerdir.

Robotik alanında, Mikada ve ark. (2020), YOLOv3 ve CNN kullanarak, cerrahi bir robotun iğne yerleştirme ve geri çekmeyi öngördüğü bir metot önermişlerdir. İnsan iş birliği ile yapılan deneylerin sonuçları, önerilen yöntemin etkinliğini göstermiştir. Kim (2020), yapay sinir ağı tabanlı yüz algılama, konuşma ve metin tanıma modelinin patentini almıştır. Prasad vd. (2019), VGGFace kullanarak kusurlu yüzler, farklı açıdan çekilen pozlar gibi durumlarda yüz tanımda başarılı sonuçlar elde ettiklerini belirtmişlerdir.

3.1.3. Derin öğrenme mimari yapısı

Derin öğrenme işleminin başarılı ve kaliteli sonuçlar üretmesi için büyük verilerle eğitilmesi gerekmektedir. Derin öğrenme ve makine öğrenimi arasındaki fark, derin öğrenmede oluşturulan ağın birden fazla sinir ağı katmanına sahip olmasıdır. Öğrenme aşaması sınıflardan oluşur. Alt katmanların özellikleri daha az belirginken, alt katmanların birleştirilmesiyle oluşturulan üst katmanların özellikleri daha belirgindir. Kullanılan sinir ağı modeli problem durumuna göre değişiklik göstermektedir. Bu bölümde derin öğrenmede kullanılan sinir ağları ve yöntemler açıklanmıştır.

Denetimli sinir ağları;

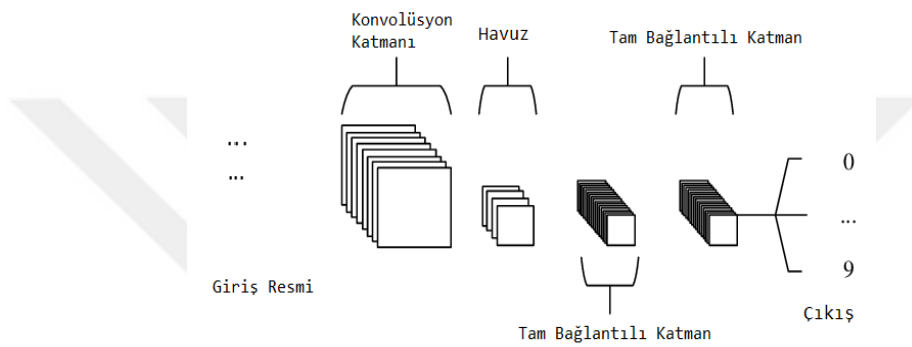
- CNN (Convolutional Neural Networks, Konvolüsyonel Sinir Ağları) ve
- RNN (Recurrent Neural Network, Tekrarlayan Sinir Ağı)'dir.

Denetimsiz sinir ağları;

- RBM (Restricted Boltzmann Machine, Kısıtlı Boltzmann Makineleri),
- LSTM (Long Short-Term Memory Networks, Uzun Kısa Vadeli Hafıza Ağları),
- DAE (Denoising Autoencoder, Derin Otomatik Kodlayıcılar) ve
- DBN (Deep Belief Network, Derin İnanç Ağı)'dir.

3.1.3.1. Konvolüsyonel sinir ağı

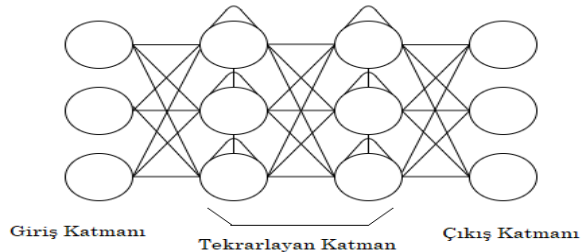
Konvolüsyonel sinir ağı elle özellik çıkarma ihtiyacını ortadan kaldırarak verilen verilerden öğrenen denetimli bir sinir ağıdır (O'Shea ve Nash, 2015). Nesne tanıma, sınıflandırma işlemleri için yeniden eğitilerek önceden oluşturulan ağların kullanılmasına olanak tanır. GoogleNet, AlexNet (Alex Network, Alex Ağı), VGG, Clarifai, CNN mimarisine dayalı olarak geliştirilen yöntemler örnek olarak alınabilir. Şekil 3.2'de konvolüsyonel sinir ağına ait mimari gösterilmiştir.



Şekil 3.2. Konvolüsyonel sinir ağı mimarisi (O'Shea ve Nash, 2015)

3.1.3.2. Tekrarlayan/yineleyen sinir ağı

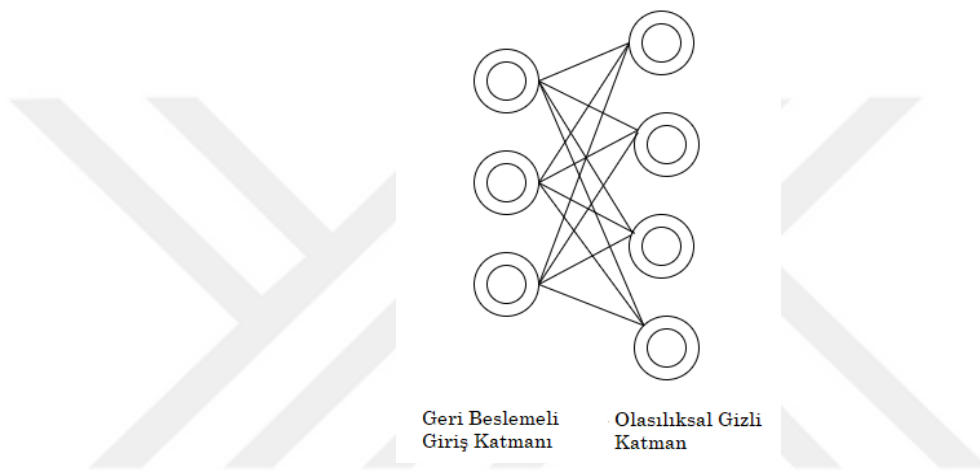
Tekrarlayan sinir ağı sıralı verileri kullanan denetimli bir sinir ağıdır (Medsker ve Jain, 2001). CNN'ler gibi eğitim verileri kullanılarak öğrenme gerçekleştirir. Uzun süreli kısa süreli bellek, iki yönlü yinelenen sinir ağı, kapalı yinelenen hücre, iki yönlü uzun süreli kısa süreli bellek, nöronları dolaşan ağ mimarisine dayalı olarak geliştirilen yöntemlere örnek olarak verilebilir. Şekil 3.3'te tekrarlayan sinir ağına ait mimari gösterilmiştir.



Şekil 3.3. Tekrarlayan sinir ağı mimarisi (Leijnen ve Veen, 2020)

3.1.3.3. Sınırlı boltzmann makineleri

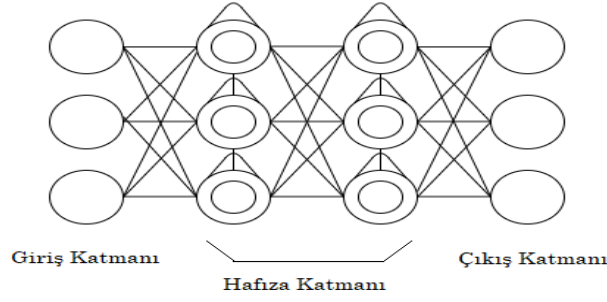
Sınırlı boltzmann makineleri katman iç bağlantısı olmayan, tekrarlayan, denetimli ve denetimsiz öğrenen bir sinir ağıdır (Shetty vd., 2020). Regresyon, sınıflandırma, boyutluluk azaltma, özellik öğrenme gibi alanlarda kullanılır. DBN, BM (Boltzmann Machines, Boltzman Makinesi), RBM mimarisine dayanan yöntemlere örnek olarak alınabilir. Şekil 3.4'te sınırlı boltzmann makinelerine ait mimari gösterilmiştir.



Şekil 3.4. Sınırlı boltzmann makineleri sinir ağı mimarisi (Leijnen ve Veen, 2020)

3.1.3.4. Uzun kısa vadeli hafıza ağları

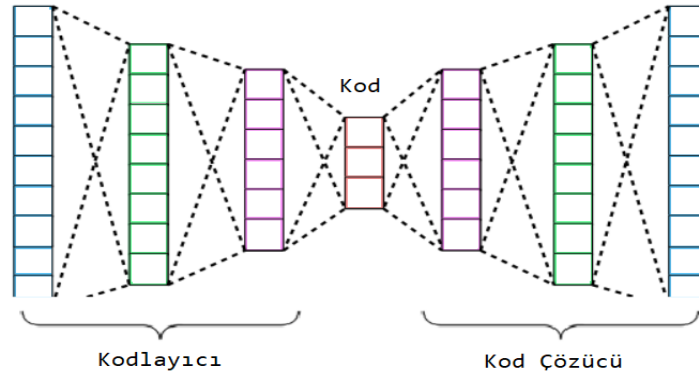
Uzun kısa vadeli hafıza ağları dil modelleme, resim alt yazısı, el yazısı üretme gibi alanlarda kullanılan bir tür tekrarlayan denetimsiz sinir ağı mimarisidir (Shetty vd., 2020). Amaç tekrarlayan sinir ağlarında kaybolan gradyanlar nedeniyle bu sorunu çözerek uzun vadeli bağımlılıkları hatırlamak ve bilginin kalıcı olmasını sağlamaktır (Hochreiter ve Schmidhuber, 1997). Şekil 3.5'te uzun kısa vadeli hafıza ağlarına ait mimari gösterilmiştir.



Şekil 3.5. Uzun kısa vadeli hafıza ağı mimarisi (Leijnen ve Veen, 2020)

3.1.3.5. Derin oto-kodlayıcılar

Derin oto-kodlayıcılar çıktı katmanındaki girdiyi yeniden üretmek için kullanılan ileri beslemeli denetimsiz bir sinir ağı mimarisidir (Şeker vd., 2017). Çıktı katmanındaki nöron sayısı, girdi katmanındaki nöron sayısı ile aynıdır. DAE mimarisine dayalı olarak geliştirilen yöntemlere örnek olarak seyrek otomatik kodlayıcı, gürültü giderme otomatik kodlayıcı, varyant otomatik kodlayıcı verilebilir. Şekil 3.6'da derin oto-kodlayıcılara ait mimari gösterilmiştir.

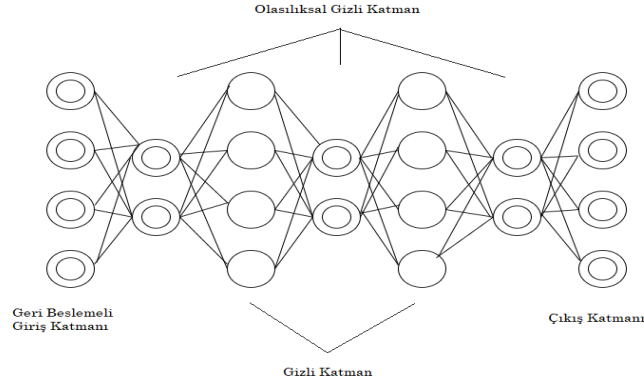


Şekil 3.6. Derin oto-kodlayıcı sinir ağı mimarisi (Liu vd., 2017)

3.1.3.6. Derin inanç ağları

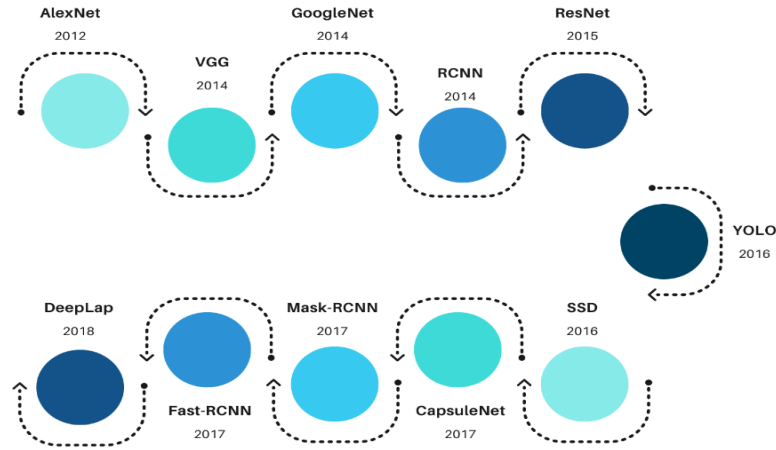
Derin inanç ağları derin sinir ağlarından farklı olarak birçok gizli katmana sahip ileri beslemeli bir denetimsiz sinir ağı modelidir. Katmanlar ön ve arka katmanlarla ilişkilendirilir ve düğümler arasında yatay iletişim yoktur. Son katmanda ise öğrenmeli kümeleme ve softmax sınıfı ile denetimsiz sınıflandırma yapılabilmektedir (Hinton vd., 2006). Softmax sınıfı, birçok sınıfın

bulunduğu sınıflandırma problemlerinde, çıktı katmanında kullanılan bir aktivasyon fonksiyonudur. Bu fonksiyon, girdi olarak bir vektör alır ve bu vektörü sınıf olasılıklarına dönüştürür. Şekil 3.7’de derin inanç ağlarına ait mimari gösterilmiştir.



Şekil 3.7. Derin inanç ağları mimarisi (Leijnen ve Veen, 2020)

Derin öğrenme mimarilerinin zaman içerisindeki gelişimi Şekil 3.8’de gösterilmiştir.



Şekil 3.8. Derin öğrenme mimarilerinin zamansal gelişimi (Saleem vd., 2019)

3.1.4. Derin öğrenme alanında kullanılan diller/kütüphaneler

Bu alanda derin öğrenme ile yapılan çalışmalarda kullanılan diller ve kütüphaneler hakkında bilgiler verilmiştir. Çizelge 3.1’de bu bilgiler hakkındaki detaylar sunulmuştur.

Çizelge 3.1. Derin öğrenmede kullanılan diller/kütüphaneler

Kütüphane	Dil	Yıl	Açıklama	GitHub Yıldız Sayısı	GPU Çalışma Zamanı Performansı (sn)
PyTorch (Collobert vd., 2017)	Python	2017	GPU (Graphics Processing Unit, Grafik işlemci birimi) desteği, hız ve esneklik	45k	2.88
Tensorflow (TensorFlow, 2020)	Python, C++	2020	Otomatik resim yazısı oluşturma yazılımı, tek bir API ile birden fazla GPU ve CPU (Central Process Unit, Merkezi İşlem Birimi)'ya dağıtma olanağı	152k	5.57
Keras (Keras: The Python Deep Learning API, 2020)	Python	2020	Aktivasyon fonksiyonları ve optimize ediciler	50.4k	-
Theano (Theano Development Team, 2016)	Python	2016	CPU ve GPU üzerinde verimli bir şekilde çalışır, matematiksel hesaplamalar kolay	9.3k	1.40
Caffe (Jia vd., 2014)	Python, C++	2016	Görüntü sınıflandırma ve görüntü bölümlleme, GPU desteği	31.2k	2.45

Çizelge 3.1'e göre GPU çalışma zamanı en hızlı Theano, en yavaş TensorFlow'dur (Yuret, 2016). Kullanım oranına bakıldığında ise Github'da yıldız derecelendirmesine göre en çok kullanılan Tensorflow olmuştur.

3.1.5. Derin öğrenme ile nesne tespiti ve nesne takibi

Görüntülerden analiz ve tespit işlemi yapmak için görüntü işleme yöntemleri kullanılır. Viola Jones (Viola ve Jones, 2001), görüntülerdeki nesneleri hızlı bir şekilde tespit eden ilk algoritmadır. Eğer takip edilecek bir nesne varsa yani videolar gibi durağan olmayan görüntülerden analiz ve tespit işlemi yapılacaksa derin öğrenme algoritmaları sıklıkla kullanılmaktadır. Derin öğrenme ile nesne tespiti ve takibi için genel olarak uygulanan adımlar sırasıyla;

- Ön işleme,
- Nesnenin tespiti,
- Nesnenin sınıflandırılması ve
- Nesnenin takibidir.

3.1.5.1. Ön işleme

Ön işleme aşaması tanımlanmış kategorilere göre hazırlanan verilerin analiz edilmeye hazır hale getirilmesi için yapılan işlemlerdir. Ön işleme, verileri daha fazla analiz için hazırlayan faaliyetleri ve sistem arızalarını düzeltmeye veya telafi etmeye çalışan adımları içermektedir. Makine öğrenimi ve yapay zekâ geliştirme hattının ilk aşamalarındandır. Veri profilini çıkarma, veri temizleme, veri azaltma, veri doğrulama adımları uygulanarak ön işleme aşaması gerçekleştirilir.

3.1.5.2. Nesne tespiti

Nesne tespiti, görüntüler veya videolarda belirli bir sınıftaki nesnelerin örneklerinin algılaması işlemidir. Nesne tespiti iki adımdan oluşur. Bunlar tek aşamalı yöntemler ve iki aşamalı yöntemlerdir. Tek aşamalı yöntemler tespit hızına öncelik verir. YOLO, SSD tek aşamalı yöntemlere örnek gösterilebilir. İki aşamalı yöntemler ise algılama doğruluğuna öncelik verir. Faster R-CNN, Mask R-CNN ve R-CNN (Region Based Convolutional Networks, Bölge-Tabanlı Evrişimli Sinir Ağı) iki aşamalı yöntemlere örnek gösterilebilir.

R-CNN, nesne algılamada ilk önce bölge teklifleri üretir. Teklif bölgeleri görüntüden kırılır ve yeniden boyutlandırılır. Ardından, CNN bu bölgeleri sınıflandırır (Girshick vd., 2016). Son olarak, bölge teklif sınırlama kutuları, CNN özellikleri kullanılarak eğitilmiş bir SVM kullanılarak sınıflandırma işlemi gerçekleştirilir (Daş ve diğerleri, 2019).

Fast R-CNN (Fast Region-Based Convolutional Neural Network, Hızlı Bölge-Tabanlı Evrişimli Sinir Ağı), nesne algılamada bölge önerileri oluşturur. Önerilen bölge önerilerini kırıp yeniden boyutlandıran R-CNN dedektörünün aksine, Fast R-CNN dedektörü tüm görüntüyü işlemektedir. R-CNN dedektörü her bölgeyi sınıflandırmak zorundayken, Fast R-CNN, her bölge önerisine karşılık gelen CNN özelliklerini bir havuzda toplar. Fast R-CNN, R-CNN'den daha verimlidir (Girshick, 2015). Sınıflandırma ise, softmax sınıfı kullanılarak gerçekleştirilir.

Faster R-CNN, nesne algılamada doğrudan ağda bölge teklifleri oluşturmak için bir RPN (Region Proposal Network, Bölge Öneri Ağı) ekler (Ren vd., 2016). RPN, nesne algılama için bağlantı kutuları kullanır. Ağda bölge teklifleri oluşturmak daha hızlıdır ve verilerinize göre daha iyi ayarlanmıştır. Sınıflandırma aşamasında bir kez giriş verilerindeki görüntüyü kullanarak nesnenin konumu hesaplanır. Paylaşılan evrişim katmanları ve derin konvolüsyonel bir ağ, nesne algılama performansını iyileştirmek için eğitilebilir (Cömert ve diğerleri, 2019).

Mask R-CNN, alan nesne algılama algoritmaları arasında en gelişmişidir. Görüntüdeki nesneleri verimli bir şekilde tespit ederken, nesnenin bulunduğu tüm pikselleri tanımlar (Omeroglu ve diğerleri, 2019) ve her örnek için yüksek kaliteli bir bölütleme maskesi oluşturur. Nesne tanıma için Faster R-CNN kullanmaktadır.

SSD, gerçek zamanlı olarak nesne tespiti için tasarlanmıştır. Faster R-CNN, sınır kutuları oluşturmak için bir bölge öneri ağı sunar ve bu kutuları nesneleri sınıflandırmak için kullanır. SSD ise nesne tanımayı aynı anda tek bir derin sinir

ağı kullanarak gerçekleştirir ve tespit hızı Faster R-CNN'den yüksektir (Liu ve diğerleri, 2016).

YOLO, tek bir sinir ağı kullanarak sorunu baştan sona optimize eder ve algılama performansına göre nesneleri doğrudan algılar (Redmon ve diğerleri, 2016). YOLO hızı, öğrenme yeteneği ve doğruluğu nedeniyle popülerdir. Trafik sinyallerini, insanları, parkmetreleri ve hayvanları tespit etmek gibi birçok farklı uygulamalarda kullanılmaktadır.

3.1.5.3. Nesne sınıflandırma

Sınıflandırma, özniteliklerine göre grup nesnelere uygulanır. Sınıflandırma işlemi yapılırken nesnenin benzersiz nitelikleri kullanılır (Hanbay ve Üzen, 2017). Renk, doku, gradyan, hareket, zaman nesne sınıflandırmasında en sık kullanılan niteliklerdir. Renk sınıflandırması etkili bir yöntemdir ancak rengin ışık duyarlılığının hesaplama süresine etkisi göz önüne alınmalıdır.

Doku da renk gibi sınıflandırmada etkili bir niteliktir ancak hesaplama süresi yüksektir. Hesaplama süresi zaman içinde birçok hareket ve değişime sahip olduğundan maliyeti yüksektir. Çizelge 3.2'de sınıflandırma yöntemlerinin niteliklerine göre maliyet ve başarı oranları hakkında bilgiler verilmiştir.

Çizelge 3.2. Sınıflandırma için kullanılan özniteliklerin kıyaslanması

Sınıflandırma Yöntemi	Maliyet	Başarı Oranı	Yorum
Şekil Tabanlı	Düşük	Orta	Işık duyarlılığı fazla
Hareket Tabanlı	Yüksek	Orta	Durağan olmayan görüntülerde nesne tespitinde yüksek başarı
Renk Tabanlı	Yüksek	Yüksek	Işık duyarlılığı fazla
Doku Tabanlı	Yüksek	Yüksek	Hesaplama süresi fazla

3.1.5.4. Nesne takibi

Nesne takibi, bir dizi görüntü veya video çerçevesinde belirli bir nesneyi veya birden fazla nesneyi bulma ve takip etme sürecidir. Bir videodaki nesneleri otomatik olarak tanımlama ve bunları yüksek doğrulukla diğer karelerde de yorumlama görevidir. Çoğu zaman, izlenen nesnenin etrafında bir gösterge bulunur; örneğin, nesneyi takip eden çevreleyen kare, kullanıcıya nesnenin ekranda nerede olduğunu gösterir. Kullanılan nesne izleme yöntemleri, kullanımı ve maliyet-başarı oranları Çizelge 3.3'te sunulmuştur. Bu yöntemler uygulama ve takip edilen nesnelerin niteliğine bağlı olarak karmaşıklıkta farklılık gösterebilir.

Çizelge 3.3. Nesne takibi yöntemleri (Hanbay ve Üzen, 2017)

Nesne Takip Yöntemi	Kullanımı	İşlem Zamanı	Başarı
Nokta tabanlı	Deterministik model, Olasılık modeli	Düşük-Orta Orta-Yüksek	Orta Orta
Çekirdek tabanlı	Çoklu görünüme dayalı, Şablon tabanlı	Düşük-Orta Orta	Düşük Orta-Yüksek
Siluet tabanlı	Kontur gelişimi, Şekil eşleştirme	Orta Yüksek	Orta-Yüksek Yüksek

Nesne takibi, bir videodaki nesneleri tanımlamak ve bunları yüksek doğrulukla bir dizi yörüngede izlemek için kullanılır. Nesne takibindeki asıl zorluk, hesaplama verimliliği ve performans arasında bir denge kurmaktır. Işık hassasiyetinin olmadığı nesnelerin takibinde nesneler nokta olarak gösterilir ve nokta tabanlı yöntem kullanılır. Nesne sadece ilk sahnede tespit edilip ardından kareler ve dikdörtgenler gibi şekiller kullanarak nesne takip edilecekse çekirdek tabanlı yöntemler kullanılmaktadır. Katı olmayan nesneler için, algılanan çerçevede arka planın görünmesi sorunu ortaya çıkabilir. Eğer nesne takibi nesnenin silueti kullanılarak yapılacaksa siluet tabanlı yöntem tercih edilir. Sınır tabanlı (Ojha ve Sakhare, 2015), kontur tabanlı (Jacob ve Anitha, 2012) ve bölge tabanlı izleme (Deori ve Meitei, 2014) nesne izlemede kullanılan diğer yöntemlere örnek olarak verilebilir.

3.1.5.5. Nesne tespiti ve takibi veri setleri

Geliştirilecek uygulamada tespit ve takip edilecek nesnenin özelliklerine göre kullanılması gereken veri kümesi değişiklik göstermektedir. Küçük nesnelerin tespiti ve takibinde sıklıkla Caltech101/256 (Griffin vd., 2007), TinyImage (Torralba vd., 2008), PASCAL (Pattern Analysis, Statistical Modelling and Computational Learning, Desen Analizi, İstatistiksel Modelleme ve Hesaplamalı Öğrenme) (Everingham vd., 2008), MSRC (Microsoft Research Cambridge) (Shotton vd., 2006) veri kümeleri kullanılmaktadır. Bunlara ek olarak nesnelerin kümelenmesinde, sınıflandırılmasında ImageNet (Deng vd., 2009) veri kümesi kullanılabilir.

ESP (Extended SP, Genişletilmiş SP) (Ahn ve Dabbish, 2004), paylaşımına açık olmayan ve çevrimiçi oyunlarda kullanıcıların etiketlediği resimlerden oluşan veri setidir. Nesnelerin tespiti ve takibinde kullanılan ve nesnelere ait etiketleri, konum bilgilerini ve renk detaylarını içeren Lotus Hill ve LabelMe (Russell vd., 2008; Yao vd., 2007) yaklaşık 636.748'den fazla görüntü ve video karesinden oluşmaktadır.

PASCAL-VOC (PASCAL Visual Object Classes, PASCAL Görsel Nesne Sınıfları) (Everingham vd., 2010), MS-COCO (Microsoft Common Objects in Context, Microsoft Ortak Bağlamdaki Nesneler) (Lin vd., 2014), KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute at Chicago, Karlsruhe Teknoloji Enstitüsü ve Toyota Teknolojik Enstitüsü'nün Ortak Nesneleri) (Geiger vd., 2012) şiddet unsurları gibi küçük nesneleri içeren veri kümelerindendir. MS-COCO nesne, kategori algılama uygulamaları için 91 kategori ve 328 binden fazla görüntü içermektedir. Aynı zamanda 2,5 milyon etiketli verilerden oluşmaktadır. KITTI, nesne algılama uygulamalarında kullanılan 7481 eğitim görüntüsü içermektedir.

3.2. Zararlı İçerik Tespiti Sisteminin Uygulanması

İş ve eğitim hayatında zararlı unsurlar içeren videolarla karşılaşmak kolay ve sıkıntılı bir durum haline gelmiştir. Tüm bu durumlardan dolayı video paylaşım platformlarının kendi politikalarına ek olarak internet tabanlı tüm videolarda belirlenen kötü unsurları anlık olarak tespit eden ve filtreleyen bir sisteme ihtiyaç duyulmaktadır.

Tez çalışmasında videodaki eylemleri, nesneleri tanımak için eğitilmiş modeller kullanılarak yeni bir sistem geliştirilmiştir. Geliştirilen sistem sadece çocukların güvenliği için değil, iş hayatında ya da eğitim hayatında da kolaylıkla kullanılabilir. Sonuç olarak video akışının olduğu her alanda meydana gelebilecek zararlı içeriklerin engellenmesi amaçlanmaktadır. İnternet üzerindeki videolarda hangi içeriklerin filtreleneceği yapılan anket çalışması ile belirlenmiştir.

3.2.1. Anket sorularının hazırlanması ve deney sonuçlarına göre kategorilerin belirlenmesi

Bu alanda kullanıcılara zararlı içeriklerin tespitine yönelik anket uygulanmıştır. Toplam 9 sorudan oluşan ankete 688 kişi katılım sağlamıştır. Her bir soru ayrı ayrı analiz edilerek tez çalışmasında kullanılacak içerik tespiti gerçekleştirilmiştir.

Şekil A.1’de bu ankete katılım oranına bakıldığında %50 oranı ile erkek katılımcı sayısının daha fazla olduğu görülmüştür. Şekil A.2’de bu ankete katılan kişilerin yaş aralığına bakıldığında en büyük oranın %38.8 ile 36-50 yaş aralığı olduğu görülmüştür. Şekil A.3’te ankete katılan kişilerin meslek dağılımlarına bakıldığında en büyük oranın %35.6 ile memur olduğu görülmüştür. Şekil A.4’te ankete %69.5 oranı ile evli bireylerin daha fazla katılım sağladığı görülmüştür. Şekil A.5’te ankete katılan kişilerin %67.2’sinin çocuk sahibi olduğu görülmüştür. Şekil A.6’da ankete katılan kişilerin %86.9’u internet üzerinde izlediği videolarda kötü unsurları görmekten rahatsız olmaktadır.

Diğer cevaplar kısmındaki cevaplara bakıldığında ise çoğunluğun sadece şiddet ve uygunsuz görüntülerden rahatsız olduğu görülmüştür. Şekil A.7'ye göre ankete katılan kişilerin %92.2'si zararlı içeriklerin izlenen videolarda tespit edilip filtrelenmesini istemektedir. Diğer cevaplar kısmındaki yorumlar incelendiğinde özellikle çocuğu olan bireylerin kendi onayı ile bu uygulamanın zararlı içerikleri tespit etmesinin daha uygun olduğunu ifade etmişlerdir. Bu ankete katılan kullanıcıların Alkol, sigara, açık içerik ve şiddet barındıran içerikler için verdiği oylara göre kategorilerin oranları Şekil A.8'de listelenmiştir. 1-5 arası (1: En az, 5: En fazla) derecelendirmelerine göre internet üzerinde izlenen videolarda görmekten rahatsız oldukları unsurlarda en büyük oranı “Şiddet Barındıran İçerikler” almıştır. Tez çalışmasında alkol, sigara, açık içerik ve şiddet barındıran içerikler için nesne tanıma işlemi gerçekleştirilecektir.

3.2.2. Veri setlerinin oluşturulması

Anketten elde edilen sonuçlara göre “Şiddet İçeren Unsurlar”, “Sigara”, “Açık İçerik” ve “Alkol” kategorilerindeki içeriklerinin tespitlerinin yapılmasına karar verilmiştir. Her kategori için %80 eğitim ve %20 doğrulama veri seti oluşturulmuştur. Şiddet içeriklerinin tespitinde tehlikeli unsurlara göre veri setleri oluşturulmuştur. Model, videolarda bulunan tüm giyilebilir veya kişisel tehlikeli unsurları algılayacak şekilde tasarlanmıştır. Bu unsurlar şunları içerir:

- Tüfekler (ari-dasci/OD-WeaponDetection, 2021, Google Images, 2021),
- Tabancalar (ari-dasci/OD-WeaponDetection, 2021, Pistols Dataset, 2021, Google Images, 2021),
- Potansiyel silah olan veya şiddet içeren aletler: bıçak (ari-dasci/OD-WeaponDetection, 2021).

Çizelge 3.4'te şiddet unsuru veri türleri ve sayıları gösterilmiştir.

Çizelge 3.4. Şiddet unsurlarına ait eğitim ve test veri sayısı

Veri Türü	Eğitim Verisi	Doğrulama Verisi	Test Verisi Video
Tüfek	707	180	1
Tabanca	1425	356	1
Potansiyel silah olan veya şiddet içeren aletler: Bıçak	1825	444	1
Toplam	3957	980	3

Benzer çalışmalar incelendiğinde oluşturulan modellerin genellikle gerçek veriler üzerinde denendiği görülmüştür. Tez çalışmasında oluşturulan eğitim modelinin sentetik verilerde de aynı doğruluk oranında çalışıp çalışmayacağının tespiti için tabanca verisine ait sentetik veri seti (Giddens, 2021) oluşturulmuştur. Böylece hem oyun, çizgi film gibi sentetik veriler içeren hem de gerçek nesneler içeren videolar üzerinde başarılı bir şekilde nesne tespiti yapmak planlanmıştır. Çizelge 3.5'te tabanca veri türüne ait çeşitli oyun sahnelerinden toplanan sentetik veri sayıları gösterilmektedir.

Çizelge 3.5. Sentetik tabanca eğitim ve test veri sayıları

Veri Türü	Eğitim Verisi	Doğrulama Verisi	Toplam	Test Verisi Video
Tabanca	1000	250	1250	1

Sigara tespiti için çeşitli kaynaklardan toplanan verilerle sigara (Lee vd., 2019, Khan, 2020) nesnesine ait veri seti oluşturulmuştur. Çizelge 3.6'da sigara nesnesine ait veri sayıları gösterilmektedir.

Çizelge 3.6. Sigara nesnesine ait eğitim ve test verileri sayısı

Veri Türü	Eğitim Verisi	Doğrulama Verisi	Toplam	Test Verisi Video
Sigara	2379	590	2969	5

Alkol (Alcohol Pictures, 2021) nesnesine ait oluşturulan veri seti bilgileri Çizelge 3.7'de gösterilmiştir.

Çizelge 3.7. Alkol nesnesine ait eğitim ve test verileri sayısı

Veri Türü	Eğitim Verisi	Doğrulama Verisi	Test Verisi Video
Bira	503	125	1
Şişe	574	143	1
Viski	209	52	1
Kokteyl	604	151	1
Şarap-Şampanya	382	100	1
Toplam	2272	571	5

Açık içeriklere sahip görüntülere ait veri seti GitHub (Bazarov, 2019) üzerinde uygunsuz görüntülerin linklerini içeren dosyalardan oluşturulan bir script aracılığı ile toplanmıştır. Bu veri seti 3204 doğrulama, 801 eğitim veri setinden oluşmaktadır.

3.2.3. Algoritmaların testi ve performanslarının karşılaştırılması

Videolar üzerinde gerçek zamanlı nesne tespit yöntemlerinin kullanılması özellikle ağ bağlantılı sistemlerde yavaşlığa neden olmaktadır. Tez çalışmasında hem zararlı içeriklerin yüksek doğrulukla tespitini sağlayan hem de kullanılan nesne tespit modellerin meydana getirdiği yavaşlığı iyileştirmek için bir yöntem önerilmiştir. Çeşitli kategorilerde toplanan veri setleri etiketlenerek eğitim modelleri oluşturulmuştur. Oluşturulan eğitim modellerinin en iyi sonucu veren dosyası ONNX (Open Neural Network Exchange, Açık Yapay Sinir Ağı Değişimi) formatına dönüştürülerek Asp.Net Core ile geliştirilen uygulamada kullanılmıştır.

Derin öğrenme teknikleri kullanılarak geliştirilen bu uygulama aşağıda belirtilen süreçleri içermektedir:

1. Veri setinin hazırlanması
2. Verilerin etiketlenmesi
3. Modelin eğitilmesi
4. Modelin değerlendirilmesi ve iyileştirilmesi
5. Uygulamanın geliştirilmesi ve iyileştirilmesi

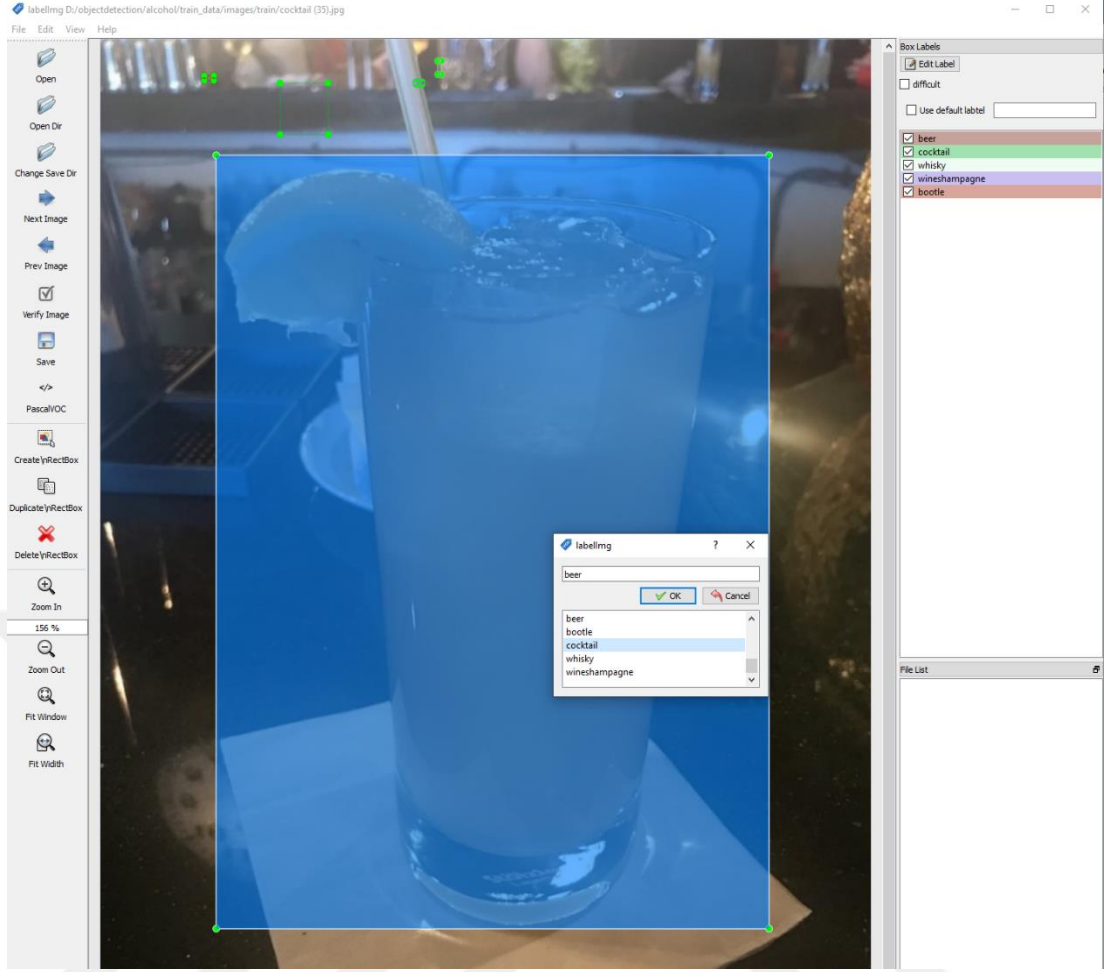
Sigara nesnesi tespiti için eğitim modelinin oluşturulması aşamasında veri seti hazırlanırken görüntülerde bulunan nesneleri etiketlemek için LabelImg (tztalin/labelImg) kullanılmıştır. Sigara nesnesi tespiti için sırasıyla YOLOv5, Faster R-CNN ve SSD modelleri denenmiş ve en yüksek başarı oranına sahip olan model tez çalışması kapsamında kullanılmıştır.

Sigara nesnesinin tespiti için:

- Parametre ve etiket bilgisi: Parametre sayısı 1 ve etiketi “**cigarette**” şeklinde oluşturulmuştur.

Eğitim modelleri için YOLO kullanılacaksa nesnelerin koordinat ve sınıf sırası bilgilerini gösteren dosya txt (Text File, Metin Dosyası) uzantılı, Faster R-CNN ve SSD kullanılacaksa xml (Extensible Markup Language, Genişletilebilir İşaretleme Dili) uzantılı olmalıdır. Bu dosyalar ile eğitim ve test verilerine ait kayıt dosyaları oluşturulacak ve eğitim aşamasında kullanılacaktır. LabelImg ile etiketleme işlemi yapılırken mutlaka format belirtilmelidir.

Şekil 3.9’da LabelImg ile etiketlemenin nasıl yapıldığını gösteren bir örnek verilmiştir. Etiketlenen veriler eğitim aşamasına hazır hale getirilmek için ön işlemlerden geçirilmektedir. Her kategori için 80 eğitim, 20 doğrulama verisi ayrılmıştır. Şekil 3.12’de gösterildiği gibi deneme-yanılma yoluyla hiperparametre ayarları yapılarak en iyi sonucu veren değerler toplam veri seti üzerine uygulanmıştır. Hiperparametre değerleri veri seti boyutu, model karmaşıklığı, donanım kaynakları ve hedeflenen performans faktörlerine bağlıdır.



Şekil 3.9. LabelImg ile örnek nesne etiketleme görüntüsü

YOLOv5 (ultralytics/yolov5, 2021), Faster R-CNN (tensorflow-gardener, 2021) ve SSD (tensorflow-gardener, 2021) ile kendi özel veri setimize ait eğitim modelleri Google Colab (Google Colab, 2021) kullanılarak oluşturulmuştur. Google hesabı ile Google Colab 12 saat boyunca ücretsiz bir GPU sağlamaktadır. Google Colab, Cuda (Compute Unified Device Architecture, Birleşik Hesaplama Mimarisi), Torch ve diğer bazı bağımlılıklar ile önceden yüklenmiş olarak gelmektedir. Hangi model eğitim modeli için kullanılacaksa ihtiyaç duyduğu bağımlılıkları ek olarak yüklenmelidir.

Eğitim modelinde kullanılacak dosyada eğitim ve doğrulama verilerinin yolu, sınıf sayısı ve etiket bilgileri yer almaktadır. Sigara nesnesine ait örnek YAML (YAML Ain't Markup Language, YAML İşaretleme Dili Değildir) dosyası Şekil 3.10'da, ptxt dosyası Şekil 3.11'de gösterilmiştir.

```
# train and val data as 1) directory: path/images/, 2) file:
path/images.txt, or 3) list: [path1/images/, path2/images/]
train: ../train_data/images/train/
val: ../train_data/images/val/ |

# number of classes
nc: 1

# class names
names: [ 'cigarette' ]
```

Şekil 3. 10. Sigara nesnesine ait örnek YAML dosyası

```
item {
    id: 1
    name: 'cigarette'
}
```

Şekil 3. 11. Sigara nesnesine ait örnek ptxt dosyası

Eğitim aşamasında sırasıyla YOLOv5, Faster R-CNN ve SSD kullanılmıştır. YOLOv5 için model eğitilirken YAML dosyası ve kullanılacak model belirtilir. Şekil 3.12’de sigara nesnesi için oluşturduğumuz custom_data.yaml (Şekil 3.10) kullanılarak YOLOv5 ile eğitim modelinin oluşturulması için kullanılan kod örneği gösterilmiştir.

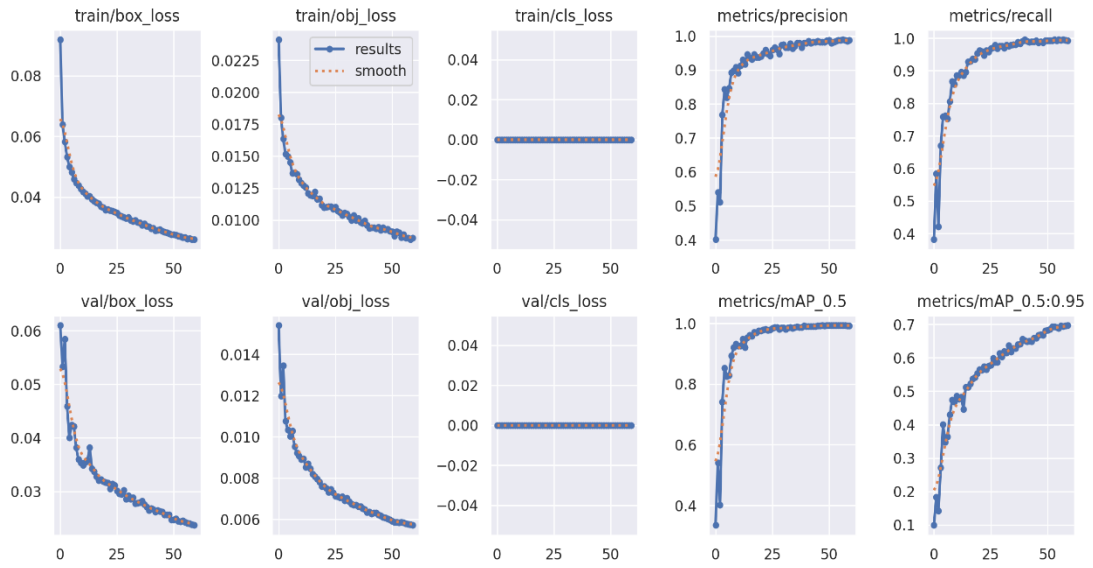
```
# Train YOLOv5s
!python train.py --img 640 --batch 16 --epochs 60 --data custom_data.yaml --weights yolov5s.pt --nosave --cache
```

Şekil 3.12. Sigara nesnesine ait eğitim modelinin oluşturulması için kullanılan kod

Şekil 3.12’de görüldüğü gibi birden çok parametre bulunmaktadır. Bunlar:

- **img:** Giriş görüntü boyutunu tanımlamada
- **batch:** Parti boyutunu belirlemede
- **epochs:** Eğitim sayısını tanımlamada
- **data:** YAML dosyamızın yolunu ayarlama
- **weights:** Ağırlıkların kaydedileceği yolu belirlemede
- **nosave:** Yalnızca son kontrol noktasını kaydetmede
- **cache:** Daha hızlı eğitim için görüntüleri önbelleğe almada kullanılan parametredir.

Sigara nesnesine ait YOLOv5 ile oluşturulan eğitim modelinin kayıpları, performans ölçümleri, mAP değerleri Şekil 3.13'te gösterilmiştir.



Şekil 3.13. Sigara nesnesine ait kayıp ve performans ölçümleri

Şekil 3.13'e göre oluşturulan eğitim modeli %98.3 tahmin oranına sahiptir. Sınırlayıcı kutu, nesne kaybı ve sınıf kaybı sonuçlarına bakıldığında modelin başarılı olduğunu anlayabilmemiz için kayıp değerlerinin 0 değerine yaklaşması gerekmektedir. Sınırlayıcı kutu kaybı fonksiyon değeri 0.09'dan 0.02 değerinin altına düşmüştür. Nesne kaybı fonksiyonu değeri 0.022'den 0.01 değerinin altına düşmüştür. Sigara nesnesi için 1 tane sınıfımız olduğundan dolayı sınıf kaybı bulunmamaktadır. Kesinlik ve geri çağırma değerlerine göre modelin başarılı olduğunu anlayabilmemiz için bu değerlerin 1 değerine yaklaşması gerekmektedir. Performans ölçümlerinde kesinlik ve geri çağırma değerleri 0.3 değerinden 0.9 değerinin üstüne çıkmıştır. mAP değeri 0.98-0.99 aralığında hareket etmektedir. Bu sonuçlar, modelin başarılı şekilde eğitildiğini göstermektedir.

YOLO, Faster R-CNN ve SSD ile eğitilen modelin doğruluğunu hesaplamak için mAP kullanılmıştır. mAP modeli değerlendirmede kullanılan popüler bir performans metriğidir. Kullanılan matematiksel formüller;

$$\text{Kesinlik} = \frac{\text{Doğru Pozitif}}{\text{Doğru Pozitif} + \text{Yanlış Pozitif}} \quad (3.1)$$

$$\text{Geri Çağırma} = \frac{\text{Doğru Pozitif}}{\text{Doğru Pozitif} + \text{Yanlış Negatif}} \quad (3.2)$$

Kesinlik ve geri çağırma değerleri, veri setindeki gerçek etiketler ve modelin tahmin ettiği etiketlerin karşılaştırılmasıyla bulunur. Burada Doğru Pozitif değeri, veri setindeki doğru etiketin doğru tahmin edilmesini; Yanlış Pozitif, yanlış etiketin doğru tahmin edilmesini; Doğru Negatif, doğru etiketin yanlış tahmin edilmesini ve Yanlış Negatif, yanlış etiketin yanlış tahmin edilmesini göstermektedir.

$$F1 = 2 \times \frac{\text{Kesinlik} \times \text{Geri Çağırma}}{\text{Kesinlik} + \text{Geri Çağırma}} \quad (3.3)$$

F1 değeri, modelin ne kadar doğru tahmin yaptığını ve gerçek nesneleri ne kadar iyi tespit ettiğini tek bir ölçümde birleştirir.

$$AP = \sum_{i=1}^N P \text{ Num}(\text{objects}) \quad (3.4)$$

Her sınıf için tek tek geri çağırma değerlerindeki kesinliğin ortalaması alınarak AP (Average Precision, Ortalama Hassasiyet) değeri hesaplanmaktadır.

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (3.5)$$

AP değerlerinin ortalamasının alınmış hali mAP değerini vermektedir. Alkol, sigara, şiddet unsurları ve açık içerik kendi içerisinde farklı sınıf sayılarına sahiptir. Eğitim modellerinde aşırı öğrenme olmaması için hiperparametreler deneme-yanılma yoluyla belirlenmiştir. Her biri 16 parti boyutu, 60 eğitim sayısı ve 640 görüntü boyutu ayarlanarak eğitilmiştir. Bu verilere göre her kategori için lokalizasyon kaybı;

$$L_{lok} = \lambda_{koordinat} \sum_{i=1}^{N_{\text{eğitim}}} \sum_{j=1}^{S^2} \sum_{k=1}^B y_{ij}^{\text{obj}} [(x_{ij}^k - \hat{x}_{ij}^k)^2 + (y_{ij}^k - \hat{y}_{ij}^k)^2] + \lambda_{koordinat} \sum_{i=1}^{N_{\text{eğitim}}} \sum_{j=1}^{S^2} \sum_{k=1}^B y_{ij}^{\text{obj}} \left[\left(\sqrt{w_{ij}^k} - \sqrt{\hat{w}_{ij}^k} \right)^2 + \left(\sqrt{h_{ij}^k} - \sqrt{\hat{h}_{ij}^k} \right)^2 \right] \quad (3.6)$$

Bu denklem, nesne algılama modelini optimize etmek için eğitim sırasında kayıp fonksiyonu olarak kullanılmaktadır. Bu denklemdeki bileşenler, sınırlayıcı kutuların tahminlerinin doğruluğunu değerlendirmek için kullanılır. Bu kayıp, ağırlıklı olarak x_{ij}^k ve y_{ij}^k gibi koordinat tahminlerindeki hata ile sınırlayıcı kutuların genişliği ve yüksekliğindeki hataların toplamıdır. Bu hatalar, gerçek ve tahmini değerler arasındaki farkın karelerinin toplamı ile hesaplanmaktadır. $\lambda_{koordinat}$, lokalizasyon kaybının toplam kayba olan katkısını ayarlamak için bir hiperparametredir. S, resimdeki ızgara hücrelerinin sayısıdır, B ise ızgara hücre başına düşen sınırlayıcı kutuların sayısıdır. N görüntüden oluşan bir veri kümesinde $N_{\text{eğitim}}$ eğitim için kullanılmaktadır. İlk terim, x ve y koordinatları arasındaki farkların karesini hesaplar. İkinci terim ise genişlik ve yükseklik arasındaki farkın karekökünü alır. Burada, w_{ij}^k ve h_{ij}^k gerçek genişlik ve yükseklik değerlerini, $\hat{w}_{ij}^k$ ve $\hat{h}_{ij}^k$ ise tahmin edilen genişlik ve yükseklik değerlerini ifade eder. Her kategori için güven kaybı;

$$L_{\text{güven}} = \sum_{i=1}^{N_{\text{eğitim}}} \sum_{j=1}^{S^2} \sum_{k=1}^B y_{ij}^{\text{obj}} \left[1_{\{C_i^k=c\}} \cdot \text{IoU}_{ij}^k - \alpha \cdot \log p_{ij}^{\text{obj}} \right] + \sum_{i=1}^{N_{\text{eğitim}}} \sum_{j=1}^{S^2} \sum_{k=1}^B (1 - y_{ij}^{\text{obj}}) \cdot \alpha \cdot \log(1 - p_{ij}^{\text{obj}}) \quad (3.7)$$

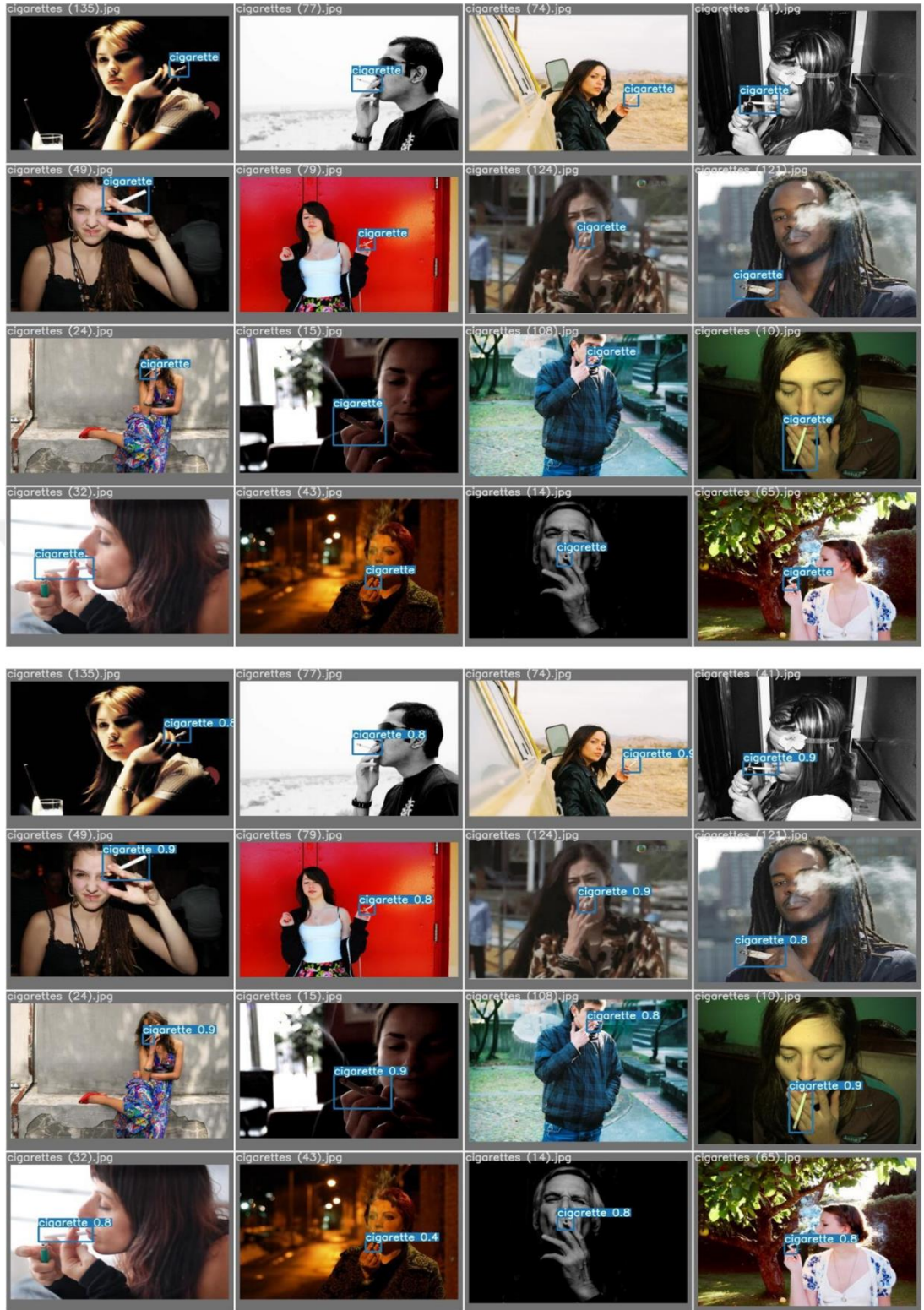
Burada, C_i^k , i. görüntüde k. sınırlayıcı kutu için tahmin edilen sınıfı göstermektedir. IoU_{ij}^k , i. resimde j. nesne için k. tahmin edilen sınırlayıcı kutu ile gerçek değerin kesişme oranını göstermektedir. α , güven kaybının toplam kayba katkısını ayarlayan bir hiperparametredir. Çizelge 3.8’de mAP kullanılarak bulunan sonuçlar gösterilmiştir.

Çizelge 3.8. Sigara nesnesine ait eğitim modellerinin başarı oranları

Yöntem	mAP
YOLOv5	0.983
Faster R-CNN	0.765
SSD	0.453

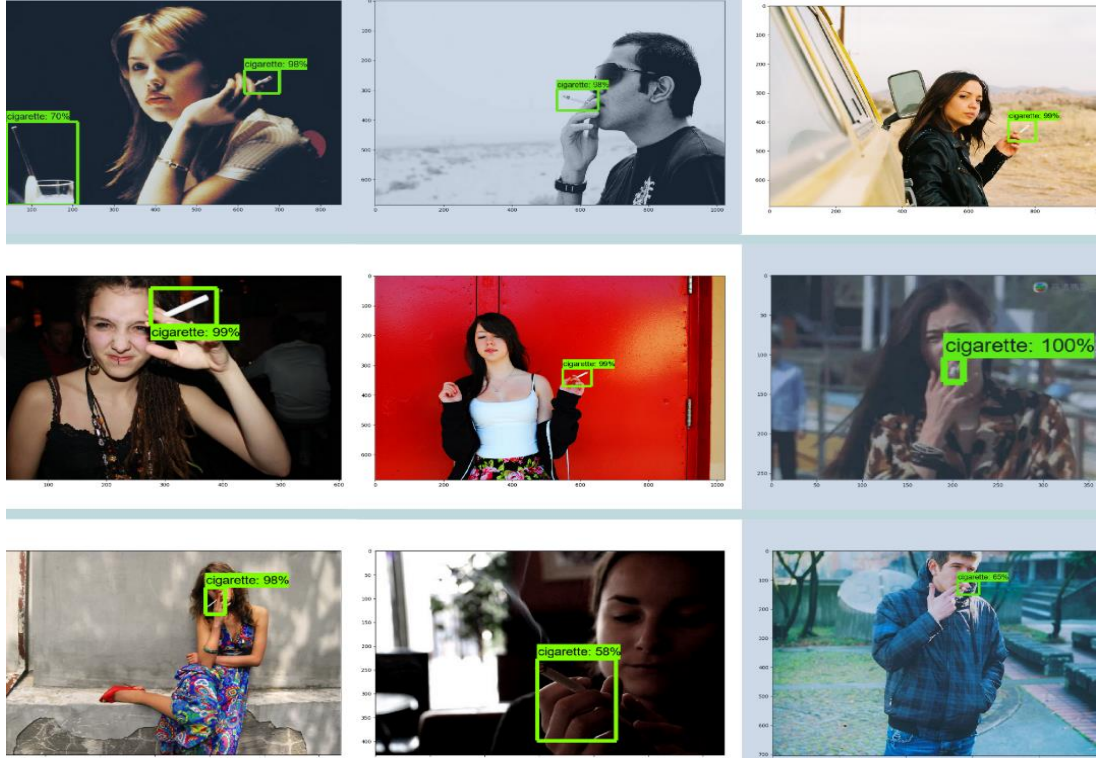
Çizelge 3.8'e göre YOLOv5 nesne tespitinde en yüksek başarı oranına sahip model olduğu görülmüştür. Çalışmada şiddet, açık içerik ve alkol unsurları tespitinde de YOLOv5 kullanılmıştır.

YOLOv5 ile oluşturulan eğitim modeli video ve görüntü veri setleri ile denenmiştir. Sigara nesnesine ait etiketlenmiş örnek veriler ve oluşturulan eğitim modelin tahmin sonuçları Şekil 3.14'te gösterilmiştir.



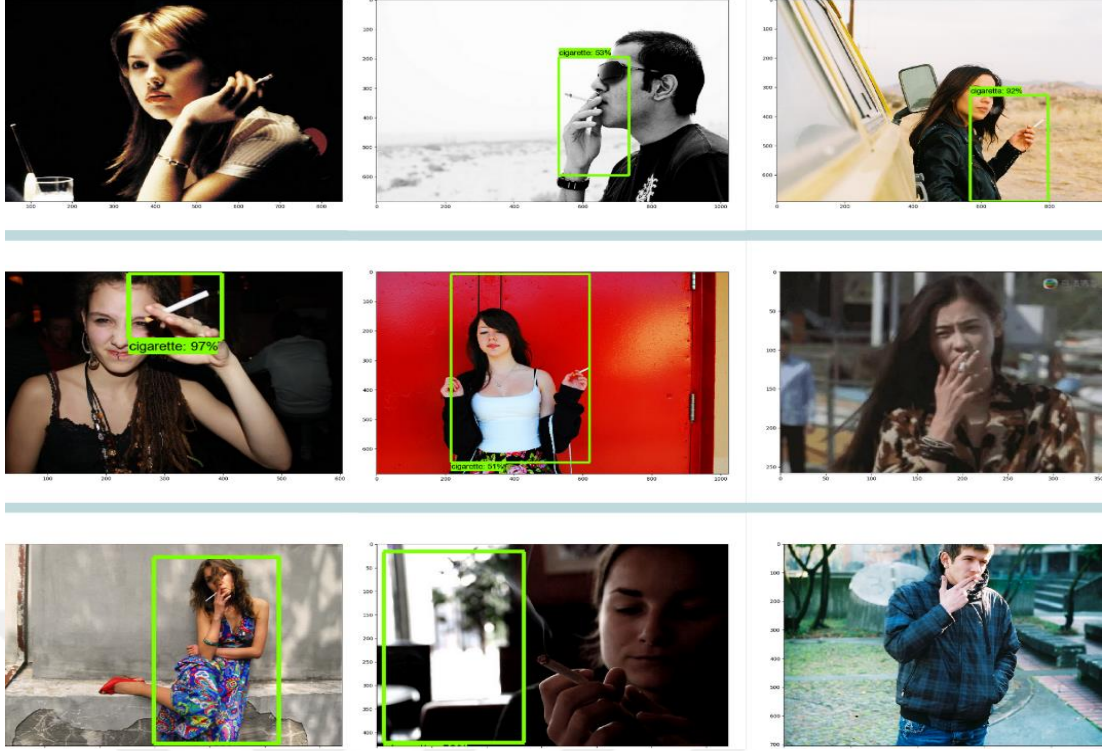
Şekil 3.14. Sigara nesnesine ait etiketlenmiş veriler ve modelin tahmin sonuçları

YOLOv5 ile oluşturulan eğitim modelinin düşük çözünürlüklü videolarda bile başarılı sonuçlar aldığı görülmüştür. Şekil 3.14'te verilen etiketlenmiş veriler Faster R-CNN ile eğitilen modelle test edildiğinde elde edilen tahmin görüntüleri Şekil 3.15'te gösterilmiştir.



Şekil 3.15. Sigara nesnesine ait Faster R-CNN modelinin tahmin sonuçları

Şekil 3.14'te verilen etiketlenmiş veriler SSD ile eğitilen modelle test edildiğinde elde edilen tahminlere ait görüntüler Şekil 3.16'da gösterilmiştir.



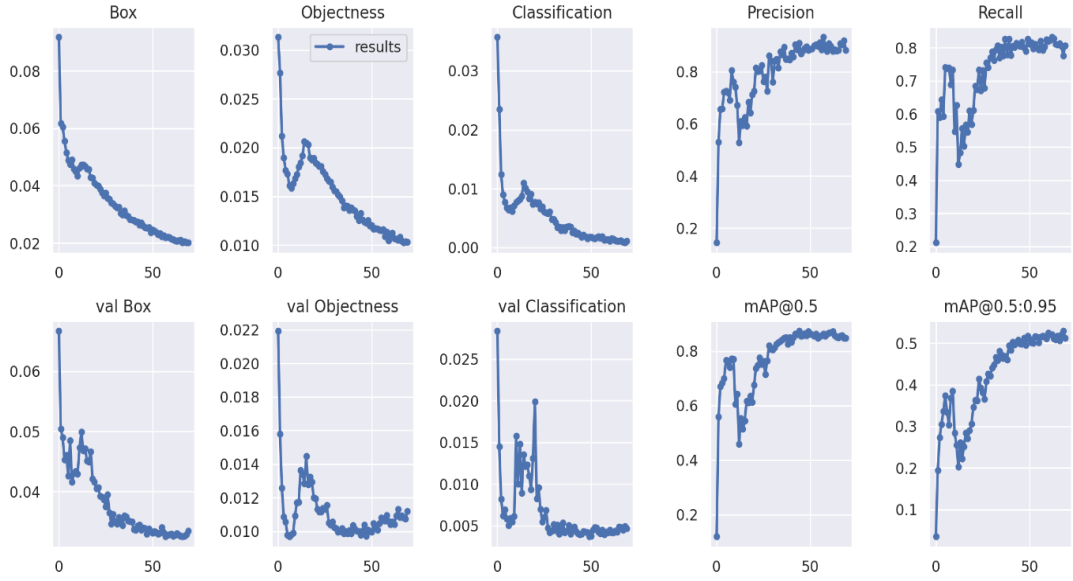
Şekil 3.16. Sigara nesnesine ait SSD modelinin tahmin sonuçları

Sigara nesnesi tespitinde YOLOv5, SSD ve Faster R-CNN kullanılmış ve en yüksek başarı oranı YOLOv5 ile alındığı için diğer kategorilerde de bu model kullanılmıştır.

Şiddet unsurlarının tespiti için veri seti hazırlanırken:

- Parametre ve etiket bilgisi: Parametre sayısı 3 ve etiketler “**pistol**”, “**knife**”, “**rifle**” şeklinde oluşturulmuştur.

Oluşturulan eğitim modelinin kayıp ve performans grafikleri Şekil 3.17’de gösterilmiştir.



Şekil 3.17. Şiddet unsurları nesnesine ait kayıp ve performans ölçümleri

Şekil 3.17'ye göre model %84.8 doğru tahmin oranına sahiptir. Geri çağırma ve tahmin değerlerine bakıldığında yanlış-pozitiflerin sayısının fazla olması yani şiddet unsurları içermeyen nesneleri de şiddet unsuru şeklinde tespit etmesi başarı oranını düşürmektedir. Oluşturulan eğitim modeline veri türlerine çok benzer nesneler gönderildiğinde bu nesneleri yanlış tespit edeceğinden başarı oranı düşmektedir. Tehlikeli unsur türlerine benzer veriler bulunarak bunlar da veri setine dâhil edilmiştir ve eğitim modeli yeniden oluşturulmuştur. Eklenen nesneler; silikon tabancası (Google Images, 2021), telefon (MakeMl, 2021). Amaç yanlış-pozitiflerinin sayısını azaltarak eğitim modelinin başarı oranını arttırmaktır. Şiddet unsurları için oluşturulan veri setine Çizelge 3.9'da gösterilen veriler de eklenerek eğitim modeli yeniden oluşturulmuştur.

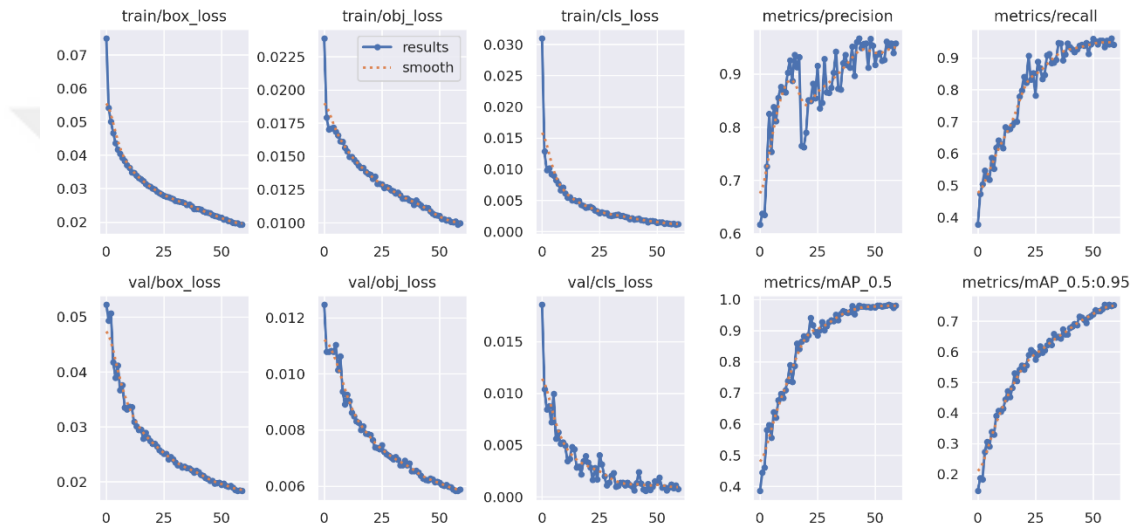
Çizelge 3.9. Benzer nesneleri içeren veri türlerine göre eğitim ve test verileri sayısı

Veri Türü	Eğitim Verisi	Doğrulama Verisi	Toplam	Test Verisi Video
Telefon	575	134	709	1
Silikon Tabancası	40	10	50	1

Eğitim modelinin oluşturulma aşamaları yeni verilere göre sırasıyla yeniden uygulanmıştır. Buna göre;

- Yeni parametre ve etiket bilgisi: Parametre sayısı 5 ve etiketler “**pistol**”, “**smartphone**”, “**knife**”, “**silicongun**”, “**rifle**” şeklinde oluşturulmuştur.

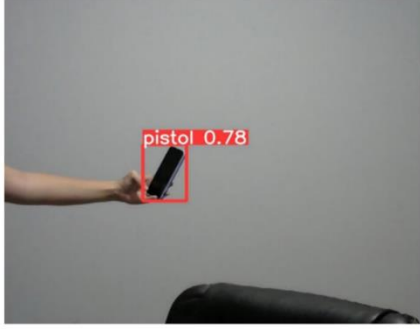
Oluşturulan yeni eğitim modelinin kayıp ve performans grafikleri Şekil 3.18’de gösterilmiştir.



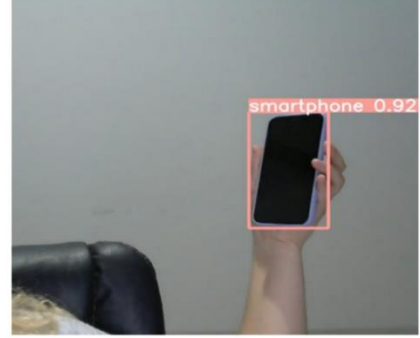
Şekil 3.18. Şiddet unsurları nesnesine ait son kayıp ve performans ölçümleri

Şekil 3.18’e göre model %97.1 doğru tahmin oranına sahiptir. Tespit edilecek nesnelere ait benzer unsurların eğitim modeline eklenmesi başarı oranını arttırmıştır. Şekil 3.19’da gösterildiği gibi benzer nesne yeni eğitim modeli ile test edildiğinde doğru tahmin gerçekleşmiştir.

Şiddet Unsurları Tespiti
(Benzer Nesneler Olmadan)



Şiddet Unsurları Tespiti
(Benzer Nesneler Eklenererek)



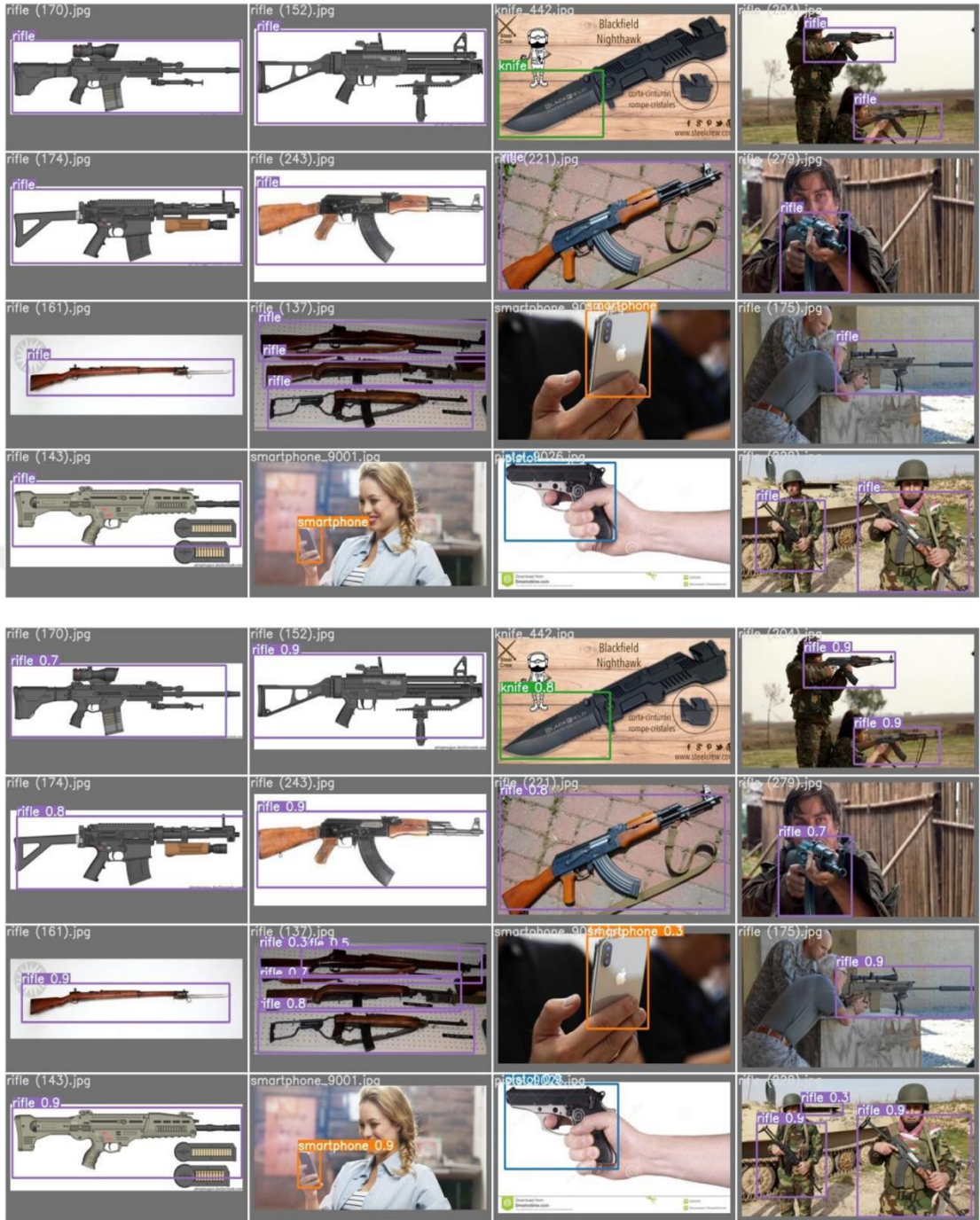
Şekil 3.19. Eğitim modelinin test edilmesi

YOLOv5 ile eğitilen model sentetik veriler üzerinde denendiğinde de tespit işlemi gerçekleşmiştir. Fakat sentetik verilerde bulunan nesnelerin çok küçük olması, net olmaması yanlış-pozitif sayısının da artmasına neden olmaktadır. Karanlık arka planı olmayan, net görüntüler üzerinde eğitim modeli kullanılabilir ya da bu veriler de eğitim setine eklenerek yeni bir model oluşturulabilir. Oluşturulan eğitim modelinin sentetik veriler üzerinde yaptığı tahminlere ait görüntüler Şekil 3.20’de gösterilmiştir.



Şekil 3.20. Sentetik verilerde şiddet unsurlarının tespiti

Şekil 3.21’de YOLOv5 formatında etiketlenmiş veriler ve eğitim modelinin bu verilerdeki tahmin sonuçları gösterilmiştir.

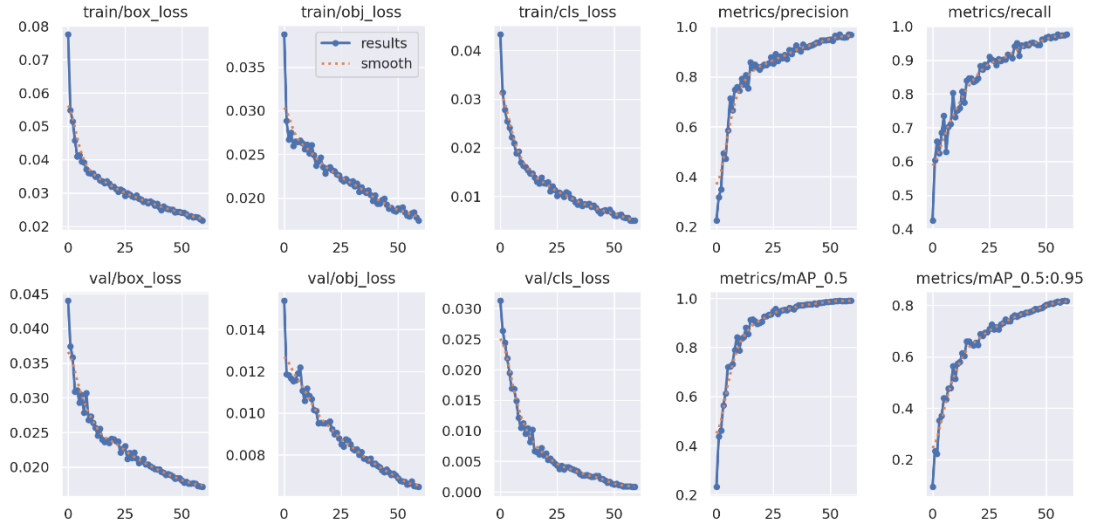


Şekil 3.21. Şiddet unsurlarına ait etiketlenmiş veriler ve modelin tahmin sonuçları

Alkol unsurlarının tespiti için veri seti hazırlanırken:

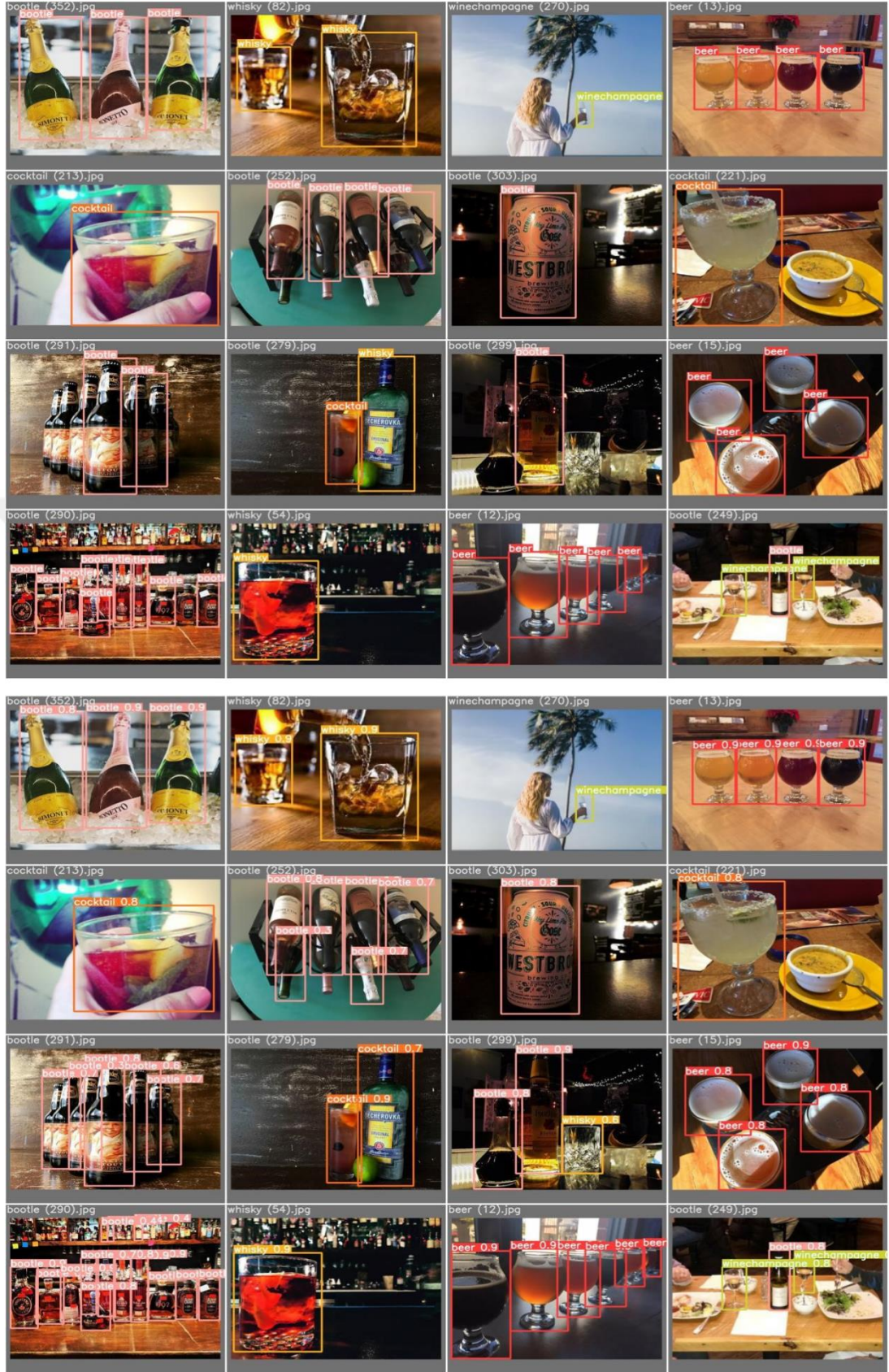
- Parametre ve etiket bilgisi: Parametre sayısı 5 ve etiketler “**beer**”, “**bootle**”, “**cocktail**”, “**whisky**” ve “**wineshampagne**” şeklinde oluşturulmuştur.

Oluşturulan eğitim modelinin kayıp ve performans grafikleri Şekil 3.22’de gösterilmiştir.



Şekil 3.22. Alkol unsurları nesnesine ait kayıp ve performans ölçümleri

Şekil 3.22’ye göre oluşturulan eğitim modeli %97.1 doğru tahmin değerine sahiptir. Alkol nesnesine ait etiketlenmiş örnek veriler ve oluşturulan eğitim modelin tahmin sonuçları Şekil 3.23’te gösterilmiştir.



Şekil 3.23. Alkol unsurlarına ait etiketlenmiş veriler ve modelin tahmin sonuçları

Kola, fanta gibi alkol içermeyen nesneler de eğitim modelinde tespit edilmektedir. Bu da eğitim modelinin başarı oranını düşürmektedir. Şiddet unsuru içeren nesnelerde yapıldığı gibi burada da alkol nesnelere benzeyen nesneler eğitim modeline eklenerek yanlış-pozitif sayısı azaltılıp başarı oranı arttırılabilir. Alkol nesnesine ait yanlış-pozitive bir örnek Şekil 3.24'te gösterilmiştir.

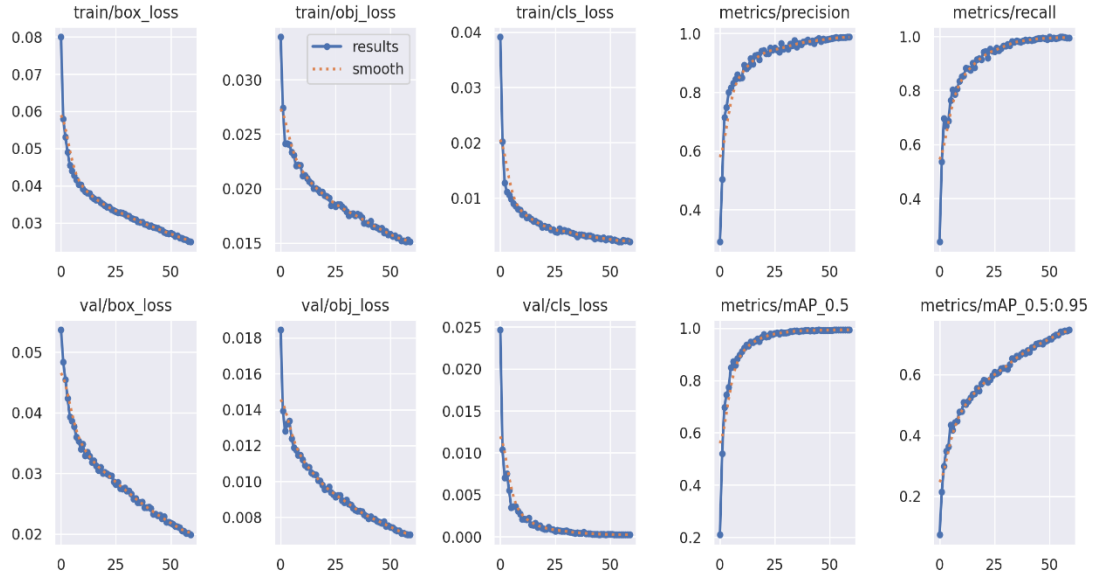


Şekil 3.24. Alkol unsurları test verilerinde yanlış-pozitif çıkan örnek görüntü

Açık içeriklerin tespiti için veri seti hazırlanırken:

- Parametre ve etiket bilgisi: Parametre sayısı 1 ve etiketler “**Explicit Content**” şeklinde oluşturulmuştur.

Oluşturulan eğitim modelinin kayıp ve performans grafikleri Şekil 3.25’de gösterilmiştir.



Şekil 3.25. Açık içerikli nesnelere ait kayıp ve performans ölçümleri

Şekil 3.25'e göre oluşturulan eğitim modeli %98.4 doğru tahmin oranına sahiptir. Tez çalışmasında sigara, şiddet unsurları, açık içerik ve alkol nesnesi tespitinde elde edilen başarı oranları Çizelge 3.10'da birleştirilerek verilmiştir.

Çizelge 3.10. Kategorilere göre başarı oranları

Kategori	mAP			YOLOv5		
	SSD	Faster R-CNN	YOLOv5	Kesinlik	Geri Çağırma	F1
Sigara	0.453	0.765	0.983	0.984	0.992	0.987
Şiddet Unsurları	-	-	0.971	0.961	0.935	0.947
Alkol	-	-	0.971	0.967	0.973	0.969
Açık İçerik	-	-	0.984	0.973	0.984	0.978

Çizelge 3.10'a göre eğitim modeli oluşturulurken kullanılacak model belirlenmeden önce sigara nesnesi SSD, Faster R-CNN ve YOLOv5 ile eğitilmiştir. YOLOv5 en hızlı ve en iyi sonucu verdiği için diğer nesneler de YOLOv5 ile eğitilmiştir. “-” sütundaki nesnelerin algoritma ile denenmediğini göstermektedir. Tez çalışmasında YOLO kullanılarak zararlı içeriklerin yüksek doğrulukla ve güvenli bir şekilde tespit edilmesi hedeflenmiştir. Eğitilen modeller web kamerası ve video üzerinde denendiğinde karelerin yavaş bir

şekilde geldiği ve hız problemi olduğu görülmüştür. Bu yüzden gerçek zamanlı işlemeyi sağlamak için YOLO nesne algılama çekirdeğinin ön ucundan bağımsız çalışan ve karelerin alınma ve gösterilme hızında iyileştirilme sağlayan ek bir yöntem ihtiyacı duyulmuştur.

3.2.4. Derin öğrenme tabanlı video oynatıcı yazılımının geliştirilmesi

Uygulama, Visual Studio platformu üzerinde C# programlama dili kullanılarak geliştirilmiştir. YOLOv5 ile eğitilen modeller .pt uzantılı dosyalardır. "pt" kısaltması, "PyTorch" teriminin kısaltmasıdır. Bu yüzden .pt uzantılı eğitim modelleri başka platformlarda da kullanılabilmesi açısından ONNX, .pb, tflite ve tfjs dosyalarına dönüştürülmüştür. "pb" kısaltması, "Protobuf" veya "Protocol Buffers" teriminin kısaltmasıdır. ".tflite" uzantısı, "TensorFlow Lite" teriminin kısaltmasıdır. ".tfjs" uzantısı, "TensorFlow.js" teriminin kısaltmasıdır. Tez çalışmasında ONNX modelleri kullanılarak videolardaki belirlenen nesneleri algılayan bir .NET web uygulaması oluşturulmuştur. Şekil 3.26'da .pt uzantılı eğitim modeli kullanılarak ONNX, .pb, tflite ve tfjs uzantılı eğitim modellerinin nasıl oluşturulduğunu gösteren kod verilmiştir.

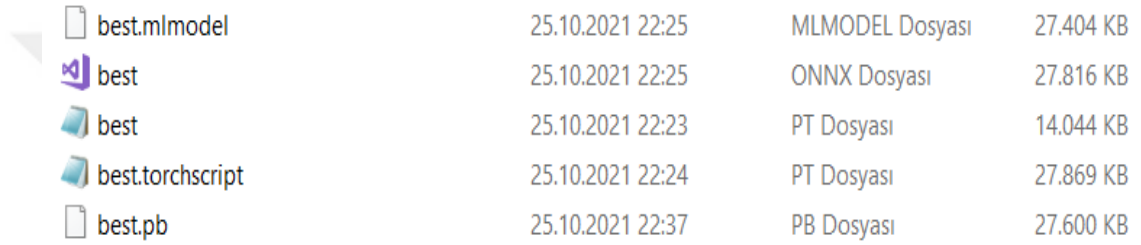
```
%%shell
export PYTHONPATH="$PWD"
rm -rf runs
for m in yolov5s; do
  python train.py --weights $m.pt --epochs 60 --img 640 --data custom_data.yaml
  python train.py --weights '' --cfg /content/yolov5/data/custom_data.yaml --epochs 60 --img 640 --device 0
  for d in 0 cpu; do
    python detect.py --weights runs/train/exp/weights/best.pt --source /content/train_data/images/val
    python val.py --weights runs/train/exp/weights/best.pt --source /content/train_data/images/val
  done
done
python hubconf.py
python models/yolo.py --cfg /content/yolov5/data/custom_data.yaml
python models/tf.py --weights runs/train/exp/weights/best.pt
python export.py --img 640 --batch 1 --weights runs/train/exp/weights/best.pt --include torchscript onnx coreml saved_model pb tflite tfjs
```

Şekil 3.26. Eğitim modelinin farklı uzantılı eğitim modellerine dönüştürülmesini sağlayan kod

Şekil 3.26'ya göre ilk for döngüsünde kendi verilerimizle YOLOv5 kullanılarak .pt uzantılı eğitim modeli oluşturulmaktadır. İkinci for döngüsünde oluşturulan

eğitim modelleri ile nesne tespit işlemleri gerçekleştirilerek modelin doğruluk oranı test edilmektedir.

export.py dosyası kullanılarak oluşturulan .pt uzantılı eğitim modeli başka platformlarda kullanılmak üzere farklı uzantılı eğitim modellerine dönüştürülmüştür. Şekil 3.26'da gösterilen kod her kategori için ayrı ayrı uygulanmaktadır. Eğitim, test ve dönüştürme işlemleri her kategori için yaklaşık 4 saat sürmektedir. Şekil 3.27'de Şekil 3.28'e uygulanan koddan sonra bir kategori için oluşan dosyalar ve bilgileri gösterilmiştir.



best.mlmodel	25.10.2021 22:25	MLMODEL Dosyası	27.404 KB
best	25.10.2021 22:25	ONNX Dosyası	27.816 KB
best	25.10.2021 22:23	PT Dosyası	14.044 KB
best.torchscript	25.10.2021 22:24	PT Dosyası	27.869 KB
best.pb	25.10.2021 22:37	PB Dosyası	27.600 KB

Şekil 3.27. Eğitim modelleri ve dosya bilgileri

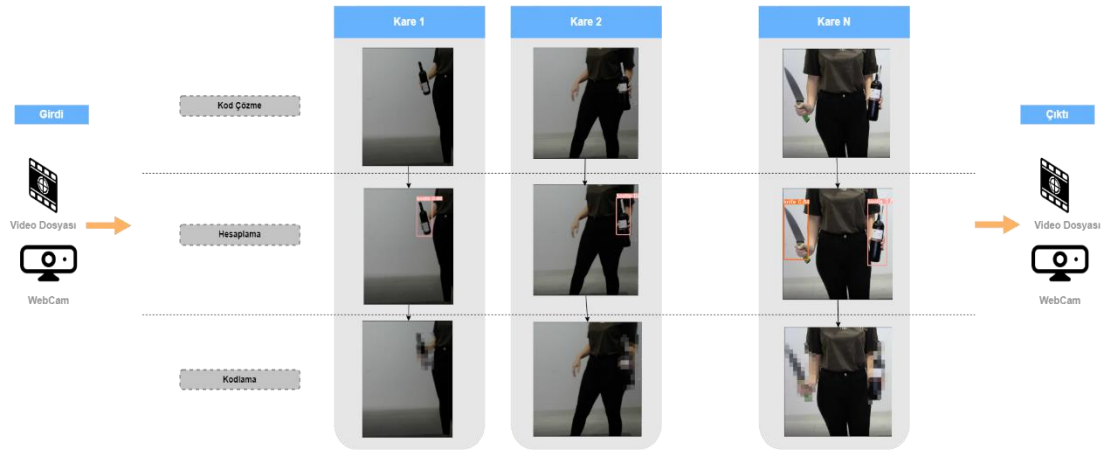
DeepCens adını verdiğimiz platform girdi olarak görüntü, web kamerasından elde edilen görüntü ve video dosyalarını kabul etmektedir. Görüntü üzerinde eğer tespit edilecek nesne varsa hesaplama ve kodlama işlemleri uygulanmaktadır. Video işleme, içerisinde bulundurduğu tüm karelere kod çözme, hesaplama ve kodlama işlemlerinin uygulanması olarak özetlenebilir. Her kare sıkıştırılmış bir biçimde elde edilir. Kod çözme bu karelerin ham biçimine dönüştürülmesi işlemidir. Hesaplama, karelerde yapılması gereken tüm işlemleri içerir. Kodlama ise işlenen tüm karelerin sıkıştırılmış haline dönüştürülme aşamasıdır. DeepCens ile tüm bu adımlar uygulanarak hız ve doğruluk arasında denge kurulması amaçlanmıştır. Şekil 3.28'de YOLOv5 algoritması ile oluşturulan klasik nesne tespiti mimarisi gösterilmiştir.



Şekil 3.28. Klasik nesne tespiti için kullanılan mimari

Şekil 3.28'e gösterilen klasik yaklaşımda girdiler karelere ayrılır ve her kare sırasıyla işlenir. Çıktı dosyasının elde edilmesi için her dosyanın birleştirilmesi gerekir. Bu da hesaplama süresini arttırdığı için yavaşlığa neden olmakta ve gerçek zamanlı işlemeyi daha zor bir hale getirmektedir.

Zararlı içeriklerin gerçek zamanlı tespit edilmesini sağlayabilmek için her karede yapılan işlemleri parçalara ayırarak paralel hale getiren bir yaklaşım önerilmiştir. Yaklaşım Şekil 3.28'de gösterilen mimariye entegre edilmiştir. Şekil 3.29'da uygulamada kullanılan mimari gösterilmiştir.

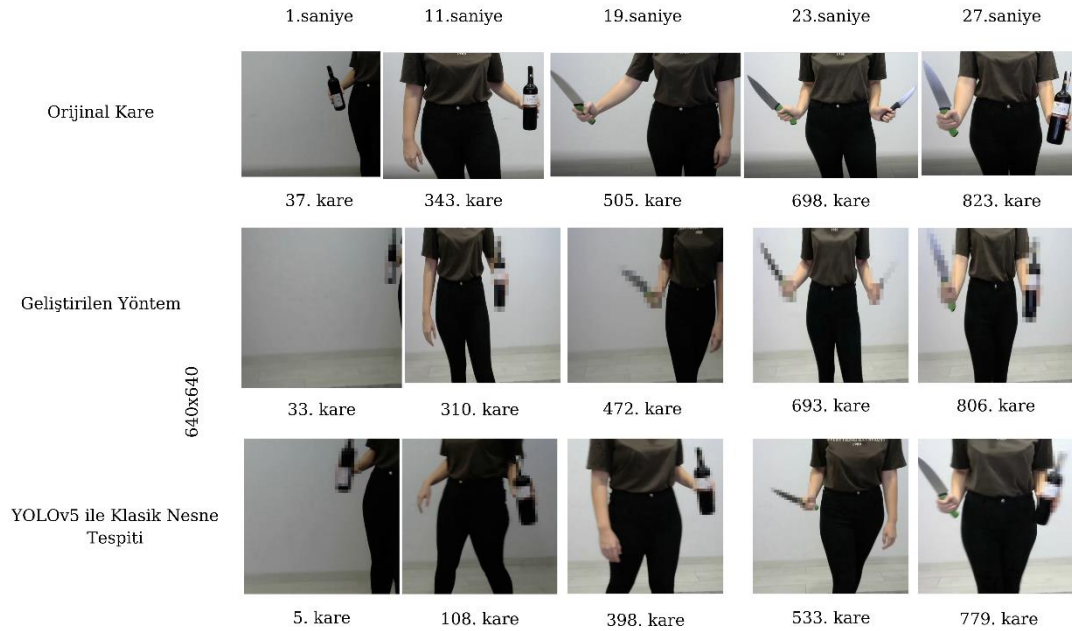


Şekil 3.29. Önerilen yöntem ile nesne tespiti için kullanılan mimari

Şekil 3.29’da işlemler üç boru hattına ayrılmıştır. Boru hatları daha fazla işlemlere ayrılabilir fakat bu da işlem zamanını arttıracaktır. Her bir boru hattında meydana gelen işlem süresi hesaplanarak performans analizi yapılır ve çıktı FPS (Frames Per Second, Saniyedeki Kare Sayısı)’si oluşturulur. Elde edilen FPS değeri orjinal videonun FPS değerine ne kadar yakınsa gerçek zamanlı işleme hızı o kadar iyileştirilmiş olur. Tez çalışmasında gerçek zamanlı performans elde etmek için farklı platformlara entegre edilebilen iki farklı boru hattı tabanlı yaklaşım önerdik: Tek Aşamalı Sansür ve İki Aşamalı Sansür.

3.2.4.1. Tek aşamalı sansür

Tek aşamalı sansür yöntemi klasik nesne tespit adımlarını paralel hale getirerek videolar üzerinde nesne tespit hızını önemli oranda arttırmıştır. Şekil 3.30’da oluşturulan eğitim modelleri ile yapılan klasik nesne tespit mimarisi ve önerilen yöntem ile web kamerası üzerinde nesne tespiti işlemi gerçekleştirildiğinde belirtilen sürelerde elde edilen görüntüler gösterilmiştir.



Şekil 3.30. Web kamerası kaynağından gelen video girişinden nesne algılama işlemi sırasında çekilen fotoğraflar

Şekil 3.30'da 1, 11, 19, 23 ve 27 saniye sonrasında videodan gelen gerçek görüntüler, bu görüntülerin klasik nesne tespit mimarisi ve önerilen yöntemle nesne tespitinde elde ettiği görüntüler gösterilmiştir. Şekilde görüldüğü gibi seri işleme ile elde edilen sonuçların hızının çok yavaş olduğu ve geçen sürede geriden geldiği, önerilen yöntemin ise klasik YOLOv5 mimarisi ile nesne tespitinde meydana gelen hız kaybını iyileştirdiği görülmüştür. Denklem 3.8'de videoların FPS değerlerinin hesaplanmasını gösteren formül gösterilmiştir.

$$\text{FPS} = \frac{\text{Toplam Kare Sayısı}}{\text{Video İşlenme Süresi}} \quad (3.8)$$

Denklem 3.8'e göre her video için kullanılan eğitim modellerinin FPS değerleri hesaplanmıştır. Hesaplanan değerler modelin FPS kayıp oranını hesaplamak için kullanılmaktadır. Denklem 3.9'da eğitim modellerinin FPS kayıp oranını hesaplamak için kullanılan formül gösterilmiştir.

$$\text{Kayıp Oranı} = \frac{\text{Orjinal Video FPS} - \text{İşlenmiş Video FPS}}{\text{Orjinal Video FPS}} * 100 \quad (3.9)$$

Denklem 3.9'a göre işlenmemiş video FPS'si ve eğitim modelleriyle işlenen videoların FPS değerleri hesaplanarak kayıp oranları hesaplanmıştır. Çizelge 3.11'de örnek videolar üzerinde nesne tespit modelleri çalıştırıldığında elde edilen FPS performans değerleri gösterilmiştir. Bu değerler kullanılan bilgisayar ve model özellikleri baz alınarak saniyede işlenen kare sayılarına göre hesaplanmaktadır. Çizelge 3.11'de verilen değerlere göre tek aşamalı sansür yönteminin kayıp oranı klasik nesne tespit yöntemine oranla oldukça düşüktür.

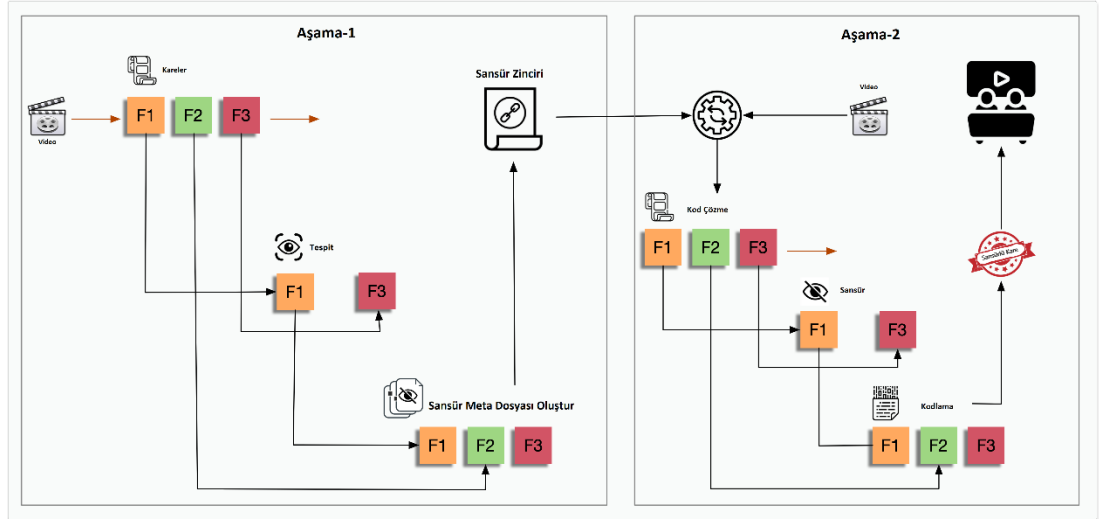
Çizelge 3.11. Kullanılan yöntemlerin FPS performans karşılaştırması

Kategori	Kare Sayısı	Görüntü Çözünürlüğü	Orjinal Video FPS	Klasik Nesne Tespiti FPS/Kayıp Oranı (%)		Tek Aşamalı Sansür FPS/Kayıp Oranı (%)	
Sigara	1492	640x640	29.84	7	76.54	22.95	23.08
Şiddet Unsurları	1219	640x640	24.38	4.06	83.34	19.34	20.67
Alkol	1252	640x640	25.04	5	80.03	20.8	19.8
Açık İçerik	647	640x640	30.8	6	80.5	23.9	22.4

Çizelge 3.11’de CPU üzerinde çalıştırılan kategorilere ait modellerin FPS performansları karşılaştırılmıştır. CPU FPS performans kıyaslaması için, Windows 10 işletim sistemi ve 16 GB RAM (Random Access Memory, Rastgele Erişim Belleği)’e sahip Intel Core i7 7700HQ CPU’lu bir bilgisayar kullanılmıştır. Her kategori için seçilen örnek video üzerinde yapılan denemelerde önerilen yöntemin gerçek zamanlı nesne tespitinde rahatlıkla kullanılabildiği görülmüştür. Çizelge 3.11’e göre örnek videolar için önerilen yöntem sigara kategorisi için %53.46, şiddet unsurları için %62.67, alkol için %60.23, açık içerik için %58.1 daha iyi performans göstermiştir. Web kameralarından gelen görüntüler üzerine önerilen yöntem entegre edildiğinde FPS kaybı yaşanmamakta ve gerçek zamanlı nesne tespit işlemi gerçekleştirilmektedir. YOLO modellerinin gerçek zamanlı performansı için GPU kullanılması önerilmektedir (Mohod vd., 2022). Tek aşamalı sansür ile videolarda meydana gelen titremelerin ve FPS kayıplarının engellenmesi için iki aşamalı sansür yöntemi geliştirilmiştir.

3.2.4.2. İki aşamalı sansür

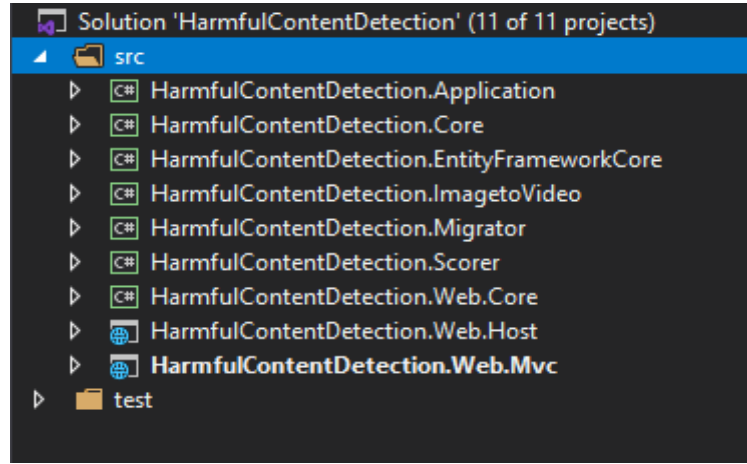
İki aşamalı sansür yöntemi özellikle büyük videolar için tasarlanmıştır ve sansürü iki aşamada gerçekleştirir. Şekil 3.31’de iki aşamalı sansür yöntemine ait akış diyagramı verilmiştir.



Şekil 3.31. İki aşamalı sansür akış diyagramı

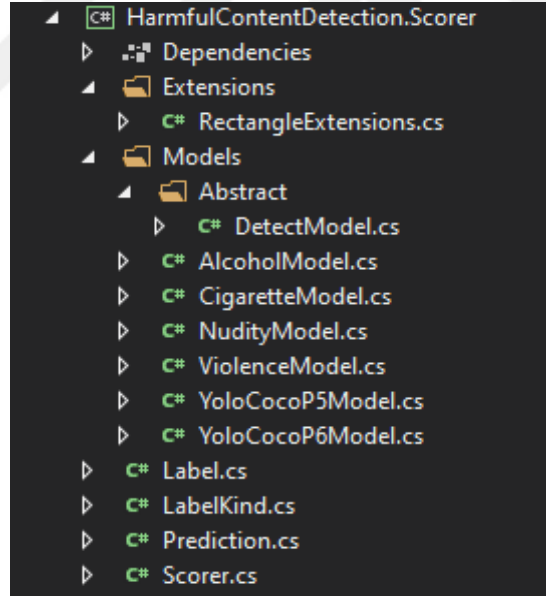
Şekil 3.31’de gösterilen yöntemle göre ilk aşamada, video karelere ayrılmakta ve her kareye zararlı içerik tespiti uygulanmaktadır. Zararlı içeriğin konumlarını ve kare numaralarını içeren bir meta veri, yani sansür zinciri oluşturulmaktadır. Bu sansür zinciri daha sonra kullanılmak üzere arabelleğe alınmaktadır. İkinci aşamada arabelleğe alınan sansür zinciri bellekten okunmakta, zincirdeki konumlara ve kare numaralarına göre bulanıklaştırma yapılarak sansürlenmiş video oynatılmaktadır. Sonuç olarak, orijinal FPS korunarak kesintisiz video oynatımı sağlanmaktadır. Sansür zinciri daha sonra kullanılmak üzere bir dosya olarak da kaydedilmektedir. Zararlı içeriğe sahip sahnelerin zaman damgaları ve konumları düzenlenebilir, güncellenebilir ve değiştirilebilir. Sonuç olarak tek aşamalı sansür yöntemi görüntüler ve web kameralarında, iki aşamalı sansür yöntemi ise videolarda FPS kaybını sıfıra indirerek gerçek zamanlı sansür işlemi yapmaktadır.

Visual Studio’da oluşturulan projeye ait ekran görüntüsü Şekil 3.32’de gösterilmiştir.



Şekil 3.32. Proje yapısına ait ekran görüntüsü

HarmfulContentDetection.Scorer, oluşturulan ONNX modellerine ait sınıfları, tahmin ve skorların parametrelerini içeren sınıfları ve tahmini dikdörtgene çevirmeye yarayan sınıfları ve metotları içermektedir. Şekil 3.33'te HarmfulContentDetection.Scorer projesine ait ekran görüntüsü gösterilmiştir.



Şekil 3.33. Yolov5Net.Scorer sınıfları

ONNX modellerine ait sınıflar her kategori için Netron (Netron, 2021) kullanılarak bu modele ait parametreler ile oluşturulmalıdır. Netron, makine öğrenimi modellerinin görselleştirilmesi ve analizi için kullanılan açık kaynaklı bir yazılımdır. Şekil 3.34'te sigara nesnesine ait örnek ONNX parametreleri gösterilmiştir.

MODEL PROPERTIES ×

format

ONNX v6

producer

pytorch 1.9

imports

ai.onnx v13

INPUTS

images

name: **images**

-

type: float32[1,3,640,640]

OUTPUTS

output

name: **output**

-

type: float32[1,25200,6]

350

name: **350**

-

type: float32[1,3,80,80,6]

416

name: **416**

-

type: float32[1,3,40,40,6]

482

name: **482**

-

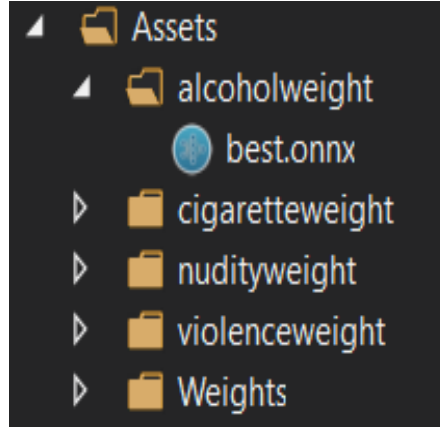
type: float32[1,3,20,20,6]

Şekil 3.34. Sigara nesnesine ait oluşturulan ONNX modeli

3.2.4.3. Web uygulamasının oluşturulması ve test edilmesi

Tez çalışmasında belirlenen kategorilerde toplanan veri setleri etiketlenerek eğitim modelleri oluşturulmuş ve sonuçları verilmiştir. Oluşturulan eğitim modellerinin en iyi sonucu veren dosyası ONNX formatına dönüştürülerek Asp.Net Core ile geliştirilen uygulamada kullanılmıştır. Web uygulamasında kullanılacak olan eğitim modelleri ONNX formatına dönüştürülerek Assets

klasörü altında kategori bazlı olarak toplanmıştır. Şekil 3.35'te uygulamada kullanılan en iyi eğitim modelleri gösterilmiştir.



Şekil 3.35. Uygulamada kullanılan kategorilere ait modeller

Eğitim modellerinin kullanılabilmesi için oluşturulacak sınıflarda kullanılan parametrelerin değerleri Netron kullanılarak belirlenmiştir. Her kategoriye ait parametre değerleri değişmektedir. Başarı oranını etkileyeceği için parametrelerin dikkatli bir şekilde oluşturulması gerekmektedir.

Nesne tespitinde kullanılacak model yükseklik, genişlik, hassasiyet, etiket bilgileri, derinlik, şekil gibi birçok parametreye sahiptir. Uygulamada görüntülerde nesnelerin tespitini gerçekleştirmek için modellerin aktivasyonunda sigmoid fonksiyonu kullanılmıştır. Sigmoid fonksiyonu ağırlık veriler arasındaki doğrusal olmayan ilişkileri öğrenmesini sağlamak için yapay sinir ağlarındaki nöronlar içindeki doğrusal olmayan bir aktivasyon işlevi olarak kullanılır. Sigmoid fonksiyonunda her görüntüye ait bilgiler kullanılarak görüntülerde bulunan nesnelerin tespit edildiği ve işaretlendiği kodda belli hassasiyet değeri altında kalan veriler işaretlenmemektedir.

3.2.4.4. Uygulamaya ait ekran görüntüleri

Uygulamada kullanılan veri setleri araştırmacılarla paylaşıldığı için projede giriş paneli bulunmaktadır. Sadece giriş yapan kullanıcılar sisteme erişebilmektedir. Şekil 3.36'da giriş paneli gösterilmiştir.

HarmfulContentDetection

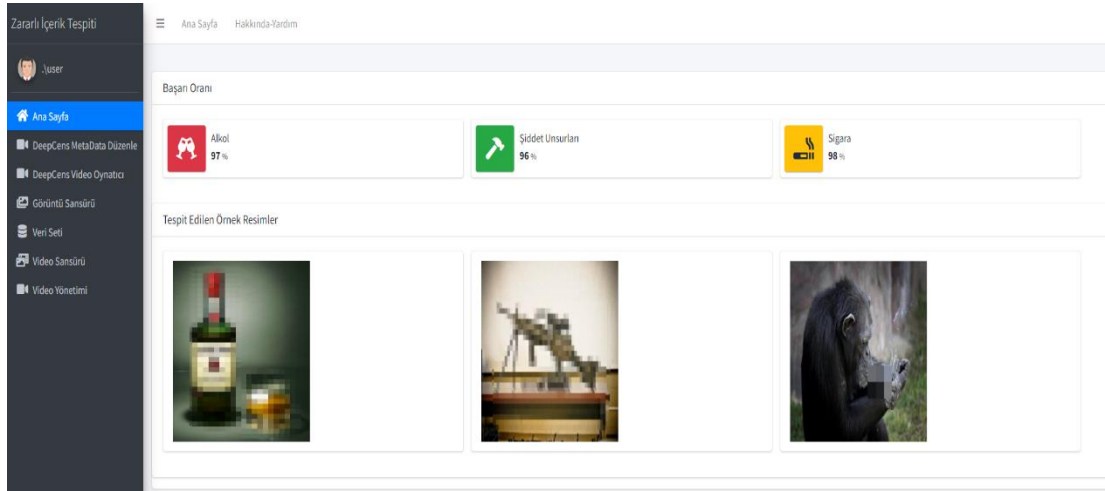
Giriş

☐ Beni hatırla

© 2022 HarmfulContentDetection. **Version** 5.5.0.0 [20220211]

Şekil 3.36. Giriş paneli

Uygulamada İngilizce ve Türkçe dil desteği bulunmaktadır. Şekil 3.37’de geliştirilen menülere ait ekran görüntüsü gösterilmiştir.

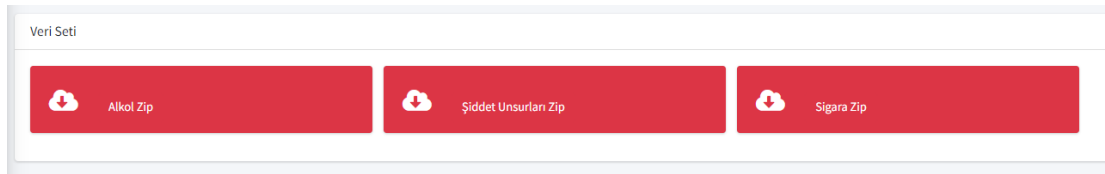


Şekil 3.37. Uygulamada kullanılan modüller

Şekil 3. 37'ye göre uygulamada 6 ana modül bulunmaktadır.

- Görüntü Sansürü modülü, görüntüler üzerinde servis vermektedir ve seçtiğiniz kategorilere göre o görüntülerdeki nesneleri algılayarak nihai görüntüyü kullanıcılara sunmaktadır. Görüntü ve kategoriye seçtikten sonra Tespit Et butonuna tıklamanız gerekmektedir.
- Veri Seti modülü, bu uygulamadaki modelleri oluşturmak için kullanılan verileri içerir. Bu alanda çalışma yapmak isteyen araştırmacılar için veri setleri paylaşılmıştır.

Şekil 3.38'de uygulamada kullanılan kategorileri görebileceğimiz ve veri setlerini indirebileceğimiz sayfaya ait ekran görüntüsü verilmiştir.



Şekil 3.38. Uygulamada kullanılan veri seti indirme ekranı

Eğitim modeli oluşturulurken kullanılan veri setleri eğitim ve test klasörleri olacak şekilde ayrılmıştır. Bu alanda çalışma yapmak isteyen araştırmacılar bu verileri indirerek kullanabilmektedirler.

- DeepCens Video Oynatıcı modülü, listelenen videoların seçilen kategorilere göre gerçek zamanlı izlenmesini sağlar. Bu modülde her videoya ait Video Yönetimi modülünde video yüklenirken oluşturulan meta dosyaları yer almaktadır. Bu sayede kullanıcı istediği kategoriye/kategorileri seçerek nesne algılamalı videoları gerçek zamanlı olarak izleyebilmektedir.
- Video Sansürü modülü, web kamerasından elde edilen görüntüler üzerindeki ve yüklenen videolarda bulunan nesneleri algılamaya yönelik bir modüldür.

Video Sansürü modülünde yer alan uygulamalar;

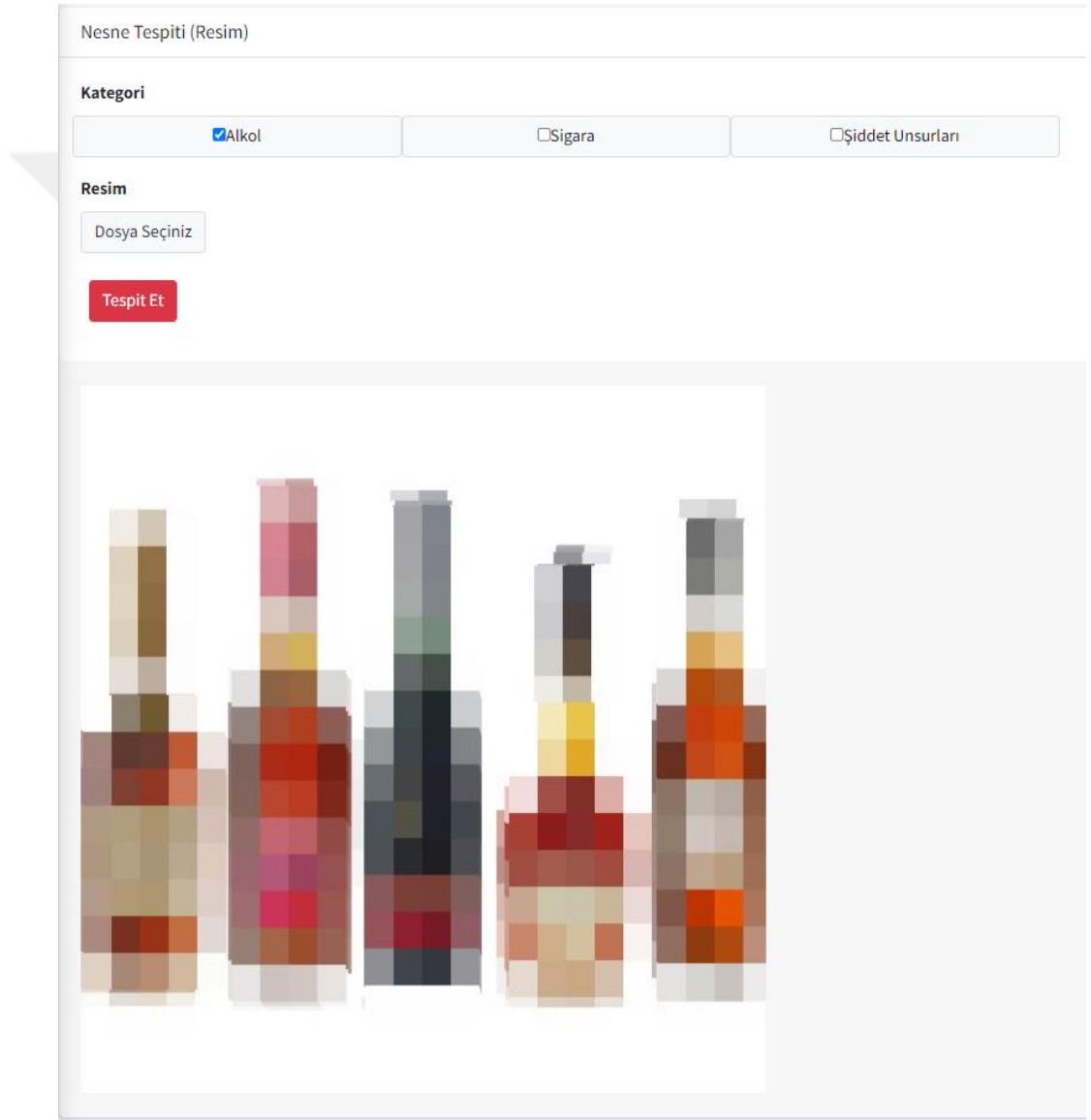
- Gerçek Zamanlı (Real-Time): Yüklenen videolar üzerine klasik YOLOv5 uygulayarak nesne tespit işlemi gerçekleştiren modüldür.
- WebCam: Web kamerasından elde edilen görüntüler üzerine klasik YOLOv5 uygulayarak nesne tespit işlemi gerçekleştiren modüldür.
- Gerçek Zamanlı Boru Hattı (Real Time Pipeline): Yüklenen videolar üzerinde önerilen yöntem uygulanarak daha hızlı nesne tespit işlemi gerçekleştiren modüldür.
- WebCam Boru Hattı (WebCam Pipeline): Web kamerasından elde edilen görüntüler üzerine önerilen yöntem uygulanarak daha hızlı nesne tespit işlemi gerçekleştiren modüldür.

Video ve web kamerası üzerinden elde edilen görüntüler üzerinde hız ve doğruluk olarak nasıl farklılıklar olduğunun net anlaşılması için iki yöntemde uygulamada gösterilmiştir.

- Video Yönetimi modülü yükleyeceğiniz videoya belirtilen tüm kategorilerde nesne algılama modellerini uygular ve videoya ait meta dosyalarını oluşturur. Meta dosyaları tespit edilen nesne adlarını ve koordinat bilgilerini içermektedir. Kaydedilen video DeepCens Video Oynatıcı modülünde listelenmektedir. Videoyu yükledikten sonra Tespit Et butonuna tıklamanız gerekir.

- DeepCens MetaData Düzenleme modülü, Video Yönetimi modülünden elde edilen meta dosyalarının listelendiği ve düzenlendiği bir modüldür. Meta dosyasından istenmeyen veriler silinerek veya yeni veriler eklenerek gerçek zamanlı izleme doğruluğu artırılabilir.

Şekil 3.39’da görüntüler üzerinde nesne tespiti yapmamızı sağlayan modüle ait ekran görüntüsü verilmiştir.

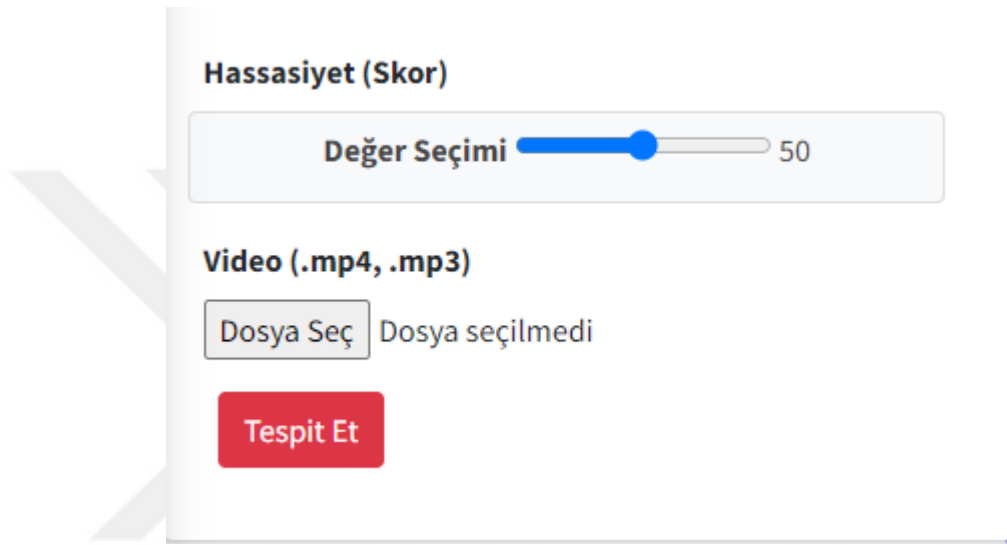


Şekil 3.39. Uygulamada görüntüler üzerinde nesne tespiti yapan modül

Şekil 3.39’da Dosya Seç kısmından seçilen resme, kategori kısmından seçilen kategoriye göre nesne tespit işlemi uygulanmaktadır. Tespit Et butonuna

tıklandıktan sonra nesne tespiti gerçekleştirilerek son görüntü ekrana getirilmektedir. Uygulamada birden fazla kategori seçerek nesne tespit işlemi gerçekleştirilebilir. Eğer hiç kategori seçilmezse orijinal görüntü ekranda gösterilmektedir.

Şekil 3.40'ta videolar üzerinde nesne tespiti yapmamızı sağlayan ve videoya ait meta dosyasının oluşturulduğu modüle ait ekran görüntüsü verilmiştir.



The image shows a user interface for video object detection. It features a slider for 'Hassasiyet (Skor)' (Sensitivity Score) with a value of 50. Below this is a section for 'Video (.mp4, .mp3)' with a 'Dosya Seç' (Select File) button and the text 'Dosya seçilmedi' (File not selected). At the bottom is a red button labeled 'Tespit Et' (Detect).

Şekil 3.40. Uygulamada videolar üzerinde nesne tespiti yapan modül

Şekil 3.40'ta, Dosya Seç kısmından seçilen videonun her karesine, hassasiyet değeri (skor) kısmından seçilen değere ve tüm kategorilere göre nesne tespit işlemi uygulanmaktadır. Seçilen hassasiyet değerinin üzerinde tespit edilen nesnelerin konumun kare numarası ve koordinat bilgileri "Tespit Et" butonuna tıklandıktan sonra arka planda bir dosyaya yazılmaktadır. Nesne tespiti gerçekleşmiş videonun orijinal hali ve meta dosyası DeepCens Video Oynatıcı modülünde listelenmektedir. İşlem, videonun son karesine kadar uygulandığı için uzun sürmektedir ve son video ekrana gelene kadar beklenmesi gerekmektedir. Video, ses dosyasıyla birleştirilerek ekranda gösterilmektedir.

4. ARAŞTIRMA BULGULARI VE TARTIŞMA

Tez çalışmasında, video ve görsellerdeki zararlı içerikleri belirli kategorilere göre tespit etmek amacıyla derin öğrenme yöntemleri kullanılarak bir uygulama geliştirilmiştir. Uygulama geliştirilmeden önce, öncelikle CNN tabanlı yöntemler ve RNN tabanlı yöntemlerle yapılan nesne algılama uygulamaları incelenmiştir. Ardından, ön işleme, nesne algılama, sınıflandırma ve nesne izleme adımları belirlenmiştir.

Ön işleme aşaması, bir sonraki adım için algılanacak nesnelerin veri setinin hazırlanmasından oluşmaktadır. Kullanılacak veri seti, tanımlanan nesnelere göre değişir. Araştırma yapıldığı dönemde ağırlıklı olarak tespit aşamasında kullanılan algoritmalar Faster R-CNN, SSD ve YOLO olarak belirlenmiştir. Eğitim modelleri oluşturulurken veri sayısının fazla olması başarı oranını arttırarak sistemin çalışmasını yavaşlatacaktır. Başarı oranının yüksek olmasının istenildiği uygulamalarda Faster R-CNN, hızlı bir şekilde nesneleri algılamak için ise SSD kullanılmaktadır. Hem yüksek başarı oranı hem de hızın önemli olduğu uygulamalarda ise YOLO sıklıkla kullanılan bir yöntemdir. Tespit edilecek nesne özellikle renk ve doku bazlı yöntemlerle sınıflandırıldığında daha yüksek başarı oranları sağladığı tespit edilmiştir. Uygulanacak sinir ağı modeli de hangi nesnenin izlendiğine bağlı olarak değişir. Sigara nesnesi tespiti için eğitim modeli oluşturulurken SSD, Faster R-CNN ve YOLOv5 modelleri için parti boyutu değeri 16, eğitim sayısı değeri 60, görüntü boyutu 640 olarak ayarlanmıştır. Modellerin başarı oranları ise her sınıf için tek tek geri çağırma değerlerindeki kesinliğin ortalamasının ortalaması alınarak hesaplanmıştır. En yüksek tahmin oranı YOLOv5 ile alındığı için diğer kategorilerde bu modelle oluşturulmuştur. Tez çalışmasında kullanılan kategoriler kullanıcılara uygulanan anket ile belirlenmiştir. 688 kişinin katılım sağladığı anket sonucuna göre sigara, alkol, açık içerikler ve şiddet unsurlarına ait nesneler eğitim modeli için kullanılmıştır. Her kategori için eğitim süresi 12 saat sürmüştür. Geliştirilen uygulamada açık içerikler için %98.4, alkol için %97.1, sigara için %98.3 ve şiddet unsurları için %97.1 tahmin oranı elde edilmiştir. Web kamerasından alınan görüntüler, telefon ile belirlenen kategorilerle çekilen videolar ve

sentetik veriler üzerinde deneyler yapılarak geliştirilen sistemin performansı test edilmiştir. Küçük nesnelerin tespitinde eğitim modelleri oluşturulurken birbirine benzemeyen nesnelerin yanlış tespit edildiği görülmüştür. Bu yüzden eğitim modellerinin başarı oranını arttırmak için şiddet unsurlarının tespitinde kullanılan veri setine yanlış tespit edilen nesneler de dahil edildiğinde yanlış-pozitif sayısının azaldığı görülmüştür. Diğer tüm modeller içinde aynı yöntem uygulanarak başarı oranı arttırılabilir. Karanlık arka planı olmayan, net görüntüler ve sentetik veriler üzerinde eğitim modeli rahatlıkla kullanılabilir.

Oluşturulan eğitim modelleri, diğer platformlarda kullanılmak üzere farklı uzantılara dönüştürülmüştür. Her uzantı, eğitim modeliyle aynı doğruluğa sahiptir. Geliştirilen uygulamada, yüklenen video karelere bölünür ve her kareye eğitim modeli uygulanarak nesne tespiti yapılır. Anında nesne algılama uygulamasında video, kullanıcıya orijinal halinden daha yavaş görünür. Çalışmamızda zararlı içeriklerin yüksek doğrulukla ve gerçek zamanlı tespit edilmesi amaçlanmıştır. Özellikle ağ tabanlı sistemlerden gelen videolarda gerçek zamanlı nesne tespiti yapmanın zor olduğu görülmüştür. Durağan olmayan görüntülerde YOLOv5 ile nesne tespiti yapılmak istenildiğinde videonun her karesine seri bir şekilde nesne tespit modelleri uygulanmaktadır. Bu da gerçek zamanlı bir görünüm sağlamamaktadır. Bu problemi çözmek için YOLOv5 modeline entegre edilen tek aşamalı ve iki aşamalı iki yöntem geliştirilmiştir. Tek aşamalı yöntem web kameralarında gerçek zamanlı nesne tespit işlemi gerçekleştirirken videolar üzerinde titremeler olduğu görülmüştür. Tek aşamalı yöntem videolar üzerinde klasik nesne tespit yöntemlerine göre %58.6 daha hızlı çalışmaktadır. İki aşamalı yöntem, tek aşamalı yöntemin videolar üzerinde meydana getirdiği FPS kayıplarını sıfıra indirmek için geliştirilmiştir ve başarılı bir şekilde gerçek zamanlı nesne tespit işlemi gerçekleştirmektedir. Önerilen yöntemler diğer nesne tespit algoritmalarına da kolaylıkla entegre edilebilir. Zararlı içeriklerin tespiti için hız ve başarı oranını arttırılmış ve bu modelin birçok platformda kolaylıkla kullanılabilirliği sağlanmıştır.

5. SONUÇ VE ÖNERİLER

Kategori belirleme için yapılan çalışmalar incelendiğinde özellikle çocuklar için internet üzerindeki kötü içerikli videolar tehlike unsuru olmaktadır. Bu alanda yapılan çalışmaların genel olarak videoları sınıflandırma işlemi için olduğu görülmüştür. Örneğin YoutubeKids bir API sağlamadığı için önerilen yöntemler Youtube videolarında uygulanarak test edilmiştir. Yani YoutubeKids üzerinde hala kötü unsurlu videolara rastlanmaktadır. Tez çalışmasında ise geliştirilen platformda yüklenen videolar, görüntüler ve web kamerasından açılan canlı yayınlar üzerinde seçilen kategorilere göre anlık ve çoklu tespit işlemi yapılmaktadır. Kategoriler kullanıcılara düzenlenen anket ile belirlenmiştir. Açık içerik, sigara, alkol ve şiddet unsurları tez çalışmasında kullanılan kategorilerdir.

Nesne tespiti farklı platformlarda gerçek zamanlı yapabilmek için YOLO mimarisine boru hattı mimarisi entegre edilen iki yöntem geliştirilmiştir. Görüntüler ve web kamerasından gelen görüntüler üzerinde tek aşamalı yöntem gerçek zamanlı nesne tespiti yaparak FPS kaybını 0'a indirmiştir. İki aşamalı yöntem ise videolar üzerinde FPS kaybını 0 yaparak gerçek zamanlı nesne tespit ve sansürleme işlemlerini gerçekleştirmektedir. İki yöntemde model bağımsız gerçek zamanlı sansür yapabilmektedir. Bu yöntemle kullanıcıların kendi modellerini entegre ederek farklı platformlarda farklı uygulamalar yapması amaçlanmıştır. Gelecek çalışmalarda oluşturulan platform Seç İzle sistemlerine geliştirilecek API'ler vasıtasıyla entegre olabilir ve içerik sağlayıcılara hizmet verebilecek hale getirilebilir. Başarı oranını etkileyen faktörler de uygulamaya dahil edilerek yanlış-pozitif sayısı azaltılabilir. Uyuşturucu madde, kan gibi zararlı içeriklerin kategori sayısı arttırabilir. Geliştirilen platformda nesnelerin videolarda tespit edilmesi ve sansürlenmesine odaklanılmıştır. Gelecek çalışmalarda videolarda ses işlemede yapılarak duyulması istenilmeyen ifadeler sansürlenebilir. Tez çalışmasında, YOLOv5 modelinin algılama yeteneği ve sağlamlılığı geliştirilerek doğru ve gerçek zamanlı zararlı içerik algılaması sağlanmıştır. Bir sonraki adımda geliştirilen yöntemle birlikte zararlı içerik tespiti modelleri telefon, tablet gibi gömülü cihazlara da yerleştirilebilir.

KAYNAKLAR

- Alcohol Pictures, 2021. Eriřim Tarihi: 10.05.2021. Unsplash Home: <https://unsplash.com/s/photos/alcohol>
- Alghowinem, S., 2018. A safer YouTube kids: An extra layer of content filtering using automated multimodal analysis. Advances in Intelligent Systems and Computing.
- Amidi, A., Amidi, S., 2020. Derin Öğrenme El Kitabı. Derin Öğrenme El Kitabı: <https://stanford.edu/~shervine/l/tr/teaching/cs-229/cheatsheet-deep-learning>
- ari-dasci/OD-WeaponDetection, 2021. Eriřim Tarihi: 27.03.2021. Github: <https://github.com/ari-dasci/OD-WeaponDetection>
- Arslankara, V., Usta, E., 2020. Lise Öğrencilerinde Sanal Risk Algısı: Problemlı İnternet Kullanımı ve Eleřtirel Düşünme Bağlamında Bir Arařtırma. DergiPark, 134-153.
- Bahhar, C., Ksibi, A., Ayadi, M., Jamjoom, M. M., Ullah, Z., Soufiene, B. O., Sakli, H., 2023. Wildfire and Smoke Detection Using Staged YOLO Model and Ensemble CNN. Electronics, 12(1), 228.
- Bazarov, E., 2019. nsfw-data-source-urls. Github: <https://github.com/EBazarov/nsfw-data-source-urls>
- Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B., 2016. Simple online and realtime tracking. Proceedings-International Conference on Image Processing, ICIP, 3464–3468, doi: 10.1109/ICIP.2017.8296962.
- Brunetti, A., Buongiorno, D., Trotta, G., Bevilacqua, V., 2018. Computer vision and deep learning techniques for pedestrian detection and tracking: A survey. Neurocomputing, doi: 10.1016/j.neucom.2018.01.092.
- Castillo, A., Tabik, S., Pérez, F., Olmos, R., Herrera, F., 2019. Brightness guided preprocessing for automatic cold steel weapon detection in surveillance videos with deep learning. Neurocomputing, doi:10.1016/j.neucom.2018.10.076.
- CBINSIGHTS, C., 2019. The Race For AI: Here Are The Tech Giants Rushing To Snap Up Artificial Intelligence Startups.
- Chaudhary, S., Khan, M., Bhatnagar, C., 2018. Multiple Anomalous Activity Detection in Videos. Procedia Computer Science, 336-345, doi: 10.1016/j.procs.2017.12.045.

- Collobert, R., Farabet, C., Kavukcuoğlu, K., 2017. Torch | Scientific computing for LuaJIT. NIPS Workshop on Machine Learning Open Source Software.
- Cömert, O., Hekim, M., Adem, K., 2019. Faster R-CNN Kullanarak Elmalarda Çürük Tespiti. Uluslararası Mühendislik Araştırma ve Geliştirme Dergisi, 335-341, doi: 10.29137/umagd.469929.
- Daş, R., Polat, B., Tuna, G., 2019. Derin Öğrenme ile Resim ve Videolarda Nesnelerin Tanınması ve Takibi. Fırat Üniversitesi Mühendislik Bilimleri Dergisi, 571-581, doi: 10.35234/fumbd.608778.
- De Azevedo Kanehisa, R., De Almeida Neto, A., 2019. Firearm Detection using Convolutional Neural Networks. ICAART 2019 - Proceedings of the 11th International Conference on Agents and Artificial Intelligence, doi:10.5220/0007397707070714.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Li, F., 2009. ImageNet: a Large-Scale Hierarchical Image Database. 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25, doi: 10.1109/cvpr.2009.5206848.
- Deng, L., Yu, D., 2013. Deep learning: Methods and applications. Foundations and Trends in Signal Processing, 197-387, doi: 10.1561/20000000039.
- Deori, B., Meitei, D., 2014. A survey on moving object tracking in video. International Journal on Information Theory, 31-46, doi: 10.5121/ijit.2014.3304.
- Dhanwal, S., Bhaskar, V., Agarwal, T., 2020. Automated Censoring of Cigarettes in Videos Using Deep Learning Techniques. Decision Analytics Applications in Industry, 339-348, doi:10.1007/978-981-15-3643-4_26.
- Egiazarov, A., Mavroeidis, V., Zennaro, F., Vishi, K., 2019. Firearm detection and segmentation using an ensemble of semantic neural networks. 2019 European Intelligence and Security Informatics Conference (EISIC), doi:10.1109/EISIC49498.2019.9108871.
- Everingham, M., Van Gool, L., Williams, C., Winn, J., Zisserman, A., 2008. The PASCAL Visual Object Classes Challenge 2008 (VOC) Results, <http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html>
- Fernandez-Carrobles, M., Deniz, O., Maroto, F., 2019. Gun and Knife Detection Based on Faster R-CNN for Video Surveillance. Iberian Conference on Pattern Recognition and Image Analysis, 441-452, doi:10.1007/978-3-030-31321-0_38.
- G. Feller, A. Ventimiglia, 2020. Vidangel: Content filtering technologies, religion, and American copyright law. Internet Histories, 5(1), 8-29.

Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? the KITTI vision benchmark suite. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 3354-3361, doi: 10.1109/CVPR.2012.6248074.

Giannakopoulos, T., Pikrakis, A., Theodoridis, S., 2010. A multimodal approach to violence detection in video sharing sites. Proceedings-International Conference on Pattern Recognition.

Giddens, K., 2021. Erişim Tarihi: 29.03.2021. World's First Synthetic Gun Detection Dataset from Edgecase.ai. Noteworthy The Journal Blog: <https://blog.usejournal.com/worlds-first-synthetic-gun-detection-dataset-from-edgecase-ai-dbe3ea8eeb7e>

Girshick, R., 2015. Fast R-CNN. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 1440-1448, doi: 10.1109/ICCV.2015.169.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2016. Region-Based Convolutional Networks for Accurate Object Detection and Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 142 – 158, doi: 10.1109/TPAMI.2015.2437384.

Google Colab, 2021. Erişim Tarihi: 25.04.2021. Google Colab: <https://colab.research.google.com/notebooks/intro.ipynb>

Google Images, 2021. Erişim Tarihi: 27.03.2021. Google Images: https://www.google.com/search?q=rifles&tbm=isch&ved=2ahUKEwiVxuTM29DvAhU47bsIHWcWB2UQ2-cCegQIABAA&oq=rifles&gs_lcp=CgNpbWcQAzICCAAYAggAMgQIABAeMgQIABAeMgQIABAeMgQIABAeMgQIABAeMgQIABAeMgQIABAeMgQIABAeOgQIIxAnOgUIABCxAzoICAAsQMogwE6BAGAEENQ8TdYkU5g4k9oAXA_A

Gracová, M., 2022. Effect of The Quarantine on Television Viewership. Megatrendy a médiá, 41-48.

Griffin, G., Holub, A., Perona, P., 2007. Caltech-256 object category dataset. Caltech mimeo.

Han, G., Li, Q., Zhou, Y., Duan, J., 2019. Rapid Cigarette Detection Based on Faster R-CNN. 2019 IEEE Symposium Series on Computational Intelligence, SSCI 2019, doi:10.1109/SSCI44817.2019.9003034.

Hanbay, K., Üzen, H., 2017. Nesne tespit ve takip metotları: Kapsamlı bir derleme. Türk Doğa ve Fen Dergisi.

Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., . . . Larochelle, H., 2017. Brain tumor segmentation with Deep Neural

- Networks. Medical Image Analysis, 18-31, doi: 10.1016/j.media.2016.05.004.
- Hinton, G., Osindero, S., Teh, Y.-W., 2006. A fast learning algorithm for deep belief nets. Neural Computation, 1527-1554, doi: 10.1162/neco.2006.18.7.1527.
- Hochreiter, S., Schmidhuber, J., 1997. Long Short-Term Memory. Neural Computation, 1735-1780, doi: 10.1162/neco.1997.9.8.1735.
- Image Moderation Api Demo, 2021. Erişim Tarihi: 13.05.2021. Sightengine: <https://sightengine.com/demo>
- Jacob, A., Anitha, J., 2012. Inspection of various object tracking techniques. International Journal of Engineering and Innovative Technology, 118-124, doi: 10.17605/OSF.IO/Y5K3H.
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., . . . Darrell, T., 2014. Caffe: Convolutional Architecture for Fast Feature Embedding. MM '14: Proceedings of the 22nd ACM international conference on Multimedia, doi: 10.1145/2647868.2654889.
- Justin, L., Sydney, M., 2017. Developing a Real-Time Gun Detection Classifier. Stanford.
- Kalaman, S., Keklik, C., Özkul, R., 2019. Türkiye'nin Akdeniz Bölgesi'ndeki Ortaokul Öğrencilerinin İnternete Yönelik Epistemolojik İnançları. Ege Üniversitesi İletişim Fakültesi Yeni Düşünceler Hakemli E-Dergisi.
- Kanna, V. J., Ebenezer Raj, S., Meena, M., Meghana, S., Mansoor Roomi, S., 2020. Deep Learning Based Video Analytics for Person Tracking. International Conference on Emerging Trends in Information Technology and Engineering, ic-ETITE 2020, doi: 10.1109/ic-ETITE47903.2020.173.
- Keras: The Python Deep Learning API, 2020. Erişim Tarihi: 25.04.2021 Keras: The Python Deep Learning API: <https://keras.io/>
- Khan, A., 2020. Erişim Tarihi: 25.04.2021. Dataset containing smoking and not-smoking images (smoker vs non-smoker). Mendeley Data: <https://data.mendeley.com/datasets/7b52hhzs3r/1>
- Kim, D., 2020. Deep learning Method For Voice Recognition Model And Voice Recognition Device Based On Artificial Neural Network.
- Kim, S., Nam, J., Ko, B., 2018. Online Tracker Optimization for Multi-Pedestrian Tracking Using a Moving Vehicle Camera. IEEE Access, 48675-48687, doi: 10.1109/ACCESS.2018.2867621.

- Lee, H., aus10powell, & Arsen., 2019. Erişim Tarihi: 24.03.2021. Cigarette Smoker Detection. Kaggle: <https://www.kaggle.com/vitaminc/cigarette-smoker-detection>
- Lei jnen, S., Veen, F., 2020. The Neural Network Zoo. Proceedings, 9.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., . . . Zitnick, C., 2014. Microsoft COCO: Common Objects in Context. European Conference on Computer Vision, 740-755.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., C. Berg, A., 2016. SSD: Single Shot MultiBox Detector. European Conference on Computer Vision, 21-37, doi: 10.1007/978-3-319-46448-0_2.
- Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., Alsaadi, F., 2017. A survey of deep neural network architectures and their applications. Neurocomputing, doi: 10.1016/j.neucom.2016.12.038.
- Lu, Y., Lu, C., Tang, C., 2017. Online Video Object Detection Using Association LSTM. Proceedings of the IEEE International Conference on Computer Vision, 2344-2352, doi: 10.1109/ICCV.2017.257.
- Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W., Zhao, X., Kim, T.-K., 2014. Multiple Object Tracking: A Literature Review. Computer Vision and Pattern Recognition, doi: 10.1016/j.artint.2020.103448.
- MakeML, 2021. Erişim Tarihi: 29.03.2021. Smartphones: <https://makeml.app/datasets/smartphones>
- Medsker, L., Jain, L., 2001. Recurrent Neural Network Design and Applications. Boca Raton London New York Washington, D.C.: CRC Press.
- Mikada, T., Kanno, T., Kawase, T., Miyazaki, T., Kawashima, K., 2020. Suturing Support by Human Cooperative Robot Control Using Deep Learning. IEEE Access, 167739-167746, doi: 10.1109/ACCESS.2020.3023786.
- Mohod, N., Agrawal, P., Madaan, V., 2022. YOLOv4 Vs YOLOv5: Object Detection on Surveillance Videos. In Advanced Network Technologies and Intelligent Computing: Second International Conference, ANTIC 2022, Varanasi, India, December 22-24, Proceedings, Part II. Cham: Springer Nature Switzerland, 654-665.
- Netron, Erişim Tarihi: 10.01.2021. Netron: <https://netron.app/>
- Ojha, S., Sakhare, S., 2015. Image processing techniques for object tracking in video surveillance-a survey. In Pervasive Computing (ICPC), 2015 International Conference on, 1-6, doi: 10.1109/PERVASIVE.2015.7087180.

- Olmos, R., Tabik, S., Herrera, F., 2018. Automatic handgun detection alarm in videos using deep learning. *Neurocomputing*, 66-72.
- Olmos, R., Tabik, S., Lamas, A., Pérez-Hernández, F., Herrera, F., 2019. A binocular image fusion approach for minimizing false positives in handgun detection with deep learning. *Information Fusion*, doi:10.1016/j.inffus.2018.11.015.
- Omeroglu, A., Kumbasar, N., Oral, E., Ozbek, I., 2019. Mask R-CNN Algoritması ile Hangar Tespiti. 27th Signal Processing and Communications Applications Conference (SIU), doi: 10.1109/siu.2019.8806552.
- O'Shea, K., Nash, R., 2015. *An Introduction to Convolutional Neural Networks. Neural and Evolutionary Computing.*
- P. Bernstein, J., Mendez, B., Sun, P., Liu, Y., Shang, Y., 2017. Using deep learning for alcohol consumption recognition. 2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC). doi:10.1109/CCNC.2017.7983278.
- Pain, S., Roy, S., Sarma, M., Samanta, D., 2023. Detection of alcoholism by combining EEG local activations with brain connectivity features and Graph Neural Network. *Biomedical Signal Processing and Control*, 85, 104851.
- Papadamou, K., Papasavva, A., Zannettou, S., Blackburn, J., Kourtellis, N., Leontiadis, I., . . . Sirivianos, M., 2019. Disturbed YouTube for Kids: Characterizing and Detecting Inappropriate Videos Targeting Young Children. *Proceedings of the International AAAI Conference on Web and Social Media*, 522-533.
- Park, S., K. Kim, U. Oh, 2022. Understanding the user preferences in the types of video censorship. *International Journal of Internet, Broadcasting and Communication*, 14(2), 147-161.
- Pérez-Hernández, F., Tabik, S., Lamas, A., Olmos, R., Fujita, H., Herrera, F., 2020. Object Detection Binary Classifiers methodology based on deep learning to identify small objects handled similarly: Application in video surveillance, Knowledge Based Systems, doi:10.1016/j.knosys.2020.105590.
- Perez, M., Avila, S., Moreira, D., Moraes, D., Testoni, V., Valle, E., ... & Rocha, A., 2017. Video pornography detection through deep learning techniques and motion information. *Neurocomputing*, 230, 279-293. doi: 10.1016/j.neucom.2016.12.017.
- Pistols Dataset, Erişim Tarihi: 27.03.2021. Roboflow: <https://public.roboflow.com/object-detection/pistols/1>

- Prasad, P., Pathak, R., Gunjan, V., Rao, H., 2019. Deep Learning Based Representation for Face Recognition. ICCCE, 419-424, doi: 10.1007/978-981-13-8715-9_50.
- Realtime Image & Video Moderation API, Eriřim Tarihi: 15.11.2021. TupuTech: <https://us.tuputech.com/>
- Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You Only Look Once: Unified, Real-Time Object Detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 779-788, doi: 10.1109/CVPR.2016.91.
- Ren, S., He, K., Girshick, R., Sun, J., 2016. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1137 - 1149, doi: 10.1109/TPAMI.2016.2577031.
- Russell, B., Torralba, A., Murphy, K., Freeman, W., 2008. LabelMe: A database and web-based tool for image annotation. International Journal of Computer Vision, 157-173, doi: 10.1007/s11263-007-0090-8.
- Saleem, M., Potgieter, J., Arif, K., 2019. Plant Disease Detection and Classification by Deep Learning. Plants, 8(11):468, doi: doi:10.3390/plants8110468.
- Samal, S., Zhang, Y. D., Gadekallu, T. R., Nayak, R., & Balabantaray, B. K., 2023. SBMYv3: Improved MobYOLov3 a BAM attention-based approach for obscene image and video detection. Expert Systems, e13230. doi: 10.1111/exsy.13230.
- Shetty, D., Varma, J., Navi, S., Ahmed, M., 2020. Diving Deep into Deep Learning: History, Evolution, Types and Applications. The International Journal on Media Management, 2278-3075, doi: 10.35940/ijitee.A4865.019320.
- Shotton, J., Winn, J., Rother, C., Criminisi, A., 2006. TextonBoost: Joint Appearance, Shape and Context Modeling for Multi-class Object Recognition and Segmentation. European Conference on Computer Vision, 1-15, doi: 10.1007/11744023_1.
- Sorin, V., Barash, Y., Konen, E., Klang, E., 2020. Deep Learning for Natural Language Processing in Radiology Fundamentals and a Systematic Review. Journal of the American College of Radiology, 639-648.
- Sun, W., Zheng, B., Qian, W., 2017. Automatic feature learning using multichannel ROI based on deep structured algorithms for computerized lung cancer diagnosis. Computers in Biology and Medicine, 530-539, doi: 10.1016/j.compbimed.2017.04.006.
- řeker, A., Diri, B., Balık, H., 2017. Derin Öğrenme Yöntemleri ve Uygulamaları Hakkında Bir İnceleme. Gazi Mühendislik Bilimleri Dergisi, 47-64.

- Tan, F., 2022. Harmful-Content-Detection. Turkey. Github: <https://github.com/gulsahtan>
- Tao, C., Zhang, J., Wang, P., 2016. Smoke Detection Based on Deep Convolutional Neural Networks. Proceedings - 2016 International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration, ICIICII 2016. doi:10.1109/ICIICII.2016.0045.
- tensorflow-gardener, Erişim Tarihi: 20.04.2021. tensorflow / models. Github: https://github.com/tensorflow/models/tree/master/research/object_detection
- TensorFlow, 2020. TensorFlow: <https://www.tensorflow.org/>
- Theano Development Team, 2016. Theano: A Python framework for fast computation of mathematical expressions. arXiv:1605.02688.
- Toğaçar, M., Ergen, B., 2019. Biyomedikal Görüntülerde Derin Öğrenme ile Mevcut Yöntemlerin Kıyaslanması. Fırat Üniversitesi Mühendislik Bilimleri Dergisi.
- Toğaçar, M., Ergen, B., Sertkaya, M., 2019. Zatürre Hastalığının Derin Öğrenme Modeli ile Tespiti. Fırat Üniversitesi Mühendislik Bilimleri Dergisi.
- Torralba, A., Fergus, R., Freeman, W., 2008. 80 million tiny images: A large data set for nonparametric object and scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1958-1970, doi: 10.1109/TPAMI.2008.128.
- tzutalin/labelImg, Erişim Tarihi: 24.03.2021. GitHub: <https://github.com/tzutalin/labelImg>
- ultralytics/yolov5, Erişim Tarihi: 20.04.2021. GitHub: <https://github.com/ultralytics/yolov5>
- Verma, G., Dhillon, A., 2017. A Handheld Gun Detection using Faster R-CNN Deep Learning. ACM International Conference Proceeding Series, 84-88, doi:10.1145/3154979.3154988.
- Viola, P., Jones, M., 2001. Rapid object detection using a boosted cascade of simple features. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, doi: 10.1109/cvpr.2001.990517.
- Von Ahn, L., Dabbish, L., 2004. Labeling images with a computer game. Conference on Human Factors in Computing Systems Proceedings, 319-326, doi: 10.1145/985692.985733.

- Wang, D., 1998. Unsupervised video segmentation based on watersheds and temporal tracking. IEEE Transactions on Circuits and Systems for Video Technology, 539-546, doi: 10.1109/76.718501.
- Yang, S., Bailey, E., Yang, Z., Ostrometzky, J., Zussman, G., Seskar, I., Kostic, Z., 2020. COSMOS Smart Intersection: Edge Compute and Communications for Bird's Eye Object Tracking. IEEE Annual Conference on Pervasive Computing and Communications Workshops (PerCom), 1-7, doi: 10.1109/PerComWorkshops48775.2020.9156225.
- Yao, B., Yang, X., Zhu, S., 2007. Introduction to a large-scale general purpose ground truth database: Methodology, annotation tool and benchmarks. Lecture Notes in Computer Science, 169-183, doi: 10.1007/978-3-540-74198-5_14.
- Yin, Z., Wan, B., Yuan, F., Xia, X., & Shi, J., 2017. A Deep Normalization and Convolutional Neural Network for Image Smoke Detection. IEEE Access. doi:10.1109/ACCESS.2017.2747399.
- Yuret, D., 2016. Eriřim Tarihi: 10.03.2021. Julia ve Knet ile Derin Öğrenmeye Giriř. Julia ve Knet ile Derin Öğrenmeye Giriř: <http://www.denizyuret.com/2016/09/julia-ve-knet-ile-derin-ogrenmeye-giris.html>
- Zhang, L., Gray, H., Ye, X., Collins, L., Allinson, N., 2019. Automatic individual pig detection and tracking in pig farms. Sensors (Switzerland), doi: 10.3390/s19051188.

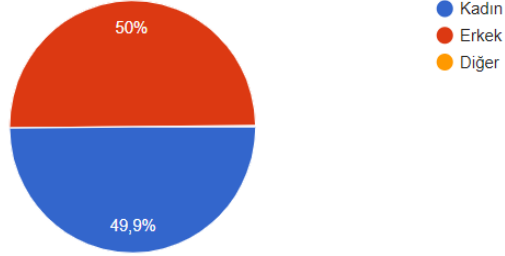
EKLER

EK A. Anket Soruları



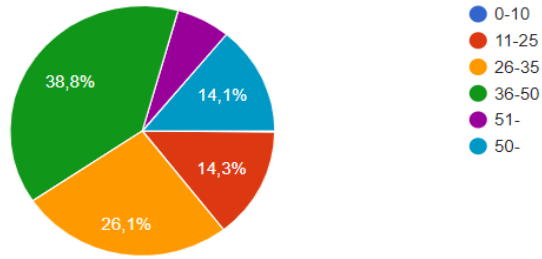
EK A. Anket Soruları

Cinsiyetiniz
688 yanıt



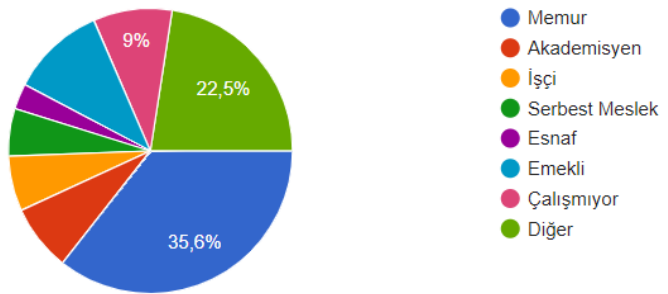
Şekil A.1. Cinsiyetiniz

Yaş Aralığınız
683 yanıt



Şekil A.2. Yaş aralığınız

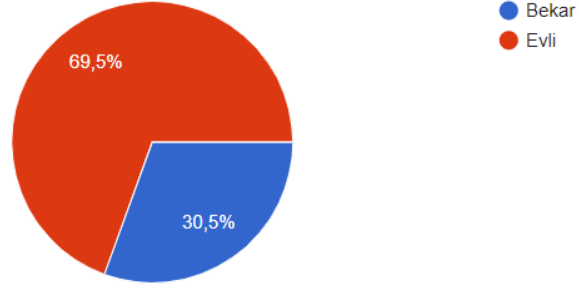
Mesleğiniz
688 yanıt



Şekil A.3. Mesleğiniz

Medeni Durumunuz

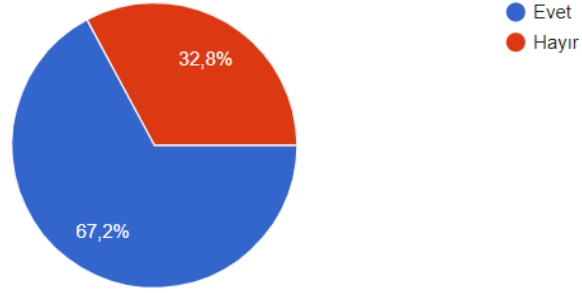
688 yanıt



Şekil A.4. Medeni durumunuz

Çocuğunuz var mı?

688 yanıt



Şekil A.5. Çocuğunuz var mı?

İnternet üzerinde izlediğiniz videolarda kötü unsurları görmekten rahatsız oluyor musunuz?
(Şiddet, uygunsuz görüntüler, alkol, sigara vs.)

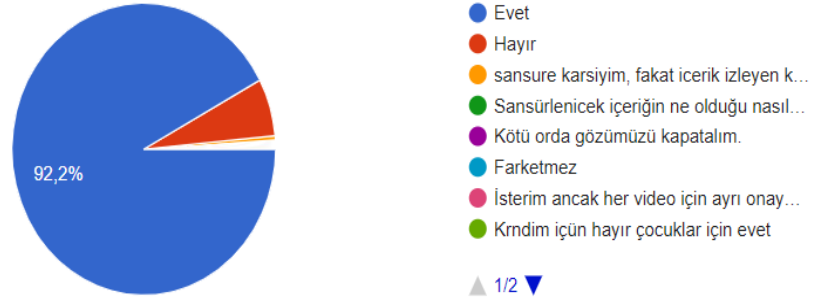
688 yanıt



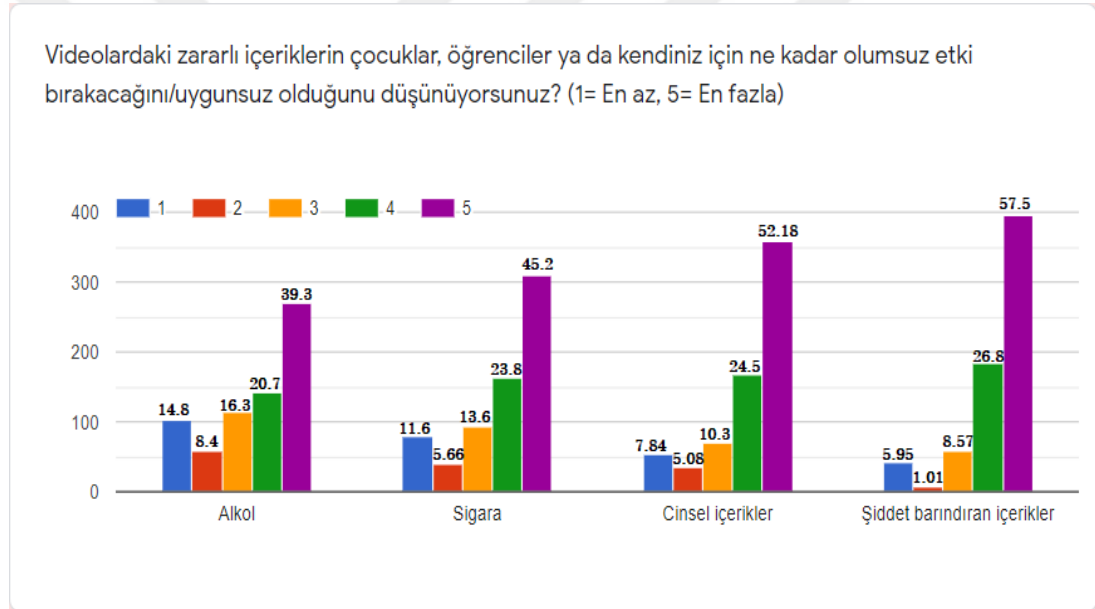
Şekil A.6. Kötü unsurları görmekten rahatsız olma durumu

İnternet üzerinde izlenen herhangi bir videoda çocuklar, öğrenciler ya da kendiniz için zararlı olabilecek içerik/içeriklerin anlık olarak tespit edilip filtrelenmesini ister misiniz?

688 yanıt



Şekil A.7. Kötü unsurların filtrelenmesini ya da filtrelenmemesini gösteren şekil



Şekil A.8. Kötü unsurların oranları

İzlediğiniz içeriklerde çocuklar, öğrenciler ya da kendiniz için olumsuz etki bırakacağını veya uygunsuz olduğunu düşündünüz başka unsurlar var mı?

Uzun yanıt metni

Şiddet, alkol, sigara

1 yanıt

Bazı programlar da şiddet içerikli ,aile içi tartışma ve uygunsuz davranışlar konu edilmekte ve bu yaş sınırı olmaksızın sergilenmektedir.bu gibi yayınların çok olması toplum tarafından izleniyor ve bence zamanla sıradanlaşıyor böylece normalleşiyor ,asla normal olmayan ve ahlak kurallarına aykırı olan durumşiddet ve ahlaksızlık öğretiliyor topluma.. örneğin kadın cinayetleri yayınlandıkca sıklaştı .. teşekkürler..

1 yanıt

Kadına şiddet içerikli film ve diziler,silahların çatışmaların havada uçuştığı film ve diziler,evlilik dışı cinselliğin ön planda olduğu dizi ve filmlerin maalesef çocuklarda olumsuz etkiler bıraktığını düşünüyorum

1 yanıt

Evet diziler çukur , arıza , eskiya dünyaya hükümdar olmaz gibi mafya dizileri çocukları yada gençlerimizi ateşli silahlar kesici aletlere özendiriyor malesef...

1 yanıt

Dizi ve filmlerdeki, kadın erkek ilişkilerindeki cinsel içerikler, küfürlü ve aşırı argo sözler. Türk toplumunun değer yargılarına ters ilişkişer. Eşcinselliği hoş gösteren diziler. Kavgı, dövüş, işkence, kolayca insan öldürme uyuşturucu baronluğu gibi mafya durumlarını normal yaşantının bir parçasıymış gibi göstermek, sanki bu durumların karşılığı yokmuş devlet polis yokmuş, yapanın yaptığı yanına kar kalıyormuş gibi diziler. Örnek Çukur dizisi.

1 yanıt

Şekil A.9. Şiddet içeriklerinin filtrelendiğini isteyen kullanıcı yorumları

Uyuşturucu, silahlı, zengin fakir ayrımı

1 yanıt

Kumar ve diğer oyunları içeren reklamlar

1 yanıt

Toplumu eğitici bilgi ve görgü kuralları

1 yanıt

Mafya, silah, dolandırıcılık, hırsızlık

1 yanıt

Şekil A.10. Şiddet içeriklerinin filtrelendiğini isteyen kullanıcı yorumları

Şiddet içerikli diziler , alkole özendirilen videolar , cinselliğe yönelik videolar uygunsuzdur

1 yanıt

Cinayet,hırsızlık,boşanma,kadına şiddet,

1 yanıt

Şiddet içeren oyunlar ve savaş oyunları

1 yanıt

Şekil A.11. Şiddet içeriklerinin filtrelendiğini isteyen kullanıcı yorumları