



**T.C.
BURSA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

**TEKSTİL ÜRETİMİNDE DERİN ÖĞRENME TABANLI
OTOMATİK HATA TESPİT SİSTEMİ**

YÜKSEK LİSANS TEZİ

Ahmet METİN

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı

Nisan 2024

T.C.
BURSA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

TEKSTİL ÜRETİMİNDE DERİN ÖĞRENME TABANLI
OTOMATİK HATA TESPİT SİSTEMİ

YÜKSEK LİSANS TEZİ

Ahmet METİN
(19376482010)

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı

Tez Danışmanı: Prof. Dr. Haydar ÖZKAN

Nisan 2024





20.04.2016 tarihli Resmi Gazete’de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, Bursa Teknik Üniversitesi’nin abonesi olduğu intihal yazılım programı kullanılarak Lisansüstü Eğitim Enstitüsü’nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

İNTİHAL BEYANI

Bu tezde görsel, işitsel ve yazılı biçimde sunulan tüm bilgi ve sonuçların akademik ve etik kurallara uyularak tarafımdan elde edildiğini, tez içinde yer alan ancak bu çalışmaya özgü olmayan tüm sonuç ve bilgileri tezde kaynak göstererek belgelediğimi, aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim.

Öğrencinin Adı Soyadı: Ahmet METİN

İmzası:



Aileme,

ÖNSÖZ

Bu tez çalışmamda; bana destekleri için danışman hocam Prof. Dr. Haydar ÖZKAN'a ve eğitim hayatım boyunca desteklerini esirgemeyen aileme sonsuz teşekkürlerimi sunuyorum.

Nisan 2024

Ahmet METİN
(Araştırma Görevlisi)

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	vii
İÇİNDEKİLER	viii
KISALTMALAR	x
SEMBOLLER	xii
ÇİZELGE LİSTESİ.....	xiii
ŞEKİL LİSTESİ.....	xiv
ÖZET.....	xvii
SUMMARY	xviii
1. GİRİŞ	1
1.1 Tezin Amacı	2
1.2 Tezin Katkısı	3
1.3 Literatür Araştırması	4
2. DERİN ÖĞRENME ESASLARI.....	17
2.1 Evrişimsel Sinir Ağları.....	18
2.1.1 Filtreler ve evrişim işlemi	18
2.1.2 Dolgu ve adımlama	19
2.1.3 Havuzlama	20
2.1.4 Yaygın doğrusal olmayan aktivasyon fonksiyonları.....	21
2.1.5 Tam bağlantılı katmanlar	24
2.1.6 Düzleştirme	25
2.1.7 Optimizasyon	25
2.1.8 Aşırı uyumluluk	26
2.1.9 Bırakma.....	27
2.1.10 Kayıp fonksiyonu	28
2.1.11 Yığın normalizasyonu	28
3. GÖRÜNTÜ İŞLEME ALGORİTMALARI	29
3.1 Medyan Filtresi	29
3.2 Görüntü Keskinleştirme	30
3.3 Görüntü Doğrusal Kontrast Germe	31
3.4 Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme (CLAHE)	32
4. DERİN ÖĞRENME ALGORİTMALARI	34
4.1 VGG Evrişimsel Ağı	34
4.2 Yoğun Bağlantılı Evrişimsel Ağlar	35
4.3 Üretken Çekişmeli Ağlar.....	37
4.4 Süper Çözünürlüklü Üretken Çekişmeli Ağ.....	38
4.4.1 Karşıt kayıp	39
4.4.2 Algısal kayıp	39
4.4.3 İçerik kaybı	40
4.4.4 Toplam kayıp	40
4.5 Geliştirilmiş Süper Çözünürlüklü Üretken Çekişmeli Ağlar	41

4.6 YOLOv8 Mimarisi	42
4.6.1 Omurga mimarisi	43
4.6.2 Boyun mimarisi.....	44
4.6.3 CSP bağlantı modülü	44
4.6.4 Nesne algılama kafası	45
4.6.5 Eğitim geliştirmeleri	45
5. MATERİYAL VE METOT	46
5.1 Veri Setleri	49
5.1.1 TILDA veri seti	49
5.1.2 Bu çalışmada oluşturulan veri seti	53
5.2 Deney Seti Donanım Sistemi	57
5.2.1 Raspberry Pi 4B	59
5.2.2 Raspberry Pi kamera kartı v1.3	60
5.2.3 Raspberry Pi 12,3 MP sensörlü yüksek kaliteli kamera.....	61
5.2.4 Yüksek başarılı hesaplamalı laboratuvarı ve kişisel bilgisayar.....	62
5.3 Deney Seti Yazılım Sistemi	62
5.3.1 Kütüphane kurulumu.....	62
5.3.2 ESRGAN işlemi	63
5.3.3 Görüntüyü lokal bölgelere ayırma	67
5.3.4 Etiketleme	68
5.3.5 Ön işleme	70
5.3.6 YOLOv8 ile hata tespiti	72
5.3.7 Yeniden yapılandırma	73
5.3.8 Hiperparametre seçimi	74
6. BULGULAR VE TARTIŞMA	78
6.1 Değerlendirme Metrikleri.....	78
6.1.1 YOLOv8 başarımlı metrikleri	78
6.1.2 K-Katlamalı çapraz doğrulama (K-Fold cross validation).....	80
6.1.3 ESRGAN başarımlı metrikleri	80
6.2 Kumaş Hata Tespit Sonuçlarının Elde Edilmesi	84
6.2.1 Kumaşta delik ve kesik hataları	84
6.2.2 Yağ lekeleri ve renk hataları	86
6.2.3 İplik hatası.....	87
6.2.4 Yabancı obje hatası	88
6.3 Uç Cihazlarda Gerçek Zamanlı Çalışma	89
6.4 ESRGAN Boyut Azaltma.....	91
6.5 TILDA Veri Seti ile Sonuçlar	97
6.5.1 TILDA yeniden yapılandırılan görsel sonuçları	112
6.6 Yeni Veri Seti ile Sonuçlar.....	114
6.6.1 Koyu kumaş yeniden yapılandırılan görsel sonuçları	120
6.6.2 Açık kumaş yeniden yapılandırılan görsel sonuçları	123
7. SONUÇLAR VE ÖNERİLER	126
KAYNAKLAR	129
ÖZGEÇMİŞ.....	134

KISALTMALAR

Adagrad	: Uyarlanabilir Gradyan Algoritması
Adam	: Uyarlanabilir Moment Tahmini
AE	: Oto Enkodlayıcılar
AHE	: Geleneksel Uyarlanabilir Histogram Eşitleme
AI	: Yapay Zekâ
BA-TCB	: Aşağıdan Yukarıya Yol Büyütme Aktarım Bağlantı Bloğu
BN	: Yığın Normalizasyonu
CLAHE	: Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme
CMOS	: Tamamlayıcı Metal Oksit Yarı İletkeni
CNN	: Evrimsel Sinir Ağı
CPU	: Merkezi İşlem Birimi
CSI	: Kamera Seri Arayüzü
CSP	: Cross-Stage Partial
CU-Nets	: Birleştirilmiş U-Netler
dB	: Desibel
DCGAN	: Derin Evrimsel Üretken Çekişmeli Ağ
DIoU-NMS	: Mesafe-IoU tabanlı Maksimum Olmayan Bastırma
ESA	: Evrimsel Sinir Ağları
ESRGAN	: Gelişmiş Süper Çözünürlük Üretken Çekişmeli Ağlar
FCCA	: Tam Evrimsel Kanal Dikkat
FCN	: Tam Evrimsel Ağlar
FPN	: Özellik Piramidi Ağı
FPS	: Frame Per Second (Saniyedeki Kare Sayısı)
GAN	: Oto Kodlayıcı ve Üretken Çekişmeli Ağ
GLM	: Genelleştirilmiş Doğrusal Modeller
GPU	: Grafik İşlem Birimi
HPC	: Yüksek Performanslı Hesaplama Laboratuvarı
IoT	: Nesnelerin İnterneti
LBP	: Yerel İkili Örüntü
MAE	: Ortalama Mutlak Hata

MSCDAE	: Çok Ölçekli Evrişimli Denoising Otomatik Kodlayıcı
MSE	: Ortalama Kare Hatası
MSSIM	: Ortalama Yapısal Benzerlik
NMS	: Maksimum Olmayan Bir Bastırma
PRAN-Net	: Ön Çapa Konvolüsyonel Sinir Ağı
PSNR	: Tepe Sinyal-Gürültü Oranı
R-CNN	: Bölgesel Tabanlı CNN
RCT	: Rastgele Kontrollü Çalışma
ReLU	: Doğrultulmuş Lineer Birim
RMSprop	: Kök Ortalama Kare Yayılımı SRGAN
RPDNet	: Tekrarlanan Desen Kusur Ağı
RRDB	: Artık İçinde Artık Yoğun Blok
SGD	: Stokastik Gradyan İnişi
SRGAN	: Süper Çözünürlüklü Üretken Çekişmeli Ağ
SSIM	: Yapısal Benzerlik İndeksi
TIFF	: Etiketli Görüntü Dosyası Formatı
VAE	: Varyasyonel Oto Enkodlayıcılar
VGG	: Visual Geometry Group (Görsel Geometri Grubu)
WACNet	: Geniş ve Kompakt Ağ

SEMBOLLER

β	: Artık ölçekleme parametresi
e	: Euler sayısı
ϕ	: Euler'in totient fonksiyonu
σ	: Sigma parametresi



ÇİZELGE LİSTESİ

Sayfa

Çizelge 5.1 : TILDA verisindeki desensiz kumaş tipleri, hata türleri ve sayıları.....	50
Çizelge 5.2 : TILDA verisindeki desenli kumaş tipleri, hata türleri ve sayıları.....	51
Çizelge 5.3 : ESRGAN modeli seçilen hiperparametre değerleri ve açıklamaları....	75
Çizelge 5.4 : YOLOv8 modeli seçilen hiperparametre değerleri ve açıklamaları.....	76
Çizelge 6.1 : Uç cihazda üretici modele göre hız ve parametre performansı	91
Çizelge 6.2 : Uç cihazların YOLOv8 modeline göre hız performansı	91
Çizelge 6.3 : Üretici kısımdaki RRDB blok sayısının azaltılması ve performansı ...	96
Çizelge 6.4 : Görüntü işleme metotları ile her hatanın sınırlayıcı kutu sonuçları	97
Çizelge 6.5 : Görüntü işleme metotları ile her hatanın sınırlayıcı maske sonuçları.	98
Çizelge 6.6 : Görüntü işleme ile toplam hataların sınırlayıcı kutu sonuçları	99
Çizelge 6.7 : Görüntü işleme ile toplam hataların sınırlayıcı maske sonuçları	99
Çizelge 6.8 : Üretilen veri setleri ile her hata sınıfının sınırlayıcı kutu tahminleri.....	115
Çizelge 6.9 : Üretilen veri setleri ile her hata sınıfının maske kutu tahminleri	115
Çizelge 6.10 : TILDA ağırlıkları ile eğitilen YOLOv8 sınırlayıcı kutu sonuçları	116
Çizelge 6.11 : TILDA ağırlıkları ile eğitilen YOLOv8 sınırlayıcı maske sonuçları	116
Çizelge 6.12 : TILDA ağırlıkları ile yeni kumaş sınırlayıcı kutu sonuçları	117
Çizelge 6.13 : TILDA ağırlıkları ile yeni kumaş sınırlayıcı maske sonuçları.....	118
Çizelge 6.14 : Farklı kumaşlarda toplam hataların sınırlayıcı kutu sonuçları	118
Çizelge 6.15 : Farklı kumaşlarda toplam hataların sınırlayıcı maske sonuçları	119

ŞEKİL LİSTESİ

	<u>Sayfa</u>
Şekil 2.1 : Tam bağlantılı katmanlar	24
Şekil 2.2 : (a) tam bağlantılı katman, (b) bırakma katmanı	27
Şekil 4.1 : VGG19 mimarisi.	34
Şekil 4.2 : DenseNet mimarisinin bağlantı şekli	35
Şekil 4.3 : GAN mimarisinin çalışma düzeni.	36
Şekil 4.4 : SRGAN'da artık bloğundaki BN katmanları kaldırıldı	40
Şekil 4.5 : RRDB bloğu yapısı	41
Şekil 4.6 : YOLOv8 mimarisi.....	42
Şekil 5.1 : Tezin ana aşamaları	46
Şekil 5.2 : TILDA kumaş örnekleri (a) c1r1 (b) c1r3 (c) c2r2 (d) c2r3 (e) c3r1 (f) c3r3 (g) c4r1 (h) c4r3 gösterilmiştir.	49
Şekil 5.3 : TILDA veri setinde kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri.	52
Şekil 5.4 : Düşük çözünürlüklü kamera ile oluşturulan veri setindeki koyu kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri.	54
Şekil 5.5 : Düşük çözünürlüklü kameraya ile oluşturulan veri setindeki açık kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri.	55
Şekil 5.6 : Yüksek çözünürlüklü kamera ile oluşturulan veri setindeki koyu kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri	56
Şekil 5.7 : Yüksek çözünürlüklü kameraya ile oluşturulan veri setindeki açık kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri	57
Şekil 5.8 : Proje de kullanılan kişisel bilgisayar ve hatalı açık kumaş görseli	58
Şekil 5.9 : Proje de kullanılan Raspberry Pi 4b ile kamerası ve hatalı açık renk kumaş görseli.....	59
Şekil 5.10 : Proje de kullanılan Raspberry Pi 4B	60
Şekil 5.11 : Proje de kullanılan Raspberry Pi kamera ve entegresi	61
Şekil 5.12 : IMX477 sensörlü kamera ve PT361060M3MP12 CS montaj lensi.....	61
Şekil 5.13 : Yüksek başarılı bilgisayarda bulunan kütüphane versiyonları.....	63
Şekil 5.14 : ESRGAN eğitim ve test işlemleri	64
Şekil 5.15 : 416x416 piksel çözünürlüğe sahip orijinal görüntü	65
Şekil 5.16 : 1664x1664 piksel çözünürlüğe sahip yapay görüntü	66
Şekil 5.17 : Veri setinden rastgele seçilmiş orijinal görsel.....	67
Şekil 5.18 : Orijinal görüntünün çözünürlüğü artırılarak ve lokal bölgelere ayrılarak elde edilmiş hali	68
Şekil 5.19 : (a) Orijinal görsel, (b) bölütlenmiş görsel.....	69

Şekil 5.20 : Lokal bölgelere ayrılmış görüntüdeki hatalarının koordinatlarını çıkarma.....	70
Şekil 5.21 : TILDA veri setinde, (a) açık renk kumaş veya aydınlatılmış delik hatası görseli, (b) koyu renk kumaş veya aydınlatma yapılmadan alınmış delik hatası görseli.....	71
Şekil 5.22 : (a) Orijinal görüntü, (b) CLAHE yöntemi sonucu, (c) görüntü doğrusal kontrast germe yöntemi sonucu ve (d) ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe yöntemlerinin ortak kullanımı	72
Şekil 5.23 : YOLOv8 modelinden alınan görsel sonuçların yeniden yapılandırılmış hali.....	74
Şekil 6.1 : Görsel bilgi doğruluğunun çalışma mekanizması	82
Şekil 6.2 : (a) Orijinal görüntüler ile (b) CLAHE ile elde edilen görüntüler ile (c) görüntü doğrusal kontrast germe ile elde edilen görüntüler ile ve (d) sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe ile elde edilen görüntüler ile eğitilen modellerin delik hatasına göre sonuçları.	85
Şekil 6.3 : (a) Orijinal görüntüler ile (b) CLAHE ile elde edilen görüntüler ile (c) görüntü doğrusal kontrast germe ile elde edilen görüntüler ile ve (d) sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe ile elde edilen görüntüler ile eğitilen modellerin yağ lekesine karşı sonuçları	86
Şekil 6.4 : (a) Orijinal görüntüler ile (b) CLAHE ile elde edilen görüntüler ile (c) görüntü doğrusal kontrast germe ile elde edilen görüntüler ile ve (d) sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe ile elde edilen görüntüler ile eğitilen modellerin iplik hatasına karşı sonuçları	87
Şekil 6.5 : (a) Orijinal görüntüler ile (b) CLAHE ile elde edilen görüntüler ile (c) görüntü doğrusal kontrast germe ile elde edilen görüntüler ile ve (d) sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe ile elde edilen görüntüler ile eğitilen modellerin yabancı obje hatasına karşı sonuçları	88
Şekil 6.6 : 1 RRDB bloğuna sahip üreticinin toplam iterasyon sayısına karşılık gelen kayıp fonksiyonu grafiği.....	92
Şekil 6.7 : 2 RRDB bloğuna sahip üreticinin kayıp fonksiyonu	93
Şekil 6.8 : 4 RRDB bloğuna sahip üreticinin kayıp fonksiyonu	93
Şekil 6.9 : 8 RRDB bloğuna sahip üreticinin kayıp fonksiyonu	94
Şekil 6.10 : Orijinal RRDB bloğuna sahip üreticinin kayıp fonksiyonu	95
Şekil 6.11 : Sınırlayıcı kutunun F1 skoru ve güven düzeyleri arasındaki ilişkisini gösteren grafik	100
Şekil 6.12 : Sınırlayıcı kutunun kesinlik ve güven arasındaki ilişkisini gösteren grafik.....	101
Şekil 6.13 : Sınırlayıcı kutunun kesinlik ve duyarlılık arasındaki ilişkisini gösteren grafik.....	102
Şekil 6.14 : Sınırlayıcı kutunun duyarlılık ve güven arasındaki ilişkisini gösteren grafik.....	103
Şekil 6.15 : Karışıklık matrisini gösteren grafik.....	105
Şekil 6.16 : YOLOv8 modelini veri dağılımı sonuçlarına ait grafikleri (a) bar grafik, (b) ısı haritası, (c) ve (d) dağılım grafikleri	106

Şekil 6.17 : YOLOv8 sonuçlarına ait epok sayısına karşılık a) kayıp fonksiyonu değişimi b) başarımlı değişimi	107
Şekil 6.18 : Sınırlayıcı maskenin F1 skoru ve güven düzeyleri arasındaki ilişkisini gösteren grafik	108
Şekil 6.19 : Sınırlayıcı maskenin kesinlik ve güven düzeyleri arasındaki ilişkisini gösteren grafik	109
Şekil 6.20 : Sınırlayıcı maskenin kesinlik ve duyarlılık arasındaki ilişkisini gösteren grafik	110
Şekil 6.21 : Sınırlayıcı maskenin duyarlılık ve güven düzeyi arasındaki ilişkisini gösteren grafik	111
Şekil 6.22 : (a) TILDA verisinde bulunan orijinal delik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen delik hatası görüntüsü	112
Şekil 6.23 : (a) TILDA verisinde bulunan orijinal yağ lekesi hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen yağ lekesi hatası görüntüsü	113
Şekil 6.24 : (a) TILDA verisinde bulunan orijinal iplik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen iplik hatası görüntüsü	113
Şekil 6.25 : (a) TILDA verisinde bulunan orijinal obje hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen obje hatası görüntüsü	114
Şekil 6.26 : (a) Koyu Kumaş verisinde bulunan orijinal delik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen delik hatası görüntüsü	121
Şekil 6.27 : (a) Koyu Kumaş verisinde bulunan orijinal iplik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen iplik hatası görüntüsü	121
Şekil 6.28 : (a) Koyu Kumaş verisinde bulunan orijinal yağ lekesi hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen iplik hatası görüntüsü	122
Şekil 6.29 : (a) Koyu Kumaş verisinde bulunan orijinal obje hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen obje hatası görüntüsü	122
Şekil 6.30 : (a) Açık Kumaş verisinde bulunan orijinal delik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen delik hatası görüntüsü	123
Şekil 6.31 : (a) Açık Kumaş verisinde bulunan orijinal iplik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen iplik hatası görüntüsü	124
Şekil 6.32 : (a) Açık Kumaş verisinde bulunan orijinal yağ lekesi hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen yağ lekesi hatası görüntüsü	124
Şekil 6.33 : (a) Açık Kumaş verisinde bulunan orijinal obje hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen obje hatası görüntüsü	125

TEKSTİL ÜRETİMİNDE DERİN ÖĞRENME TABANLI OTOMATİK HATA TESPİT SİSTEMİ

ÖZET

Kumaş hata tespiti, tekstil kalite güvencesi kapsamında üretim standartlarını korumak için tekstil malzemelerindeki düzensizlikleri veya anormallikleri tespit etmeyi amaçlayan önemli bir alandır. Bu hataların tespiti manuel denetimlerle başlamıştır. Bilgisayar görmesinin yaygınlaşmasıyla hata tespit algoritmaları üretilmeye başlanmıştır ve başarımları günden güne artmaktadır. Eş zamanlı olarak istatistiksel metodolojiler ve özellik çıkarımıyla desteklenen desen tanıma algoritmaları, normal ve kusurlu kumaş desenlerini ayırt edebilir duruma gelmiştir. Son yıllarda, özellikle Evrişimli Sinir Ağları (CNN'ler) ve Üretken Çekişmeli Ağlar (GAN'lar) olmak üzere derin öğrenme tekniklerinin entegrasyonu, karmaşık kalıpların ve anormalliklerin analizinde önemli oranda yüksek seviyede doğruluk, verimlilik ve ölçeklenebilirlik sunarak yapı hatası tespitinde devrim yaratmıştır. Yine de bu alanda, kumaş tipolojilerine özgü değişkenliği, ortam aydınlatmasındaki bozulmaları ve ölçeklenebilirlik hususlarını kapsayan zorluklar devam etmektedir.

Bu tezde, temel kumaş hatası örnekleri için yeni bir veri seti (YVS) oluşturulmuş ve ek olarak da TILDA veri setinden yararlanılmıştır. YVS ve TILDA veri setindeki görüntüler 416x416 boyutundadırlar. Yüksek çözünürlüklü kamera ile düşük çözünürlüklü kamera arasındaki farkları gidermek, kumaş üzerindeki hataları ayrıntılı görebilmek ve özellikle de küçük boyutlu hataların gözden kaçmasını engellemek için görüntüler gelişmiş süper çözünürlüklü üretken çekişmeli ağ (ESRGAN) modeli ile 4 kat büyütülmüştür. Bu işlemi gerçekleştirebilmek için öncelikli olarak görüntüler 4 kat küçültülerek ESRGAN modeli ile eğitilmiş süper çözünürlüklü yeni görüntüler elde edilmiştir. Eğitim sonunda elde edilen model ağırlıkları kullanılarak, test aşamasında 416x416 boyutundaki orijinal görüntüler, 4 kat büyütülerek 1664x1664 boyutunda yeni süper çözünürlüklü görüntüler elde edilmiştir. Böylece küçük boyutlardaki hatalar daha görünür hale gelmiştir. Daha sonra 1664x1664 görüntü 4x4 matris şeklinde 416x416 boyutlu 16 lokal görüntüye ayrılmıştır. Lokal bölgelere ayrılmış görüntüler, sırasıyla YOLOv8 modeli ile hata tespit işlemine tabi tutulmuş sonrasında test sonuçları tekrar birleştirilerek orijinal görüntü üzerindeki hatalar, global görüntü üzerinde raporlanmıştır. YOLOv8 için etiketleme işleminde doğruluk oranını artırmak ve hatayı doğru tanımlayabilmek için segmentasyon metodu kullanılmıştır. Eğitim ve test işlemleri TILDA veri seti ile ve YVS için ayrı ayrı gerçekleştirilmiştir. Sınırlayıcı maske mAP50 değerleri TILDA için %74.4, YVS'daki koyu ve açık kumaşlar için sırasıyla %92.2 ve %70.7 olarak elde edilmiştir. Ayrıca TILDA veri seti ile gerçekleştirilen ESRGAN ve YOLOv8 eğitim model ağırlıkları YVS modelinde kullanılarak da veriler test edilmiştir. Bu durumda koyu renkli kumaş veri kümesinde sınırlayıcı kutu ve maske sınırlayıcısı %99.5'lik bir mAP50 değerine ulaşırken, açık renkli kumaş veri kümesi için %90'lık bir mAP50 değeri elde edilmiştir. Önerilen sistem ile sade renkli kumaşlarda hata tespitinin başarılı bir şekilde gerçekleştirilebildiği ortaya çıkmıştır. Bundan sonraki çalışmalarda kumaş motifleri ve karmaşık desenlerin olduğu kumaşlardaki hataların tespiti için de araştırmalar yapılması gerekliliği ortaya çıkmıştır.

Anahtar kelimeler: Kumaş Üretiminde Hata Tespiti, Derin Öğrenme, Görüntü İşleme

DEEP LEARNING BASED AUTOMATED FAULT DETECTION SYSTEM IN TEXTILE PRODUCTION

SUMMARY

Fabric defect detection is an important area in textile quality assurance that aims to detect irregularities or abnormalities in textile materials to maintain production standards. The detection of these defects started with manual inspections. With the widespread use of computer vision, defect detection algorithms have started to be produced and their performance is increasing day by day. Simultaneously, pattern recognition algorithms supported by statistical methodologies and feature extraction have been able to distinguish between normal and defective fabric patterns. In recent years, the integration of deep learning techniques, in particular Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs), has revolutionised fabric defect detection, offering significantly higher levels of accuracy, efficiency and scalability in the analysis of complex patterns and anomalies. Yet challenges remain in this area, including variability inherent to fabric typologies, distortions in ambient lighting and scalability considerations. In this thesis, a new dataset (YVS) was created for basic fabric defect samples and the TILDA dataset was additionally utilised. The images in the YVS and TILDA datasets have a size of 416x416. In order to eliminate the differences between the high-resolution camera and the low-resolution camera, to see the defects on the fabric in detail and especially to prevent small-sized defects from being missed, the images were enlarged by a factor of 4 with the enhanced super-resolution generative contention network (ESRGAN) model. In order to perform this process, the images were first reduced by a factor of 4 and new super-resolution images trained with the ESRGAN model were obtained. Using the model weights obtained at the end of the training, new super-resolution images of 1664x1664 size were obtained by enlarging the original images of 416x416 size by 4 times in the test phase. Thus, small errors became more visible. Then, the 1664x1664 image was divided into 16 local images of 416x416 size as a 4x4 matrix. The images divided into local regions were subjected to the error detection process with the YOLOv8 model, respectively, and then the test results were recombined and the errors on the original image were reported on the global image. For YOLOv8, a segmentation method was used to increase the accuracy of the labelling process and to correctly identify the error. Training and testing procedures were performed separately for TILDA and YVS. The bounding mask mAP50 values were 74.4% for TILDA and 92.2% and 70.7% for dark and light fabric in YVS, respectively. In addition, the ESRGAN and YOLOv8 training model weights performed with the TILDA dataset were also tested in the YVS model. In this case, the bounding box and mask delimiter achieved a mAP50 value of 99.5% for the dark coloured fabric dataset, while a mAP50 value of 90% was obtained for the light coloured fabric dataset. It is concluded that the proposed system can successfully detect defects in plain coloured fabrics. In future studies, it is necessary to carry out researches for the detection of defects in fabrics with fabric motifs and complex patterns.

Keywords: Defect Detection in Fabric Production, Deep Learning, Image Processing

1. GİRİŞ

Tekstil endüstrisi, ilkel el dokuması kumaşlardan karmaşık tasarımlar ve desenler üreten son derece gelişmiş makinelere dönüşerek yüzyıllardır insan uygarlığının temel taşı olmuştur. Bu gelişmelere rağmen kumaş kusurları, tekstil üretiminin kaçınılmaz bir unsuru olmayı sürdürmektedir ve üreticiler, distribütörler ve tüketiciler için zorluklar yaratmaktadır. Kumaştaki bir kusur, yalnızca kumaşın estetik çekiciliğini bozmayıp, aynı zamanda fonksiyonel özelliklerini, dayanıklılığını ve genel kalitesini de etkileyebilmektedir. Bu nedenle kumaş kusurlarını anlamak, tanımlamak ve azaltmak; tekstil ürünlerinin bütünlüğünü ve pazarlanabilirliğini sağlamak açısından çok önemlidir. Kumaşlarda kusurların varlığı; hammadde tutarsızlıkları, üretim süreçleri, çevre koşulları ve insan hataları dâhil olmak üzere çeşitli faktörlerden kaynaklanabilmektedir. Bu kusurlar; delikler, lekeler, yırtıklar, düzensiz boyama ve dokuma tutarsızlıkları gibi sayısız biçimde ortaya çıkabilmektedir. Her kusur tipinin farklı nedenleri ve sonuçları olabilmektedir; bu da tespit, analiz ve çözüm için özel bilgi ve teknikler gerektirmektedir. Ekonomik açıdan bakıldığında kumaş kusurları; ürünün geri çağırılması, iade edilmesi ve itibarın zarar görmesi nedeniyle üreticiler için önemli mali kayıplara yol açabilmektedir (Kumar, 2008). Tüketiciler için kumaş kusurları; memnuniyetsizliğe, ürün ömrünün kısalmasına ve potansiyel sağlık ve güvenlik risklerine neden olabilmektedir (Ngan ve diğ., 2011).

Hassasiyet, kalite ve verimliliğin üstün olduğu tekstil üretimi alanında yapay zekanın (AI) ortaya çıkışı; inovasyon, optimizasyon ve mükemmelliğin damgasını vurduğu dönüştürücü bir çağın habercisi olmaktadır. Bu sektördeki sayısız yapay zekâ uygulaması arasında kalite güvencesini, operasyonel verimliliği ve tüketici memnuniyetini artırmak için geliştirilmiş algoritmalar, sinir ağlarından ve makine öğrenimi yöntemlerinden yararlanan kumaş hatası tespiti çok önemli bir kavram olarak ortaya çıkmaktadır. Manuel incelemelere, görsel değerlendirmelere ve subjektif değerlendirmelere dayanan geleneksel yöntemler; ölçeklenebilirlik, doğruluk, tutarlılık ve verimlilik açısından sınırlamalar sergilemektedir ve

dolayısıyla bu kısıtlamaları aşabilecek yenilikçi ve akıllı çözümlere yönelik acil bir talebi teşvik etmektedir (Rasheed ve diğ., 2020).

Makine öğrenmesi, derin öğrenme, bilgisayar görmesi ve sinir ağı mimarilerini kapsayan yapay zekâ, kumaş kusur tespitinde inovasyon için bir hızlandırıcı olarak ortaya çıkmaktadır. Yapay zekâ algoritmaları; kumaş özellikleri, desenler, dokular ve kusur belirtilerini içeren geniş veri kümelerinden yararlanarak gerçek zamanlı, otomatik ve doğru kusur tanımlama, sınıflandırma ve iyileştirme işlemlerini kolaylaştırmaktadır. Öncü algoritmalar arasında evrişimli sinir ağları (CNN), YOLO ve Faster R-CNN; çeşitli ağlardaki yapı kusurlarını tespit etme, analiz etme ve azaltma konusunda benzersiz başarı, ölçeklenebilirlik ve çok yönlülük sergilemektedir (Yue ve diğ., 2022).

Yapay zekâ yöntemleri; hesaplama teknolojileri ve sensör entegrasyonu paradigmalarındaki son gelişmeleri, kumaş hatası tespitinde yeni uygulamaları teşvik etmektedir. Eş zamanlı olarak uç cihazlardaki bilgi işlem, gerçek zamanlı işleme ve bulut tabanlı analitiklerdeki gelişmeler; üreticilere yapay zekâ odaklı kalite kontrol sistemlerini uygulama olanağı tanımaktadır, böylece operasyonel verimliliği, ürün kalitesini ve tüketici memnuniyetini artırırken israfı, geri çağırımları ve itibar risklerini en aza indirmektedir (Song ve diğ., 2021). Kumaş kusur tespitindeki yapay zekâ uygulamaları; tekstil endüstrisinde teknoloji, inovasyon ve kalite güvencesinin birleşiminin somut örneğidir. Yapay zekâ odaklı çözümler; geleneksel sınırları, yöntemleri ve sınırlamaları aşarak mükemmellik, verimlilik ve tüketici odaklılıkla karakterize edilen bir geleceğin müjdesini vermekte ve böylece sürekli gelişen küresel ortamda tekstil endüstrisinin kalıcı mirasını, geçerliliğini ve dayanıklılığını sağlamaktadır.

1.1 Tezin Amacı

Çağdaş endüstriyel uygulamalarda kusur tespit yöntemlerinin doğruluğu çok önemlidir. Endüstriler ilerledikçe, kumaş hatası tespitiyle ilgili zorluklar giderek daha karmaşık hale gelmekte ve karmaşık hesaplamalı çözümler gerektirmektedir. Bu tez, kumaş hatası tespit paradigmalarında yüksek ve düşük çözünürlüklü görüntüleme sistemleri arasındaki çözünürlük eşitsizliklerini gidermeye odaklanan araştırmaya girişmektedir. Tüm bunlara ek olarak gerçek zamanlı çalışmada elde edilen görüntülerin boyutlarının bir yerden sonra depolanamayacak kadar fazla

olması, görüntülerin düşük çözünürlüklü depolanması ve istenildiğinde büyütülüp yüksek başarımlı sonuçlar elde edilmesi de amaçlanmaktadır. Ayrıca kumaş üzerinde dikkatli bakılınca ancak fark edilebilecek küçüklükte hataların tespiti de görüntülerin büyütülerek analizlerinin gerçekleştirilmesi önem arz etmektedir. Bu kapsamda, gelişmiş süper çözünürlük üretken çekişmeli ağların (ESRGAN) konuşlandırılması yer almaktadır. Bu tez, ESRGAN'ın yeteneklerinden yararlanarak, yakalanan görüntülerin çözünürlüğünü artırmayı, böylece kusur görünürlüğünü ve netliğini artırmayı amaçlamaktadır. ESRGAN geliştirmesinin ardından iyileştirilen görüntüler, daha fazla iyileştirme için lokal bölgelere bölünerek ayrıntılı bir segmentasyon ve etiketleme sürecinden geçirilmiştir. Bölümlere ayrılmış bu görüntüler daha sonra titizlikle seçilmiş bir dizi görüntü işleme yönteminden geçerek farklı durumlara karşı hassasiyeti kontrol edilmiştir. Bu ön işleme yöntemlerinden sonra tüm iyileştirilen görüntüler gelişmiş YOLOv8 modeli kullanılarak hata tahmini test aşamasına tabi tutulmuş ve yüksek başarımlı sonuçlar elde edilmiştir. Bu tespit aşamasında veriler YVS ve TILDA veri seti ile eğitilmiş sonrasında ise eğitilen model ağırlıkları oluşturulan veri setlerinde optimum kusur tanımlama doğruluğu ve verimliliği sağlanarak test edilmiştir. ESRGAN'ın çözünürlük iyileştirmesi, görüntüleri bölütleme, görüntü işleme teknikleri ve YOLOv8'in öngörü becerisinin uyumlu bir şekilde entegrasyonu yoluyla bu tez, hesaplamalı görüntüleme ve endüstriyel kalite kontrolünde yeni ve az maliyetli bir metot belirleyerek kumaş hatası tespit paradigmalarını artırmayı amaçlamaktadır.

1.2 Tezin Katkısı

Kumaş kusur tespiti, son yıllarda derin öğrenme yöntemlerinin entegrasyonu ile önemli ilerlemeler kaydetmiştir. Bu tez çalışmasında, TILDA veri seti ve YVS kullanarak kumaş hatası tespiti detaylı bir şekilde incelenmiştir. Araştırmanın temel amaçlarından biri, TILDA verisinin ipek kumaşa özgü özelliklerinin, oluşturulan YVS kumaş hatalarına karşı nasıl bir performans göstereceğini araştırmaktır. Tezin temel katkısı, düşük maliyetli ve düşük çözünürlüklü kameraların, yüksek çözünürlüklü kameralara rakip olabilecek performans elde etmesini sağlayan bir yöntem sunmasıdır. Bu yöntem, ESRGAN modelinin hesaplama gücünden yararlanarak düşük çözünürlüklü görüntülerin kalitesini önemli ölçüde artırmaktadır. Modelin daha hızlı optimuma ulaşması için önce TILDA veri seti ile eğitim

gerçekleştirilmiş, ardından TILDA ve YVS ile test edilmiştir. Orijinal ESRGAN modeli ile görüntü boyutu 4 katına çıkartılmıştır. Bu sayede gözle görülmesi zor olabilecek hatalar kolayca belirlenebilmiştir. Bu algoritmaların yeteneklerinden yararlanılarak yapılan çalışma, kumaş kusur tespitinde yüksek doğruluk ve verimlilik elde etmeyi amaçlamaktadır. Tez çalışması, teorik katkılara ek olarak, donanım uygulamalarına da öncelik vermiştir. Bu amaçla, iki kamera arasındaki 5 kata yakın maliyet farkının başarıdan ödün vermeden giderilmesi hedeflenmiştir. Araştırma, veri setinin hazırlanmasından, yüksek ve düşük performanslı bilgi işlem kaynaklarının eğitim ve test için kullanılmasına kadar titizlikle aşamalara bölünmüştür. Bu yapılandırılmış yaklaşım, gerçek zamanlı senaryolarda kusur tespit algoritmalarının etkinliğini değerlendirmek için uç cihazların kullanıldığı zorlu test aşamalarıyla sonuçlanmıştır.

1.3 Literatür Araştırması

Kumaş kusuru tespitinde hangi yöntemin kullanılması gerektiği, çeşitli faktörlere bağlıdır ve genellikle uygulamanın gereksinimlerine ve mevcut veri setinin özelliklerine bağlı olmaktadır. Görüntü işleme, öznitelik çıkarımı ve makine öğrenimi yöntemleri genellikle birlikte kullanılır ve kumaş kusurlarını tespit etmek için etkili bir şekilde bir araya getirilebilmektedirler. Görüntü işleme yöntemleri, kumaş görüntülerini işleyerek belirli özellikleri çıkarma ve desenleri belirlemek için kullanılmaktadır. Öznitelik çıkarımı, bu işlenmiş görüntülerden belirli özniteliklerin (örneğin, kenarların yoğunluğu, desenlerin tekrarlanma oranı) çıkarılmasını sağlamaktadır. Makine öğrenimi algoritmaları, bu özniteliklerin kullanılmasıyla eğitilmekte ve kumaş kusurlarını tespit etmek için gerekli sınıflandırma görevini yerine getirmektedir. Bu bölümde daha öncesinde yapılan bu çalışmalar özetlenmiştir.

Priya ve diğ. (2011), görüntüyü bit düzlemlerine ayırarak başlayan yeni bir yaklaşım önermektedir. Düşük dereceli bit düzlemlerinin kusur konumu ve şekli hakkında hayati bilgiler içerdiğini raporlamışlardır. Sonrasında yazarlar bu kusurları görüntü içinde tam olarak bulmak için matematiksel morfoloji tekniklerini kullanmaktadırlar. Önerilen algoritma basit, verimli ve otomasyon için uygun olmaktadır. Matematiksel karmaşıklığa sahip diğer yöntemlerin aksine, hesaplama süresinde önemli iyileştirmeler de sunmaktadır.

Shi ve diğ. (2011) yaptıkları çalışmalarında, kumaş hatalarının tespiti için yerel kontrast sapmasına (LCD) dayalı yeni bir yaklaşım önerilmektedir. LCD, analiz edilen görüntü ile aynı kumaşın hatasız bir görüntüsü arasındaki dört yöndeki kontrast farkını ölçmektedir. Ayrıca hata segmentasyonu için iki seviyeli bir eşik fonksiyonu ile birlikte kullanılmaktadır. LCD yaklaşımı, dokuma dokularının ve arka plan gürültüsünün etkisini en aza indirirken kumaş kusurlarındaki kontrastı artırmaktadır. Deneysel bulgular, hesaplama karmaşıklığını azaltırken kumaş kusurlarını doğru bir şekilde bulmadaki etkinliğini göstermektedir, bu da onu gerçek zamanlı kusur algılama uygulamaları için çok uygun hale getirmektedir.

Raheja ve diğ. (2013), iki yaygın yöntemi karşılaştırarak kumaş kusurlarının tespiti için yeni bir yaklaşım sunmaktadır: Gri Seviye Eş-Oluşum Matrisi (GLCM) ve Gabor filtresi. GLCM, doku istatistiklerini çıkarmak için kullanılır ve bir sinyal grafiği oluşturmak için bunları pikseller arası mesafeye karşı çizmektedir. Bu arada, Gabor filtresi yaklaşımı, konvolüsyon yoluyla kumaş kusurlarını yakalamak için çeşitli ölçek ve yönlere sahip bir filtre bankası oluşturmaktadır. TILDA'nın tekstil doku veritabanından alınan kumaş görüntüleri üzerinde yapılan deneylerde, GLCM yöntemi, kusur tespit doğruluğu ve hesaplama verimliliği açısından Gabor filtresi yaklaşımından daha iyi performans göstermiştir.

Rebhi ve diğ. (2016) yaptıkları çalışmalarında, yerel homojenlik ve matematiksel morfoloji ilkelerini birleştiren yeni bir kumaş hatası tespit algoritması tanıtılmaktadır. Başlangıçta algoritma, H-görüntüsü olarak adlandırılan bir homojenlik görüntüsü oluşturmak için her pikselin yerel homojenliğini hesaplamaktadır. Bu işlem, kumaş içindeki potansiyel kusurlu alanları etkili bir şekilde vurgulamaktadır. Daha sonra, en uygun eşik değerini belirlemek için H-görüntüsü için klasik bir histogram hesaplanmaktadır ve H-görüntüsünün ikili bir gösterime dönüştürülmesi kolaylaştırılmıştır. Bu ikili görüntü, kusurlu alanları ve arka plan dokularını ayırt etmek için çok önemli olmaktadır. Bunu takiben, bir yapılandırma elemanı kullanılarak ikili görüntüye açma ve kapama gibi bir dizi morfolojik işlem uygulanmaktadır. Bu işlemler kusurlu alanları iyileştirir ve arka plan gürültüsünü ortadan kaldırmaya yardımcı olarak kusur tespitinin doğruluğunu artırmaktadır. Son olarak, algoritma işlenmiş görüntüden kusur bölgelerini çıkararak kapsamlı bir tespit sonucu sağlamaktadır.

Deotale ve diğ. (2019) yaptıkları çalışmalarında, görüntü işlemede doku analizi tekniklerinden yararlanan bir kumaş hatası tespit algoritması tanıtılmaktadır. Doku, görüntü analizi ve örüntü tanımda önemli bir unsurdur ve önerilen algoritma ilk olarak GLCM ve Gabor dalgacık çıkarımı kullanarak kumaş görüntüsünden doku özelliklerini çıkarmaktadır. Bu teknikleri birleştiren algoritma, kumaş görüntüsünü etkili bir şekilde yeniden yapılandırırken potansiyel kusurlu bölgeleri de vurgulamaktadır. Kumaş görüntüsünü kusurlu veya kusurlu olmayan olarak sınıflandırmak için rastgele bir karar ormanı (RDFs) sınıflandırıcısı kullanılmaktadır. RDF'ler hem sürekli hem de ayrık değişkenleri ele alma konusundaki çok yönlülüğü, aşırı uyuma karşı direnci ve büyük veri kümelerini işlemedeki verimliliği nedeniyle seçilmektedir.

Ashiba ve diğ. (2023), kumaş hatası tespitine odaklanarak doku görüntü analizi ve yüzey hatası tespiti için çift amaçlı bir kıyaslama veri seti sunmaktadır. İki çerçeve önerilmiştir: SCFC ve AHSFC. SCFC, ön işleme için Kontrast Sınırlı Histogram Eşitleme (CLAHE), ardından Optimum Eşikleme (OT) segmentasyonu ve sınıflandırma için özellik çıkarma kullanmaktadır. AHSFC, iyileştirme için A Trous Algoritması ile Homomorfik Yöntemi (HM) birleştirir, ardından OT segmentasyonu ve algılama için özellik çıkarma uygulanmaktadır. Performans ölçütleri arasında entropi, ortalama gradyan, kontrast, Sobel kenar büyüklüğü, hassasiyet, özgüllük, kesinlik, doğruluk ve tanımlama süresi yer almaktadır. Her iki teknik de kumaş hatası tespitinde başarı gösterirken, ikincisi ilkinden daha iyi performans göstermiştir.

Derin öğrenme ile kumaş hatası tespiti, tekstil malzemelerindeki kusurları, düzensizlikleri veya öznitelik çıkarımını otomatik olarak tanımlamak ve sınıflandırmak için gelişmiş sinir ağı mimarilerinin uygulanmasını içermekte ve tekstil endüstrisinde kalite kontrolü için sofistike ve verimli bir çözüm sağlamaktadır.

Jing ve diğ. (2019), CNN kullanarak otomatik kumaş hatası tespiti için bir yaklaşım sunmaktadır. Bu yaklaşım üç ana aşamadan oluşmaktadır: kumaş görüntülerinin ek açıklamalarla yerleştirilmiş yamalar halinde segmentasyonu, önceden eğitilmiş bir derin CNN üzerinde transfer öğrenme ve inceleme aşamasında kusur kategorisi ve konumu hakkında bilgi sağlayan kusur tanımlama. Aynı kumaş veri tabanları kullanılarak GLCM, RCT, LBP, LeNet, AlexNet ve VGG16 gibi yöntemlerle yapılan karşılaştırmalı değerlendirmeler, yöntemin üstün performansını vurgulamaktadır.

Çalışma özellikle, GLCM gibi manuel özellik tabanlı yöntemlerin boyutsallık sorunlarıyla karşılaşabileceğini ve %89-93 doğruluk aralığına ulaşabileceğini vurgulamaktadır. Buna karşılık önerilen yaklaşım da dahil olmak üzere transfer öğrenme yöntemleri %93-98 aralığında daha yüksek doğruluk elde etmektedir. Önerilen yaklaşım, en yüksek ortalama doğruluğa sahip yöntem olarak öne çıkmaktadır.

Liu ve diğ. (2019), hesaplama işleminin sınırlı olduğu platformlarda verimli bir şekilde yürütülmesi gereken tekstil endüstriyel ortamlarında kumaş kusurlarının tanımlanmasıyla ilgili zorluklara odaklanmıştır. Bu zorlukların üstesinden gelmek için araştırma, evrimsel bir ağı özellik haritalarındaki belirli aktivasyonlardan sorumlu girdi modellerini anlamayı amaçlayan yeni bir yöntem sunmaktadır. Bu görselleştirme tekniğinden yararlanan çalışma, tanıma doğruluğunu korurken hesaplamayı optimize etmeye odaklanarak orijinal VGG16 sinir ağı mimarisini yeniden şekillendirmiştir. Bir dizi geliştirmeden sonra, orijinal modele kıyasla hesaplama yükünde %23 ve parametrelerde %8,9 önemli bir azalma sergileyen yeni bir evrimsel ağ geliştirilmiştir. Bu geliştirilmiş sinir ağı, kaynakların kısıtlı olduğu ortamlarda kumaş hatası tanıma için özel olarak tasarlanmıştır. Bu çalışma, dekonvolüsyonel ağları birleştirerek ve görselleştirme tekniklerini kullanarak kumaş görüntüsü özellik çıkarımının verimliliğini büyük ölçüde artırmaktadır. Ortaya çıkan ve sadece 12 milyon parametre içeren sinir ağı, tekstil üretim ortamlarında kumaş hatası tanıma için özel olarak tasarlanmıştır. Bu çalışma sadece tanıma doğruluğunu ve verimliliğini artırmakla kalmayan aynı zamanda çeşitli kumaş türlerine hitap eden ve tekstil fabrikası ortamlarındaki kaynak sınırlamalarının yarattığı zorlukları ele alan yeni bir yaklaşım sunmaktadır.

Kumaş hatalarının tespitine ilişkin literatür, derin öğrenme tekniklerinden yararlanan özellikle de verimli ve doğru tespit için YOLO modellerine odaklanan çeşitli yenilikçi yaklaşımlarla dikkat çekmektedir. Bu çalışmalar, bu modellerin kumaş hatası tespit sistemlerinin kalitesini ve verimliliğini nasıl önemli ölçüde artırabileceğine dair fikir vermektedir.

Zhang ve diğ. (2018) yaptıkları çalışmalarında, iplik boyalı kumaş hatalarının tespiti için görüntü özelliklerinin çıkarılmasına yönelik yoğun emek gerektiren süreci kolaylaştırmak için yenilikçi bir yaklaşım tanıtmaktadır. Önerilen yöntem, iplik boyalı kumaş hatalarını otomatik olarak bulmak ve sınıflandırmak için YOLOV2'yi

kullanılmaktadır. Süreç; 276 iplik boyalı kumaş hatası görüntüsünün toplanması, ön işlem den geçirilmesi ve etiketlenmesi ile başlar. Kumaş hatası tespit modelleri oluşturmak için YOLO9000, YOLO-VOC ve Tiny YOLO gibi çeşitli modeller kullanılmaktadır. Titiz karşılaştırmalı analizler sonucunda, YOLO-VOC daha fazla geliştirme için en uygun aday olarak belirlenmiştir. Derin evrişimli sinir ağına, hiper parametreler optimize edilerek ince ayar yapılmıştır. Bu geliştirilmiş derin evrişimli sinir ağı, pratik kumaş görüntüleri üzerinde kapsamlı bir şekilde test edilerek etkinliği ve iplik boyalı kumaş kusur tespiti ile ilişkili işçilik maliyetlerini önemli ölçüde azaltma potansiyeli doğrulanmıştır. YOLO-VOC modeli, görüntü başına 0,023 saniye gibi düşük bir algılama süresi ile aynı zamanda kesinlik ve duyarlılık metrikleri açısından da üstün bir başarı sunmuştur. Bu da modeli oldukça verimli kılmaktadır. Bu yaklaşım sadece doğruluk gereksinimlerini karşılamakla kalmayıp aynı zamanda iplik boyalı kumaş hatası tespiti için gerçek zamanlı endüstriyel denetimlerin taleplerini de karşılamaktadır.

Jing ve diğ. (2020), gerçek zamanlı performansa odaklanarak tespit oranlarını ve kumaş ürün kalitesini iyileştirmeyi amaçlayan yenilikçi bir kumaş hatası tespit yöntemi sunmaktadır. Yöntem, YOLOv3 modelindeki iyileştirmelere dayanmakla birlikte iki temel adımı kapsamaktadır. İlk olarak, kusur boyutu tahmini için önceki karelerin optimum sayısını ve boyutlarını belirlemek üzere hedef kare boyutlarının kümelenmesini k-ortalamar algoritması ile entegre eder. İkinci olarak, düşük seviyeli görüntü özellikleri yüksek seviyeli görüntü özellikleri ile birleştirerek ve YOLO algılama katmanını değişen boyutlardaki özellik haritalarına dahil ederek gri kumaş ve örgü kumaştaki kusurları tespit etmek için uygulanabilirliği artırır. Deney sel sonuçlar, önerilen yöntemin hem gri hem de örgü kumaş için etkinliğini ve uygulanabilirliğini doğrulamaktadır.

Liu ve diğ. (2022), kumaş hatalarının tespiti için MaxPool yerine SoftPool kullanan yeni bir SPP yapısını içeren geliştirilmiş bir YOLOv4 algoritmasını tanıtmaktadır. Geliştirilmiş algoritma, özellik haritalarını etkili bir şekilde işleyerek olumsuz yan etkileri azaltmakta ve tespit doğruluğunu artırmaktadır. Ek olarak, görüntü kalitesini artırmak, parazit önleme yeteneklerini geliştirmek ve mAP'yi biraz artırmak için kontrast sınırlı uyarlanabilir histogram eşitleme uygulanmaktadır. Deney sel sonuçlar, orijinal YOLOv4'e kıyasla FPS'de yalnızca %2'lik bir düşüşle mAP'de %6'lık bir artış olduğunu ortaya koyuyor ve doğru ve hızlı kusur lokalizasyonunu mümkün

kılmaktadır. Bu model, özellikle arka plan şekillerinin yakınındaki küçük kumaş kusur alanlarını ele alarak manuel tespitteki verimsizliklerin üstesinden gelmektedir.

Xie ve Hu (2020), güçlü nesne yerleştirme yeteneklerinden ve nesne algılamadaki genellemesinden yararlanan geliştirilmiş bir RefineDet modeline dayanan sağlam bir kumaş hatası algılama yaklaşımı tanıtmaktadır. Yöntem öncelikle RefineDet'i temel model olarak kullanır ve kusurlu nesnelerin verimli ve hızlı tespiti için hem iki aşamalı hem de tek aşamalı dedektörlerin avantajlarını miras almaktadır. Ek olarak, yöntem, kusur lokalizasyon doğruluğunu artırmak için FCCA bloğu ve BA-TCB ile gelişmiş bir kafa yapısı içermektedir. Önerilen yöntem; dikkat mekanizmaları, DIOU-NMS gibi çeşitli genel optimizasyon tekniklerini içermekte ve bunların kumaş kusur lokalizasyonu görevlerindeki etkinliğini göstermektedir. Yöntem, yüksek bir algılama hızına (18 FPS'nin üzerinde) ulaşmakta ve 320x320 girdi boyutuyla mütevazı bir model parametresi boyutunu (yaklaşık 43 milyon) korumaktadır. Çoklu kusur türlerine sahip periyodik desenli kumaşlar ve kusur sınıfı başına sınırlı örnekler de dahil olmak üzere çeşitli kumaş türlerine uygulanabilmekte ve çok yönlülüğünü göstermektedir.

Li ve diğ. (2019), kompakt bir konvolüsyonel sinir ağı mimarisi kullanarak kumaş hatası tespiti için WACNet adı verilen yeni bir yaklaşım önermektedir. Önerilen ağ; çok ölçekli analiz, filtre faktörizasyonu, parametre azaltma ve çoklu konum havuzlama gibi teknikler kullanarak çok katmanlı algılayıcı ile mikro mimariler kullanmaktadır. Bu teknikler, kompakt bir model içinde tespit doğruluğunu artırmayı amaçlamaktadır. Kenar sarkması, kenar çapağı ve kırık uç gibi yaygın kusurları içeren bir kumaş veri kümesi üzerindeki deneysel sonuçlar, WACNet'in önemli ölçüde daha küçük bir model boyutuyla algılama doğruluğu açısından ana akım evrimsel sinir ağı mimarilerinden daha iyi performans gösterdiğini ortaya koymaktadır. Ağ, yalnızca kumaş hatası tespitinde değil aynı zamanda genel veri kümeleri üzerinde nesne tanımda da umut verici sonuçlar sergilemektedir. Deneyler, WACNet'in kumaş kusurlarını hem tek tek hem de toplu olarak ele almadaki etkinliğini ortaya koymakta ve bu da onu çeşitli uygulamalarda kaynak sınırlı sistemler için potansiyel bir çözüm haline getirmektedir.

Peng ve diğ. (2020), mevcut derin öğrenme algoritmalarının küçük ve aşırı en-boy oranlı kumaş hatalarını tespit etmedeki sınırlamaları ele almaktadır. PRAN-Net adı verilen yeni bir çözüm tanıtılmaktadır. PRAN-Net, inceleme süresini azaltırken

kumaş hatası tespit doğruluğunu ve verimliliğini artırmayı amaçlamaktadır. Bu yaklaşım birkaç temel teknikten yararlanmaktadır. İlk olarak küçük kusurlar hakkında ayrıntılı bilgileri korumak için çok ölçekli özellik haritalarını seçmek için bir feature pyramid networks (FPN) kullanmaktadır. İkinci olarak PRAN-Net, sabit çapalar yerine kumaş hatası temel gerçek kutularına dayalı seyrek önceki çapalar üretmek için yeni bir yaklaşım sunmaktadır. Bu adaptasyon, aşırı kusurların daha hassas ve verimli bir şekilde tespit edilmesini sağlamaktadır. Son olarak, kumaş kusurlarının konumunu sınıflandırmak ve iyileştirmek için bir sınıflandırma ağı kullanılmaktadır. Deneysel sonuçlar PRAN-Net'in etkinliğini doğrulayarak özellikle küçük ve aşırı en-boy oranı kusurları için otomatik kumaş kusur tespiti için umut verici bir çözüm olarak konumlandırmaktadır. Yöntem, daha az sayıda önceki çapa ile bile gerçek zamanlı algılama yeteneklerini korumaktadır.

Wu ve diğ. (2021), Faster R-CNN çerçevesine dayanan geniş ve hafif bir ağ sunarak yaygın kumaş kusurlarını tespit etmek için yenilikçi bir çözüm önermektedir. Amaç, yüksek tespit doğruluğunu korurken hesaplama maliyetlerini en aza indirmek ve böylece sınırlı kaynaklara sahip donanım platformları için uygun hale getirmektir. Önerilen ağ mimarisi, farklı kusur boyutlarını verimli bir şekilde barındırmak için çok ölçekli konvolüsyon çekirdeklerini kullanan genişletilmiş bir konvolüsyon modülü ekleyerek özellik çıkarımını geliştirmektedir. Üst düzey semantik özelliklerin atlama bağlantıları aracılığıyla ince taneli ayrıntılarla bütünleştirilmesi, çeşitli ölçeklerdeki kumaş kusurlarını tespit etmek için gerekli çok ölçekli birleştirme özelliklerini sağlamaktadır. Algoritma, çok ölçekli kumaş kusurlarını etkili bir şekilde tespit etmek için farklı boyutlarda birden fazla sınırlayıcı çerçeve kullanılır. Deneysel sonuçlar kırık iplikler, kaymalar, delikler, tiftik topları, düğümler, çizikler ve lekeler gibi yaygın kumaş kusurlarını tespit etmede %97'yi aşan bir doğruluk oranına işaret ederken, model boyutunu küçülterek kumaş kalite kontrolü için değerli bir varlık haline getirmektedir.

Kumaş hata tespitinde, derin öğrenmede segmentasyon yapılarının kullanılması, tekstil numunelerindeki hataların hassas bir şekilde bölgesel tanımlanmasını sağlayan ve böylece tekstil endüstrisindeki kalite kontrol süreçlerinin verimliliğini artıran güçlü bir yaklaşım sağlamaktadır.

Huang ve Xiang (2022), tekstil endüstrisinde kalite kontrolün sürdürülmesi için çok önemli olan kumaş hatalarının piksel düzeyinde tespiti için tasarlanmış semantik bir

segmentasyon ağı olan RPDNet'i sunmaktadır. RPDNet, "tekrarlanan desen kusur ağı" anlamına gelir ve kumaşlardaki kusurları tespit etmek için bir yaklaşım sunmaktadır. Yöntem, kumaş görüntülerindeki periyodik desenleri tanımlamak için CNN'e dayalı bir tekrarlanan desen dedektörü kullanmaktadır. Bu bilgi daha sonra ağı üst düzey bir semantik alanda yönlendirmek için kullanılmakta ve periyodik özellikleri tanıma ve potansiyel kusur alanlarını vurgulama yeteneğini geliştirmektedir. Model, verimli ve hafif kumaş hatası tespiti sağlamak için iki gelişmiş mimariyi, DeeplabV3+ ve GhostNet'i birleştirmektedir. Sonuçlar, RPDNet'in kumaş kusurlarını yüksek doğruluk ve hızla etkili bir şekilde tespit ettiğini ve endüstriyel uygulamalarda gerçek zamanlı tespit gereksinimlerini karşıladığını göstermektedir. FCN, UNet ve Mobile-Unet gibi diğer derin öğrenme yöntemleriyle yapılan karşılaştırma, RPDNet'in özellikle kumaş dokusu ayrıntılarını tespit etme ve çıktı piksellerindeki sınıf hatalarını azaltma açısından üstün performans sunduğunu göstermektedir. Bununla birlikte, makale tekrarlanan desensiz görüntülerin işlenmesindeki bir sınırlamayı vurgulamaktadır. Ağ, tekrarlanan desenleri tespit etmede başarılı olsa da desen tekrarlarının daha küçük boyutta olması nedeniyle tekrarlanmayan desenlerin yakalanmasını zorlaştırmaktadır.

Rong-qiang ve diğ. (2021), kumaş kusurlarını tespit etme özel görevi için CU-Net adlı geliştirilmiş bir evrimsel sinir ağı önermektedir. Bu ağ, klasik U-Net mimarisinin temelleri üzerine çeşitli geliştirmelerle inşa edilmiştir. Kayda değer bir gelişme, bir dikkat mekanizmasının dahil edilmesidir. Ayrıca, eğitim sırasında yeni bir bileşik kayıp fonksiyonu kullanılmaktadır. CU-Net'in etkinliğini değerlendirmek için yazarlar, test için halka açık AITEX kusurlu kumaş veri setini kullanmaktadır. Deneysel sonuçlar, diğer mevcut tespit yöntemleriyle karşılaştırıldığında, onları sırasıyla %4,8 ve %2,3 oranında aşan önemli gelişmeleri temsil etmektedir. Modelin eğitim ve test süreçlerini hızlandırmak için yazarlar, özellik kanalı budama ve alt örnekleme işlemlerini gerçekleştirmektedir. Özellikle, özellik kanallarının sayısı standart U-Net'te; 64, 128, 256, 512, 1024 iken CU-Net'te; 32, 64, 128, 256'ya düşürülmüştür. Kumaş hatası tespiti esasen ikili bir sınıflandırma görevi olduğundan, bu azaltma stratejik olarak yerelleştirmeye öncelik vermek için yapılmıştır. Sonuç olarak lokalizasyona daha fazla önem verilmekte ve minimize edilmiş alt örnekleme daha yüksek lokalizasyon hassasiyetine yol açmaktadır.

Jing ve diğ. (2022), uçtan uca kusur segmentasyonu için son derece verimli bir konvolüsyonel sinir ağı olan Mobile-Unet tanıtmıştır. Örnek dengesizliğini gidermek için medyan frekans dengeleme kayıp fonksiyonunu içerir ve model karmaşıklığını ve boyutunu önemli ölçüde azaltarak derinlemesine ayrılabilir konvolüsyon kullanılır. Mobile-Unet bir kodlayıcı ve kod çözücüden oluşmaktadır. Kodlayıcı MobileNetV2 özellik çıkarıcısını kullanırken, kod çözücü giriş ve çıkış boyutlarını koruyan beş dekonvolüsyon katmanına sahiptir. Ağ iki aşamaya ayrılmıştır: kusur sınıflandırması için ön eğitim ve kusur tespiti için semantik segmentasyon. Mobile-Unet modeli, FCN, U-net, SegNet gibi mevcut derin öğrenme tabanlı görüntü segmentasyon modellerinden ve LGM-FC ve PTIP gibi geleneksel yöntemlerden daha iyi performans göstermektedir. Piksel düzeyinde segmentasyona ulaşarak kusur ayrıntılarını korumada üstündür. Mobile-Unet'in hesaplama verimliliği, derinlik bazlı konvolüsyonun geleneksel konvolüsyonla karşılaştırılmasıyla gösterilmiş ve çeşitli donanım platformlarında üstün performans göstermiştir. Mobile-Unet daha az sayıda parametre ve daha küçük bir model boyutu ile tasarlanmıştır. Bu da daha hızlı çıkarım sürelerine ve gerçek zamanlı uygulamalar için daha uygun olmasına katkıda bulunmaktadır. Algılama hızı 140 FPS ile endüstriyel uygulamalardaki gerçek zamanlı algılama kriterlerini karşılamaktadır.

Bai ve Jing (2023), pratik uygulamalarda veri dengesizliği ve gerçek zamanlı performans gibi sorunları ele almaktadır. Bu makale uçtan uca kusur segmentasyonu için tasarlanmış yüksek gerçek zamanlı bir konvolüsyonel sinir ağı olan Mobile-Deeplab'ı tanıtmaktadır. Önerilen model, uzun vadeli bağımlılığı ve küresel bağlamsal bilgi yakalamayı geliştirmek için bir ESA bloğu içermekte ve düşük kontrastlı küçük kusurların daha iyi tanımlanmasını ve segmentasyonunu sağlamaktadır. Karışık kayıp fonksiyonu, özellikle düşük kontrastlı kusurlar için performansı daha da iyileştirmektedir. Deneysel sonuçlar, Mobile-Deeplab'in diğer segmentasyon modellerinden daha iyi performans gösterdiğini, daha eksiksiz kusur ayrıntılarını koruduğunu ve çeşitli kusur türleri ve dokularda iyi performans elde ettiğini göstermektedir. MioU, recall, precision ve F1-Score gibi değerlendirme metrikleri, özellikle dengesiz veri kümeleri için önerilen algoritmanın etkinliğini desteklemektedir. Ek olarak, çalışma farklı kayıp fonksiyonlarının eğitim ve model performansı üzerindeki etkisini araştırmakta ve sınıf dengesizliği sorunlarını ele almada karma kayıp fonksiyonunun etkinliğini vurgulamaktadır. Önerilen yöntem,

diğer modellere kıyasla azaltılmış parametreler, FLOP'lar ve model boyutu ile kanıtlandığı üzere üstün verimlilik sergilemektedir. Özellikle model yüksek algılama hızlarına ulaşarak endüstriyel gerçek zamanlı gereksinimleri karşılamaktadır.

Oto kodlayıcı ve GAN mimarilerini içeren kumaş hatası tespit yöntemlerine geçiş yapan bu yenilikçi yaklaşımlar, tekstil endüstrisinde kalite kontrolüne uygulanan derin öğrenme alanında önemli bir adımı temsil etmektedir. Denetimsiz öğrenme modelleri olan otomatik kodlayıcılar, kumaş görüntülerinden karmaşık özelliklerin ve temsillerin çıkarılmasında önemli bir rol oynar ve geleneksel yöntemlerden kaçabilecek ince kusurların belirlenmesine yardımcı olmaktadır. Öte yandan, üretken çekişmeli ağlar, sentetik örneklerin gerçekçiliğini artırmak için bir üreteç ve ayırt edici etkileşiminden yararlanarak benzersiz bir dinamik sunar ve modelin sınırlı kusurlu veriler üzerinde daha sağlam bir şekilde eğitilmesini sağlamaktadır.

Seker ve diğ. (2016), tekstil imalat endüstrisinde kumaş hatalarının tespiti için geleneksel manuel denetim yöntemlerini denetimsiz makine öğrenimi ile değiştiren yenilikçi bir yaklaşıma odaklanmaktadır. Kumaş hatası tespiti için derin öğrenmenin, özellikle de Autoencoder olarak bilinen derin bir sinir ağının uygulanmasını tanıtmaktadır. Autoencoder, giriş verilerini hem sıkıştırılmış hem de genişletilmiş formlarda temsil edecek şekilde tasarlanmıştır. Kumaş hatası tespitinde kabul edilebilir başarı seviyelerine ulaşabilmektedir. Çalışma, girdi değerlerini ve hiper parametreleri ayarlayarak özellik çıkarma performansını artırmaya odaklanmaktadır. Derin sinir ağı, iki oto kodlayıcı ve ardından denetimli bir sınıflandırma softmax katmanından oluşmakta ve genel doğruluk oranı %88'e ulaşmaktadır. Özellikle, yırtılma ve leke kusurlarını tespit etmede başarılı olsa da yanlış hizalama hataları için tespit doğruluğu nispeten daha düşüktür. Yazarlara göre bu tutarsızlık, giriş görüntüleri otomatik kodlayıcı kullanılarak sıkıştırıldığında yanlış hizalama hatalarının kaybolma eğilimine bağlanabilmektedir.

Mei ve diğ. (2018) tarafından önerilen yöntem, tekstil imalat endüstrisinde kalite kontrolün önemli bir yönü olan kumaş hatalarının tespiti bağlamında, MSCDAE adlı yenilikçi bir denetimsiz öğrenme tabanlı otomatik yaklaşımı tanıtmaktadır. Önerilen MSCDAE yöntemi, görüntü yamalarını çoklu Gauss piramit seviyelerinde yeniden yapılandırmak için bir konvolüsyonel denoising otomatik kodlayıcı ağı kullanır ve ilgili çözünürlük kanallarından algılama sonuçlarını sentezlemektedir. Her görüntü yamasının yeniden yapılandırma artığı, kusur tahmini için piksel bazında bir gösterge

görevi görür ve bu göstergelerin farklı çözünürlük seviyelerinde entegrasyonu nihai denetim sonucunu üretmektedir. MSCDAE modelini karakterize eden birkaç önemli avantaj vardır. İlk olarak, minimum miktarda hatasız numune ile eğitilebildiği için veri kıtlığına karşı dayanıklılık göstermektedir. Bu da bol miktarda hatalı numune toplamının zor olduğu durumlarda önemli bir avantajdır. İkinci olarak çok modlu entegrasyon stratejisi, her çözünürlük seviyesindeki sonuçları ayrı olarak ele alarak sağlamlığı ve doğruluğu artırmaktadır. Üçüncü olarak model, basitten karmaşık desenlere kadar değişen karmaşıklıkları ele alarak çeşitli tekstil kumaşları arasında uyarlanabilirlik göstermektedir. Modelin otomatik kumaş hatası tespiti alanındaki çok ölçekli yaklaşımı, sağlamlığı ve uyarlanabilirliği umut verici bir ilerleme olarak konumlandırılmaktadır.

Wei ve diğ. (2021), pamuklu tekstil fabrikalarında otomatik kumaş hatası tespitinin önemine değinmekte ve bu hataların otomatik olarak sınıflandırılmasındaki zorlukları araştırmaktadır. Kusur örneklerinden oluşan kapsamlı bir veri kümesi toplamının zorluğu göz önüne alındığında çalışma CNN'lere dayalı denetimsiz öğrenme algoritmalarını karşılaştırmakta, AE ve VAE'ye odaklanmaktadır. Performansı artırmak için MSSIM, Lp-mesafe kayıp fonksiyonuna ek olarak bir ağ eğitim kayıp fonksiyonu olarak tanıtılmıştır. VAE'ye dayanan önerilen sistem, gerçek zamanlı gereksinimleri karşılayan Nvidia'nın Jetson TX2'sinde uygulanmıştır. Araştırma, kendi geliştirdiği bir kumaş örneği kütüphanesini TILDA genel kütüphanesi ile birleştirerek güvenilir bir genel veritabanı eksikliği sorununu ele almaktadır. Önerilen algoritma, VAE araştırmasını görüntü benzerliği karşılaştırması, parlaklık, doku yapısı ve kontrasttaki geliştirmelerle birleştirerek endüstriyel senaryolarda gerçek zamanlı algılamayı iyileştirmektedir. Jetson TX2'nin hızlandırıcı olarak kullanılması, endüstriyel ortamlarda kumaş kusurlarının tespiti için gerçek zamanlı işlem yapılmasını sağlayarak yapay görmenin tekstil üretim süreçlerine dahil edilmesine yönelik devam eden eğilime katkıda bulunmaktadır.

Yang ve diğ. (2020), özellikle anomaliler küçük olduğunda ve ön bilgi olmadığında, görüntülerdeki anormal bölgeleri otomatik olarak tespit etme gibi zorlu bir görevi ele almaktadır. Anomaliler, üretim ürünlerindeki kusurlar veya görüntülerdeki herhangi bir düzensizlik olabilmektedir. Bu makale, görüntülerdeki küçük, yerel bölgelerdeki anormallikleri tespit etmek ve segmentlere ayırmak için etkili bir denetimsiz anormallik segmentasyonu yaklaşımı sunmaktadır. Önerilen yöntem, bir görüntüdeki

alt bölgeler için uzamsal bağlama duyarlı temsiller oluşturmak üzere önceden eğitilmiş derin evrimsel ağları kullanan çok ölçekli bir bölgesel özellik oluşturu sunmaktadır. Bu bölgesel temsiller hem yerel özellikleri hem de çok katmanlı uzamsal bağlamı yakalayarak anomali tespiti için oldukça ayırt edici ve avantajlı hale getirmektedir. Görüntülerdeki anormal bölgeleri tespit etmek için yazarlar, hızlı özellik yeniden yapılandırması için bu tanımlayıcı bölgesel özelliklerden yararlanan derin ama verimli bir evrimsel otomatik kodlayıcı tasarlanmıştır. Makale, daha küçük ölçeklerin anomali tespiti için değerli bilgiler içerdiğini ve daha büyük ölçeklerin zaten gerekli verileri yakalayabileceğini öne sürmektedir. Çıkarım hızı açısından, yöntem 12 farklı özellik ölçeğinde bile saniyede 100 karenin (100 Fps) üzerinde etkileyici sonuçlar ortaya koymaktadır. Bu yüksek hız, yaklaşımı gerçek dünya uygulamaları için pratik hale getirmektedir.

Liu ve diğ. (2019), kumaş hatalarını otomatik olarak tespit etmeyi ve farklı uygulama senaryolarında çeşitli kumaş dokularına uyum sağlamayı amaçlayan GAN'lara dayalı kumaş hatası tespitine yenilikçi bir yaklaşım sunmaktadır. Sistemin çekirdeği, farklı kusur türlerini tanımlayabilen kumaş kusur tespiti için tasarlanmış özel bir derin semantik segmentasyon ağıdır. Kumaş dokusu çeşitliliği sorununu ele almak için yazarlar çok aşamalı GAN tabanlı bir yaklaşım kullanmaktadır. Bu yaklaşım ilk olarak koşullu kusur dağılımlarını koşullu bir GAN ile modelleyerek yeni bir kumaş dokusu sunulduğunda makul kusurlu yamaların oluşturulmasına olanak tanımaktadır. Daha sonra, üretilen kusurları kumaş içindeki belirli konumlara entegre etmek için GAN'lara dayalı bir birleştirici ağ kullanılmaktadır. Çok aşamalı GAN çerçevesi, derin semantik segmentasyon ağını sürekli olarak günceller ve ince ayarlar, değişen koşullara uyarlanabilirliğini korur ve hem basit hem de karmaşık kumaş dokularında etkili kusur tespiti sağlamaktadır. Deneysel sonuçlar, sistemin verimliliğini, sağlamlığını ve çeşitli kumaş türlerine ve kusur ölçeklerine uyarlanabilirliğini doğrulamaktadır.

Hu ve diğ. (2020), özellikle kumaşlar olmak üzere dokulu malzemelerdeki yüzey kusurlarını tespit etmek için yenilikçi bir denetimsiz yaklaşım tanıtmaktadır. Bu yaklaşım, farklı dokulara ve sınırlı kusurlu örnek verisine sahip tekstillerdeki kusurları tespit etme zorluğunu ele almaktadır. Yazarlar, yeni tanıtılan bir kodlayıcı bileşeni ile bir DCGAN önermektedir. Kodlayıcının rolü çok önemlidir. Çünkü kusurları ortadan kaldırmak için giriş görüntülerini yeniden yapılandırmakta ve

yeniden yapılandırma sırasında yalnızca normal dokuları korumaktadır. Bu yeniden yapılandırma, orijinal görüntüden çıkarıldığında potansiyel kusur bölgelerini etkili bir şekilde vurgulayan bir kalıntı haritasının oluşturulmasıyla sonuçlanmaktadır. Bu yaklaşımın en dikkat çekici yönü, hatasız örneklerden küçük görüntü yamaları kullanılarak denetimsiz bir şekilde eğitilebilmesi ve eğitim sırasında insan açıklamasına veya etiketli temel gerçeğe olan ihtiyacı ortadan kaldırmasıdır. Çalışma; aydınlatma koşullarındaki değişiklikler, görüntü bulanıklığı ve çeşitli doku yapıları gibi zorlukların varlığında bile modelin sağlamlığını, verimliliğini ve yüksek tespit doğruluğunu göstermektedir.

Bu çalışmada literatürden farklı olarak öncelikle gözle görülmesi zor olabilecek küçük hataların tespitini kolaylaştırabilmek için ve aynı zamanda farklı çözünürlükteki kameralarla alınan görüntülerde yüksek çözünürlüklü standardizasyon oluşturabilmek için görüntüler ESRGAN modeli 4 kat büyütülmüştür. Elde edilen görüntüler YOLOv8 modeli ile test edilerek yüksek başarılı hata sınıflandırması gerçekleştirilmiştir.

2. DERİN ÖĞRENME ESASLARI

Yapay zekanın göze çarpan bir alt kümesi olan derin öğrenme, hesaplamalı sistemlerin otonom öğrenme yetenekleri kazanmasını ve ampirik verilerden bilgi çıkarmasını kolaylaştıran dönüştürücü bir çerçeve görevi görmektedir. Temel olarak veri kümeleri içindeki gizli kalıpları ayırt etme konusunda becerikli hesaplamalı mimariler oluşturmaya odaklanmaktadır. Böylece özerk karar verme ve tahmin yeteneklerini mümkün kılmaktadır. "Derin Öğrenme" terimi, özellikle yapay sinir ağlarındaki ilerlemeleri ifade etmek üzere 2000 yılında ortaya çıktı. 2006'daki önemli atılımlar, çok katmanlı ileri beslemeli sinir ağlarının etkinliğini ortaya çıkarmış ve 2012'de trafik işareti tanıma ve tıbbi görüntüleme gibi karmaşık görevlerle ilgili yarışmalarda derin öğrenme yöntemlerinin hakimiyetiyle sonuçlanmıştır (LeCun ve diğ., 2015).

Makine öğreniminin daha geniş alanında derin öğrenme, titiz veri meta-analizi ile başlayan farklı bir yöntemi temsil etmektedir. Duyusal girdilerin öğrenmeyi kolaylaştırdığı insan bilişine benzer şekilde, hesaplama sistemleri bilgi edinimi için geniş veri kümelerini özümseyecek şekilde tasarlanmıştır. Makine öğrenimi yöntemleri ağırlıklı olarak denetimli ve denetimsiz yöntemlere ayrılmaktadır. Denetimli öğrenmede makineler, başlangıçta ilgili etiketlerden yoksun, açıklanmalı veri kümeleri üzerinde eğitime tabi tutulmaktadır. Bu sistemler, yinelemeli geri bildirim mekanizmaları aracılığıyla tahmine dayalı modelleri geliştirip ardından yeni, açıklamasız veri örneklerine etiket atama konusundaki yeterliliğini göstermektedir. Bunun tersine, denetimsiz öğrenme algoritmaları, açıklamasız veri kümeleri üzerinde çalışarak buradaki içsel yapıları ve kalıpları ayırt etmeye çalışmaktadır. Bu tür yöntemler, açık kategorizasyon veya etiketlemeden yoksun veriler içindeki gizli yapıları açığa çıkarmaya çalışmaktadır. Denetimsiz öğrenmedeki en önemli görevlerden biri, veri örneklerinin içsel benzerlikler ve farklılıklara göre sistematik olarak gruplandırıldığı kümelemeyle ilgilidir (Goodfellow ve diğ., 2016).

Derin öğrenme, özünde birden fazla işlem katmanı ile karakterize edilen hesaplamalı çerçeveleri kolaylaştırarak çeşitli soyutlama düzeylerinde karmaşık veri temsillerinin

elde edilmesini sağlamaktadır. Başlangıçtaki hesaplama engellerine rağmen bu alan, gelişen bilimsel yayınlar külliyyatının da doğruladığı gibi katlanarak çoğalmaya tanık olmuştur. Doğası gereği insan beyninin nörobiyolojik temellerinden ilham alan derin öğrenme algoritmaları hem denetimli hem de denetimsiz öğrenme yöntemleri aracılığıyla çalışmaktadır. Derin öğrenmenin artan önemi; yöntemleri, hesaplama tekniklerini, donanım mimarilerini ve geniş veri kümelerini kapsayan ilerici gelişmelere atfedilebilir. Yüksek performanslı CPU'ların ve GPU'ların vazgeçilmezliği, hacimli veri kümelerinin etkili bir şekilde işlenmesi ve böylece hızlandırılmış hesaplama veriminin sağlanması için vazgeçilmez hale gelmiştir (Wang ve diğ., 2019). Derin öğrenmenin doğal cazibesi, doğrusal olmayan veri dönüşümlerini gerçekleştirme ve katmanlı mimariler aracılığıyla karmaşık ilişkileri ustaca modelleme konusundaki becerisinden kaynaklanmaktadır. Derin öğrenme çerçeveleri tarafından geliştirilen bu tür hiyerarşik temsiller, özellik çıkarma mekanizmalarını kolaylaştırarak gelişmiş veri temsili doğruluğu ve sınıflandırma etkinliği sağlamaktadır.

2.1 Evrişimsel Sinir Ağları

Derin öğrenme alanında bir köşe taşı olan CNN, görsel bilgi işleme alanında devrim yaratmıştır. Derin öğrenmeye olan ilginin artması, CNN'lerin görsel verilerin karmaşıklığını çözmedeki başarısıyla yakından ilişkilidir. Geleneksel sinir ağları görüntü ile ilgili görevlerde zorlanırken, CNN'ler hiyerarşik özellik çıkarımı için özel katmanlar sunarak bir yöntem değişikliği getirmiştir. İnsan görsel sistemini taklit etmek üzere tasarlanan CNN'ler, görüntü sınıflandırma ve nesne tespitinden karmaşık görsel tanıma zorluklarına kadar çeşitli görevler için güçlü bir araç olarak ortaya çıkmıştır (O'Shea ve Nash, 2021). Bu tez çalışması, CNN'lerin derin öğrenme alanındaki temel ilkelerini, mimari inceliklerini ve dönüştürücü uygulamalarını incelemektedir.

2.1.1 Filtreler ve evrişim işlemi

Giriş verileri tipik olarak bir görüntüyü temsil eden bir 3D dizisidir. Renkli görüntüler söz konusu olduğunda üç kanala (kırmızı, yeşil, mavi) sahip olup gri tonlamalı görüntüler için tek bir kanala sahiptir. Kernel olarak da bilinen filtreler, giriş verileri üzerinde kayan konvolüsyon işleminde kullanılan küçük boyutlu

matrislerdir (Cui ve diğ., 2017). Filtreler; giriş verilerindeki kenarlar, köşeler veya dokular gibi belirli özellikleri tespit etmekten sorumludur. CNN'lerde bu filtreler, ağıın eğitim sırasında optimize ettiği öğrenilebilir parametrelerdir. Filtre, giriş verilerine giriş uzamsal boyutları (genişlik ve yükseklik) boyunca kaydırılarak uygulanmaktadır. Her pozisyonda filtre değerleri ile örtüşen giriş değerleri arasında eleman bazında bir çarpma işlemi gerçekleştirilip sonuçlar toplanmaktadır. Bu toplamın sonucu çıkış özellik haritasında tek bir eleman oluşturmaktadır. İşlem filtrenin giriş verileri üzerindeki her konumu için tekrarlanıp sonuçta özellik haritası olarak bilinen yeni bir matris elde edilmektedir. Evrişimsel ağ terminolojisinde, evrişimin ilk değişkeni olan I görüntüsü girdi olarak ve ikinci değişkeni olan K fonksiyonu kernel olarak adlandırılmaktadır. Çıktı ise öznitelik haritası olarak adlandırılmaktadır.

$$(I * K)(i, j) = \sum_m \sum_n I(i + m, j + n) \cdot K(m, n) \quad (2.1)$$

Denklem 2.1'de (i, j) çıktı özellik haritasındaki konumu ve (m, n) filtrenin indislerini temsil etmektedir. Eğitim sırasında, ağıın ağırlıkları hesaplanan kayba dayalı olarak geriye yayılım yoluyla ayarlanıp ağıın doğru tahminler yapma becerisini kademeli olarak geliştirmektedir. Çıkış katmanı, CNN'in son katmanı olup giriş verilerinden öğrenilen özelliklere dayalı olarak ağıın tahminlerini veya sınıflandırmalarını üretmektedir.

2.1.2 Dolgu ve adımlama

Dolgu (Padding) genellikle sınırlardaki uzamsal bilgileri korumak için konvolüsyondan önce giriş verilerine uygulanmaktadır. Bir katmanın bir önceki katmanla aynı yükseklik ve genişliğe sahip olması için girişin etrafına fazladan piksel eklenmesini içerir, bu da filtre görüntü boyunca kayarken bilgi kaybını önlemeye yardımcı olmaktadır. Adım (Stride), filtrenin giriş verileri üzerinde hareket ettiği adım boyutunu tanımlamaktadır. Daha büyük bir adım, daha küçük bir çıktı boyutuyla sonuçlanmaktadır. Adım, çıktı özellik haritasının uzamsal boyutlarını etkilemektedir. S adımlı konvolüsyon işlemi için matematiksel gösterim aşağıdaki gibi ifade edilebilmektedir:

$$O = \frac{(I - F)}{S} + 1 \quad (2.2)$$

Denklem 2.2’de O çıktı boyutu, I girdi boyutu, F filtre (kernel) boyutu, S adımdır. CNN’lerde, farklı özellikleri yakalamak için paralel olarak birden fazla filtre kullanılır ve filtre başına bir tane öznitelik haritası çıkarmaktadır. Sonuçlar bir 3D çıktı tensörü oluşturmak için istiflenmektedir. Konvolüsyon işlemini aktivasyon fonksiyonları ve derin hiyerarşik temsiller oluşturmak için havuzlama ve tam bağlı katmanlar gibi ek katmanlar takip etmektedir.

2.1.3 Havuzlama

Havuzlama işlemleri, CNN’lerin temel bileşenleri olup uzamsal boyutların azaltılmasında, hesaplama yükümü, bellek kullanımını ve parametre sayısını azaltmak dolayısıyla aşırı uydurma riskini sınırlamak için görüntü girdisini alt örnekleme işlemidir. Havuzlama işlemleri, CNN’lerdeki evrimsel katmanların çıktılarına uygulanır. Birincil amaç, temel bilgileri korurken girdi özellik haritalarının uzamsal boyutlarını sistematik olarak azaltmaktır. İki yaygın havuzlama işlemi türü vardır: maksimum havuzlama ve ortalama havuzlama. Maksimum havuzlama, bir grup komşu pikselden maksimum değeri seçmektedir. CNN’lerdeki en yaygın havuzlama işlemi olup yerel bir alıcı alan içindeki en göze çarpan özelliklerin yakalanmasına yardımcı olmaktadır. Ortalama havuzlama, yerel bir komşuluktaki piksellerin ortalama değerini hesaplamaktadır. Girdinin yumuşatılmış bir temsili sağlar ve belirli özellikleri aşırı vurgulamaya daha az eğilimlidir. Geron’a (2019) göre ortalama havuzlama katmanları eskiden çok popüler olmasına karşın günümüzde insanlar daha çok maksimum havuzlama katmanlarını kullanmaktadır. Çünkü daha iyi bir performans göstermektedir. Bu şaşırtıcı gelebilmektedir. Çünkü ortalamayı hesaplamak maksimumu hesaplamaktan genellikle daha az bilgi kaybettirmektedir. Maksimum Havuzlama için, 2 adımlı 2x2 havuzlama penceresi için ilgili denklem aşağıdaki gibi gösterilebilir:

$$H(i, j) = \max (I(2i, 2j), I(2i, 2j + 1), I(2i + 1, 2j), I(2i + 1, 2j + 1)) \quad (2.3)$$

Denklem 2.3’de $I(i, j)$ giriş özellik haritasını temsil eder. $2i$ ve $2j$ indisleri, girdiyi hem dikey hem de yatay yönde 2 kat aşağı örnekleme için kullanılır. Maksimum havuzlama işlemi girişin 2x2’lik bir penceresine uygulanır. Aşağı örneklenmiş çıktıdaki (i, j) konumundaki her pencere için, girdinin karşılık gelen 2x2 bölgesindeki dört değer arasından maksimum değer alınır. Yani havuzlama (i, j) orijinal giriş görüntüsünde $(2i, 2j)$ merkezli girişin 2x2 penceresindeki maksimum değerdir. Bu

işlem, uzamsal boyutları azaltırken her yerel bölgeden en önemli bilgileri korumak için kullanışlıdır. Bu da ağı hesaplama açısından daha verimli ve girdideki küçük ötelemelere karşı değişmez hale getirmektedir.

Çağdaş sinir ağı mimarilerinde sıkça karşılaşılan bir katman da küresel ortalama havuzlama katmanıdır (Hsiao ve diğ., 2019). Çalışma mekanizması, geleneksel havuzlama çekirdeklerinden önemli ölçüde farklıdır. Özellikle, bu katman her bir öznitelik haritasındaki tüm öğelerin ortalama değerini hesaplar. Sonuç olarak bu, her örnek için tekil bir sayısal çıktı üretilmesine neden olmaktadır. Bu sürecin doğası gereği indirgeyici olduğunun ve öznitelik haritasında önemli bir bilgi kaybına yol açtığıнын altını çizmek zorunludur. Bilgi kaybı eğilimine rağmen, küresel ortalama havuzlama katmanı benzersiz özellikleri sayesinde bir çıktı katmanı olarak fayda sağlamaktadır. Tekil sayısal çıktı, öznitelik haritasının yoğunlaştırılmış bir temsilini kapsar ve böylece sinir ağı mimarilerinde ayırt edici bir özellik olarak hizmet etmektedir.

2.1.4 Yaygın doğrusal olmayan aktivasyon fonksiyonları

Doğrusal olmayan aktivasyon fonksiyonlarının birincil rolü, doğrusal yapıda olan sinir ağına doğrusal olmama özelliğini katmaktır. Bu doğrusal olmama durumu, doğrusal dönüşümlerin karmaşık örüntüleri yakalama kabiliyetleri sınırlı olduğundan, ağı karmaşık yapılara sahip verilerden öğrenmesi ve bunlara uyum sağlaması için çok önemlidir. Aktivasyon fonksiyonlarının dikkatli bir şekilde seçilmesi derin ağlarda çok önemlidir. Uygun aktivasyon fonksiyonlarının seçilmesi, kaybolan veya patlayan gradyanlar gibi sorunların azaltılmasına yardımcı olarak istikrarlı ve etkili eğitime katkıda bulunur.

2.1.4.1 Doğrultulmuş lineer birim (ReLU)

ReLU aktivasyon fonksiyonu, sinir ağına doğrusal olmayan bir özellik katarak verilerdeki karmaşık ilişkileri ve örüntüleri modellemesine olanak tanınmaktadır. Basitliği ve kaybolan gradyan sorunu gibi sorunları hafifletmedeki etkinliği nedeniyle derin öğrenme mimarilerinde yaygın olarak kullanılmaktadır. ReLU işlevi hesaplama açısından verimlidir ve ağı eğitim süreci sırasında karmaşık özellikleri öğrenme ve temsil etme becerisini kolaylaştırmaktadır (Li ve Yuan, 2017).

$$\text{ReLU: } f(x) = \max(0, x) \quad (2.4)$$

Denklem 2.4'te 0, x girişı arasındaki maksimum değeri alma işlemini temsil eder. Daha basit bir ifadeyle, x girişı pozitifse, ReLU işlevi x çıktısı vermekte ve x negatif veya sıfırsa, sıfır çıktısı vermektedir.

2.1.4.2 Sigmoid

Lojistik fonksiyon olarak da bilinen sigmoid fonksiyonu, herhangi bir gerçek değerli sayıyı (0,1) aralığına eşler. Bu özellik onu özellikle ikili sınıflandırma modellerinin çıktı katmanında kullanışlı kılar. Burada amaç bir girdinin iki sınıftan birine ait olup olmadığını tahmin etmektir (Han ve Moraga, 1995). Ayrıca, sinir ağı mimarilerine doğrusal olmama özelliği katarak eğitim süreci sırasında verilerdeki karmaşık ilişkileri ve örüntüleri modellemelerine olanak tanır.

$$\text{Sigmoid: } f(x) = \frac{1}{1 + e^{-x}} \quad (2.5)$$

Denklem 2.5, belirli bir x girdisi için sigmoid fonksiyonunun çıktısını temsil eder. Matematiksel ifade şu şekilde ayrıştırılabilir: e, yaklaşık olarak 2,71828'e eşit olan matematiksel sabiti temsil eder. e^{-x} , -x'in üstel fonksiyonunu hesaplar. $1 + e^{-x}$ paydadır ve kesrin değerinin her zaman 1'den büyük olmasını sağlar. Son olarak, $\frac{y}{1+e^{-x}}$ sigmoid fonksiyon çıktısıdır.

2.1.4.3 Hiperbolik tanjant (Tanh)

Tanh, giriş değerlerini -1 ile 1 arasına sıkıştırarak negatif korelasyonları yakalamasını sağlar. Verilerin hem pozitif hem de negatif değerlere sahip olduğu senaryolarda kullanışlıdır.

$$\text{Hiperbolik Tanjant (tanh): } f(x) = \frac{e^{2x} - 1}{e^{2x} + 1} \quad (2.6)$$

Denklem 2.6'da e, yaklaşık olarak 2,71828'e eşit olan matematiksel sabiti ifade eder. e^{2x} ifadesi, $2x$ 'in üstel fonksiyonunu hesaplar ve $e^{2x}-1$ payı, bu üstel fonksiyon ile 1 arasındaki farkı temsil eder. $\frac{e^{2x}-1}{e^{2x}+1}$ kesrinin tamamı, (-1,1) aralığında çıktı değerleri üretir ve bu da onu hiperbolik bir tanjant fonksiyonu yapar. Bu fonksiyon genellikle sinir ağlarında, özellikle de gizli katmanlarda aktivasyon fonksiyonu olarak kullanılır. Rolü, ağa doğrusal olmayan bir özellik katarak modelin eğitim süreci boyunca verilerdeki karmaşık örüntüleri ve ilişkileri yakalamasını sağlamaktır.

2.1.4.4 Sızdıran ReLU

Sızdıran ReLU, girdi negatif olduğunda küçük bir gradyana izin vererek "ölen ReLU" sorununu ele alır ve nöronların inaktif hale gelmesini önlemektedir.

$$\text{Sızdıran ReLU: } f(x) = \max(\alpha x, x) \quad (2.7)$$

Denklem 2.7'de iki değer arasındaki eleman bazında maksimum işlemi temsil eder: αx ve x , burada α bir sabittir. Bu formülde, karşılık gelen her αx ve x değer çifti için, fonksiyon ikisinden büyük olanı verir. Matematiksel olarak şu şekilde tanımlanır:

$$f(x) = \begin{cases} \alpha x, & \text{if } \alpha x \geq x \\ x, & \text{if } x > \alpha x \end{cases} \quad (2.8)$$

Denklem 2.8'de α parametresi bir ölçeklendirme faktörü olarak işlev görür. α parametresi 1'den büyük olduğunda, fonksiyon x değerlerini ölçeklendirerek girdiyi etkili bir şekilde büyütür. Eğer α parametresi 1'e eşit veya 1'den küçükse, fonksiyon x 'in orijinal değerlerini korur. Bu tür bir işlem sinir ağlarında yaygın olarak kullanılıp genellikle sızdıran ReLU olarak adlandırılan bir α parametresi ile düzeltilmiş doğrusal birim (ReLU) olarak bilinir. Ağa doğrusal olmayan bir özellik katarak eğitim süreci sırasında verilerdeki karmaşık ilişkileri öğrenmesine ve temsil etmesine olanak tanmaktadır (Dubey ve Jain, 2019). Denklem 2.8'de α , negatif giriş değerleri için fonksiyonun eğimini kontrol edip "ölen ReLU" sorununu ele almak için geri yayılma sırasında küçük, sıfır olmayan bir gradyan sağlamaktadır.

2.1.4.5 Softmax

Softmax fonksiyonu makine öğreniminde, özellikle de bir sinir ağının çıkış katmanında, gerçek sayılardan (logit) oluşan bir vektörü birden fazla sınıf üzerinde bir olasılık dağılımına dönüştürmek için yaygın olarak kullanılır.

$$\text{Softmax : } f(x)_i = \frac{e^{x_i}}{\sum_j e^{x_j}} \quad (2.9)$$

Denklem 2.9'da e^{x_i} giriş vektöründeki her x_i öğesinin üstel değerini hesaplar. $\sum_j e^{x_j}$ girdi vektöründeki tüm öğelerin üstellerinin toplamını hesaplar. Bu toplam bir normalleştirme faktörü olarak işlev görür. $\frac{e^{x_i}}{\sum_j e^{x_j}}$ her bir x_i elemanının üstel değerini

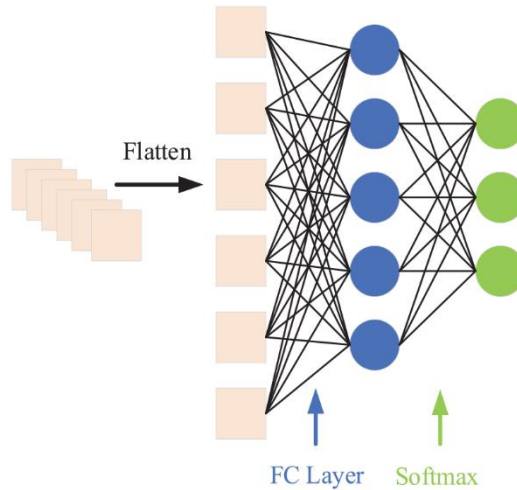
$\sum_j e^{x_j}$ 'i normalleştirme faktörüne bölün. Bu adım, elde edilen değerlerin toplamı 1

olduğu için geçerli bir olasılık dağılımı oluşturmasını sağlar. Softmax işlevi esasen giriş logitlerini olasılıklara dönüştürerek orijinal vektörde daha yüksek değerlere sahip öğelere daha yüksek olasılıklar atar. Özellikle amacın bir girdi örneğini birkaç olası sınıftan birine atamak olduğu çok sınıflı sınıflandırma problemlerinde kullanışlı olmaktadır (Dunne ve Campbell, 1997).

Sonuç olarak doğrusal olmayan aktivasyon fonksiyonları derin öğrenme modellerinin başarısında kritik bir rol oynamaktadır. Sinir ağlarının verilerdeki karmaşık örüntüleri ve ilişkileri yakalamasını sağlayarak sınıflandırma, regresyon ve özellik çıkarma gibi çeşitli görevler için gerekli olan karmaşık eşlemelerin öğrenilmesini kolaylaştırırlar.

2.1.5 Tam bağlantılı katmanlar

Sinir ağları alanında, yoğun katmanlar olarak da adlandırılan tam bağlantılı katmanlar, ağ genelinde bilginin işlenmesi ve entegrasyonunda önemli bir etkiye sahip olan çok önemli bir mimari bileşen oluşturmaktadır. Tam bağlantılı katmanların belirleyici özelliği, her nöronun bir önceki katmandaki her nöronla karmaşık bir şekilde birbirine bağlanmasıdır. Tam bağlantılı katman şekil 2.1’de gösterilmektedir.



Şekil 2.1 : Tam bağlantılı katmanlar (Zhu ve diğ., 2022).

“Tam bağlantılı” isimlendirmesi, yoğun bir şekilde örülmüş bir ağla sonuçlanan kapsamlı sinaptik bağlantı ağından türetilmiştir. Tam Bağlı Katmanın matematiksel gösterimi $y=W \cdot x+b$ denkleminde özetlenmiştir; burada x , m uzunluğundaki giriş vektörünü (bir önceki katmandaki nöron sayısını temsil eder), W ise $n \times m$

boyutundaki ağırlık matrisini ve b ise bias vektörünü ifade etmektedir. W ağırlık matrisi, nöronlar arasındaki bağlantıların gücünü belirleyen sinaptik ağırlıkları kapsar. Bu ağırlıklar eğitim aşamasında öğrenme ayarlamalarına tabi tutularak fark edilebilir veri modellerine uyarlanır. Bias vektörü b , her nörona öğrenilebilir ek bir parametre sağlayarak sinir ağına uyarlanabilirliğini ve ifade kapasitesini artırır. Genellikle sinir ağı mimarilerinin son bölümlerinde sınıflandırıcılar veya regresyon başlıkları olarak kullanılan tam bağlantılı katmanlar, önceki katmanlar tarafından çıkarılan üst düzey özellikleri kesin tahminlere dönüştürmede oldukça başarılıdır. Bununla birlikte, bunların kullanılması parametrelerin genişlemesine ve hesaplama taleplerinin artmasına neden olur. Tam bağlantılı katmanlar, eğitim sırasında nöronları rastgele devre dışı bırakarak sağlamlığı artıran bırakma katmanları da dahil olmak üzere çeşitli biçimlerde ortaya çıkar. Ek olarak, bazı mimariler işlevlerini daha da çeşitlendirmek için tam bağlantılı katmanlar içinde küresel ortalama havuzlama içerir.

2.1.6 Düzleştirme

Düzleştirme (Flatten) katmanının birincil amacı, çok boyutlu giriş verilerini tek boyutlu bir dizi veya vektöre dönüştürmektir. Düzleştirme katmanı tipik olarak evrişimsel katmanlar ile tam bağlı katmanlar arasına yerleştirilir. Şekil 2.1'de düzleştirme katmanı gösterilmektedir. Evrişimsel katmanlar uzamsal hiyerarşiler üzerinde çalışarak örüntüleri ve özellikleri yakalarken, tam bağlı katmanlar sınıflandırma veya regresyon gerçekleştirmek için düzleştirilmiş bir girdi gerektirir. Düzleştirme katmanı sinir ağı mimarilerinde ayrı bir katman olarak uygulanır ve öğrenilebilir herhangi bir parametre içermez. Tamamen bir yeniden şekillendirme işlemidir. Düzleştirme işlemi, öğelerin sırasının korunmasını içermektedir. Girdi (m,n) boyutlarında 2 boyutlu bir dizi ise, burada ' m ' satır sayısı ve ' n ' sütun sayısıdır. Düzleştirilmiş çıktı $(m*n)$ boyutunda 1 boyutlu bir dizi olacaktır.

2.1.7 Optimizasyon

Derin öğrenme alanında optimizasyon, sinir ağlarını eğitmenin çok önemli bir yönüdür. Optimize edici, hatayı veya kayıp fonksiyonunu en aza indirmek için eğitim süreci sırasında modelin parametrelerini ayarlamaktan sorumlu kilit bir bileşendir. Bir optimize edicinin birincil amacı, tahmin edilen çıktı ile gerçek hedef değerler

arasındaki tutarsızlığı azaltmak için bir sinir ağının ağırlıklarını ve önyargılarını iteratif olarak güncellemektir. Çoğu optimize edici, gradyan inişinin temel prensibi üzerine inşa edilmiştir (Shrestha ve Mahmood, 2019). Gradyan, kayıp fonksiyonundaki en dik artışın yönünü temsil eder. Optimize edici, kayıp fonksiyonunun minimumuna doğru iteratif olarak ilerlemek için bu bilgiyi kullanır. Optimize edicilerle ilişkili kritik bir parametre öğrenme oranıdır. Her iterasyon sırasında atılan adımların boyutunu belirler. Uygun bir öğrenme oranının seçilmesi, aşırıya kaçmadan veya yerel minimumlara takılmadan yakınsamanın sağlanması için çok önemlidir. Her biri kendine özgü özelliklere sahip çeşitli optimizasyon algoritmaları önerilmiştir. Yaygın olarak kullanılan bazı optimize ediciler şunlardır:

- SGD: Eğitim verilerinin rastgele örneklenmesi ile temel gradyan inişi.
- Adagrad: Geçmiş gradyan bilgilerine dayanarak her parametre için öğrenme oranlarını uyarlar.
- RMSprop: Öğrenme oranlarını son gradyanların büyüklüğüne göre ayarlar.
- Momentum: Temel gradyan inişine geçmiş gradyanların hareketli ortalamasını ekleyen bir geliştirmedir. Salınımların üstesinden gelmeye yardımcı olur ve özellikle seyrek gradyanların varlığında yakınsamayı hızlandırır.
- Adam: Her parametre için öğrenme oranlarını uyarlayarak momentum ve RMSprop unsurlarını birleştirir.

Birçok modern optimize edici, her parametre için adım boyutunu ayrı ayrı ayarlayan uyarlanabilir öğrenme oranlarını içerir. Bu uyarlanabilirlik genellikle daha hızlı yakınsama ve daha iyi performans sağlar. Bir optimize edicinin etkinliği, kayıp fonksiyonunun global minimumuna yakınsama kabiliyeti ve girdi verilerindeki değişikliklere karşı sağlamlığı ile ölçülür. Optimize ediciler, öğrenme oranları ve momentum katsayıları gibi hiper parametreler sunar. Bu hiper parametrelerin ince ayarını yapmak, belirli görevler için sinir ağlarını optimize etmenin önemli bir parçasıdır.

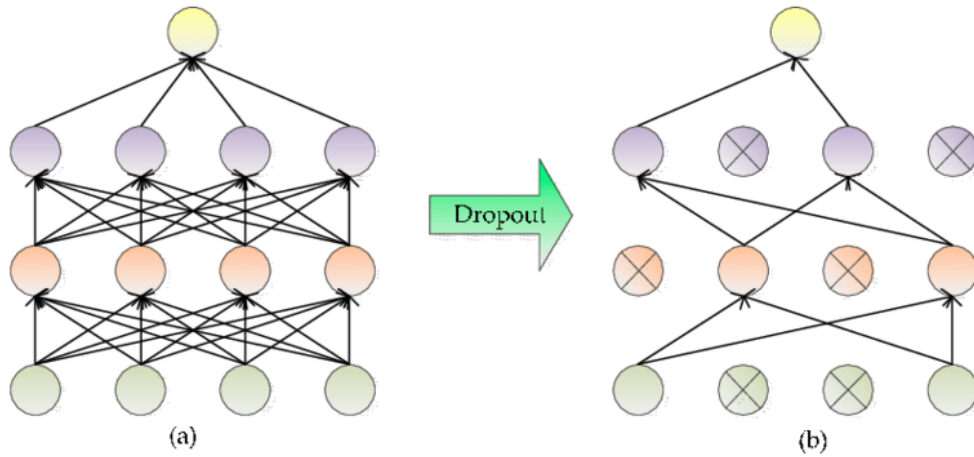
2.1.8 Aşırı uyumluluk

Aşırı uyum (overfitting), makine öğreniminde bir modelin eğitim verilerini çok iyi öğrendiği, altta yatan örüntüler yerine gürültü ve rastgele dalgalanmaları yakaladığı bir olgudur. Model aşırı karmaşık olduğunda, eğitim veri setinin boyutuna göre çok fazla parametreye sahip olduğunda ortaya çıkmaktadır (Li ve diğ., 2019). Model son

derece esnek hale gelir ve benzersiz özellikleri ve aykırı değerleri de dahil olmak üzere eğitim verilerine çok yakından uyar. Sonuç olarak model yeni, görülmemiş veriler üzerinde düşük performans gösterir çünkü esasen eğitim setini ezberlemiştir. Aşırı uyumun yaygın belirtileri arasında eğitim ve doğrulama/test performansı arasında büyük bir eşitsizlik yer alır. Model, eğitim verilerinde üstün performans gösterirken, görülmeyen veri kümelerinde doğruluğu önemli ölçüde düşer. Derin sinir ağlarının genellikle devasa parametre uzayı göz önüne alındığında, aşırı uyum derin öğrenmede önemli bir sorundur. Bırakma, düzenli hale getirme tekniği ve toplu normalleştirme gibi derin öğrenmeye özgü teknikler, aşırı uyumun azaltılmasında hayati bir rol oynamaktadır.

2.1.9 Bırakma

Bırakma (dropout), genelleme performanslarını artırmak ve aşırı uyumu azaltmak için sinir ağlarında kullanılan bir düzenleme tekniğidir. Bırakmanın birincil amacı, eğitimin her iterasyonu sırasında nöronların bir alt kümesini rastgele devre dışı bırakarak karmaşık ortak adaptasyonları önlemektir. Bu, daha sağlam ve genelleştirilmiş bir model oluşturulmasına yardımcı olur. Bu işlem şekil 2.2.b'de gösterilmektedir.



Şekil 2.2 : (a) tam bağlantılı katman, (b) bırakma katmanı (Zhang ve diğ., 2019).

Eğitim sırasında her iterasyonda, ağıdaki nöronların rastgele bir kısmı devre dışı bırakılır veya sıfıra ayarlanır. Bu süreç stokastiktir ve çıkarılan nöron kümesi iterasyonlar arasında değişir. Çıkarım aşamasında, tahminler yapılırken, ağı tamamı bırakma olmadan kullanılır. Ancak öğrenilen ağırlıklar, bırakma olasılığını hesaba katacak şekilde ölçeklendirilerek tutarlılık sağlanır. Bırakma, tam bağlantılı

katmanların yanı sıra evrişimli katmanlara da uygulanabilir. Evrişimsel katmanlarda, bırakma maskesi genellikle tek tek nöronlar yerine tüm özellik haritalarına uygulanır.

2.1.10 Kayıp fonksiyonu

Maliyet fonksiyonu veya amaç fonksiyonu olarak da bilinen bir kayıp fonksiyonu, tahmin edilen ve gerçek değerler arasındaki eşitsizliği yakalamak ve optimizasyon algoritmasının en aza indirmeye çalıştığı sayısal bir değer sağlamaktır. Kayıp fonksiyonunun en aza indirilmesi yeni, görülmemiş verilere iyi genelleme yapan bir modelle sonuçlanır. Bir kayıp fonksiyonunun seçimi, makine öğrenimi görevinin niteliğine bağlıdır. Regresyon problemleri genellikle MSE veya MAE kullanırken, sınıflandırma görevleri sıklıkla çapraz entropi kaybını kullanır (Xu ve diğ., 2018). Problemin özellikleri, veri dağılımı ve model mimarisi seçimi etkiler.

2.1.11 Yığın normalizasyonu

Yığın normalizasyonu (BN), derin sinir ağlarında iç kovaryant kayması sorununu azaltmak için kullanılan bir tekniktir. İç eş değişken kayması, bir sinir ağının katmanlarına girdilerin dağılımı eğitim sırasında değiştiğinde ortaya çıkar. Bu kayma, her katmanın değişen girdi dağılımına sürekli olarak uyum sağlaması gerektiğinden modelin yakınsamasını yavaşlatabilir (Santurkar ve diğ., 2018). BN'de, eğitim sırasında her mini parti için, her katmanın aktivasyonlarının (çıktılarının) ortalaması ve standart sapması hesaplanır. Aktivasyonlar daha sonra ortalama çıkarılarak ve standart sapmaya bölünerek normalleştirilir. Bu normalleştirme, bir katman içindeki her bir özelliğe (nöron) bağımsız olarak uygulanır. Belirli bir nöron için bir x girdisi verildiğinde, normalleştirilmiş $\hat{x}$ çıktısı denklem 2.11'deki gibi hesaplanır.

$$\hat{x} = \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} \quad (2.10)$$

Denklem 2.11'de μ ortalama, σ^2 varyans ve ϵ sayısal kararlılık için eklenen küçük bir sabittir. BN, kaybolan ve patlayan gradyan sorunlarını hafifleterek daha yüksek öğrenme oranlarının kullanılmasını sağlar. Bu da eğitim sürecinin yakınsamasını hızlandırır.

3. GÖRÜNTÜ İŞLEME ALGORİTMALARI

Görüntü işleme alanı, karmaşık tıbbi teşhislerden sıkı endüstriyel kalite güvence protokollerine kadar çeşitli uygulamaları kapsayan, çağdaş teknolojik yeniliklerin temelini oluşturan vazgeçilmez bir temel olarak hizmet vermektedir. Bu geniş alanda, kumaş kusurlarının titizlikle tanımlanması ve incelikli analizi büyük önem kazanmakta ve benzersiz doğruluk, güvenilirlik ve tutarlılık sağlamak için görüntü işleme yöntemlerini uyarlanması önem arz etmektedir (Mahajan ve diğ., 2009). Aydınlatma kusurları, diğer fark edilebilir anormalliklerin yanı sıra düzensiz aydınlatma düzenleri, göze çarpan gölge kusurları ve ışık yansımadaki ara sıra tutarsızlıklar gibi olguları kapsayan çok yönlü bir düzensizlik spektrumunu kapsar. Bu karmaşık zorluklarla yüzleşmek, karmaşık aydınlatma düzensizliklerinin yol açtığı kumaş görünümündeki ince değişiklikleri ustalıkla ayırt etmek ve tasvir etmek için titizlikle hazırlanmış gelişmiş görüntü işleme tekniklerinin kullanılmasını zorunlu kılmaktadır (Singh ve diğ., 2021). Bu zorunluluğa uygun olarak, takip eden tez, tekstil yüzeylerinde ortaya çıkan aydınlatma kusurlarının tespiti, karmaşık karakterizasyonu ve etkili bir şekilde azaltılması için titizlikle tasarlanmış bir dizi görüntü işleme tekniğini titizlikle tanımlayıp değerlendirerek titiz ve kapsamlı bir araştırmaya girişmektedir.

3.1 Medyan Filtresi

Medyan filtresi, gürültüyü azaltmak özellikle de görüntüye rastgele koyu ve parlak pikseller sokan gürültüyü ortadan kaldırmak için görüntü işlemede yaygın olarak kullanılan bir tekniktir. Bu filtre, her piksel değerini komşuluğundaki medyan değerle değiştirerek çalışır ve kenarları veya diğer önemli görüntü özelliklerini bulanıklaştırmadan izole edilmiş gürültüyü etkili bir şekilde yumuşatmaktadır. Görüntüdeki her piksel için o pikselin etrafında bir pencere tanımlanıp bu pencere kare veya daire gibi çeşitli şekil ve boyutlarda olabilmektedir. Diğer bazı yumuşatma filtrelerinin aksine medyan filtresi görüntüdeki kenarları ve ayrıntıları koruma eğilimindedir. Bu da onu yapısal bilgilerin korunmasının önemli olduğu uygulamalar için uygun hale getirmektedir. $I(x,y)$, görüntüdeki (x,y) koordinatlarındaki bir

pikselin yoğunluğunu temsil ettiği düşünülürse (x,y) pikselinin etrafında $(2n+1) \times (2n+1)$ büyüklüğünde bir komşuluk göz önüne alırsak, medyan filtre işlemi matematiksel olarak denklem 3.1'deki gibi gösterilmektedir.

$$I_{\text{filtrelenmiş}}(x, y) = \text{Medyan}(I(x - n, y - n), I(x - n, y - n + 1), \dots, I(x + n, y + n)) \quad (3.1)$$

Denklem 3.1'de, $I_{\text{filtrelenmiş}}(x,y)$, (x,y) pikselindeki filtrelenmiş yoğunluk değeri olup medyan fonksiyonu, pencere içindeki tüm piksellerin yoğunluklarından medyan değerini hesaplamaktadır.

3.2 Görüntü Keskinleştirme

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix} \quad (3.2)$$

Denklem 3.2'deki filtre, bir görüntüdeki kenarları geliştirmek için tasarlanmıştır. 5'in merkezi değeri, merkezi piksele daha fazla ağırlık verirken, çevredeki -1'in negatif değerleri onun etrafında keskinleştirme etkisi yaratır. 0 değerleri, merkezi pikselin yoğunluğunu etkilemeden filtrenin yapısını koruyan yer tutucular görevi görmektedir. Bu filtreyi bir $I(x,y)$ görüntüsüne uygulamak için bir evrişim işlemi gerçekleştirilir. Her pikseldeki (x,y) evrişimi denklem 3.3'deki gibi uygulanmaktadır.

$$I_{\text{filtrelenmiş}}(x, y) = \sum_{i=-1}^1 \sum_{j=-1}^1 \text{çekirdek}(i, j) \times I(x + i, y + j) \quad (3.3)$$

Denklem 3.3'teki $\text{çekirdek}(i, j)$, filtrenin (i,j) koordinatlarındaki değerlerini temsil eder. $I(x+i,y+j)$, (x,y) pikselinin etrafındaki komşuluktaki piksel yoğunluklarıdır. Filtre öncelikle görüntü keskinleştirme için kullanılır. Filtre, merkezi pikseli ve onun yakın komşularını vurgulayarak görüntü ayrıntılarını geliştirip, kenarları ve dokuları daha belirgin hale getirmektedir. Bu özel çekirdekle evrişim, küçük boyutundan dolayı hesaplama açısından basittir. Bu da onu gerçek zamanlı veya kaynak kısıtlı uygulamalar için nispeten verimli kılmaktadır.

3.3 Görüntü Doğrusal Kontrast Germe

Görüntü işleme bağlamında "uzatılmış görüntü", kontrast uzatma veya histogram uzatma olarak bilinen bir işlemde geçmiş bir görüntüyü ifade etmektedir. Görüntünün başlangıçta karanlık olan bölgeleri daha da koyulaşarak daha zengin gölgelere veya daha derin siyahlara yol açabilmektedir. Benzer şekilde, zaten parlak olan alanlar daha da parlak hale gelecek ve bu da daha iyi vurgulamalarla sonuçlanacaktır. Bir görüntüyü genişletmenin temel amacı, mevcut piksel değerlerinin tam dinamik aralığını kullanarak görsel görünümünü iyileştirmektir. Bu geliştirme özellikle kontrastın düşük görüldüğü, farklı özellikler veya ayrıntılar arasında ayırım yapılmasını zorlaştıran görüntülerde yararlı olabilecektir. Piksel değerlerinin histogramı genişletildiğinde, görüntüdeki daha ince ayrıntılar daha ayırt edilebilir hale gelir ve genel olarak daha keskin bir görünüm elde edilir. En yaygın yöntem piksel değerlerinin doğrusal dönüşümünü içermektedir. Orijinal yoğunluk değerlerinin mevcut aralıktan daha geniş bir aralığa eşlenmesiyle kontrast etkili bir şekilde artırılır.

$$uzatGör = (orjGör - minDeğ) * \frac{(yMaxDeğ - yMinDeğ)}{(maxDeğ - minDeğ)} + yMinDeğ \quad (3.4)$$

Denklem 3.4'te $orjGör$ değişkeni tarafından temsil edilen bir görüntü üzerinde doğrusal bir genişletme işlemi gerçekleştirilmektedir. Bu işlem genellikle görüntü işlemede, bir görüntünün piksel yoğunluk değerlerini $yMinDeğ$ ve $yMaxDeğ$ tarafından tanımlanan yeni bir aralığa genişleterek kontrastını geliştirmek için kullanılmaktadır.

$$\text{Ölçeklendirme Faktörü} = \frac{(yMaxDeğ - yMinDeğ)}{(maxDeğ - minDeğ)} \quad (3.5)$$

Denklem 3.5'te kesir doğrusal dönüşümün ölçeklendirme faktörünü veya eğimini temsil etmektedir. Uzatılmış değerlerin $yMinDeğ$ ve $yMaxDeğ$ arasındaki yeni aralığa nasıl eşleneceğini belirlemektedir.

$$(orjGör - minDeğ) * \text{Ölçeklendirme Faktörü} \quad (3.6)$$

Denklem 3.5'te elde edilen kesirin denklem 3.6'da ölçeklendirme faktörü ile çarpılması piksel değerlerinin $y_{MinDeğ}$ ve $y_{MaxDeğ}$ arasında istenen aralığa orantılı olarak dönüştürülmesini sağlamaktadır. Son olarak denklem 3.6'da elde edilen sonuca $y_{MinDeğ}$ eklenerek dönüştürülen değerler, $y_{MinDeğ}$ 'den başlayarak istenen aralıkta yer alacak şekilde kaydırılmaktadır.

3.4 Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme (CLAHE)

CLAHE, Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme anlamına gelir. Geleneksel Uyarlanabilir Histogram Eşitleme (AHE) yönteminin bir uzantısıdır ve özellikle görüntülerdeki yerel kontrastın artırılmasında etkilidir. CLAHE, görüntünün tamamını dikkate alan global histogram eşitlemesinden farklı olarak görüntüyü daha küçük bölgelere veya döşemelere böler ve histogram eşitlemesini bu döşemelerin her birine ayrı ayrı uygulamaktadır (Reza, 2004). AHE'nin zorluklarından biri, nispeten homojen bölgelerde yerel kontrastı aşırı artırarak gürültüyü artırabilmesidir. Bu sorunu çözmek için CLAHE, kontrast geliştirme belirlili bir sınır dahilinde sınırlandırmasını sağlayan bir kontrast sınırlama mekanizması içermektedir. Görüntü, önceden tanımlanmış boyuttaki örtüşmeyen döşemelere bölünüp her döşeme için bir histogram $H(z)$ hesaplanır. Burada z , o döşemedeki yoğunluk seviyelerini temsil etmektedir. Kümülatif dağılım fonksiyonu $CDF(z)$, $H(z)$ histogramından hesaplanmaktadır. Daha sonra her döşemedeki piksel değerleri denklem 3.7'deki işlem uygulanır.

$$I_{eşitlenmiş}(x, y) = CDF(I(x, y)) \times L \quad (3.7)$$

Denklem 3.7'de $I(x, y)$, döşeme içindeki (x, y) konumundaki pikselin orijinal yoğunluk değeridir. $CDF(I(x, y))$, $I(x, y)$ 'deki kümülatif dağılım fonksiyonunun değeridir. L mümkün olan maksimum yoğunluk seviyesidir (8 bitlik bir görüntü için genellikle 255). Eşitlemeden sonra histogram önceden tanımlanmış bir kontrast sınırını aşan değerlerle sonuçlanırsa (genellikle döşemedeki toplam piksel sayısının bir kısmına ayarlanmaktadır), piksel değerleri yeniden dağıtılır. Bu yeniden dağıtım, gürültünün çoğalmasını önlerken yerel kontrastın artırılmasını sağlamaktadır. CLAHE, gürültüyü aşırı derecede yükseltmeden yerel ayrıntıları ve kontrastı geliştirme konusunda üstün bir performans sergilemektedir. Bu da onu özellikle

aydınlatmada önemli deęişimlerin olduęu görüntülerde, global histogram eşitleme yöntemlerine kıyasla üstün sonuçlar üretebilmektedir.

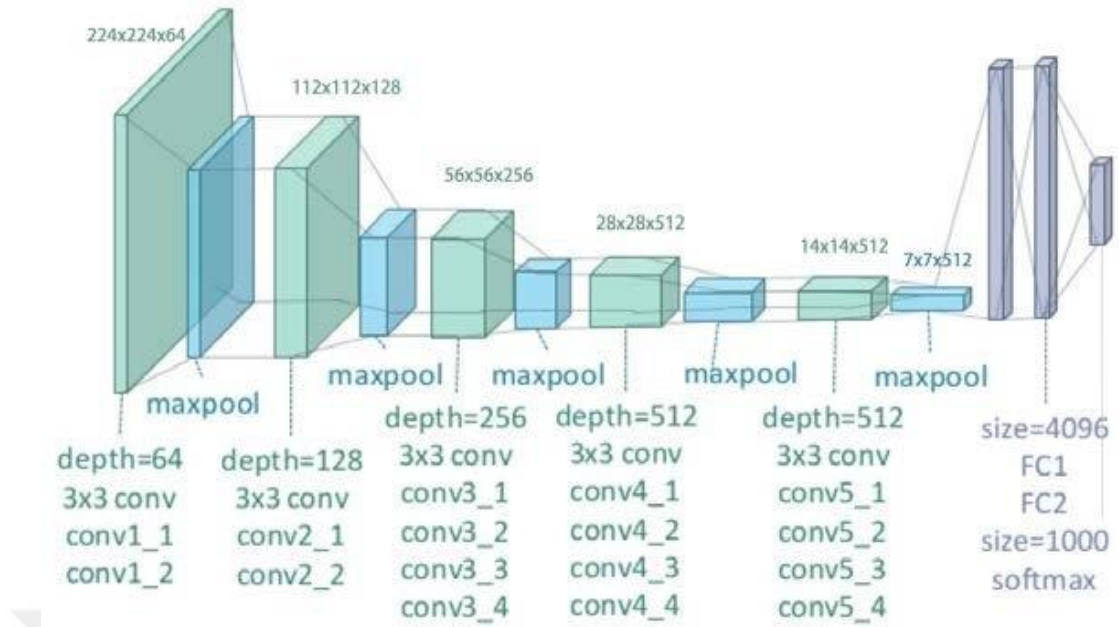


4. DERİN ÖĞRENME ALGORİTMALARI

Kumaş hatalarının tespiti, tekstil imalat endüstrisinde kalite kontrolün kritik bir yönüdür. Geleneksel manuel görsel denetim yöntemleri, özellikle uzun vadeli endüstriyel uygulamalarda verimsizlik ve sınırlı hassasiyetten muzdariptir. Derin öğrenme algoritmalarının ortaya çıkmasıyla, kumaş hatası tespitinde verimlilik ve doğruluk açısından umut verici sonuçlar sunan bir yöntem değişikliği olmuştur. Derin öğrenmenin başarısına rağmen, kumaş hatası tespiti benzersiz zorluklar ortaya çıkarmaktadır. Etiketli kusur örneklerinin azlığı, veri dengesizliği ve gerçek zamanlı işleme gereksinimleri, araştırmacılar tarafından ele alınan önemli zorluklar arasındadır. Bu tez çalışması, kumaş hatası tespiti için kullanılan çeşitli derin öğrenme algoritmalarının kapsamlı bir incelemesini sunmakta ve bunların güçlü yönlerini, zorluklarını ve ilerlemelerini vurgulamaktadır.

4.1 VGG Evrişimsel Ağı

Oxford Üniversitesi'ndeki Görsel Geometri Grubu'nun öncülüğünü yaptığı VGG modeli, görüntü sınıflandırma görevlerinde basitliği ve etkinliği ile tanınan öncü bir derin evrişimli sinir ağı mimarisi olarak öne çıkmaktadır. VGG16 ve VGG19 gibi varyantlar önerilmiş olup her biri uzamsal alt örnekleme için stratejik olarak yerleştirilmiş maksimum havuzlama katmanlarının ardından kompakt 3x3 filtrelerle donatılmış bir dizi evrişimsel katman içeren tek tip bir yapı ile ayırt edilmektedir. Modelin dikkat çekici derinliği ve yüksek boyutlu temsili, görüntülerdeki karmaşık hiyerarşik özellikleri ayırt etmesini sağlamaktadır (Shaka ve Pawar, 2018). VGG'nin mimari temeli, doğrusal olmayan özellik öğrenimini teşvik eden ReLU aktivasyonları ile serpiştirilmiş çoklu evrişim katmanlarına dayanmaktadır. Mütevazı filtrelerin tekrarlayan uygulaması, modelin hesaplama verimliliğini korurken daha büyük alıcı alanlara etkili bir şekilde yaklaşmasını sağlayan ayırt edici bir özelliktir. Makul bir şekilde dahil edilen maksimum havuzlama katmanları, uzamsal özetleme ve değişmezliğe katkıda bulunarak modelin çeşitli uzamsal ölçeklerdeki kalıpları tanıma becerisini artırmaktadır.



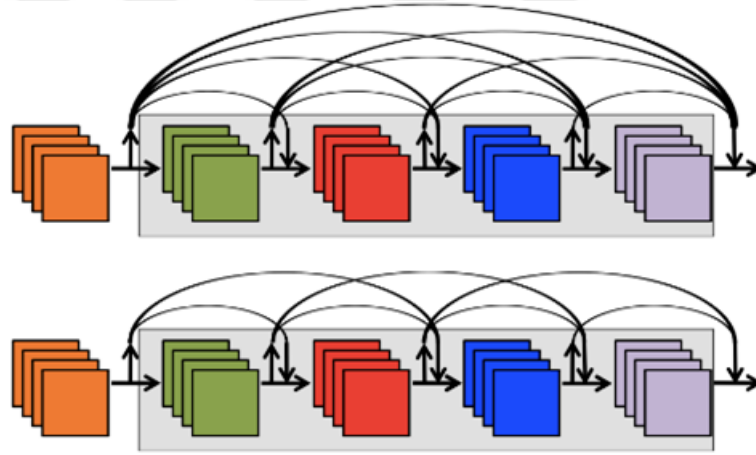
Şekil 4.1 : VGG19 mimarisi (Zheng ve diğ., 2018).

Şekil 4.1’de ayrıca VGG, mimarisinin sonuna doğru tam bağlantılı katmanları entegre ederek bir sınıflandırıcının kritik rolüne hizmet etmektedir. VGG mimarisinin karmaşık olmayan ve homojen yapısı, kavrama ve eğitim kolaylığı sağlamaktadır. Basit tasarımına rağmen VGG, karşılaştırmalı görüntü sınıflandırma görevlerinde rekabetçi bir performans sergileyerek derin evrimsel çerçevesinin etkinliğini doğrulamıştır. Bu mimari açıklık ve performans, VGG’yi derin öğrenme araştırmaları ve uygulamalarındaki sonraki gelişmeleri önemli ölçüde etkileyen temel bir model olarak konumlandırmıştır.

4.2 Yoğun Bağlantılı Evrimsel Ağlar

Yoğun bağlantılı evrimsel ağların kısaltması olan DenseNet, görüntü sınıflandırma görevleri için tasarlanmış derin bir evrimsel sinir ağı mimarisidir. DenseNet’in tanımlayıcı özelliği yoğun bağlantı modelidir. Geleneksel bir evrimli sinir ağında, her katman yalnızca bir sonraki katmana bağlıdır. DenseNet’te her katman ileri beslemeli bir şekilde sonraki tüm katmanlara bağlanmaktadır. Bu yoğun bağlantı, özelliklerin yeniden kullanılmasına olanak tanımakta ve ağ boyunca bilgi akışını teşvik etmektedir. Ayrıca kaybolan gradyan sorununun ele alınmasına da yardımcı olmaktadır (Huang ve diğ., 2017). DenseNet, birden fazla yoğun bağlantılı katmandan oluşan yoğun bloklardan oluşmaktadır. Bir yoğun bloğun girdisini X ve çıktısını H_l olarak gösterelim; burada l yoğun blok içindeki katman indeksidir. X

olarak gösterilen yoğun bloğun girdisi, yoğun bloktaki önceki tüm katmanlardan gelen özellik haritalarının $X=[H_0, H_1, \dots, H_{l-1}]$ birleştirilmesidir. Yoğun blok içindeki her katmanın çıktısı $H_l = H_{l-1} \oplus F_l(H_{l-1})$ ile elde edilmektedir. Burada $\oplus$ birleştirmeyi ve F_l katman l tarafından uygulanan doğrusal olmayan dönüşümü temsil etmektedir. Daha basit bir ifadeyle her katmanın çıktısı, önceki tüm katmanların özellik haritalarının kendi doğrusal olmayan dönüşümünün çıktısı ile birleştirilmesiyle elde edilmektedir. Bir blok içindeki yoğun bağlantı, son derece ayırt edici özelliklerin verimli bir şekilde öğrenilmesini kolaylaştırmaktadır. Yoğun bloklar arasında, özellik haritalarının uzamsal boyutlarını azaltmak için geçiş blokları kullanılmaktadır. Bu genellikle toplu normalizasyon, 1x1 evrimsel katmanlar ve ortalama havuzlama kombinasyonu ile elde edilmektedir. Geçiş blokları, öğrenilen özellikleri korurken hesaplama maliyetini ve parametre sayısını azaltmaya yardımcı olmaktadır.

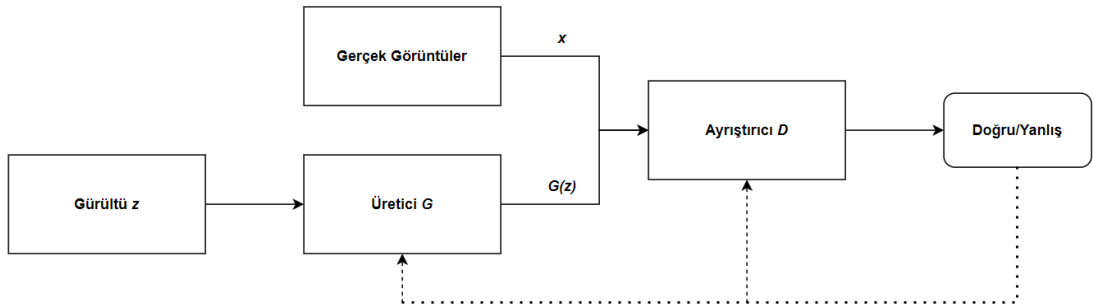


Şekil 4.2 : DenseNet mimarisinin bağlantı şekli (Hess, 2018).

Şekil 4.2’de her yoğun blok içinde genellikle darboğaz katmanları kullanılmaktadır. Bu katmanlar 1x1 konvolüsyon ve ardından 3x3 konvolüsyondan oluşmaktadır. Bu, girdi özellik haritalarının boyutluluğunu azaltmaya yardımcı olarak modeli hesaplama açısından daha verimli hale getirmektedir. Büyüme oranı, yoğun bir bloktaki her katmanın kaç yeni özellik haritasına katkıda bulunduğunu belirleyen bir hiper parametredir. Daha yüksek bir büyüme oranı daha etkileyici bir modele yol açar ancak hesaplama gereksinimlerini de artırmaktadır. DenseNet tipik olarak bir global ortalama havuzlama katmanı ve ardından sınıflandırma için tam bağlı bir katman ile sonlanmaktadır. Yoğun bağlantı, daha az parametre ile oldukça karmaşık örüntülerin öğrenilmesinde faydalı olan özelliklerin yeniden kullanılmasını sağlamaktadır.

4.3 Üretken Çekişmeli Ağlar

Üretken Çekişmeli Ağlar (GAN) arkasındaki temel fikir, iki sinir ağı, bir üretici (G) ve bir ayırıcı (D) arasındaki rekabetçi bir süreçten yararlanarak gerçekçi veriler oluşturmak için üretken bir model eğitmektir (Goodfellow ve diğ., 2020). Bir GAN'daki üretici, yeni veri örnekleri oluşturmaktan sorumludur. Rastgele gürültü üreterek başlar ve gerçek verilere benzemek için çıktısını aşamalı olarak iyileştirmektedir. Üretecin amacı, gerçek örneklerden ayırt edilemeyen veriler üretmektir. Ayırıcı, gerçek veriler ile üretici tarafından üretilen veriler arasında ayırım yapan ikili bir sınıflandırıcı görevi görmektedir. Amacı verilen bir girdinin gerçek mi yoksa sentetik mi olduğunu doğru bir şekilde tespit etmektir. Eğitim sırasında ayırt edici, üretece geri bildirim sağlayarak çıktısını iyileştirmesi için ona rehberlik etmektedir. Eğitim süreci G ve D arasında düşmanca bir oyun içermektedir. GAN mimarisine ait akış şeması aşağıdaki şekilde verilmiştir (Şekil 4.3).



Şekil 4.3 : GAN mimarisinin çalışma düzeni (Wang ve diğ., 2017).

Şekilde 4.3'teki D, üreticinin gerçekçi örnekler ürettiği ve ayırıcının gerçek ve üretilen veriler arasında güvenilir bir ayırım yapamadığı bir Nash dengesi ile sonuçlanmaktadır. Üreteç giderek daha ikna edici veriler üretmeyi amaçlarken ayırt edici de gerçek ve üretilen örnekler arasında ayırım yapma konusunda daha becerikli olmaya çalışmaktadır. Bu çekişmeli etkileşim hem üreticinin hem de ayırt edicinin zaman içinde gelişmesiyle sonuçlanmaktadır. Veri dağılımını p_d üreticinin dağılımını p_g olarak kabul ettiğimiz durumda amaç bir minimaks oyunu olarak formüle edilmektedir. Üreteç bu amacı minimize etmeyi amaçlarken, ayırıştırıcı maksimize etmeyi amaçlamaktadır:

$$\min_{\theta_g} \max_{\theta_d} V(D, G) = E_{x \sim p_{data}} [\log D(x; \theta_d)] + E_{z \sim p_z} [\log (1 - D(G(z; \theta_g); \theta_d))] \quad (4.1)$$

Denklem 4.1'e bakıldığında üretici, rastgele z gürültüsünü bir $P(z)$ ön dağılımından almaktadır. Aynı zamanda üretici, rastgele gürültü z 'yi sentetik verilere dönüştüren θ_g parametrelili bir sinir ağıdır. Ayırıcı, gerçek veri dağılımı verisi p_{data} verisinden veya $G(z; \theta_g)$ 'den üretilen veriden x gerçek verisini almaktadır. x ise girdisinin gerçek mi yoksa üretilmiş mi olduğunu gösteren bir olasılık çıktısı veren θ_d parametrelili bir sinir ağıdır. Her eğitim iterasyonunda; ayırıcı, gerçek ve üretilen örnekleri doğru şekilde sınıflandırma olasılığını en üst düzeye çıkarmak için eğitilmektedir. Eş zamanlı olarak üretici, ayırt edicinin doğru bir sınıflandırma yapma log-olasılığını en aza indirecek şekilde eğitilmektedir.

4.4 Süper Çözünürlüklü Üretken Çekişmeli Ağ

SRGAN, görüntü süper-çözünürlüğü görevi için tasarlanmış bir GAN türüdür. Görüntü süper-çözünürlüğü, tipik olarak piksel boyutlarını artırarak bir görüntünün çözünürlüğünü ve kalitesini geliştirmeyi içermektedir. SRGAN'ın birincil amacı, düşük çözünürlüklü muadillerinden yüksek çözünürlüklü görüntüler üretmektir. Derin öğrenmeden, özellikle de GAN'lardan yararlanarak geleneksel enterpolasyon tekniklerinin ötesine geçmektedir (Ledig ve diğ., 2017). SRGAN bir üretici ağ ve bir ayırıcı ağdan oluşmaktadır. Üretici düşük çözünürlüklü bir giriş görüntüsü alır ve yüksek çözünürlüklü bir çıktı oluşturur. Düşük çözünürlükten yüksek çözünürlüğe karmaşık eşlemeleri öğrenmek için tipik olarak bir CNN mimarisi kullanır. Ayırıcı, veri kümesindeki yüksek çözünürlüklü görüntüler ile üretici tarafından üretilen yüksek çözünürlüklü görüntüler arasında ayırım yapmayı amaçlamaktadır. Bu karşıt eğitim süreci, üreticinin daha gerçekçi ve görsel olarak çekici yüksek çözünürlüklü görüntüler oluşturmaya yardımcı olmaktadır. Standart GAN çerçevesine benzer şekilde, SRGAN'ın eğitimi, üretici ve ayırıcı arasında bir minimax oyunu içermektedir. Üretici, gerçek yüksek çözünürlüklü görüntülerden ayırt edilemeyen yüksek çözünürlüklü görüntüler üretmeye çalışırken, ayırıcı da gerçek ve üretilen görüntüler arasında doğru sınıflandırma yapmayı amaçlamaktadır. SRGAN'lar, basit enterpolasyon yöntemlerinin sınırlamalarını aşarak gerçekçi ayrıntılara sahip yüksek çözünürlüklü görüntüler oluşturabilmektedir. SRGAN'lar, özellikle derin mimarileriyle, hesaplama açısından yoğun olabilmekte ve eğitim için güçlü donanım gerektirmektedir. Algısal kayıp kullanımı, üretilen görüntülerin görsel kalitesini

artırarak sadece keskin olmalarını değil aynı zamanda algısal olarak gerçek yüksek çözünürlüklü görüntülere benzer olmalarını sağlamaktadır.

$$x_{yüksek} = G(x_{düşük}, \theta_G) \quad (4.2)$$

Denklem 4.2’de G , θ_G parametrelili üreteç fonksiyonudur. Üreteç, düşük çözünürlüklü bir $x_{düşük}$ görüntüsünü yüksek çözünürlüklü bir $x_{yüksek}$ görüntüsüne dönüştürmeyi amaçlamaktadır. Ayırıcı, gerçek yüksek çözünürlüklü görüntüler $x_{gerçek}$ ile üretilen yüksek çözünürlüklü görüntüler x_{gen} arasında ayırım yapmaya çalışmaktadır.

$$D = (x; \theta_D) \quad (4.3)$$

Denklem 4.3’te D , θ_D parametrelili ayırıcı fonksiyondur. $x_{gerçek}$ önceden verilmiş gerçek bir yüksek çözünürlüklü görüntüdür. x_{gen} , üretilmiş bir yüksek çözünürlüklü görüntüdür.

4.4.1 Karşıt kayıp

Karşıt kayıp (Adversarial loss), üreteç ve ayırıştırıcı arasında bir minimaks oyunu içermektedir. Üreteç aşağıdaki hedefi minimize etmeyi amaçlarken ayırıştırıcı maksimize etmeyi amaçlamaktadır.

$$\mathcal{L}_{adv}(G, D) = E_{x_{gerçek}}[\log D(x; \theta_d)] + E_{x_{düşük}} \left[\log \left(1 - D(G(x; \theta_g); \theta_d) \right) \right] \quad (4.4)$$

Denklem 4.4’teki $\mathcal{L}_{adv}$, karşıt kayıptır. $E_{x_{gerçek}}$, gerçek yüksek çözünürlüklü görüntüler üzerindeki beklentiyi ifade etmektedir. $x_{düşük}$, düşük çözünürlüklü görüntüler üzerindeki beklentiyi ifade etmektedir. İlk terim, ayırt ediciyi gerçek yüksek çözünürlüklü görüntüleri doğru şekilde sınıflandırmaya teşvik etmektedir. İkinci terim, üreticiyi, ayırt edicinin gerçek olarak sınıflandırdığı yüksek çözünürlüklü görüntüler üretmeye teşvik etmektedir.

4.4.2 Algısal kayıp

SRGAN genellikle oluşturulan ve gerçek yüksek çözünürlüklü görüntüler arasındaki algısal farkı ölçen bir algısal kayıp fonksiyonu kullanmaktadır. Algısal kayıp, yalnızca piksel bazlı kayıp fonksiyonlarına dayanmak yerine algısal benzerliği yakalamak için önceden eğitilmiş bir sinir ağı (VGG gibi) tarafından çıkarılan üst düzey özellikleri dikkate almaktadır (Johnson ve diğ., 2016). Algısal kayıp

içerisindeki VGG kaybı, büyük bir görüntü veri kümesi üzerinde önceden eğitilmiş olan VGG evrişimli sinir ağının mimarisine dayanmaktadır. Eğitilmekte olan modelin aktivasyonları ile ilgili katmanlardaki VGG ağının aktivasyonları arasındaki farkı ölçer. Bu, modeli VGG ağı tarafından öğrenilen özelliklere benzer özellikler öğrenmeye teşvik etmek için faydalı olabilir.

$$\mathcal{L}_{\text{perceptual}}(G) = \frac{1}{N} \sum_{i=1}^N \left\| \text{VGG} \left(G(x_{\text{düşük}}) \right)_i - \text{VGG}(x_{\text{gerçek}})_i \right\|_2^2 \quad (4.5)$$

Denklem 4.5'teki $\mathcal{L}_{\text{perceptual}}$, algısal kayıptır. N , VGG ağındaki katman sayısıdır. $\text{VGG} \left(G(x_{\text{düşük}}) \right)_i$ ve $\text{VGG}(x_{\text{gerçek}})_i$ sırasıyla oluşturulan ve gerçek görüntüler için i katmanındaki özellik haritalarıdır.

4.4.3 İçerik kaybı

İçerik kaybı olarak hareket eden standart L1 kaybıdır, oluşturulan görüntünün temel gerçeğe benzer içeriğe sahip olmasını teşvik etmektedir. L1 kaybı, tahmin edilen değerler ile gerçek değerler arasındaki mutlak farkı ölçer. Genellikle amacın sıkıştırılmış veya bozulmuş bir versiyondan orijinal görüntüyü kurtarmak olduğu görüntü yeniden yapılandırma gibi görevlerde kullanılır. (Sanakoyeu ve diğ., 2018). Denklem 4.6'da $\phi(x)$ içerik özelliklerini çıkaran bir fonksiyonu göstermektedir.

$$\mathcal{L}_{\text{content}}(G) = \frac{1}{M} \sum_{j=1}^M \left\| \phi \left(G(x_{\text{düşük}}) \right)_j - \phi(x_{\text{gerçek}})_j \right\|_2^2 \quad (4.6)$$

Denklem 4.6'daki $\mathcal{L}_{\text{content}}$, içerik kaybıdır. M içerik katmanlarının sayısıdır. $\phi \left(G(x_{\text{düşük}}) \right)_j$ ve $\phi(x_{\text{gerçek}})_j$ sırasıyla oluşturulan ve gerçek görüntüler için j katmanındaki içerik özellikleridir.

4.4.4 Toplam kayıp

SRGAN'ı eğitmek için nihai hedef, düşmanca kayıp ve algısal/içerik kayıplarının bir kombinasyonudur.

$$\mathcal{L}_{\text{SRGAN}}(G, D) = \lambda_{\text{adv}} \mathcal{L}_{\text{adv}}(G, D) + \lambda_{\text{perceptual}} \mathcal{L}_{\text{perceptual}}(G) + \lambda_{\text{content}} \mathcal{L}_{\text{content}}(G) \quad (4.7)$$

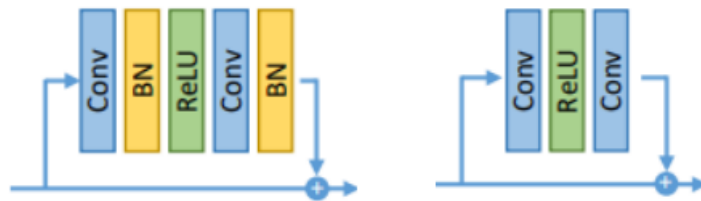
Denklem 4.7'de λ_{adv} , $\lambda_{\text{perceptual}}$ ve λ_{content} , her bir kayıp teriminin önemini kontrol eden hiper parametrelerdir. Tüm formülasyona ek olarak Gradyan cezası da

kullanılmaktadır, bazı eğitim algoritmalarında eğitim sürecinin istikrarını artırmaya yardımcı olmak için kullanılır. Gradyanları çok büyük hale gelirse ayırıcıyı cezalandırır, bu da ayırıcının eğitim verilerine aşırı uyum sağlamasını önlemeye yardımcı olabilir. Bu formülasyon, SRGAN'ın düşük çözünürlüklü girdilerden yüksek kaliteli, gerçekçi yüksek çözünürlüklü görüntüler üretmek için düşmanca eğitimi algısal ve içerik kayıplarıyla nasıl birleştirdiğinin özünü yakalamaktadır.

4.5 Geliştirilmiş Süper Çözünürlüklü Üretken Çekişmeli Ağlar

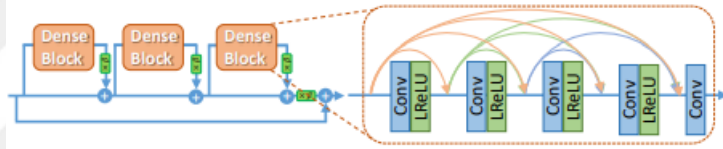
ESRGAN, özellikle süper çözünürlüklü görüntü üretimi alanında SRGAN'a göre kayda değer bir ilerlemeyi temsil etmektedir. SRGAN, görüntüleri yükseltmekten sorumlu bir üretici ağ ve gerçek ve üretilen görüntüler arasında ayırım yapmakla görevli bir ayırt edici ağ kullanarak süper çözünürlük elde etmek için rakip eğitimin entegrasyonuna öncülük etmiştir. ESRGAN, üretilen görüntülerin genel kalitesini artırmak için mimaride, kayıp fonksiyonlarında ve eğitim yöntemlerinde iyileştirmeler yaparak bu temel üzerine inşa edilmiştir. ESRGAN'daki önemli bir gelişme, daha sofistike bir ağ tasarımının benimsenmesinde yatmaktadır. SRGAN eğitim için karşıt kayıp ve MSE kaybının bir kombinasyonunu kullanırken ESRGAN algısal ve bağlamsal kayıp fonksiyonlarının iyileştirilmesine daha fazla önem vererek farklılaşmaktadır. Bu stratejik ayarlama, yalnızca görsel gerçekçilik sergileyen değil aynı zamanda önemli içerik ayrıntılarını da koruyan görüntüler üretmeyi amaçlamaktadır (Wang ve diğ., 2018).

SRGAN tarafından üretilen görüntülerin kalitesini yükseltmek için ağ mimarisinde iki önemli değişiklik uygulanmıştır. Tüm BN katmanlarının çıkarılması ve orijinal temel bloğun RRDB ile değiştirilmesi aşağıdaki şekilde verilmiştir (Şekil 4.4).



Şekil 4.4 : SRGAN'da artık bloğundaki BN katmanları kaldırıldı (Wang ve diğ., 2018).

Şekil 4.4'te de görüldüğü gibi bölüm 2.1.10'da belirtildiği gibi toplu normalleştirme katmanları, eğitim sırasında özellikleri normalleştirmek için ortalama ve standart sapma değerlerini kullanır. Eğitim sırasında üretilen ortalama ve standart sapma değerleri daha sonra tahminleri kolaylaştırmak için test sırasında toplu normalleştirme katmanları tarafından kullanılır. İlk ESRGAN araştırmalarının bulgularına göre, eğitim ve test veri kümeleri arasında istatistiksel olarak önemli farklılıklar olduğunda, toplu normalleştirme katmanları genellikle istenmeyen yapılara neden olmaktadır. Yine, araştırmacıların ilk ESRGAN yaklaşımına ilişkin bulgularına göre, GAN çerçevesinde daha derin ağlarla çalışmak, istenmeyen yapılar yaratma şansını artırmaktadır. BN katmanlarının kaldırılmasının performansı artırdığı, hesaplama karmaşıklığını azalttığı ve çeşitli ağ mimarilerinde bellek kullanımını azalttığı gözlemlenmiştir. ESRGAN, bir RRDB ekleyerek üretici ağına önemli bir yapısal değişiklik getirmektedir. Bu değişiklik aşağıda gösterilmiştir. (Şekil 4.5)



Şekil 4.5 : RRDB bloğu yapısı (Wang ve diğ., 2018).

Şekil 4.5'te gösterilen ve bölüm 4.2'deki DenseNet mimarisinden esinlenen bu blok, blok içindeki tüm katmanları doğrudan birbirine bağlayarak ağ kapasitesini artırır ve eğitimi kolaylaştırmaktadır. Orijinal SRGAN'ın artık bloğundan bu ayrılış, ağın derinliğini ve karmaşıklığını artırarak daha iyi performansa katkıda bulunmaktadır. Eğitim sürecinin zorluklarına rağmen, önceki araştırmalar daha derin ve daha bağlantılı bir ağ kullanmanın daha yüksek sonuçlar sağladığını göstermiştir. Bu çalışmada tanıtılan RRDB fikri, daha derin ağların daha doğru bulgular verebileceği fikrine dayanmaktadır. Aynı zamanda, RRDB'nin entegrasyonu, orijinal artık bloğa kıyasla daha karmaşık bir yapı sunmakta ve böylece ağın genel performansını artırmaktadır.

4.6 YOLOv8 Mimarisi

YOLO serisinin son ürünü olan YOLOv8, gerçek zamanlı, doğru nesne algılama ve hata tespit özellikleriyle ünlüdür. YOLOv8, tek aşamalı nesne algılama yöntemini

mimarilerinde bir köşe taşı olarak durmaktadır. CSPDarknet53, özellik çıkarma görevlerindeki etkinliği ile tanınan yenilikçi bir CNN mimarisi olarak ortaya çıkmaktadır. CSP Darknet53, girdi verilerinden hiyerarşik özellikler çıkarmak için küçük filtreli ardışık evrimsel katmanlar kullanmaktadır. Bu tasarım tercihi hem hesaplama verimliliğine hem de modelin ince taneli ayrıntıları yakalama kapasitesine katkıda bulunmaktadır.

4.6.2 Boyun mimarisi

YOLOv8 boyun mimarisi, omurga tarafından alınan özellikleri nesne tanımaya uygun tek bir gösterimde birleştirmeyi amaçlamaktadır. YOLOv8, boyun mimarisi olarak bir FPN kullanmaktadır. FPN, çeşitli boyutlarda özellik haritaları oluşturan yukarıdan aşağıya bir rotadan ve omurga özelliklerini yukarıdan aşağıya yol özelliklerine bağlayan bir yan bağlantıdan oluşmaktadır. Bu, YOLOv8'in farklı boyut ve en boy oranlarına sahip nesneleri tanımasını sağlamaktadır. FPN, özellik haritalarının birden fazla ölçeğini entegre ederek ağın hem küresel hem de yerel bağlamsal bilgileri etkili bir şekilde yakalamasını sağlamaktadır (Lin ve diğ., 2017).

4.6.3 CSP bağlantı modülü

CSPDarknet53'ün ayırt edici özelliği CSP bağlantı modülünün dahil edilmesidir. Bu modül, ağın farklı aşamaları arasında kısmi bağlantılar sunarak gelişmiş bilgi akışını teşvik eder ve gradyan yayılımını desteklemektedir (Wang ve diğ., 2021). Bölüm 4.2'de bahsedilen DenseNet'in öznitelik yeniden kullanım özelliklerinin avantajlarını korumaktadır. Ancak aynı zamanda gradyan akışını keserek aşırı miktarda yinelenen gradyan bilgisini önlemektedir. CSP modülü, bilginin hem kısmi bağlantılar hem de normal bağlantılar üzerinden akmasına izin vererek paralel işlemeyi teşvik etmektedir. Bu çift yollu yapı, bilginin farklı aşamalarda eş zamanlı olarak işlenmesini sağlayarak ağın anlamlı özellikleri yakalama ve tutma yeteneğini artırmaktadır. CSP bağlantıları, kaybolan gradyan sorunu gibi sorunları hafifleterek modelin karmaşık hiyerarşik özellikleri etkili bir şekilde öğrenmesini sağlamaktadır. CSP bağlantı modülü, sinir ağlarının özellik çıkarma yeteneklerini önemli ölçüde etkilemektedir. Modül, gelişmiş bilgi alış-verişini kolaylaştırarak ağın karmaşık hiyerarşik özellikleri ve kalıpları yakalamasını sağlar. Bu, özellikle nesne algılama ve görüntü segmentasyonu gibi ayrıntılı özellik temsilleri gerektiren görevlerde faydalı olmaktadır.

4.6.4 Nesne algılama kafası

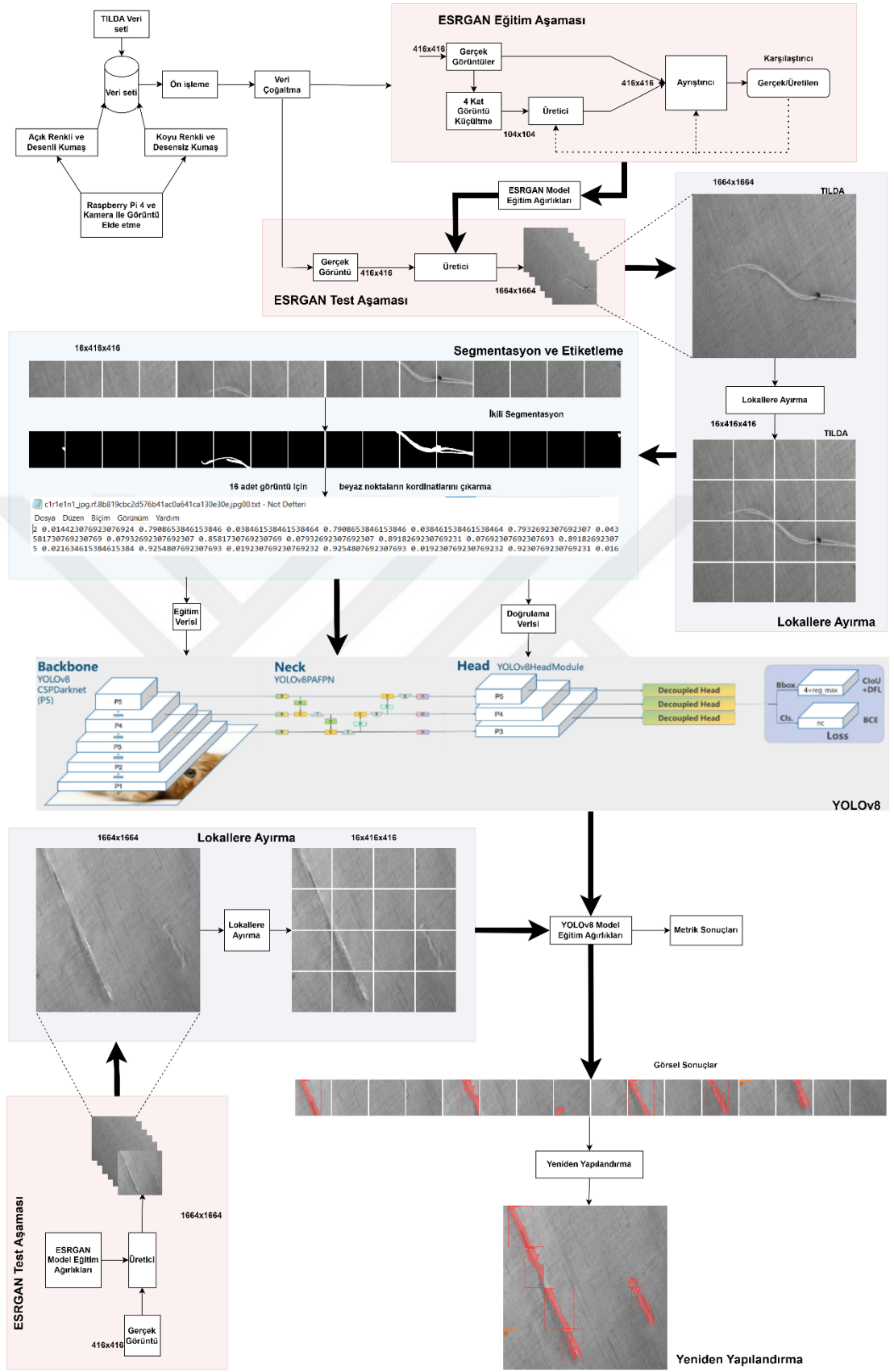
YOLOv8'in algılama kafası; sınırlayıcı kutu koordinatlarını, sınıf olasılıklarını ve nesne güven puanlarını tahmin etmekten sorumlu bir algılama kafası kullanmaktadır. YOLOv8 tespit kafası, YOLOv5 tespit kafasının çapa bazlı tahmin, dinamik çapa ataması ve odak kaybı gibi gelişmiş yeteneklere sahip değiştirilmiş bir versiyonudur. Çapa tabanlı tahmin, önceden belirlenmiş bağlantı kutuları koleksiyonuna göre öğe konumlarını tahmin etmeye yönelik bir tekniktir. Dinamik bağlantı atama yaklaşımı, veri kümesindeki öğelerin dağılımına bağlı olarak bağlantı kutularının boyutunu ve en boy oranını değiştirmektedir. Modelin performansını artırmak için odak kaybı, kategorize edilmesi zor örneklerle daha büyük ağırlıklar eklemektedir. Ayrıca başlık, çok ölçekli algılama için tasarlanmış olup modelin farklı boyutlardaki nesneleri etkili bir şekilde ele almasını sağlamaktadır.

4.6.5 Eğitim geliştirmeleri

YOLOv8'i eğitmek için SGD versiyonu olan AdamW kullanılmaktadır. AdamW, modelin eğitim verilerine gereğinden fazla uymasını önleyen bir düzenleme stratejisi olan ağırlık azalmasını içeren bir Adam varyasyonudur (Loshchilov ve Hutter, 2017). Ayrıca, eğitim verilerinin çeşitliliğini artırarak modelin genelleme yeteneklerini artırmak için rastgele kırpma, ölçeklendirme ve çevirme gibi veri artırma teknikleri de kullanılmaktadır. YOLOv8, mozaik veri artırımı, odak kaybı ve dinamik hiper parametre ayarı dahil olmak üzere gelişmiş eğitim stratejileri içerir ve gelişmiş model yakınsamasına ve sağlamlığına katkıda bulunmaktadır. Resimdeki nesnelerin konumunun ve sınıfının tespit kafası tarafından tahmin edilmesinin ardından, güven düzeyi düşük tespitleri filtrelemek ve örtüşen tespitleri entegre etmek için bir işlem sonrası aşaması gerçekleştirilir. Bunu yapmak için YOLOv8 ve NMS tekniği kullanır. NMS, her nesne için en yüksek puana sahip algılamayı seçen ve geri kalanını atan bir stratejidir (Hosang ve diğ., 2017).

5. MATERYAL VE METOT

Bu bölümde tezde TILDA veri setinden ve TILDA veri setine benzer ama farklı problemlere de uyarlanabilecek, koyu renk tonunda ve benzer desene sahip bir kumaş ile aynı renk tonunda ama farklı desene sahip bir kumaştan oluşturulan 2 farklı veri setinden bahsedilmektedir. Eğitim ve test işlemlerinin yapılabilmesi için Raspberry Pi 4b uç cihazı ile NVIDIA GeForce GTX 1050 3072MiB GPU'ya sahip özel bir cihaz ve Bursa Teknik Üniversitesine bağlı yüksek başarımlı hesaplama laboratuvarında bulunan NVIDIA GeForce RTX 3090 24GB GPU'ya sahip yüksek hesaplama işlemi yapabilen cihazdan ve bu cihazlara kurulan kütüphanelerden bahsedilmektedir. Düşük çözünürlük ve yeterli kamera performansına sahip olmayan görüntülerin iyileştirilmesi için kullanılan ESRGAN algoritmasının görüntü iyileştirme de gösterdiği performanstan bahsedilmektedir. Sonrasında ise çözünürlüğü iyileştirilen görüntülerin lokal bölgelere ayrılarak daha küçük hataların daha kolay tespit edilmesinden ve her bir lokal bölge görüntülerinin ikili segmente edilerek hatalı noktaların koordinatlarının çıkarılmasından bahsedilmektedir. Görüntüler incelenirken fark edilen aydınlatma kusurlarının, görüntü işleme yöntemleri ile farklı kumaş hatalarında kaldırılması amaçlanmaktadır. Kumaş kusurlarında her bir bölgesel görüntünün tespiti için kullanılan YOLOv8 segmentasyon algoritmasının hata tespit yönteminden bahsedilmektedir. Tespit edilen her bir bölgesel görüntüdeki hataların yeniden yapılandırılarak yüksek çözünürlüklü bir halde sunumundan söz edilmektedir. Kullanılan yöntemlerin akış diyagramı Şekil 5.1'de gösterilmiştir.



Şekil 5.1 : Tezin ana aşamaları.

Şekil 5.1’de görüldüğü üzere bu tez, deneysel iş akışına katkıda bulunan sekiz temel aşamadan oluşmaktadır:

Başlangıç aşaması, çalışmada yer alan veri setlerinin seçilmesi ve bir takım ön işleme işlemlerine tabi tutularak, görüntü arttırma metotları ile çoğaltılmasıdır. Kullanılan cihazların tanıtılması ve tüm uç cihazlara gerekli kütüphanelerin kurulması da bu aşamayı içermektedir. Bu temel adım, standartlaştırılmış bir bilgi işlem ortamı oluşturarak sonraki süreçlerin kusursuz bir şekilde yürütülmesini ve tekrarlanabilirliğini sağlamaktadır.

İkinci aşama, görüntü iyileştirme hattının temel taşı olan ESRGAN modelinden çıktı oluşturmaya odaklanmaktadır. 104x104 gürültü görüntüleri ve 416x416 gerçek görüntüler ile eğitilen model, bir noktadan sonra gerçek görüntüler kadar kaliteli, çözünürlüğü 4 kat artmış sahte görüntüler üretmektedir. Eğitim aşamasında 4 kat büyümeyi öğrenen bu model, test aşamasında ise 416x416 gerçek giriş görüntüleri verilerek 1664x1664 sahte görüntüler oluşturmaktadır. Bu aşama, modelin çözünürlüğü ve görüntü kalitesini artırma da kritik bir öneme sahiptir.

Üçüncü aşama çok önemli bir unsur, görüntüyü bölgesel parçalara bölmeyi içermektedir. Bu bölgesel görüntüler ileride kullanılacak olan tespit algoritmasının çalışma hızını bölgesel parçalara ayrılmamış görüntülerin tespitine göre arttırmaktadır.

Dördüncü aşamada ise parçalara ayrılmış her bir bölüt için hatalı bölgelerinin etiketlemesi işlemlerini içermektedir. Öncelikle ikili segmentasyon işlemi uygulanan hatalı bölgeler belirginleşmektedir. Bu işlem özellikle görüntü etiketleme amacıyla görüntülerin manuel olarak siyah beyaz formata dönüştürülmesini içermektedir. Sonrasında ise siyah beyaz formattaki bu görüntülerin hatalı kısımları (beyaz) işaretlenmiş yerlerin kod tarafından yönlendirilerek koordinatlarını çıkartmaktadır. Bu segmentasyon ve etiketleme süreci, görüntülerde dikkat veya iyileştirme gerektiren bölgelerin kesin bir şekilde tanımlanmasını sağlamayı, iyileştirme sürecinin kapsamlı bir şekilde anlaşılmasına katkıda bulunmayı amaçlamaktadır.

Beşinci aşama, etiketlenmiş görüntüleri aydınlatma farklılıklarına çözüm bulmak için tasarlanmıştır. Çözüm olarak ise CLAHE, doğrusal kontrast germe ve birleşik metot adı verilen sırası ile uygulanan ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe metodu kullanılmıştır. Tüm bu işlemler veri kümesinin farklı

aydınlatma koşullarına karşı farklı kumaş hata türlerini belirginleştirmek amacıyla uygulanmıştır.

Altıncı aşama, hem görüntü işleme uygulanmış veri kümelerinin hem de işlenmemiş veri kümesinin YOLOv8 modelinde eğitimi etrafında dönmektedir. Bu aşama, modelin öğrenme yeteneklerinin ve genel performansının değerlendirilmesinde kritik bir dönüm noktasını temsil etmektedir.

Yedinci aşama ise YOLOv8 modeli ile tespit edilen her bir görüntü yamasının yeniden yapılandırılarak eski yüksek çözünürlüklü ve hataların tespiti yapılmış hale getirilmesini kapsamaktadır.

Sekizinci ve son aşama yeni etiketlenen veri kümesinin YOLOv8 modeli aracılığıyla işlenmek üzere gerçek zamanlı olarak uç cihazlara dinamik olarak dağıtılmasıdır. Bu gerçek dünya uygulama aşaması yalnızca modelin etkinliğini değerlendirmekle kalmayıp, aynı zamanda gelişmiş bilgisayarlı görme algoritmalarını gerçek zamanlı senaryolarda dağıtmanın pratik uygulanabilirliğini de vurgulamaktadır.

Bu aşamaların yapılandırılmış ve sıralı doğası, araştırmaya yönelik titiz ve sistematik bir yaklaşımın altını çizerek veri kümesi özellikleri, model performansları ve gerçek dünyaya uygulanabilirlik arasındaki karmaşık etkileşimi kapsamlı bir şekilde anlatmaktadır.

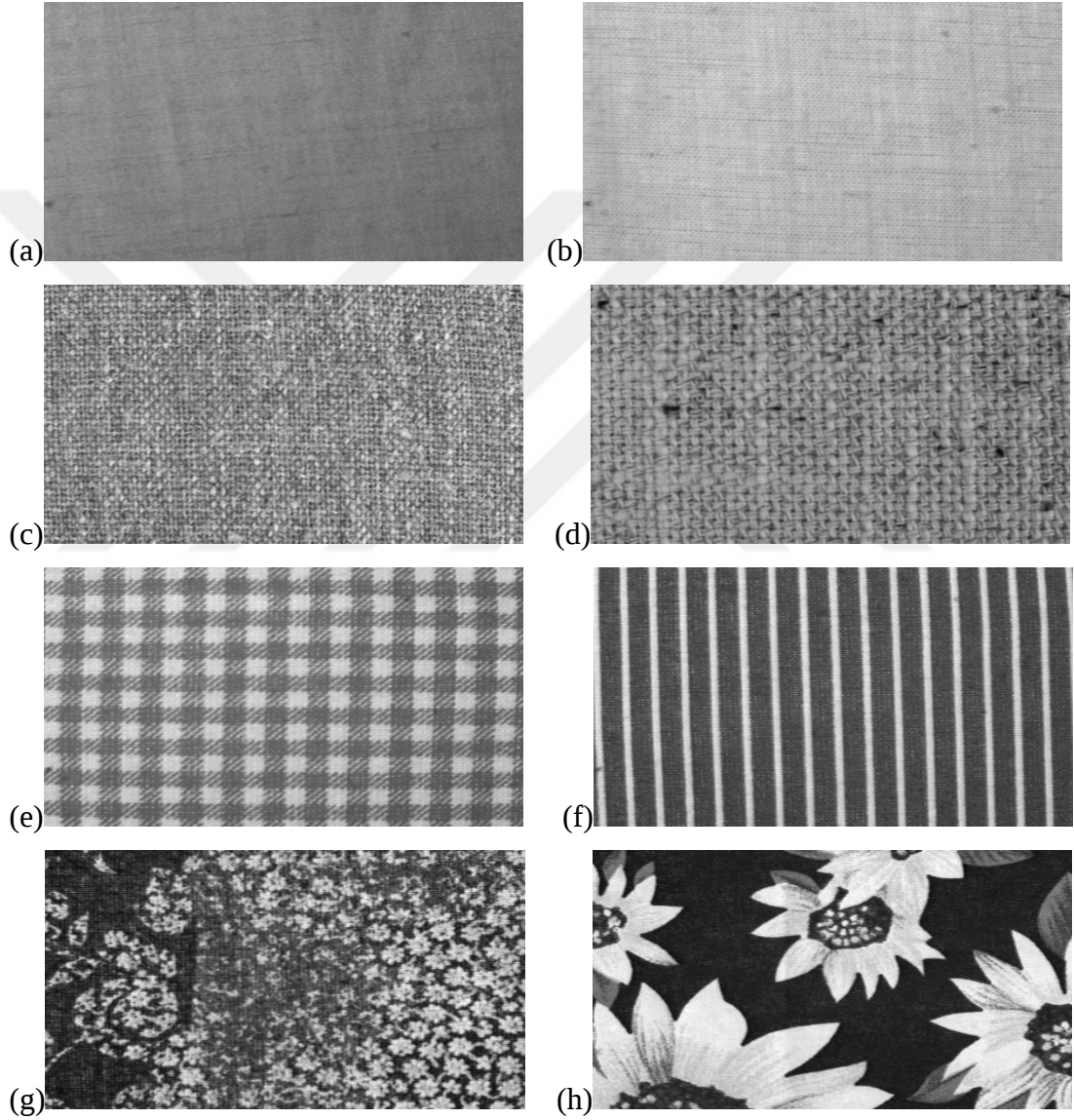
5.1 Veri Setleri

Bu tez iki farklı veri setinin kullanımıyla desteklenen, yöntemsel olarak kapsamlı bir araştırmayı üstlenmektedir. İlk veri seti olan TILDA, köklü kalitesi ve uygunluğu nedeniyle seçilmişken ikinci veri seti, çalışmanın özel gereksinimlerine uyum sağlamak üzere titizlikle hazırlanmıştır. Özel veri setinin geliştirilmesi, donanım cihazlarının dikkatli seçilmesini gerektirmiştir. Bu başlık altında veri setleri detaylandırılmıştır.

5.1.1 TILDA veri seti

Bu tez çalışması, Hamburg Teknik Üniversitesi tarafından titizlikle hazırlanan, birincil referans veri seti olarak TILDA veri setinden yararlanmaktadır. TILDA veri seti, kumaş sınıflarını kapsamlı bir şekilde temsil etmesi ve bunları yüzey yapılarının düzenliliğine göre kategorilere ayırmasıyla ünlüdür. Veri setinde cd1 ve cd2 olarak

adlandırılan iki dizin bulunmaktadır. Her iki klasörde de dört kategori yer almaktadır. cd1'deki dört grup (c1r1, c1r3, c2r2, c2r3) desensiz kumaş örneklerinden oluşurken, cd2'deki diğer dört grup (c3r1, c3r3, c4r1, c4r3) desenli örnekleri içermektedir. Her grup için her biri elli örnek içeren sekiz alt dizin oluşturulmuştur (e0, e1, e2, e3, e4, e5, e6, e7). Her türlü kusura sahip örnekler diğer alt dizinlerde bulunur, ancak hatasız örnekler e0'da bulunmaktadır. Kusursuz kumaş, doku tekrarının bozulmadığı kumaştır.



Şekil 5.2 : TILDA kumaş örnekleri (a) c1r1 (b) c1r3 (c) c2r2 (d) c2r3 (e) c3r1 (f) c3r3 (g) c4r1 (h) c4r3 gösterilmiştir.

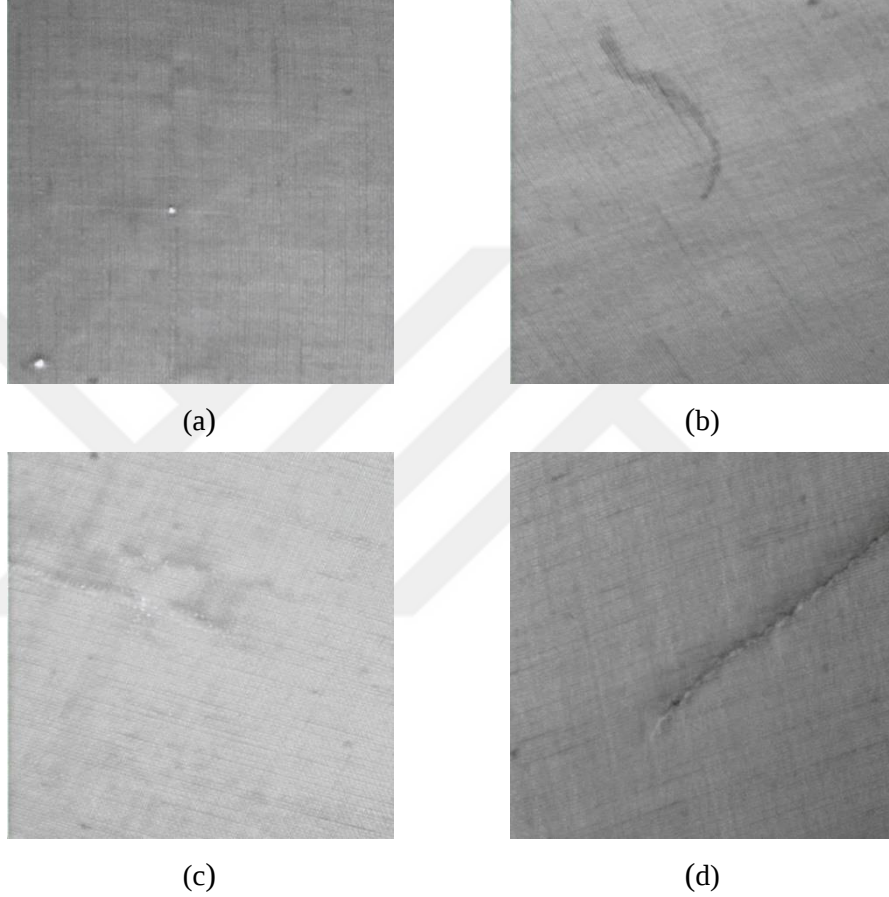
Şekil 5.2’de gösterilen TIFF formatındaki görüntüler, 768x512 piksel çözünürlük sergilemekte ve 256 gri düzeyiyle karakterize edilen monokrom gri görüntüleri göstermektedir.

Çizelge 5.1 : TILDA verisindeki desensiz kumaş tipleri, hata türleri ve sayıları.

Desensiz Kumaş				
Kumaş Türü	Grup	Sınıf	Hata Sınıf Açıklaması	Görüntü Sayısı
İpek Kumaş	c1r1	e0	Kumaşta hata yok	50
		e1	Kumaşta delik hataları	50
		e2	Yağ lekeleri ve renk hatası	50
		e3	İplik hatası	50
		e4	Kumaş dokusunda yabancı cisimler	50
		e5	Kumaşta kırıksıklıklar	50
		e6	Değişen aydınlatma koşulları	50
		e7	Kameranın eğilmesi, yerinin değişmesi	50
İpek Kumaş	c1r3	e0	Kumaşta hata yok	50
		e1	Kumaşta delik hataları	50
		e2	Yağ lekeleri ve renk hatası	50
		e3	İplik hatası	50
		e4	Kumaş dokusunda yabancı cisimler	50
		e5	Kumaşta kırıksıklıklar	50
		e6	Değişen aydınlatma koşulları	50
		e7	Kameranın eğilmesi, yerinin değişmesi	50
Yün Kumaş	c2r2	e0	Kumaşta hata yok	50
		e1	Kumaşta delik hataları	50
		e2	Yağ lekeleri ve renk hatası	50
		e3	İplik hatası	50
		e4	Kumaş dokusunda yabancı cisimler	50
		e5	Kumaşta kırıksıklıklar	50
		e6	Değişen aydınlatma koşulları	50
		e7	Kameranın eğilmesi, yerinin değişmesi	50
Yün Kumaş	c2r3	e0	Kumaşta hata yok	50
		e1	Kumaşta delik hataları	50
		e2	Yağ lekeleri ve renk hatası	50
		e3	İplik hatası	50
		e4	Kumaş dokusunda yabancı cisimler	50
		e5	Kumaşta kırıksıklıklar	50
		e6	Değişen aydınlatma koşulları	50
		e7	Kameranın eğilmesi, yerinin değişmesi	50
Toplam		1400 hatalı 200 hatasız desensiz kumaş		1600

TILDA veri setinde bulunan desensiz kumaşlardaki hata sınıflarının açıklamaları Çizelge 5.1’de gösterilmiştir.

Özellikle veri seti, her kusurlu görüntüye eşlik eden ve sonraki analizler için değerli bağlam sağlayan metinsel açıklama içermektedir. TILDA veri koleksiyonunda toplam 3228 resim ve hata açıklamalarını içeren 2828 metin dosyası yer almaktadır. Veri seti 1,27 GB'lık bir hacme sahiptir. Bu tez çalışmasında uygulamada kullanılmak için TILDA veri setinden ipek kumaşı görüntüleri ve bu görüntülerin e1, e2, e3, e4 kumaş hata görüntüleri alınmıştır.



Şekil 5.3 : TILDA veri setinde kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri.

Şekil 5.3'te TILDA veri setinden örnekler görülmektedir. Şekil 5.3.a'da kumaş üzerinde bulunan delikten, Şekil 5.3.b'de kumaş üzerinde bulunan yağ lekesinden, Şekil 5.3.c'de kumaşın iplik hatasından ve Şekil 5.3.d'de kumaş üzerinde yabancı obje bulunmasından kaynaklanan hatalar gösterilmiştir. TILDA veri setinde Şekil 5.3'te görülen hataları barındıran 896 görsel bulunmaktadır.

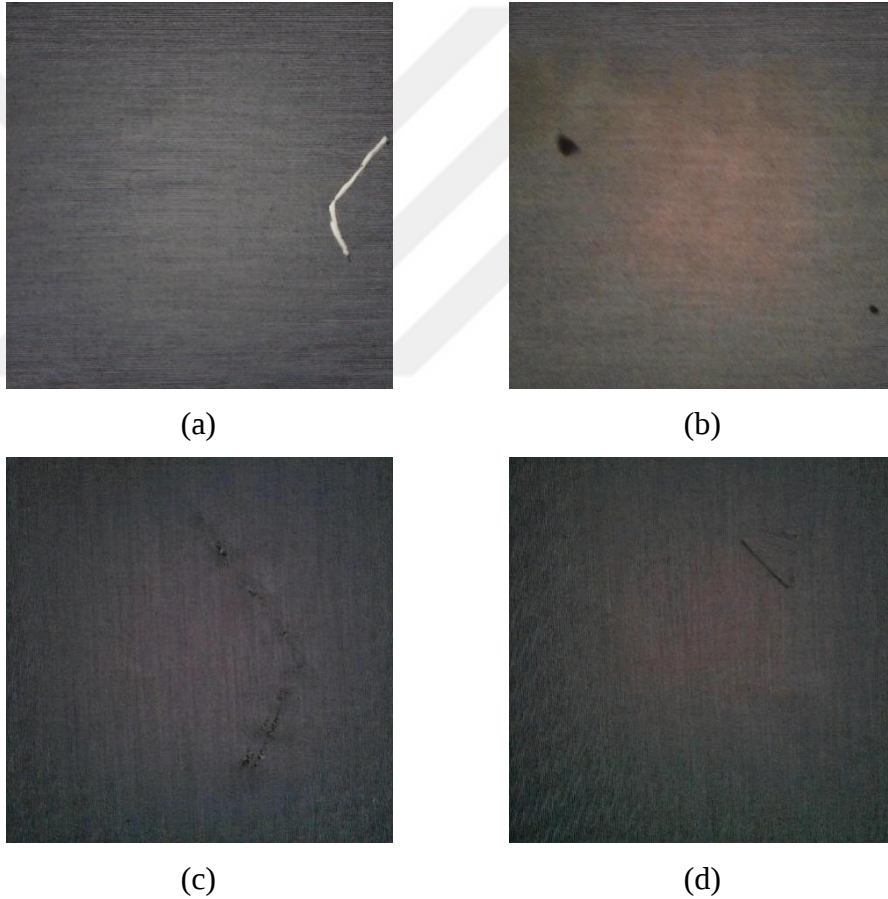
Andersen (2021), bu tezde kullanılan TILDA veri setini etiketleyerek kullanıma sunmuştur.

Bu çalışmada kullanılan veri setindeki etiket görüntülerinden faydalanılarak, her bir bölüt için tekrardan etiketleme işlemi özenle yapılmıştır. Veri seti özellikle ipek kumaş türüne odaklanmaktadır ve sekiz farklı sınıf içerisinde delik ve kesim hataları, yağ lekeleri ve renk hataları, iplik kusurları ve kumaş doku sınıfı içindeki yabancı obje örnekleri tercih edilmiştir. Model eğitimi için veri setini standartlaştırmak amacıyla, süper çözünürlük için büyütülen görüntülerin 416x416 piksele yeniden boyutlandırıldığı bir ön işleme aşaması yapılmıştır. Eş zamanlı olarak gri tonlamalı dönüşümü de içeren otomatik bir yönlendirme işlemi uygulanmıştır. Her eğitim örneğinde hem yatay hem de dikey olarak kapsamlı bir analiz gerçekleştirilerek üç farklı çıktı elde edilmiştir. Bu çıktılar hem saat yönünde hem de saat yönünün tersine düzenli dönüşleri ve 90° dönüşleri kapsayarak veri kümesini çeşitli perspektifler ve yönelimlerle zenginleştirmektedir. Verisetinin kusur sınıflarındaki zenginliği, stratejik ön işleme ve büyütme ile birleştiğinde, yalnızca eğitim için uygunluğunu arttırmakla kalmayıp, aynı zamanda ESRGAN modelinin kumaş modeli iyileştirme ve hata tespitindeki etkinliğinin kapsamlı bir şekilde araştırılmasını da sağlamaktadır.

5.1.2 Bu çalışmada oluşturulan veri seti

Bu tezde, TILDA veri setinin özellikleriyle yakından uyumlu özellikler sergileyen iki farklı kumaş tedarik edilmiştir. İlk kumaş, TILDA verileriyle benzer renk niteliklerini paylaşıırken, fark edilebilir bir kumaş deseni sunmaktadır. İkinci kumaş, daha koyu bir renk tonu ortaya koymaktadır ancak TILDA veri setini anımsatan bir kumaş desenine sahiptir. Aynı kumaş görüntü verileri hem Raspberry Pi kamera kartı v1.3 ile hem de 12,3 megapiksel Sony IMX477 sensörlü gelişmiş kameranın kullanılmasıyla, açık ve koyu kumaşların üzerinde bulunan kumaş kusurlarından elde edilmiştir. Her iki kameradan da görüntü alma işlemi doğal gün ışığında gerçekleştirilmiştir. Günün belirli vakitlerinde alınan görüntülerin aydınlatma koşulları birbirleriyle benzer olmaktadır. Görüntü alma işlemleri sırasında kamera lensinden dolayı kenarda bozulmalar meydana gelmektedir. Görüntünün kenarlarındaki bozulmalar, kırpmaya ön işleme sonrasında düzenlenmiştir. TILDA veri setine benzer olması amacıyla, delik ve kesim hataları, yağ lekeleri ve renk hataları, iplik kusurları ve kumaş doku sınıfı içindeki yabancı obje hataları oluşturulmuştur. Her iki kumaş çeşidi ve dört farklı kusur için 8'er farklı görüntü alınmıştır ve toplamda 32 görüntü elde edilmiştir. Daha önce TILDA veri kümesinde kullanılan görüntü artırma yöntemlerinden yararlanılarak benzer teknikler burada da titizlikle

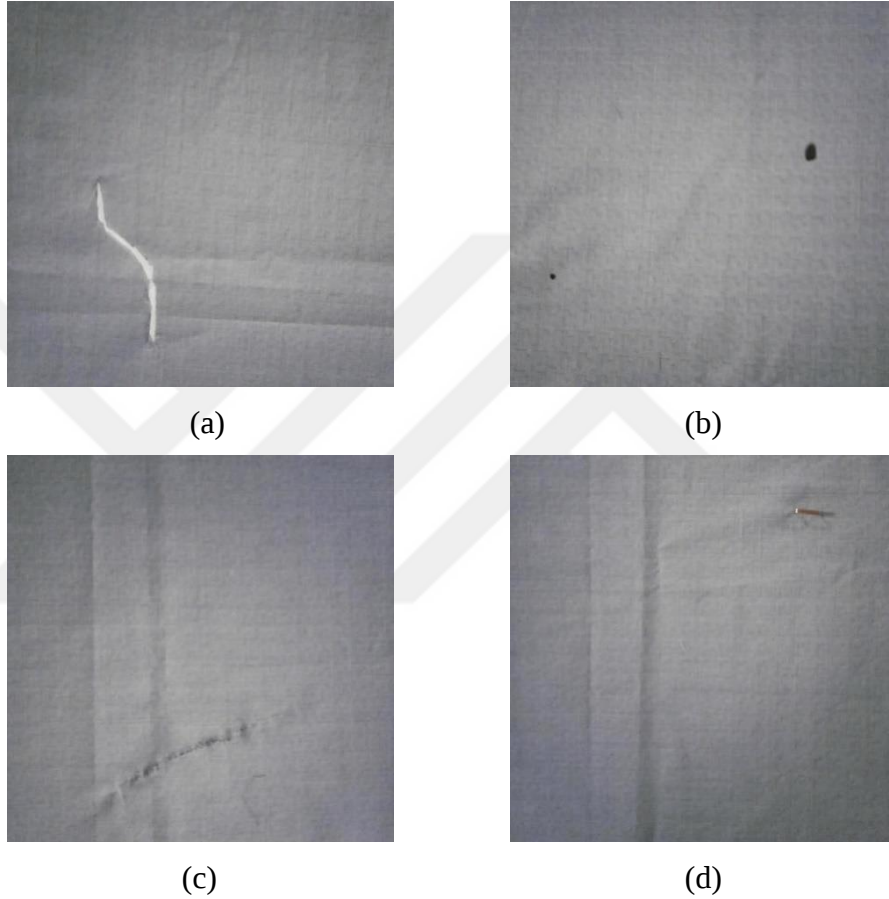
uygulanmıştır. Giriş görüntüsü, Pillow kütüphanesi kullanılarak açılmaktadır. Görüntü, yatay, dikey ve saat yönünün tersine 90 derece döndürülür. Her dönüştürülmüş görüntü, çıktı dosyasına kaydedilir. Ana döngü, belirtilen giriş klasöründeki tüm dosyaları işler. İşlenen her görüntü için bir çıktı dosyası adı oluşturulur ve dönüştürme işlevi çağrılmaktadır. 32 adet görüntü verisi, görüntü zenginleştirilmesi ile 4 kat artarak 128 adet olmuştur. Düşük çözünürlükteki kameradan elde edilen görüntülerin yarısı ESRGAN modelinin eğitim aşamasında yarısı ise test aşamasında kullanılmaktadır. Açık kumaş verisinde ve koyu kumaş verisinde 64'er adet bölütlenmeye hazır görüntü bulunmaktadır. Düşük çözünürlüklü kamera ile oluşturulan veri setindeki koyu kumaş örnekleri Şekil 5.4'te gösterilmiştir.



Şekil 5.4 : Düşük çözünürlüklü kamera ile oluşturulan veri setindeki koyu kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri.

Şekil 5.4'te düşük çözünürlüklü kamera ile oluşturulan veri setindeki koyu kumaş örnekleri görülmektedir. Şekil 5.4.a'da koyu kumaş üzerinde bulunan delikten, Şekil 5.4.b'de koyu kumaş üzerinde bulunan yağ lekesinden, Şekil 5.4.c'de koyu kumaşın iplik hatasından ve Şekil 5.4.d'de koyu kumaş üzerinde yabancı obje bulunmasından

kaynaklanan hatalar gösterilmiştir. Şekil 5.4'e bakıldığında kumaşların renginin koyu olmasına karşın, TILDA veri setinde bulunan kumaşlarla benzer bir desene sahip olduğu gözlemlenebilmektedir. Düşük çözünürlüklü kameraya ile oluşturulan veri setinde bulunan açık renk kumaşlara ait kumaş hatası örnekleri ise Şekil 5.5'te gösterilmiştir.

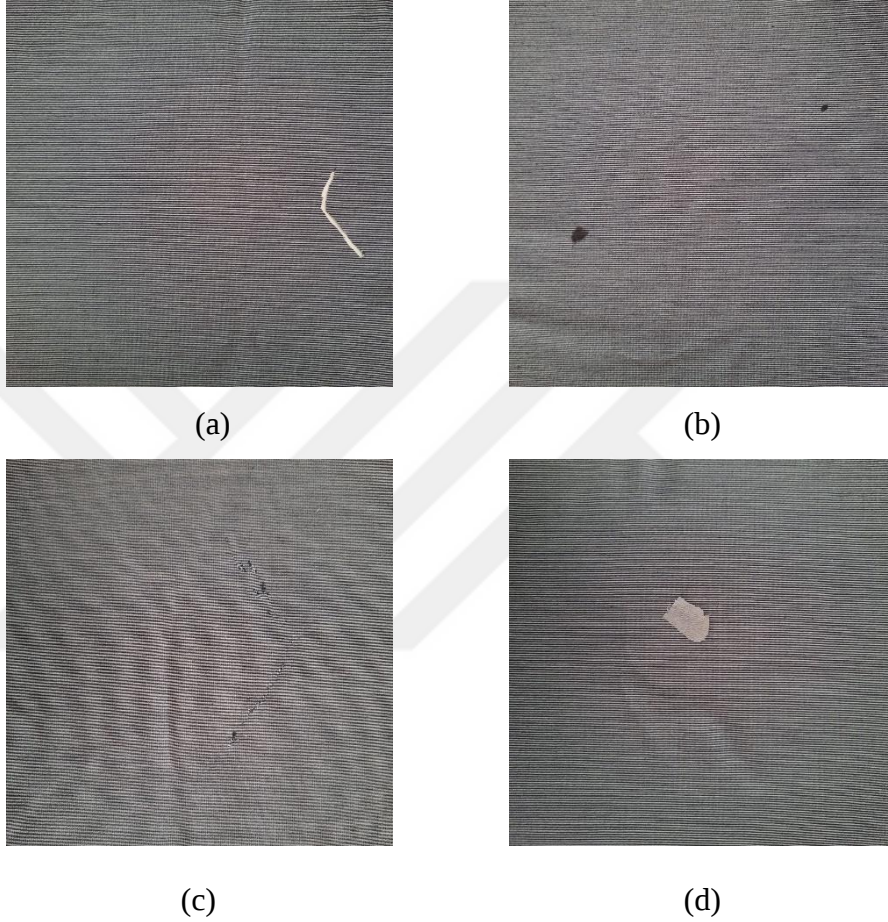


Şekil 5.5 : Düşük çözünürlüklü kamera ile oluşturulan veri setindeki açık kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri.

Şekil 5.5'te düşük çözünürlüklü kamera ile oluşturulan veri setindeki açık kumaş örnekleri görülmektedir. Şekil 5.5.a'da açık kumaş üzerinde bulunan delikten, Şekil 5.5.b'de açık kumaş üzerinde bulunan yağ lekesinden, Şekil 5.5.c'de açık kumaşın iplik hatasından ve Şekil 5.5.d'de açık kumaş üzerinde yabancı obje bulunmasından kaynaklanan hatalar gösterilmiştir.

Yüksek çözünürlükteki kameradan elde edilen görüntüler yüksek çözünürlük değerlerine sahip olduğundan ESRGAN modelinin eğitim aşamasında

kullanılmamaktadır. Çözünürlük değerleri kırıldıktan sonra 1664x1664'e tekrar boyutlandırılır. Her bir 1664x1664 yüksek çözünürlüklü görüntü 416x416'lık lokal görüntülere ayrılarak YOLOv8 modeli eğitimi için hazır hale getirilmektedir. Açık kumaş verisinde ve koyu kumaş verisinde 349'er adet 416x416 boyutunda etiketlenmiş görüntü bulunmaktadır. Yüksek çözünürlüklü kamera ile oluşturulan veri setindeki koyu kumaş örnekleri Şekil 5.6'da gösterilmiştir.



Şekil 5.6 : Yüksek çözünürlüklü kamera ile oluşturulan veri setindeki koyu kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri.

Şekil 5.6'te yüksek çözünürlüklü kamera ile oluşturulan veri setindeki koyu kumaş örnekleri görülmektedir. Şekil 5.6.a'da koyu kumaş üzerinde bulunan delikten, Şekil 5.6.b'de koyu kumaş üzerinde bulunan yağ lekesinden, Şekil 5.6.c'de koyu kumaşın iplik hatasından ve Şekil 5.6.d'de koyu kumaş üzerinde yabancı obje bulunmasından kaynaklanan hatalar gösterilmiştir. Yüksek çözünürlüklü kamera ile oluşturulan veri setinde bulunan açık renk kumaşlara ait kumaş hatası örnekleri ise Şekil 5.7'da gösterilmiştir.



(a)



(b)



(c)



(d)

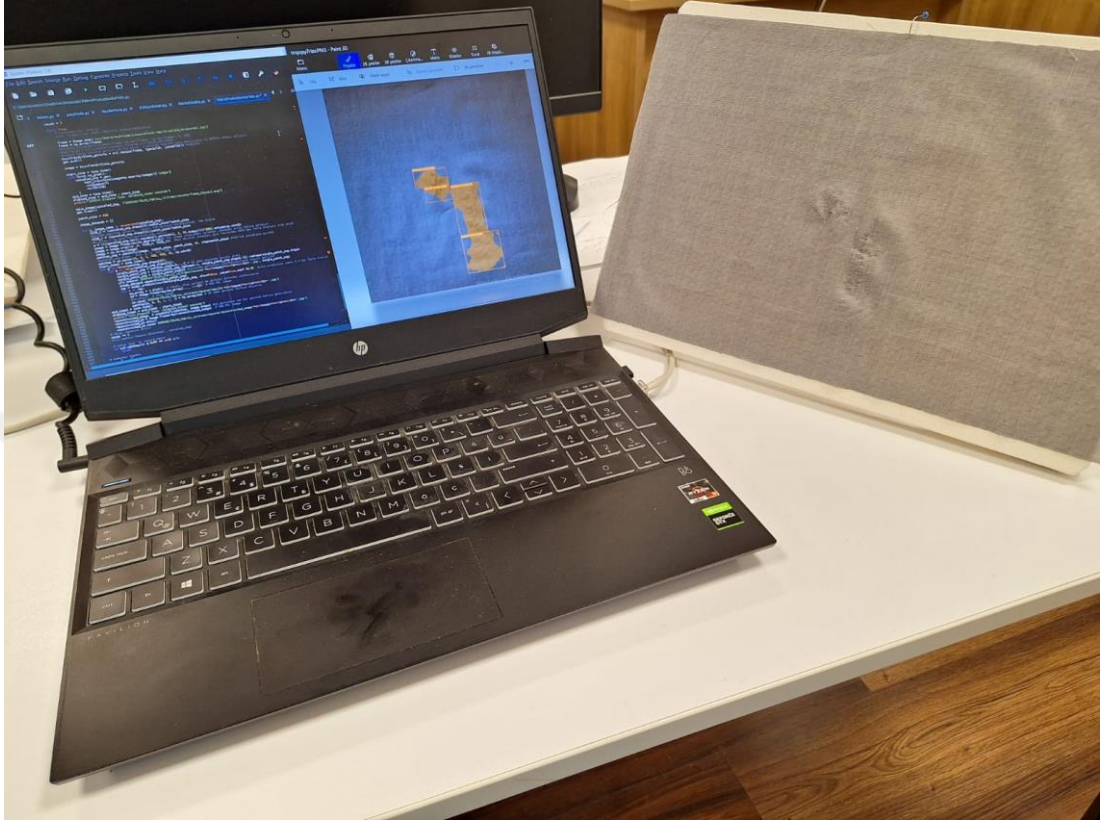
Şekil 5.7: Yüksek çözünürlüklü kamera ile oluşturulan veri setindeki açık kumaş üzerinde bulunan (a) delik hatası, (b) yağ hatası, (c) iplik hatası ve (d) obje hatası örnekleri.

Şekil 5.7’de yüksek çözünürlüklü kamera ile oluşturulan veri setindeki açık kumaş örnekleri görülmektedir. Şekil 5.7.a’da açık kumaş üzerinde bulunan delikten, Şekil 5.7.b’de açık kumaş üzerinde bulunan yağ lekesinden, Şekil 5.7.c’de açık kumaşın iplik hatasından ve Şekil 5.7.d’de açık kumaş üzerinde yabancı obje bulunmasından kaynaklanan hatalar gösterilmiştir. Şekil 5.5’e ve 5.7’ya bakıldığında kumaşların renginin TILDA veri setinde bulunan kumaşlara benzer şekilde açık olmasına karşın, TILDA veri setinde bulunan kumaşlardan farklı bir desene sahip olduğu gözlemlenebilmektedir.

5.2 Deney Seti Donanım Sistemi

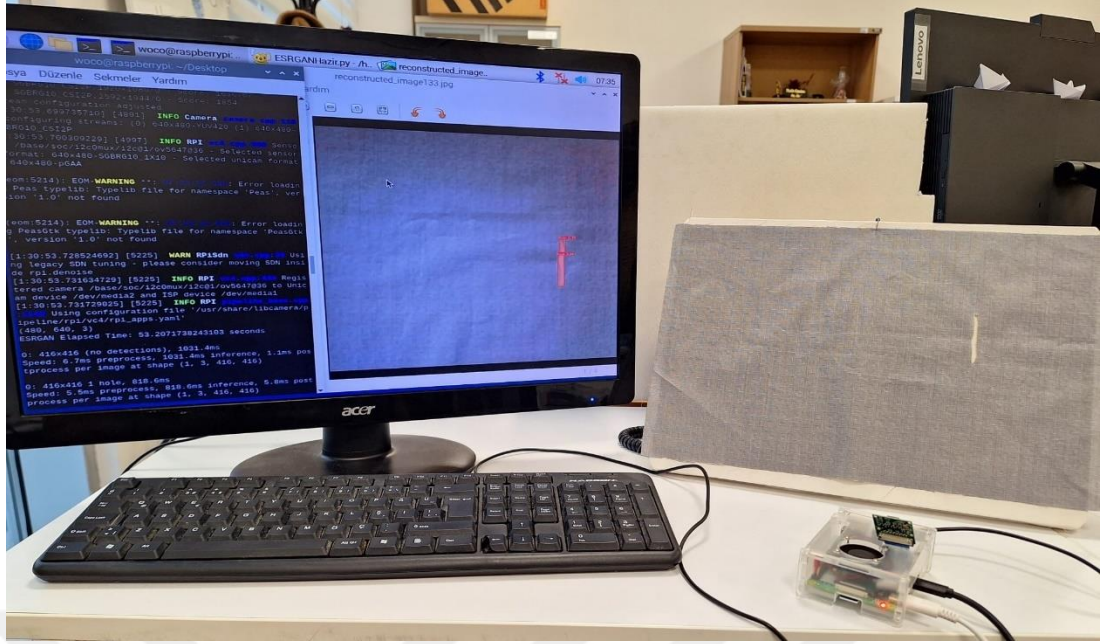
Bu tezde yeni bir veri setinin oluşturulması için fiyatı uygun ve çok yüksek gereksinimlere ihtiyaç olmayan Raspberry Pi kamerası tercih edilmiştir. Buna ek olarak hem eğitim aşamasında hem de test aşamasında birden fazla donanım

ürünlerinden faydalanılmıştır. Bunlardan ilki proje boyunca eğitim aşamasında kullanılan yüksek başarılı bilgisayar, ikincisi geliştirme aşamasından başlayarak tüm aşamalarda kullanılan kişisel bilgisayar ve üçüncüsü de Raspberry Pi'dır.



Şekil 5.8 : Proje de kullanılan kişisel bilgisayar ve hatalı açık kumaş görseli.

Şekil 5.8'de kişisel bilgisayar üzerindeki kameradan yapılan test ve gerçek zamanlı çalışma denemesinde, kumaş üzerindeki iplik hatasının tespit edildiği gözlemlenmektedir. Çözünürlük artırma işleminin evrimsel ağlar ile yapılmasıyla birlikte görüntü çözünürlükleri arttıkça buna paralel olarak hesaplama gereksinimleri de artmaktadır. Eğitim aşamasında modelin yapısını bozmadan, yığın (batch) sayısının azaltılması ile hesaplama kaynaklarından tasarruf edilebilmektedir. Bu gibi yöntemler kullanışlı olmasına rağmen yığın sayısının azaltılması, epok sayısının artmasına o da model eğitim süresine bir o kadar etki etmektedir. Kişisel bilgisayar da eğitim aşaması ancak yığın sayısı 1 veya 2 olunca çalışmakta olan ESRGAN modeli, yüksek başarılı bilgisayarlarda ise her seviyede istenilen performansta çalışmaktadır. Tüm bu eğitim aşamalarının gerçek zamanlı olarak çalışma işlemi ise Raspberry Pi üzerinde gerçekleştirilmiştir.



Şekil 5.9 : Proje de kullanılan Raspberry Pi 4b ile kamerası ve hatalı açık renk kumaş görseli.

Şekil 5.9’da Raspberry Pi 4b ile kamerası üzerinden gerçek zamanlı veri okuma ve otomatik hata tespiti işlemleri gerçekleştirilmiştir. Şekil 5.9’daki kumaş görüntüsü üzerindeki delik hatasının ekranda tespit edildiği gösterilmektedir. Raspberry Pi’da eğitim aşaması yapılmadığı gibi ESRGAN modelinin sadece üretici kısmının çalışması için modelin yapısında değişikliğe gidilip, katman sayısında azaltma yapılmıştır. Bu bölümde ilgili donanım birimleri açıklanacaktır.

5.2.1 Raspberry Pi 4B

Raspberry Pi 4B, 2019 yılında Raspberry Pi Vakfı tarafından tasarlanan kompakt, tek kartlı bir bilgisayar olarak ortaya çıkmıştır. Broadcom BCM2711 dört çekirdekli Cortex-A72 (ARM v8) 64 bit SoC ile desteklenen bu yineleme, 8 GB LPDDR4-3200 SDRAM seçenekleriyle artırılmış hesaplama yetenekleri sunarak çok yönlülüğünü artırmaktadır. 40 pinli bir GPIO başlığının dahil edilmesi, donanım arayüzünü güçlendirerek kullanıcıların robotik, ev otomasyonu ve IoT projelerine katılmasını sağlamaktadır. Uygulamalarda Raspberry Pi 4B, eğitimde, amatör çabalarda, prototip oluşturmada ve uygun maliyetli küçük ölçekli sunucu dağıtımlarında kullanım alanı bulmuştur. Kompakt form faktörü, gelişmiş özellikler ve kapsamlı topluluk desteğiyle birleştiğinde, çok çeşitli bilgisayar ve eğitim girişimleri için öne çıkan bir seçenek olarak konumlanmaktadır.

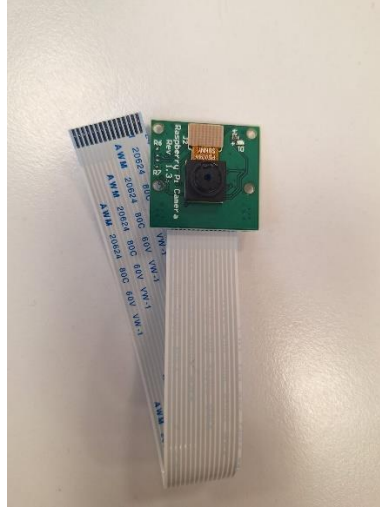


Şekil 5.10 : Proje de kullanılan Raspberry Pi 4B.

Şekil 5.10'da Raspberry Pi 4b için tasarlanan kompakt bir dış tasarım ve ısınma sorununa çözüm olarak eklenen fan görülmektedir.

5.2.2 Raspberry Pi kamera kartı v1.3

5 megapiksel çözünürlük ve 1080p video özelliğine sahip Raspberry Pi kamera kartı v1.3, Raspberry Pi tek kartlı bilgisayarlarla kusursuz entegrasyon için özel olarak tasarlanmıştır. Omnivision OV5647 sensörünü ve 3,6 mm odak uzaklığına sahip sabit odaklı lensi kullanan bu kompakt modül, Raspberry Pi kartlarındaki Kamera Seri Arayüzü (CSI) bağlantı noktasıyla doğrudan arayüz oluşturmaktadır. Uyumluluğu, sürüm 1, 2, 3 ve sonraki sürümler dahil olmak üzere çeşitli Raspberry Pi modellerini kapsamaktadır. CMOS tipi bir görüntü sensörü olarak çalışan kamera kartı, yaklaşık 54 derecelik yatay bir görüş alanı sergilemektedir. Tipik olarak Raspberry Pi kartından (3,3V) güç alan kamera kartı, fotoğraf ve video kaydından bilgisayarlı görüntü projelerine kadar çeşitli uygulamalara uygun, çok yönlü bir görüntüleme çözümüdür. Çeşitli alanlardaki projeler için temel bir bileşen sağlayan Raspberry Pi kamera kartı v1.3, resmi Raspberry Pi kamera yazılımı ve kütüphaneleri tarafından desteklenmektedir. Raspberry Pi bağlantısı için kamera kablosu gibi ek aksesuarlarla donatılmış bu kamera kartı, eğitim ve bilgisayarlı görme uygulamaları için erişilebilir ve uygun maliyetli bir araç olarak hizmet etmektedir.



Şekil 5.11 : Proje de kullanılan Raspberry Pi kamera ve entegresi.

Şekil 5.11’de Raspberry Pi için özel olarak tasarlanmış kamera görülmektedir. Ayrıca Raspberry Pi ile kolayca entegre olabilmesi için kablo entegresi de bulunmaktadır. Kamera kartı, Raspberry Pi’ye 15 pimli FFC kablosu kullanılarak bağlanmaktadır. 2592 x 1944 piksel statik resim çözünürlüğü sunan kamera, gerçek zamanlı çalışmada ise video kaydı alındığı için 640x480 piksel çözünürlüğünde saniyede 31.85 kare sayısı elde edilmektedir.

5.2.3 Raspberry Pi 12,3 MP sensörlü yüksek kaliteli kamera

Raspberry Pi 12,3 MP Sensörlü Yüksek Kaliteli Kamera, gelişmiş 12,3 megapiksel Sony IMX477 sensörünü entegre eder. Arkadan aydınlatmalı mimarisiyle tanınan bu sensör, özellikle düşük ışık koşullarında olağanüstü görüntüleme performansına katkıda bulunuyor.



Şekil 5.12 : IMX477 sensörlü kamera ve PT361060M3MP12 CS montaj lensi.

Şekil 5.12'deki Raspberry Pi kartları için özel olarak geliştirilen bu kamera modülü, Raspberry Pi ekosistemiyle kusursuz entegrasyon sağlar ve çalışması için bir C veya CS montajlı lens gerektirir. Özellikle, çeşitli değiştirilebilir lenslerle uyumluluk sağlayarak kullanıcılara çeşitli görüntüleme ihtiyaçlarına entegre olabilmektedir. Maksimum 4056 x 3040 piksel (12,3 megapiksel) statik resim çözünürlüğü sunan kamera, gerçek zamanlı çalışmada ise video kaydı alındığı için maksimum 3840 x 2160 video çözünürlüğünde saniyede 30 kare sayısı elde edilmektedir. Tez kapsamında PT361060M3MP12 CS montaj lensi birlikte kullanılmaktadır. 3,6 mm odak uzaklığına sahip olan bu lens, lens ile görüntü sensörü arasındaki mesafeyi belirler.

5.2.4 Yüksek başarılı hesaplama laboratuvarı ve kişisel bilgisayar

Eğitim aşaması için kullandığımız Bursa Teknik Üniversitesine bağlı yüksek başarılı hesaplama laboratuvarında (HPC) bulunan bilgisayarlarda ise Intel® Core™ i9-10900X, 3.70 GHz, 10 Core / 20 işlemci, toplamda 128GB DDR4 3000MHz RAM ve 2 adet NVIDIA GeForce RTX 3090 24GB 384-bit GDDR6X GPU bulunmaktadır. Modellerin geliştirme aşaması ve eğitim aşamasında ise kişisel bilgisayar ile çalışma yapılmaktadır. Kişisel bilgisayarda ise AMD Ryzen 5 4600H with Radeon Graphics 3.00 GHz işlemci, 8GB RAM ve NVIDIA GeForce GTX 1050 3072MiB GPU bulunmaktadır.

5.3 Deney Seti Yazılım Sistemi

Bu tez kullanılması planlanan kütüphanelerin bir çevre ortamı yaratılarak tüm donanım cihazlarında ortak bir şekilde kullanıma sunulmasını amaçlamaktadır. ESRGAN aracılığıyla çözünürlük iyileştirme, görüntüyü lokal bölgelere ayırma, her bir lokal bölgeyi etiketleme ve her hatalı bölgeyi işaretleyerek görüntü analizindeki kritik aşamalar gerçekleştirilmiştir. Tüm bunlara ek olarak, aydınlatma problemlerine de bir çözüm arayarak, görüntü işleme teknikleri kullanarak model başarımlarını artırılmaya çalışılmıştır.

5.3.1 Kütüphane kurulumu

Hem eğitim hem de test amacıyla stratejik olarak tasarlanmış üç farklı cihazda üç farklı deneysel ortam titizlikle oluşturulmuştur. Farklı ortamların seçimi, çeşitli

```
diğer kütüphane versiyonları gösterilmektedir.
```

```
BTU UNI
High Performance Computing Cluster - Powered by COMPACTA
```

Name	MaxTimeUnit	Nodes
short	2 hours	4 nodes
mid	1 day	4 nodes
long	7 days	4 nodes
longer	15 days	4 nodes

```
Job Scripts
-----
You can find example job scripts in: /cta/share/

Application List etc.
-----
RUN : nodeinfo
RUN : btu-info, btu-queue, btu-summary (soon!)

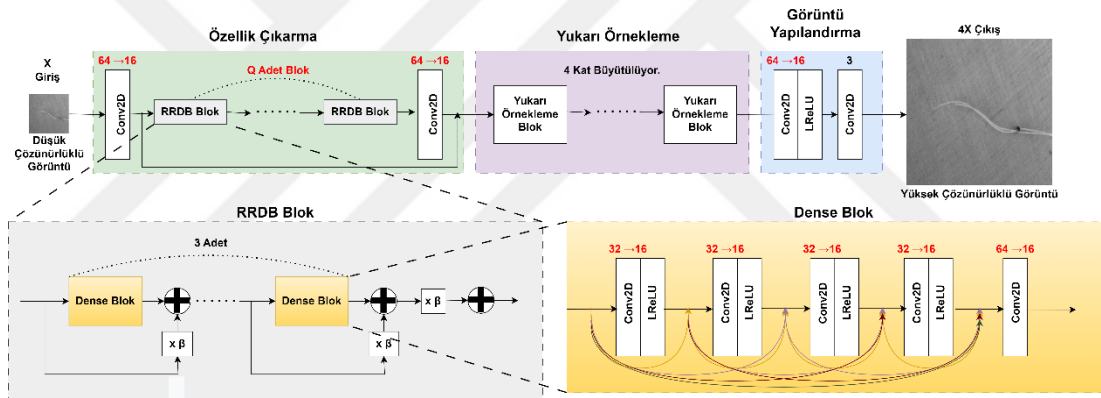
For General Help You Can:
-----
RUN : hpc-start-guide

Last login: Sun Dec 17 13:07:22 2023 from
(base) ametinfo@olovs $ source activate AmetInfo@olovs
(AmetInfo@olovs) ametinfo@logins:~$
```

Şekil 5.13'te linux işletim sistemine sahip yüksek başarılı bilgisayarlara ait giriş ekranı ve kütüphane versiyonları gösterilmektedir. Bunlara ek olarak çevreyi aktif ettiğimiz de gözlemlenmektedir.

ESRGAN modeli eğitim aşamasında, rastgele seçilen 416x416 boyutlu görüntülerin önce x4 kat küçültülerek 104x104 boyutuna getirilmesi, sonrasında küçültülmüş görüntülerden tekrar x4 kat büyütülmüş görüntülerin elde edilmesi işlemleri

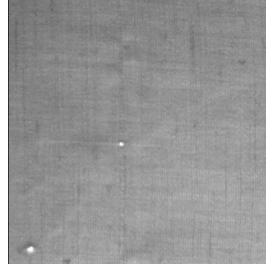
gerçekleştirilmiştir. Bu işlemde orijinal görüntü ile tahmin edilen yapay görüntü arasındaki farkın ayrıştırıcı model tarafından tahmin edilemeyecek hale gelmesine kadar sürmektedir. Bu süreç sonunda ayrıştırıcı model daha fazla öğrenemez ve üretici 416x416'lık kaliteli sahte bir görüntü oluşturur. Orijinal ESRGAN modelinden farklı olarak test işleminde değişiklik yapılarak, 416x416 girdi görüntüleri verilerek görüntüler 4 kat daha yükseltilmiş ve 1664x1664 piksel boyutuna getirilmiştir. ESRGAN modelini Raspberry Pi gibi mütevazı işleme yetenekleriyle karakterize edilen uç cihazlarıyla uyumlu hale getirme arayışında, üretici modeli içinde stratejik optimizasyonların uygulanması zorunlu hale gelmiştir. Modelin temel özelliklerinden ödün vermeden hesaplama talepleri kolaylaştırılmıştır. Bu yaklaşım, orijinal üretici modelinin ayrılmaz bir bileşeni olan Residual in Residual Dense Block (RRDB) modülü ve Dense Residual Block içindeki giriş kanallarının sayısında bir azalmayı içermektedir.



Şekil 5.14 : ESRGAN eğitim ve test işlemleri.

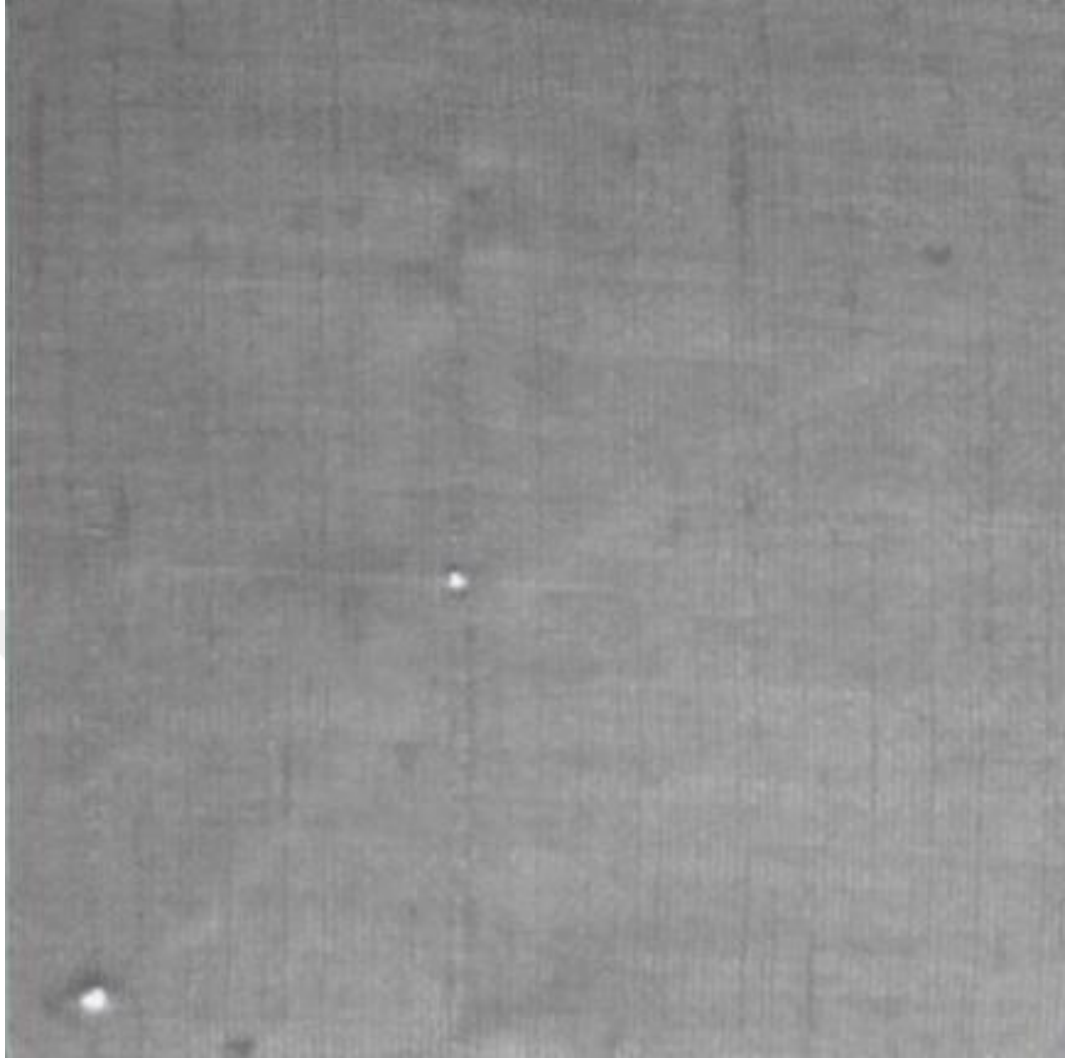
Şekil 5.14'te gösterilen kırmızı renkli yazılar değişimleri göstermektedir. Özellik çıkarma işleminde CNN'in kanal sayısı 64'ten 16'ya düşürülmüştür. Aynı işlem Dense Blok içerisinde de yapılmaktadır. Bu değişiklik, RRDB modülünün temel işlevlerini korurken hesaplama yükünü azaltmayı amaçlamaktadır. Ek olarak, RRDB bloklarının sayısında şekil 5.14'te gösterilen Q parametresinde önemli bir azalma gerçekleştirilerek sayı orijinal 23'ten hesaplama açısından daha tutumlu 1 ile 8 arasında blok değerlerine düşürülmüştür. Bu stratejik azaltma, hesaplama verimliliği ile model performansı arasında bir denge kurmayı amaçlamaktadır. Şekil 5.14'te gösterilen büyütme stratejisi, her pikseldeki ayrıntıların ayrıntı düzeyini önemli ölçüde artırarak görüntünün en küçük öğelerini bile ayırt edilebilir ve yorumlanabilir hale getirmektedir. Söz konusu görsel içeriğin daha incelikli ve gözle görülür bir

şekilde anlaşılmasını da kolaylaştırmaktadır. Piksel alanının kasıtlı olarak genişletilmesi aynı zamanda karmaşıklığın artmasına katkıda bulunmaktadır.



Şekil 5.15 : 416x416 piksel çözünürlüğe sahip orijinal görüntü.

Şekil 5.15'te orijinal boyutlarda TILDA delik hatası görüntü görünmektedir. Bu tezde veri seti, eğitim ve test amacıyla bölünmemiştir. ESRGAN modelinin geleneksel konfigürasyonuna yöntemsel bir yenilik getirilmiştir. Tipik olarak ESRGAN modelleri, daha sonra önceden belirlenmiş bir ölçeklendirme faktörü ile genişletilen bir başlangıç girdi boyutu belirlemektedir. Mevcut tezde model, 104x104 piksellik giriş boyutları kullanılarak eğitilmiş ve sonuçta 416x416 piksellik gelişmiş çözünürlük çıktısı elde edilmiştir. Görüntü üzerindeki örüntüler benzer olduğu düşünüldüğü için test aşamasında 416x416 piksel boyutunda girdiler kullanılmıştır. Üretici kısmında bu kasıtlı uyarılama, incelenen kumaş desenlerinin kendine özgü özelliklerinin titiz bir analizinden kaynaklanmıştır. Kumaş tasarımlarının nispeten basit doğası ve piksel çözünürlüğündeki görüntülerin mevcudiyeti göz önüne alındığında, test aşamasında girdi boyutlarının 416x416 olarak alınıp, çıktı olarak da görüntülerin 4 kat daha büyütülmesi amaçlanmıştır. Eğitim aşamasında orijinal görüntü ile üretilen görüntü arasındaki benzerlik fark edilemeyece kadar devam etmektedir. Tamamlanmasının ardından ESRGAN modeli, 1664x1664 piksellik artırılmış bir çıktı çözünürlüğüne ulaşmış ve kalite de önemli bir artışa işaret etmiştir. TILDA veri seti ile eğitilen ESRGAN modelinin eğitim ağırlıkları kaydedilmektedir. Kaydedilen bu ağırlıklar, modelin sıfırdan eğitilmesi yerine TILDA veri setinde öğrendiği özelliklerin üzerine, oluşturulan kumaşa ait belirli öznitelikleri öğrenmesi için tekrardan eğitilmektedir. Açık kumaş veri ve Koyu kumaş verisi ile de bu işlem tekrardan gerçekleştirilmiştir. Model TILDA veri setindeki örüntüleri öğrendiğinden dolayı, yeni veri setleri ile daha hızlı bir çıkarım da yapılabilir. Bu sayede daha az görsel veri setiyle de daha yüksek kaliteli görsellere ulaşılabilir.



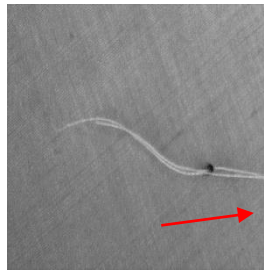
Şekil 5.16 : 1664x1664 piksel çözünürlüğe sahip yapay görüntü.

Şekil 5.16'te Şekil 5.15'teki orijinal görüntünün ESRGAN modeli büyütüldüğü görüntü verilmiştir. Bu genişletme, modelin karmaşık özellikleri tahmin etme ve oluşturma kapasitesinin altını çizerek, yalnızca güçlendirilmiş ayrıntı düzeyini yansıtmakla kalmayıp aynı zamanda yüksek çözünürlüklü görüntülerin doğasında bulunan incelikli özelliklerle de uyumlu bir çıktı üretmektedir. Bu ilerlemelerin doruk noktası, ESRGAN'ı, özellikle geleneksel enterpolasyon yöntemlerinin aksine, görüntü iyileştirme alanında önemli bir yaklaşım olarak konumlandırmaktadır (Wang ve diğ., 2018). Raspberry Pi gibi kaynakları kısıtlı cihazlarda depolama alanı kritik bir zorluk teşkil etmektedir. Her bir orijinal kumaş görüntüsü (416x416 piksel) yalnızca 18,7 kB yer kaplarken, üretken bir modelin kullanılması görüntüyü boyutunu 245 kB'ye çıkarır. Bu, görüntü başına depolama gereksinimlerinde 13 katlık önemli bir artışı temsil eder. Kümülatif olarak 448 görüntüden oluşan bir veri setine uygulandığında bu eşitsizlik 100 MB'lık önemli bir artış anlamına gelmektedir.

Ayrıca, işlemenin gerçek zamanlı doğası ve modern kameraların artan görüntü alma yetenekleri (30 fps'yi aşan) göz önüne alındığında, kümülatif depolama yükü ortaya çıkmaktadır.

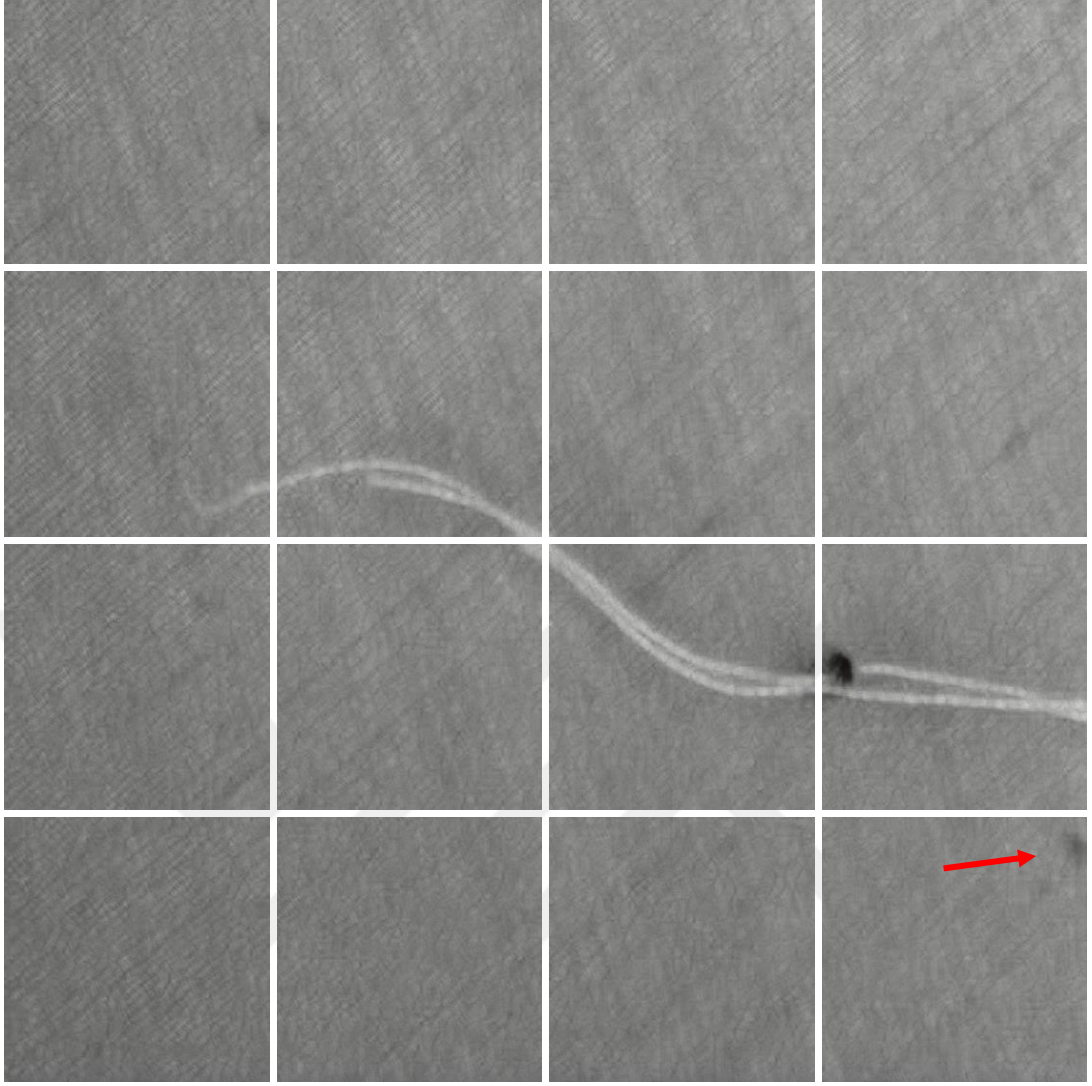
5.3.3 Görüntüyü lokal bölgelere ayırma

ESRGAN modeli tarafından oluşturulan geniş 1664x1664 görüntülerin sistematik olarak daha kolay yönetilebilir 416x416 segmentlere bölündüğü bir lokal bölgelere ayırma stratejisi kullanılmaktadır. Öncelikle veri kümesi dizini belirlenir. Belirtilen dizindeki her bir resim dosyası okunur. Bu işlem, `os.walk()` fonksiyonuyla gerçekleştirilir. Resimler, piksellerin boyutları lokal görüntü boyutuna tam bölünecek şekilde kırılır. Bu, daha küçük lokal boyutlarının çıkarılmasını sağlar ve bellek kullanımını azaltır. Kırılmış resimlerden lokal görüntüler çıkarılır. Her bir resim, belirtilen lokal boyutuyla çıkarılır ve bir dizi içinde toplanır. Bu dizi, ileride model eğitimi veya değerlendirmesi için kullanılmaktadır. Bu işlem sonunda 16 adet 416x416 görüntü elde edilmektedir. Bu bilinçli parçalama süreci, bu görüntülerin, eğitimden önce yeniden boyutlandırma aşamasına gerek kalmadan modele kusursuz şekilde entegre edilmesini kolaylaştırmaktadır. Aksi halde yüksek giriş görüntüsü düşük hesaplama işlemine sahip donanım cihazlarında ya çalışmayacak ya da gecikmeler yaşayacaktır. Bu metodolojik seçim yalnızca hesaplama verimliliğini artırmakla kalmayıp, aynı zamanda Raspberry Pi gibi mütevazı hesaplama yeteneklerine sahip işlemcilerin operasyonel kısıtlamalarına da uyum sağlamaktadır.



Şekil 5.17 : Veri setinden rastgele seçilmiş orijinal görsel. Kırmızı ok küçük hata

Şekil 5.17’de orijinal çözünürlüğünde örnek bir görsel gösterilmiştir. 416x416 boyutundaki görüntü de gözle görülmesi ilk etapta zor olabilen kırmızı ok ile işaretli yağ lekesi hatası görülmektedir. Bu görüntü Şekil 5.18’deki gibi ESRGAN ile büyütülüp lokallere ayrıldığında kırmızı ok ile gösterilen yağ lekesinin artık daha fazla gözle görülebilecek hale geldiği gözlemlenmiştir.

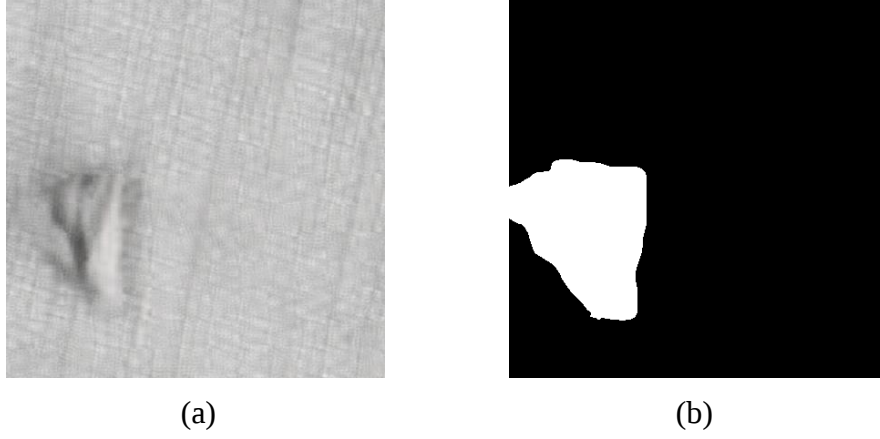


Şekil 5.18 : Orijinal görüntünün çözünürlüğü artırılarak ve lokal bölgelere ayrılarak elde edilmiş hali. (Kırmızı ok küçük hatanın büyütülmüş hali)

Bu çalışmanın önemli hedefi şekil 5.18’deki örnekteki gibi orijinal görüntüde göz ile görülmesi zor olabilecek küçüklükteki hataları büyütüp, daha görülebilir hale getirip, derin öğrenme modelinin hatayı tespit edilmesini sağlamaktır.

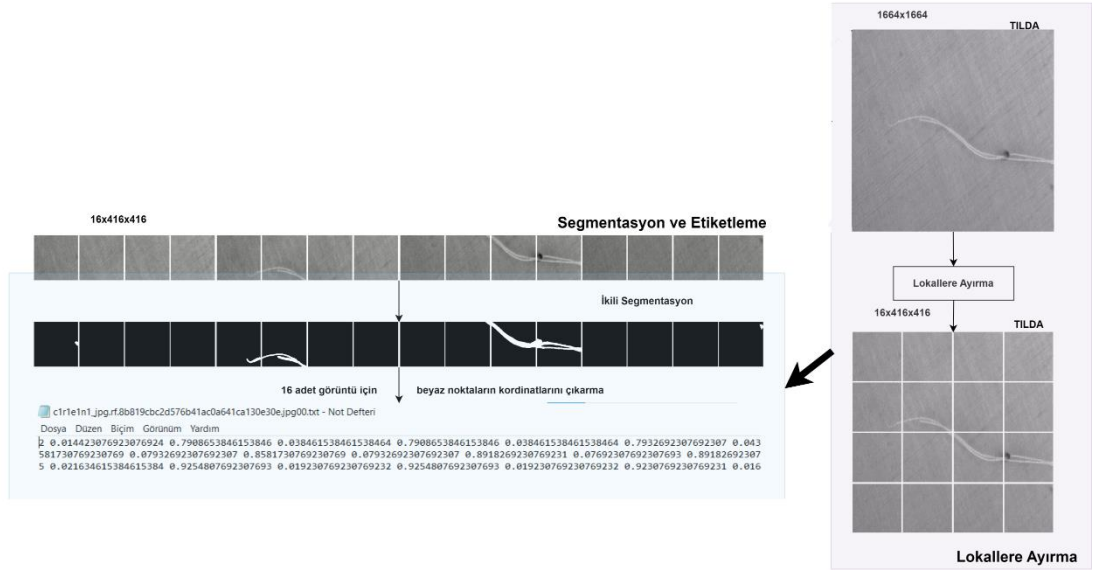
5.3.4 Etiketleme

Denetimli öğrenme alanında hem veri içeriğinin hem de istenen çıktıların derinlemesine anlaşılması esastır. Eğitim aşamasında modelin lokal bölgelere ayırma süreci, görüntülerin segmentasyonla ikili formata dönüştürülmesi ve hata tespit bölgelerinin belirgin şekilde sınırlandırılması gibi zorunlu bir adımla başlamaktadır.



Şekil 5.19 : (a) Orijinal görsel, (b) bölütlenmiş görsel.

Şekil 5.19.a’da veri setinden orijinal bir görsel gösterilmiştir. Şekil 5.19.b’de ise bu görselin üzerindeki kumaş hatasının bölütlenmiş hali yer almaktadır. İkili görüntü (binary), hataların tespit edildiği alanlar (beyazla gösterilmiştir) ile hatasız alanlar (siyahla gösterilmiştir) arasındaki kontrastı vurgulamaktadır. Her bir görüntü manuel bir şekilde siyah/beyaz olacak hale getirilmiştir. Bu işlem yapılırken paint ve fiji programlarından yararlanılmıştır. Segmentasyon için sonraki adımlar, hata sınıfının belirlenmesini ve hatanın bulunduğu etiketli bölgeyi kapsayan sınırlayıcı çokgenin koordinatlarının ayırt edilmesini içermektedir. Bu koordinatlar, etiketli verilerin YOLOv8’in segmentasyon algoritmasına entegrasyonunu kolaylaştırmada çok önemli bir rol oynamaktadır. Bunu başarmak için çalışma, giriş olarak bir görüntü yolunu ve etiket yolunu alır. Görüntü, NumPy dizisine dönüştürülür. İkili görüntüden etiketler oluşturulur, görüntünün sınırlayıcı kutusunu belirler ve bu alandaki nesnelerin koordinatlarını alır. Koordinatlar, görüntünün boyutlarına göre normalize edilir ve bir etiket satırı oluşturulur. Etiket dosyası, belirtilen yolu kullanarak oluşturulur ve etiket satırı yazılır. Ana döngü, belirtilen görüntü klasöründeki tüm dosyaları işler. Her görüntü için bir etiket dosyası oluşturulmaktadır. Bu metodik yaklaşım, bölümlere ayrılmış görüntüler içindeki hataların kapsamlı bir şekilde tanımlanmasını sağlayarak, hata sınıflarının ve bunların mekânsal dağılımının piksellerine kadar tanımlanmasına yol açmaktadır. Çıkarılan koordinatlardan yararlanan YOLOv8’in segmentasyon algoritmasının dahil edilmesi, veri kümesi içinde tespit edilen hataların tanımlanması ve karakterize edilmesinde modelin yorumlanabilirliğini ve hassasiyetini artırmaya yönelik genel hedefle uyumlu olmaktadır.

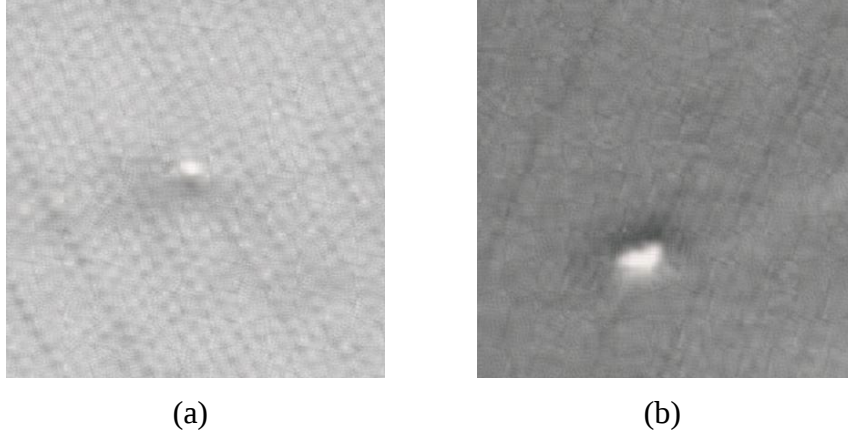


Şekil 5.20 : Lokal bölgelere ayrılmış görüntüdeki hataların koordinatlarını çıkarma.

Şekil 5.20’da lokal bölgelere ayrılmış görüntülerin her biri ikili segmentasyon işlemine tabi tutulmuştur. Hatalı beyaz bölgelerin koordinatları çıkarılmaktadır ve şekil 5.20’da görülüşü üzere format $\langle x1 \ y1 \ x2 \ y2 \ x3 \ y3... \rangle$ şeklindedir. Koordinatlar hatalı görüntünün boyutuna göre çoğalıp, azalmaktadır. İlk sayı, etiketin sınıf türünü temsil etmektedir. Diğer tüm sayılar, etiketli hatanın sınırlayıcı çokgeninin koordinatlarını temsil etmektedir.

5.3.5 Ön işleme

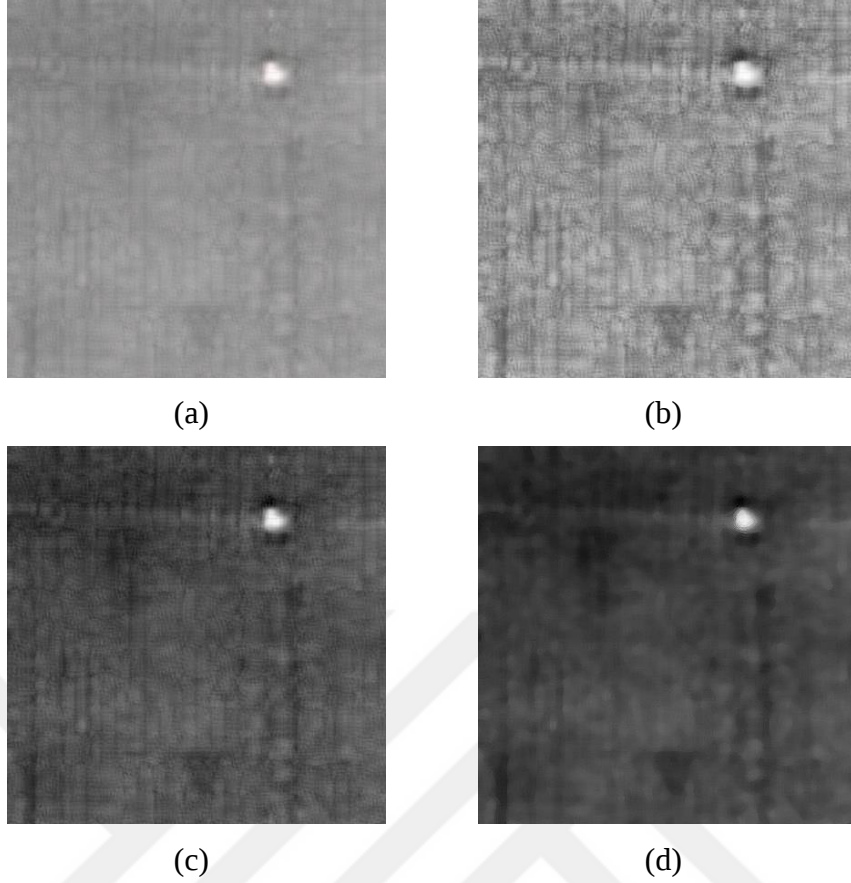
TILDA veri setindeki ipek kumaş görüntülerinin incelenmesi, açık ve koyu tonlarda iki farklı desenin varlığını ortaya koymuştur. Bu desenlerin kumaş renklerinden veya farklı aydınlatma koşullarında çekilmiş olmalarından kaynaklanması muhtemeldir. Renk ve aydınlatma çeşitliliğinin etkin tespiti engelleme potansiyeli göz önünde bulundurularak, bir dizi görüntü işleme yöntemi uygulanmıştır. Bu işlemler, kumaş renklerinin ve farklı aydınlatma senaryolarının model performansı üzerindeki etkisini azaltmayı amaçlamaktadır. Görüntü doğrusal kontrast germe, medyan filtreleme, keskinleştirme ve Kontrast Sınırlı Uyarlanabilir Histogram Eşitleme (CLAHE) bu işlemler arasında öne çıkanlardır. Farklı kumaş renk ve aydınlatma koşullarına sahip iki rastgele örnek Şekil 5.21’de sunulmuştur.



Şekil 5.21 : TILDA veri setinde, (a) açık renk kumaş ve aydınlatılmış delik hatası görseli, (b) koyu renk kumaş veya aydınlatma yapılmadan alınmış delik hatası görseli.

Şekil 5.21’de görüldüğü üzere veri setinde farklı renk veya ışıklandırmaya sahip kumaş görselleri yer almaktadır. Bir görüntünün genel kontrastını artırmaya yönelik bir yöntem olan görüntü doğrusal kontrast germe, farklı aydınlatma koşullarında görüntü dağılımlarını standartlaştırmak için kullanılmıştır. Medyan filtrenin uygulanması gürültüyü bastırarak görüntü özelliklerinin daha sağlam ve güvenilir bir şekilde temsil edilmesini sağlamaktadır. Kenarları vurgulamak ve özelliklerin belirginliğini artırmak için keskinleştirme teknikleri uygulanarak görsel netliğin iyileştirilmesine katkıda bulunulmuştur. Ek olarak, yerel kontrastı uyarlanabilir bir şekilde geliştirmek ve görüntünün farklı bölgelerindeki incelikli varyasyonları karşılamak için CLAHE özelliğinden yararlanılmıştır.

Bu görüntü işleme yöntemlerinin özenle uygulanmasına ve daha sonra modelin işlenmiş veri kümesiyle yeniden eğitilmesine rağmen, orijinal modelin öngörücü başarısını geçemeyen, kayda değer sonuçlar elde edilememiştir. Şaşırtıcı bir şekilde, bazı yöntemler segmentasyon sürecinde gelişmiş performans sergilerken, sınırlayıcı kutu tespit sürecindeki etkinliklerinin azalmasıyla ortaya çıkan istenmeyen bir sonuç ortaya çıkmıştır. Bu olgu, modelin performansının bir yönüne fayda sağlayan iyileştirmeler yanlışlıkla diğerini tehlikeye atabileceğinden, görüntü ön işlemede gerekli olan hassas dengenin altını çizmektedir. Bu yöntemlerin sonuçlarını içeren bir örnek Şekil 5.22’de verilmiştir.



Şekil 5.22 : (a) Orijinal görüntü, (b) CLAHE yöntemi sonucu, (c) görüntü doğrusal kontrast germe yöntemi sonucu ve (d) ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe yöntemlerinin ortak kullanımı.

Şekil 5.22.a’da veri setinden orijinal görsel görülmektedir. Şekil 5.22.b’de bu görselin CLAHE yönteminden çıkan sonucu yer almaktadır. Bu yöntem ile yerel kontrast düzenlenmiştir. Şekil 5.22.c’de görüntü doğrusal kontrast germe yöntemi ile genişletilmiştir. Şekil 5.22.d’de ise önce ortalama filtre yöntemi ile yumuşatılmış, sonra hata noktalarının belirgin olması için keskinleştirme uygulanmış ve en son görüntü doğrusal kontrast germe yöntemi ile genişletilmiştir.

5.3.6 YOLOv8 ile hata tespiti

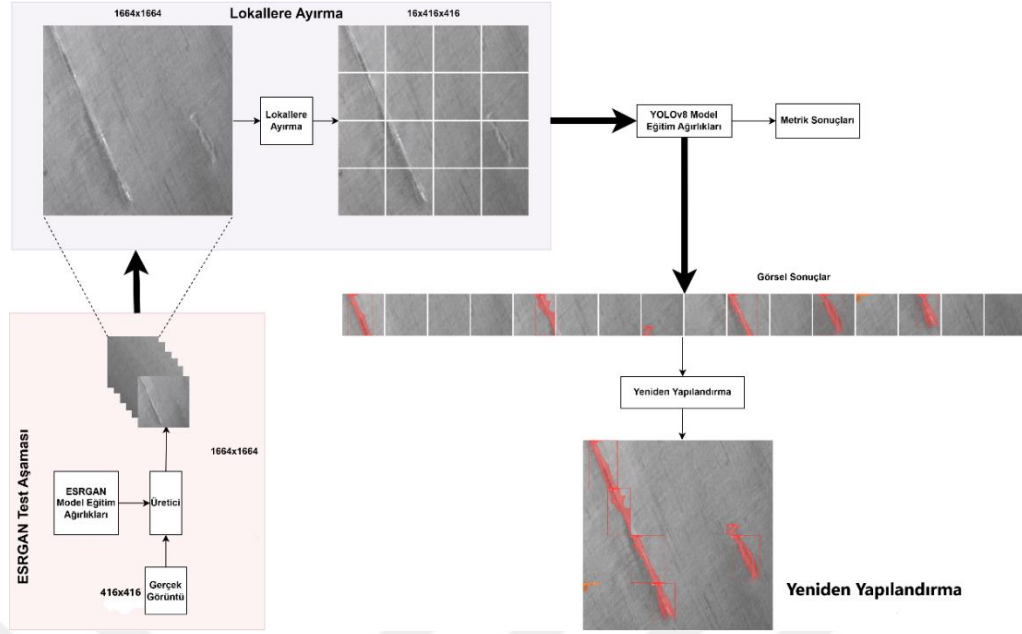
Her bir hata türü için YOLOv8 öncelikle bir sınırlayıcı kutu tahmini yapmaktadır. Bir kutu içerisinde yapılan bu tahmin kutu içerisindeki hatanın sınıflandırma başarımını göstermektedir. Yapılan bu sınırlayıcı kutu tahmininin ardından, hatalı kısmın bulunduğu yerlere de maskeleme işlemi gerçekleştirilmektedir. Örnek maskeleme işlemi piksel düzeyinde analiz gerektiren görevler için mükemmel seçimdir ve her bir hata türünü belirtmek için farklı bir renk kullanılmaktadır. Şekil 5.1’de görüldüğü üzere lokal bölgelere ayrılmış her görüntüdeki hatalı kısım için bir

adet etiket verisi elde edilmiştir. Hata bulunmayan görüntü bölütleri YOLOv8 eğitiminde kullanılmamaktadır. Lokal bölgelere ayrılmış her hatalı görüntü ve etiketleri YOLOv8 modeline eğitim aşamasında girdi olarak verilmektedir. Bu işlem eğitim aşamasında aynı zamanda görüntü arttırma metodu olarak da kullanılmaktadır. Hatalı kısımların büyük olması ve birden fazla hata olması durumunda lokallere ayrılan görüntü her bir hatayı farklı olarak analiz edecektir. Büyük olan hatalar ise lokallere ayrılarak parça parça işleme alınacaktır. Test aşamasında ise ESRGAN'dan üretilen çözünürlüğü yükseltilmiş görüntüler lokal bölgelere ayrılmaktadır. Her bir hatalı ve hatasız lokal bölge YOLOv8'e girdi olarak verilmektedir.

TILDA veri seti ile yapılan tüm bu işlemlerin model ağırlıkları kaydedilmektedir. Bu kayıt edilen ağırlıklar kullanılarak bu çalışma için oluşturulan veri setinde tekrardan eğitim işlemi gerçekleştirilmiştir. Gerçekleştirilen bu işlem sayesinde hem daha az veri seti ile hata tespiti yüksek başarımlara ulaşmakta hem de TILDA veri setindeki hataların ve örüntülerin benzer veri setlerinde tekrar öğrenilmesine gerek kalmadan model eğitim aşamasını hızlıca gerçekleştirmektedir.

5.3.7 Yeniden yapılandırma

Lokallere ayrılmış her görüntü YOLOv8 modeline girdi olarak verildikten sonra çıktı olarak sınırlayıcı kutu ve maskeleyme sonuçlarını görüntü üzerinde göstermektedir. Belirli bir sıra ile girdi olarak verilen lokallere ayrılmış görüntü parçaları, YOLOv8 çıktısı elde edildikten sonra aynı sıra ile tekrar birleştirilmektedir. Tüm bu işlemler bir kod vasıtasıyla yapılmaktadır. Önceden işlenmiş 1664x1664x3 görüntüler 16 adet 416x416x3 boyutunda yamalara ayrılır. Her yama, tahmin için bir YOLOv8 modelinden geçirilir. Sonuçlar kaydedilir ve görüntüler plt.imsave kullanılarak çizilir ve kaydedilir. Tüm yamaları işledikten sonra kod, adım boyutunun da yamalar arasında eşit olmasını sağlayarak orijinal görüntüyü yamalardan yeniden oluşturur. Bu yeniden yapılandırılmış görüntü görüntülenir ve kaydedilir.



Şekil 5.23 : YOLOv8 modelinden alınan görsel sonuçların yeniden yapılandırılmış hali.

Şekil 5.23’de tüm bu işlemler resmedilmiştir. Normal tespit işlemlerinde en köşe koordinatları alan sınırlayıcı kutu, bu metod ile birlikte lokal bölgelerdeki köşe noktalarını almaktadır. Bu işlemde sınırlayıcı kutu alanının gerektiğinden fazla olmasını önlemektedir.

5.3.8 Hiperparametre seçimi

ESRRGAN ve YOLOv8 gibi derin öğrenme modelleri, nesne algılama ve görüntü iyileştirme gibi karmaşık görevleri başarıyla gerçekleştirebilmek için belirli hiperparametrelerin dikkatlice ayarlanmasını gerektirir. Her iki modelin hiperparametre seçimleri, eğitim sürecinin başarısı üzerinde önemli bir etkiye sahiptir. Örneğin, ESRRGAN’da görüntü iyileştirme yeteneğini artırmak için yama boyutu ve öğrenme oranı gibi faktörler kritiktir. Benzer şekilde, YOLOv8’de ise giriş boyutu, eğitim süresi ve öğrenme oranı gibi parametreler, nesne tespitinin doğruluğunu ve algılama performansını belirlemede temel rol oynar. Bu hiperparametrelerin dikkatlice seçilmesi, modellerin istenilen görevleri başarıyla yerine getirmesini sağlamak için hayati öneme sahiptir. YOLO, hiperparametreleri optimize etmek için genetik algoritmalar kullanır. Genetik algoritmalar, doğal seleksiyon ve genetik mekanizmasından esinlenmiştir. Mutasyon, mevcut hiperparametrelerde küçük, rastgele değişiklikler uygulayarak ve değerlendirme için yeni adaylar üreterek hiperparametre uzayını yerel olarak aramaya yardımcı olur.

5.3.8.1 ESRGAN hiperparametre seçimi

ESRGAN'ın başarısı, eğitim sırasında ayarlanan hiperparametrelere büyük ölçüde bağlıdır. Bu parametreler, modelin karmaşıklığını, kapasitesini, öğrenme hızını ve bellek gereksinimlerini kontrol eder.

Çizelge 5.3 : ESRGAN modeli seçilen hiperparametre değerleri ve açıklamaları.

Hiperparametre	Değer	Açıklama
Eğitim Görüntü Boyutu	104x104	Eğitim sırasında kullanılan görüntü boyutu
Test Görüntü Boyutu	416x416	Test sırasında kullanılan görüntü boyutu
RRDB Sayısı (Q)	[23 - 1]	Üretici modeldeki RRDB blok sayısı
RRDB'lerdeki Kanal Sayısı	16	RRDB'lerdeki kanal sayısı
Öğrenme Oranı	1,00E-04	Modelin parametrelerini güncelleme hızı
Batch Boyutu	32	Her eğitim adımında işlenen görüntü sayısı
Epok Sayısı	1000	Tüm veri setinin sinir ağından geçirilme sayısı
Gradyan Cezası Ağırlığı	10	GP teriminin ağırlığı

Çizelge 5.3'e bakıldığında ESRGAN modelinin eğitim ve test aşamasında iki farklı giriş görüntüsü alındığı görülmektedir. Eğitim aşamasında 104x104, test kapsamında ise 416x416 olarak seçilmektedir. Daha yüksek bir çözünürlük, daha ayrıntılı görüntüler üretmeye yardımcı olabilir, ancak modelin eğitimini ve çalışmasını da zorlaştırabilir. Üretici modeldeki RRDB sayısı, modelin karmaşıklığını ve kapasitesini doğrudan etkiler. Daha fazla RRDB, daha fazla detay sunar, ancak daha fazla hesaplama ve bellek gerektirir. Bu tez kapsamında Q sayısı olarak geçen RRDB blok sayısı 23 ile 1 arasında seçim yapılmaktadır. RRDB'lerdeki Kanal Sayısı ise özellik çıkarımı ve temsilini etkiler, tez kapsamında orijinal modelde değişim gösterse de üretici boyutunda azaltmaya gidilmesi için 16'ya sabitlenmiştir. Daha yüksek kanal sayısı performansı artırabilir, ancak bellek kullanımını da artırmaktadır. Öğrenme Oranı, modelin eğitim sırasında parametrelerini ne kadar hızlı güncellediğini belirlemektedir. Çok yüksek olması kararsızlığa, çok düşük olması ise yakınsamayı yavaşlatabilmektedir. Tez kapsamında öğrenme oranı 0.0001 olarak seçilmektedir. Batch boyutu ise her eğitim adımında kaç görüntünün işleneceğini belirlemektedir. Daha büyük gruplar daha hızlı eğitim sunar, ancak daha fazla bellek gerektirmektedir. HPC'ler de 32 olarak tercih edilmekteyken, diğer cihazlarda 1 olarak seçilmektedir. Epok sayısı, tüm veri setinin sinir ağından kaç kez geçirildiğini belirlemektedir. Tez kapsamında epok sayısı 1000 olarak seçilmektedir. Gradyan cezası (GP) teriminin ağırlığını kontrol eder ve ESRGAN'ın sahte görüntüler üretmesini engellemek için kullanılan bir tekniktir. Daha yüksek bir değer, GP

teriminin daha fazla etkiye sahip olacağını ve sahte görüntü üretimini daha fazla cezalandıracağını göstermektedir. Bu, daha gerçekçi görüntüler üretmeye yardımcı olabilir, ancak modelin eğitimini de zorlaştırabilmektedir. Hiperparametrelerin doğru ayarlanması, ESRGAN'ın istenen görüntü kalitesini ve performansı elde etmesi için kritik önem taşımaktadır. Deneyler ve optimizasyon yoluyla, belirli bir veri seti ve görev için en uygun ayarlar bulunabilmektedir.

5.3.8.2 YOLOv8 hiperparametre seçimi

YOLOv8'nin hiperparametreleri, modelin nesne algılama yeteneklerini eğitmek ve değerlendirmek için kritik öneme sahiptir. Bu parametrelerin çeşitli yönlerini ele almak için detaylı bir inceleme yapıldı.

Çizelge 5.4 : YOLOv8 modeli seçilen hiperparametre değerleri ve açıklamaları.

Hiperparametre	Tanım	Değer	Anlamı
epochs	Eğitim döngüleri	500	Veri setinin üzerinden kaç kez geçileceği
patience	Erken durma sabrı	50	Val kaybı azalmadan önce durma sayısı
batch	Toplu boyutu	32	Her eğitim adımında işlenen görüntü sayısı
imgsz	Görüntü boyutu	416	Giriş görüntülerinin çözünürlüğü (416x416 piksel)
pretrained	Ön eğitilmiş	DOĞRU	Ön eğitilmiş ağırlıklar ile başla
optimizer	Optimizasyon	auto	Otomatik olarak optimizasyon algoritması seçimi
close_mosaic	Mozaik yakınlık	10	Mozaik veri büyütme için minimum görüntü sayısı
amp	Karmaşık Karışık Hassasiyet	DOĞRU	Potansiyel olarak daha hızlı eğitim için
dropout	Dropout oranı	0.0	Dropout oranı
val	Doğrulama kümesi	DOĞRU	Doğrulama kümesi kullanımı
conf	Güven skoru	None	Minimum güven skoru (varsayılan: 0.25)
iou	IoU eşik değeri	0.7	Kutu birleştirme için IoU eşik değeri
max_det	Maksimum tespit	300	Görüntü başına maksimum tespit sayısı
augment	Veri büyütme	YANLIŞ	Veri büyütme teknikleri kullanımı
nms	NMS	YANLIŞ	NMS (birbirini dışlama) kullanımı
lr0	Başlangıç öğrenme hızı	0.01	Öğrenme oranının başlangıç değeri
lrf	Son öğrenme hızı	0.01	Öğrenme oranının son değeri
momentum	Momentum	0,937	Optimizasyon için momentum değeri
weight_decay	Ağırlık çürütmesi	0,0005	Optimizasyon için ağırlık çürütmesi değeri
warmup_epochs	Isınma çağları	3.0	Öğrenme oranının kademeli artışı için epoch sayısı
warmup_momentum	Isınma momentumu	0.8	Isınma aşamasında momentum değeri
warmup_bias_lr	Isınma önyargı öğrenme hızı	0.1	Isınma aşamasında önyargı öğrenme hızı

Çizelge 5.4'e bakıldığında öncelikle, "epochs" parametresi, eğitim döngüsünün kaç kez tamamlandığını belirler ve "patience" parametresi, eğitim kaybının düzleştiği durumlarda öğrenme oranının ayarlanması için beklenen süreyi belirlemektedir. "Batch" boyutu, her bir eğitim adımında işlenen örneklerin sayısını kontrol eder ve "imgsz", hem eğitim hem de çıkarım için kullanılan giriş görüntülerinin boyutlarını belirlemektedir. "Pretrained" seçeneği, modelin önceden eğitilmiş ağırlıklarla başlatılıp başlatılmayacağını kontrol etmektedir, "optimizer" ise model parametrelerini güncellemek için kullanılan optimizasyon algoritmasını belirlemektedir. Ayrıca, öğrenme oranı, momentum, ağırlık azalma ve çeşitli veri artırma teknikleri gibi çeşitli diğer hiperparametreler, eğitim sürecinin ve modelin performansının yönetilmesinde kritik öneme sahip olmaktadır. Bu parametrelerin dikkatlice ayarlanması, optimal algılama sonuçları elde etmek için gereklidir. Örneğin, "lr0" ve "lrf" parametreleri, öğrenme hızını kontrol ederken, "momentum" ve "weight_decay" gibi diğer parametreler modelin genel stabilitesini ve performansını etkilemektedir. Bu hiperparametrelerin dikkatlice ayarlanması, modelin doğruluğunu, kararlılığını ve genelleme yeteneğini artırabilmektedir.

6. BULGULAR VE TARTIŞMA

Bu bölümde, öne çıkan iki modelin, hata tespiti kısmında uygulanan YOLOv8 ve çözünürlük arttırmak kısmında uygulanan ESRGAN modelinin sonuçlarını ve tartışmalarını kapsamlı bir analiz ile derinlemesine incelenmektedir. Değerlendirme, çeşitli ölçümleri ve hata türlerini kapsamakta ve bu modellerin kendi alanlarındaki etkinliği ve sınırlamaları hakkında fikir vermektedir.

6.1 Değerlendirme Metrikleri

Bilgisayarlı görme ve görüntü işleme alanında, gelişmiş algoritmaların etkinliğini ölçmek için model performansının ve görüntü kalitesinin değerlendirilmesi zorunludur. Bu tartışma iki kritik husus etrafında yoğunlaşıyor: Hızı ve doğruluğu ile tanınan önde gelen bir nesne ve hata algılama modeli olan YOLOv8'de kullanılan başarı ölçüm ölçümleri ve ESRGAN gibi modellerin aslına uygunluk ve yeniden yapılandırma yeteneklerinin değerlendirilmesinde öne çıkan algoritmalar bulunmaktadır.

6.1.1 YOLOv8 başarımlı metrikleri

YOLOv8'in performansının değerlendirilmesi, kesinlikten ve duyarlılıktan, mAP50 ve mAP50-95'e kadar uzanan sağlam bir ölçüm kümesine dayanıyor ve modelin hata algılama ve yerleştirme konusundaki becerisinin titizlikle değerlendirilebileceği kapsamlı bir mercek sunuyor.

6.1.1.1 Kesinlik (Precision)

Kesinlik (P), model tarafından yapılan olumlu tahminlerin doğruluğunu ölçer.

$$P = \frac{\text{Gerçek Pozitif}}{\text{Gerçek Pozitif} + \text{Yanlış Pozitif}} \quad (6.1)$$

Denklem 6.1'de gerçek pozitif tahminlerin, gerçek pozitif ve yanlış pozitif tahminlerin toplamına oranı olarak hesaplanır.

6.1.1.2 Duyarlılık (Recall)

Duyarlılık (R), modelin tüm olumlu örnekleri yakalama yeteneğini ölçer.

$$R = \frac{\text{Gerçek Pozitif}}{\text{Gerçek Pozitif} + \text{Yanlış Negatif}} \quad (6.2)$$

Denklem 6.2’de doğru pozitif tahminlerin, doğru pozitif ve yanlış negatif tahminlerin toplamına oranı olarak hesaplanır.

6.1.1.3 Birleşim üzerinde kesişme (Intersection over union)

Birleşim Üzerinden Kesişme (IoU) anlamına gelir ve özellikle sınırlayıcı kutular bağlamında hata algılama algoritmalarının doğruluğunu değerlendirmek için yaygın olarak kullanılan bir ölçüm olmaktadır. Genellikle kesinlik ve duyarlılık ölçümleriyle birlikte kullanılmaktadır. IoU, tahmin edilen sınırlayıcı kutu ile temel gerçek sınırlayıcı kutu arasındaki örtüşmeyi ölçmektedir. İki sınırlayıcı kutu arasındaki örtüşme alanının bunların birleşim alanına oranı olarak hesaplanmaktadır.

$$IoU = \frac{\text{Örtüşme Alanı}}{\text{Birleşme Alanı}} \quad (6.3)$$

Denklem 6.3’te örtüşme alanı, tahmin edilen sınırlayıcı kutu ile temel gerçek sınırlayıcı kutunun kesiştiği bölgedir. Birleşme alanı ise hem öngörülen sınırlayıcı kutunun hem de temel gerçek sınırlayıcı kutunun kapsadığı birleşik bölgedir. IoU 0 ile 1 arasında değişir; burada 0, örtüşme olmadığını ve 1, mükemmel örtüşmeyi belirtmektedir. Tahmin edilen ve temel doğruluk sınırlama kutuları arasındaki IoU önceden tanımlanmış bir eşiği aşarsa tahminin doğru olduğu kabul edilmektedir. Birden fazla hata sınıfının mevcut olduğu senaryolarda ortalama IoU, tüm sınıflardaki IoU değerlerinin ortalaması alınarak hesaplanmaktadır.

6.1.1.4 mAP50 (%50 IoU'da ortalama kesinlik)

mAP50, modelin kesinliğini farklı güven eşikleri boyunca değerlendirir ve hata algılama modellerinin genel performansını değerlendirmek için yaygın olarak kullanılmaktadır.

6.1.1.5 mAP50-95 (%50 ila %95 IoU arası ortalama kesinlik)

mAP50-95 (%50 ila %95 IoU arası Ortalama Kesinlik) %50 ila %95 arasındaki IoU eşik aralığını dikkate alarak modelin kesinliğine ilişkin daha geniş bir görünüm sağlamaktadır.

6.1.2 K-Katlamalı çapraz doğrulama (K-Fold cross validation)

K-Katlamalı Çapraz Doğrulama, tahmine dayalı modellerin performansını ve sağlamlığını değerlendirmek için makine öğrenimi ve istatistiksel modellemede yaygın olarak kullanılan bir teknik olmaktadır. Belirli bir veri alt kümesine aşırı uyum riskini azaltarak model performansına ilişkin daha sağlam bir tahmin sağlar. Tekrarlanan eğitim ve doğrulama süreci nedeniyle, özellikle büyük veri kümeleri veya karmaşık modeller için hesaplama gereksinimleri arttırmaktadır. Orijinal veri kümesini eşit (veya neredeyse eşit) boyutlu alt kümelere veya katlara bölerek çalışır. Model daha sonra 'K' kez eğitilir ve değerlendirilir; her yinelemede 'K' katlamalarından biri doğrulama seti olarak ayrılırken geri kalan 'K-1' katlamalar eğitim setini oluşturmak üzere birleştirilir. Tüm 'K' yinelemelerinin tamamlanmasının ardından, genellikle ortalama hesaplanarak, modelin genelleştirme kabiliyetinin göstergesi olan tek bir kapsamlı performans ölçümü elde etmek üzere bir araya getirilmektedir.

6.1.3 ESRGAN başarımlı metrikleri

Öğrenilmiş Algısal Görüntü Yaması Benzerliği (LPIPS), Görsel Bilgi Doğruluğu (VIF), Tepe Sinyal-Gürültü Oranı (PSNR) ve Yapısal Benzerlik İndeksi (SSIM) görüntü kalitesini çeşitli açılardan değerlendirmek için kullanılan metriklerdir. LPIPS algısal farklılıkları değerlendirmek için derin sinir ağlarından yararlanırken, VIF dalgacık alanı analizini kullanarak görsel bilgi doğruluğuna odaklanır. PSNR, bir sinyalin mümkün olan maksimum gücünün hatasına oranını ölçer ve SSIM parlaklık, kontrast ve yapıya dayalı yapısal benzerlikleri ölçer. Her bir metrik, insan algısının farklı yönlerine hitap ederek ve görüntü işleme ve bilgisayarla görme alanlarında değerli araçlar sağlayarak görüntü kalitesi değerlendirmesine ilişkin benzersiz içgörüler sunar.

6.1.3.1 Tepe sinyal gürültü oranı

Tepe Sinyal Gürültü Oranı (PSNR), yeniden yapılandırılmış veya sıkıştırılmış bir sinyalin kalitesini orijinal, sıkıştırılmamış versiyonuna kıyasla ölçmek için yaygın olarak kullanılan bir metriktir. PSNR'ın hesaplanması için referans olarak bir giriş görüntüsü alınır ve giriş görüntüsündeki gürültü seviyesini ölçmek için bir çıkış görüntüsüyle giriş görüntüsü karşılaştırılır. PSNR, desibel (dB) cinsinden ifade edilir ve aşağıdaki formül kullanılarak hesaplanır:

$$PSNR = 20 \times \log_{10} \left(\frac{255}{\sqrt{MSE(I_{giris}, I_{cikis})}} \right) \quad (6.4)$$

255 görüntünün mümkün olan maksimum piksel değeridir. MSE, original (I_{giris}) ve yeniden oluşturulmuş (I_{cikis}) görüntüler arasındaki piksel bazında karesel farkların ortalaması olan Ortalama Kare Hatasıdır. PSNR ve MSE arasında ters orantılı bir ilişki bulunmaktadır. Daha yüksek bir PSNR değeri, süper çözünürlüklü görüntünün orijinal yüksek çözünürlüklü görüntüye kalite açısından daha yakın olduğunu gösterir. Algısal önemlerine bakılmaksızın tüm piksellerin eşit derece önemli olduğunu varsaydığı için her zaman insan algısıyla uyumlu olmayabilir.

6.1.3.2 Yapı benzerlik indeksi yöntemi

Yapı Benzerlik İndeksi Yöntemi (SSIM)'de iki görüntü arasındaki benzerliğin, PSNR gibi nicel bir ölçüsünü sağlar. Ancak SSIM, yalnızca piksel bazında farklılıkları değil, aynı zamanda yapısal bilgileri ve görüntü kalitesinin algısal yönlerini de dikkate aldığı için yaklaşımda farklılık gösterir. İnsanların görüntüleri algılama biçimindeki benzerliği değerlendirmeyi amaçlayarak parlaklık, kontrast ve yapısını dikkate alır. Bu metrik, bu üç özelliğin çarpımının kombinasyonudur. Bu üç özellik aşağıdaki denklemde ifade edilmiştir.

$$SSIM(x, y) = l(x, y) \cdot c(x, y) \cdot s(x, y) \quad (6.5)$$

x ve y karşılaştırılan iki giriş görüntüsüdür. $l(x, y)$ iki görüntü arasındaki parlaklığı karşılaştırmak için kullanılır. $c(x, y)$ iki görüntünün en parlak ve en karanlık bölgesi arasındaki aralıkları ayırt etmek için kullanılır. $s(x, y)$ görüntülerin benzerliğini ve farklılığını bulmak için iki görüntü arasındaki yerel parlaklık modelini karşılaştırmak için kullanılır. Bireysel bileşenler aşağıdaki şekilde tanımlanır.

$$\text{Parlaklık Bileşeni} = l(x, y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \quad (6.6)$$

μ_x ve μ_y sırasıyla x ve y görüntülerinin ortalama parlaklık değerleridir. $\mu_x^2 + \mu_y^2$ sifıra yaklaşırken kararsızlığı önlemek için küçük bir sabit olan c_1 eklenir.

$$\text{Kontrast Bileşeni} = c(x, y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \quad (6.7)$$

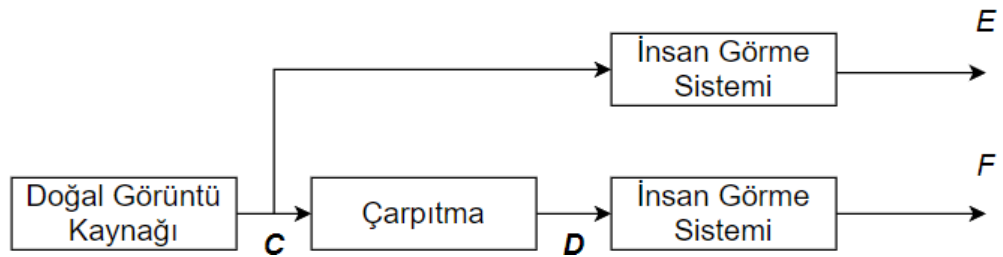
σ_x ve σ_y sırasıyla x ve y görüntülerindeki piksel yoğunluklarının standart sapmalarıdır. c_2 başka bir küçük sabittir.

$$\text{Yapısal Bileşen} = s(x, y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3} \quad (6.8)$$

σ_{xy} , x ve y'nin kovaryansıdır. c_3 küçük bir sabittir. SSIM değerleri -1 ile 1 arasında değişir, burada 1 mükemmel bir eşleşmeyi gösterir.

6.1.3.3 Görsel bilgi doğruluğu

Görsel Bilgi Doğruluğu (VIF), özellikle bozuk bir görüntünün görsel ayrıntıları bir referans görüntüye kıyasla ne kadar iyi koruduğunun değerlendirilmesinde, görüntü kalitesi değerlendirmesinde kullanılan gelişmiş bir ölçümdür. VIF, İnsan Görsel Sisteminin (HVS) özellikleriyle uyumlu olacak şekilde tasarlanmıştır ve algılanan görüntü kalitesinin daha doğru bir temsilini sağlamayı amaçlamaktadır. Dalgacık alanında çalışan VIF, çoklu çözünürlüklü analiz için yönlendirilebilir piramit veya Gauss ölçeği karışımları gibi teknikler kullanır. Yönlendirilebilir piramit, çoklu çözünürlük analizini kolaylaştıran çoklu ölçek ve yönlerde bir dizi filtredir. Dalgacık alanında çalışmanın amacı, uzamsal alana kıyasla katsayıların fazlalığını azaltmaktır. Fazlalığın azaltılması, dönüştürülen katsayılar daha az korelasyonlu olduğundan, karşılıklı bilginin daha doğru bir şekilde hesaplanmasına yardımcı olur. Algoritma, dalgacık alanında referans ve bozuk görüntü katsayıları arasındaki ilişkiyi yakalayan görsel bozulma modelini tahmin etmek için kovaryans matrisini hesaplar.



Şekil 6.1 : Görsel bilgi doğruluğunun çalışma mekanizması (Sheikh ve diğ., 2005).

C ve E arasındaki karşılıklı bilgi, beynin referans sinyalden ideal olarak çıkarabileceği bilgiyi ölçerken, C ve F arasındaki karşılıklı bilgi, bozulmuş sinyalden çıkarılabilecek ilgili bilgiyi ölçer. Sübjektif görüntü kalitesini tahmin etme konusunda dikkate değer olan VIF, görüntülerdeki iyileştirmeleri tespit edebilir ve kontrast duyarlılığı ve parlaklık dönüşümü gibi belirli algısal yönleri kasıtlı olarak göz ardı ederek görsel maskeleye modellerinden farklılık gösterir ve bu da onu görüntü kalitesi değerlendirme çalışmalarında güçlü bir araç haline getirir. VIF genel olarak 0 ile 1 arasında bir degree sahiptir ve diğer metriklere göre belirgin bir özelliği ise bozuk görüntünün görsel olarak referanstan üstün olduğu durumları da ele almaktadır.

6.1.3.4 Öğrenilmiş algısal görüntü yaması benzerliği

Öğrenilmiş Algısal Görüntü Yaması Benzerliği (LPIPS), iki görüntü arasındaki algısal benzerliği değerlendirmek için kullanılan bir metriktir. Süper çözünürlüklü görüntülerin değerlendirilmesi gibi görüntülerin algısal benzerliğin anlaşılmasının çok önemli olduğu senaryolarda özellikle yararlıdır. PSNR veya SSIM gibi insan algısıyla iyi uyum sağlayamayan geleneksel metriklerin ötesine geçmek için tasarlanmıştır. LPIPS, görüntü yamaları arasındaki algısal farklılıkları öğrenmek ve ölçmek için derin bir sinir ağı, özellikle de bir siyam ağı mimarisi kullanır. VGG, SqueezeNet veya AlexNet gibi bir görüntü sınıflandırma ağlarında, öğrenme sürecini yönlendirmek için insan algısal yargılarını içeren büyük veri kümeleri üzerinde eğitilir. Değerlendirme sırasında LPIPS, referans ve bozuk görüntülerden görüntü yamaları çiftleri alır, bunları eğitilmiş ağdan geçirir. Orijinal ve tahmin edilen görüntüden özellikler çıkarıldıktan sonra iki görüntü yamasının aktivasyonları arasındaki benzerliğin, öklid mesafesi veya kosinüs benzerliği kullanılarak karşılaştırılır. LPIPS önce tüm piksel ve katmanlardaki özellik boyutunu birim uzunluğa normalleştirir, her özelliği özelliğe özgü bir ağırlıkla ölçeklendirir ve bu ağırlıklı aktivasyonlar arasındaki L2 mesafesinin karesini değerlendirir. Bu karesel mesafelerin daha sonra görüntü boyutları üzerinde ortalaması alınır ve katmanlar üzerinde toplanır. Tüm bu işlemler aşağıdaki denklemde gösterilmektedir.

$$d_{\text{LPIPS}}(x, y) = \sum_l \frac{1}{H_l W_l} \sum_{i,j} \|w_l \odot (\hat{x}_{ij}^l - \hat{y}_{ij}^l)\|_2^2 \quad (6.9)$$

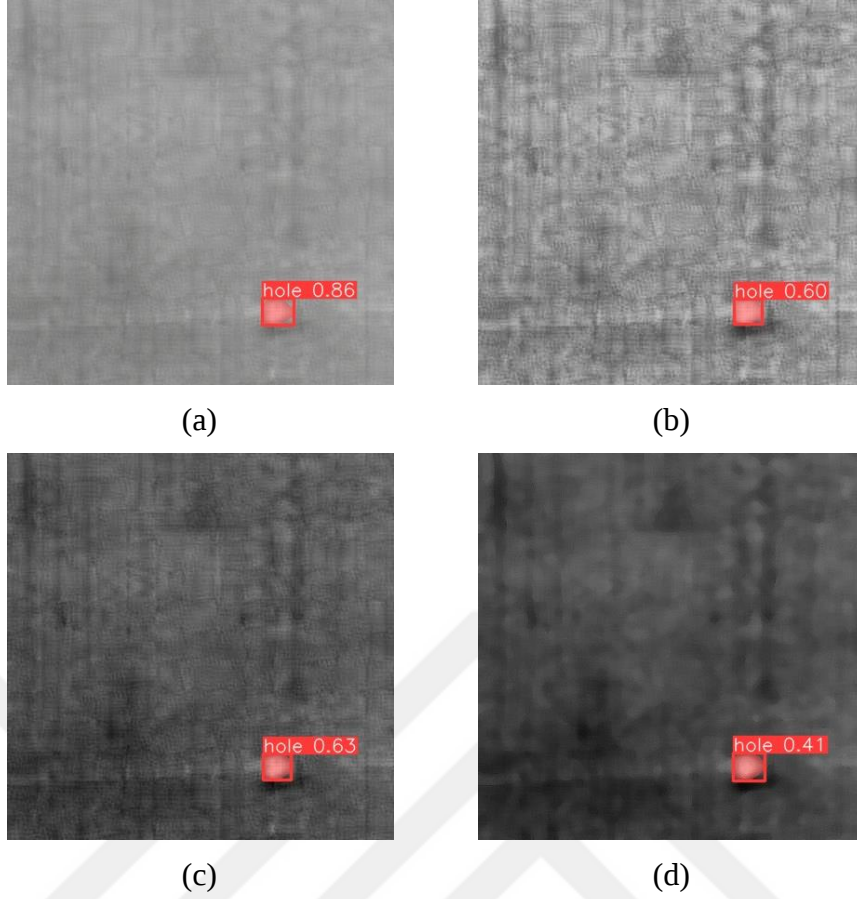
Burada $\hat{x}_{ij}^l$ ve $\hat{y}_{ij}^l$, katman l ve pikseldeki (i, j) normalleştirilmiş özellik vektörlerini gösterir, w_l katman l 'deki özelliklerin her biri için ağırlıkları içerir ve her pikseldeki özellik vektörlerini özellik ağırlıkları ile çarpar. Döndürülen değer, ağ mimarisi aracılığıyla geriye yayılır ve parametreler güncellenir, bunun üzerine tüm prosedür, yakınsamaya ulaşıldığı tahmin edilene kadar tekrarlanır ve eğitim durdurulur. Amaç, insanların görüntü kalitesindeki farklılıkları nasıl algıladığına dair daha güvenilir bir ölçü sağlamak ve bu sayede görüntü işleme ve bilgisayarla görme gibi algısal görüntü kalitesinin çok önemli olduğu uygulamalar için uygun hale getirmektir. Düşük LPIPS puanı, görüntü yamalarının algısal olarak benzer olduğu anlamına gelir.

6.2 Kumaş Hata Tespit Sonuçlarının Elde Edilmesi

Etiketlenmiş görüntüler sabit bir temel görevi görerek araştırmanın sonraki aşamaları için standartlaştırılmış bir veri seti oluşturmaktadır. Etiketli görüntüler tutarlı kalırken, görüntülerin kendilerinin farklı görüntü işleme yöntemlerinden geçtiğinin altını çizmek gereklidir. TILDA veri seti, görüntü doğrusal kontrast germe, medyan filtreleme, keskinleştirme ve CLAHE gibi işlemleri kapsayan kapsamlı bir görüntü işleme metotlarından geçmektedir. Veri seti üç farklı algoritmanın incelemesine tabi tutulmaktadır bunlar: çözünürlüklerini arttırma, yamalara ayırma ve tespittir. Farklı algoritmik çerçevelere ayrılan bu çatallanma, modelin görüntü analizinin farklı yönlerindeki performansının incelikli bir şekilde araştırılmasını sağlamaktadır. Bu kapsayıcı kategorilerin her birinde, farklı görüntü işleme yöntemlerinin modelin etkinliği üzerindeki etkisini inceleyen daha ileri bir kırılım gerçekleştirilmektedir. Bu katmanlı yaklaşım, görüntü ön işleme tekniklerinin yamalara ayırma ve tespit görevlerinin nihai sonuçları üzerindeki etkilerini ortaya çıkararak ayrıntılı bir değerlendirmeyi kolaylaştırmaktadır.

6.2.1 Kumaşta delik ve kesik hataları

Segmentasyon işlemi kapsamında, modelin kumaş görüntüsündeki delik hatalarının örneklerini doğru bir şekilde tanımlama ve izole etme yeterliliğinin titiz bir şekilde değerlendirilmesini içermektedir. Bu değerlendirmenin, işlenmemiş orijinal görüntünün yanı sıra üç farklı görüntü işleme yönteminin uygulanmasını özetleyen açıklayıcı bir örnek kullanılarak Şekil 6.1'de gösterilmektedir.

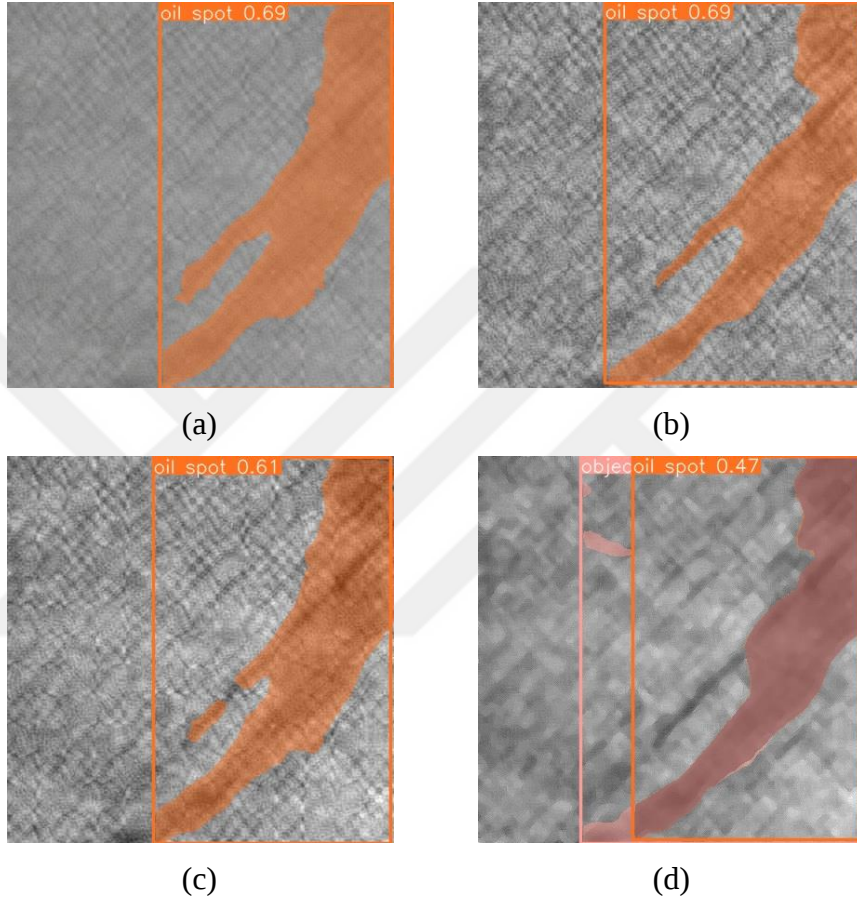


Şekil 6.2 : (a) Orijinal görüntüler ile (b) CLAHE ile elde edilen görüntüler ile (c) görüntü doğrusal kontrast germe ile elde edilen görüntüler ile ve (d) sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe ile elde edilen görüntüler ile eğitilen modellerin delik hatasına göre sonuçları.

Şekil 6.2.a’da orijinal görüntüler ile modelin eğitilmesi sonucu elde edilen sonuçlara delik hatası örneği üzerinden rastgele bir örnek görülmektedir. Bu örnekte tespit edilen delik hatasının sınıf doğruluk (Accuracy) olasılığı %86’dır.. Şekil 6.2.b’de CLAHE yönteminden geçirilerek elde edilen görüntüler ile eğitilen modelin sonucu görülmektedir. Aynı görsel için bu modelin çıktısı, delik hatasının sınıf doğruluk olasılığının %60 olduğunu göstermektedir. Şekil 6.2.c’de görüntü doğrusal kontrast germe yönteminden geçirilen görüntüler ile eğitilen modelin sonucu görülmektedir. Bu modelin aynı görsel için elde ettiği delik hatasının sınıf doğruluk olasılığı %63’tür.Son olarak şekil 6.2.d’de sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe yöntemlerinin hibrit kullanımından elde edilen görseller ile eğitilen model görülmektedir. Bu modelin sonucunda, aynı görsel için delik hatasının sınıf doğruluk olasılığı %41’dir.

6.2.2 Yağ lekeleri ve renk hataları

Segmentasyon işlemi kapsamında, modelin kumaş yüzeylerinde görülen yağ lekeleri ve renk hatalarının örneklerini doğru bir şekilde tanımlama ve izole etme yeterliliğinin titiz bir şekilde değerlendirilmesidir. Bu değerlendirmenin, işlenmemiş orijinal görüntünün yanı sıra üç farklı görüntü işleme metodolojisinin uygulanmasını özetleyen açıklayıcı bir örnek kullanılarak Şekil 6.3'te gösterilmektedir.



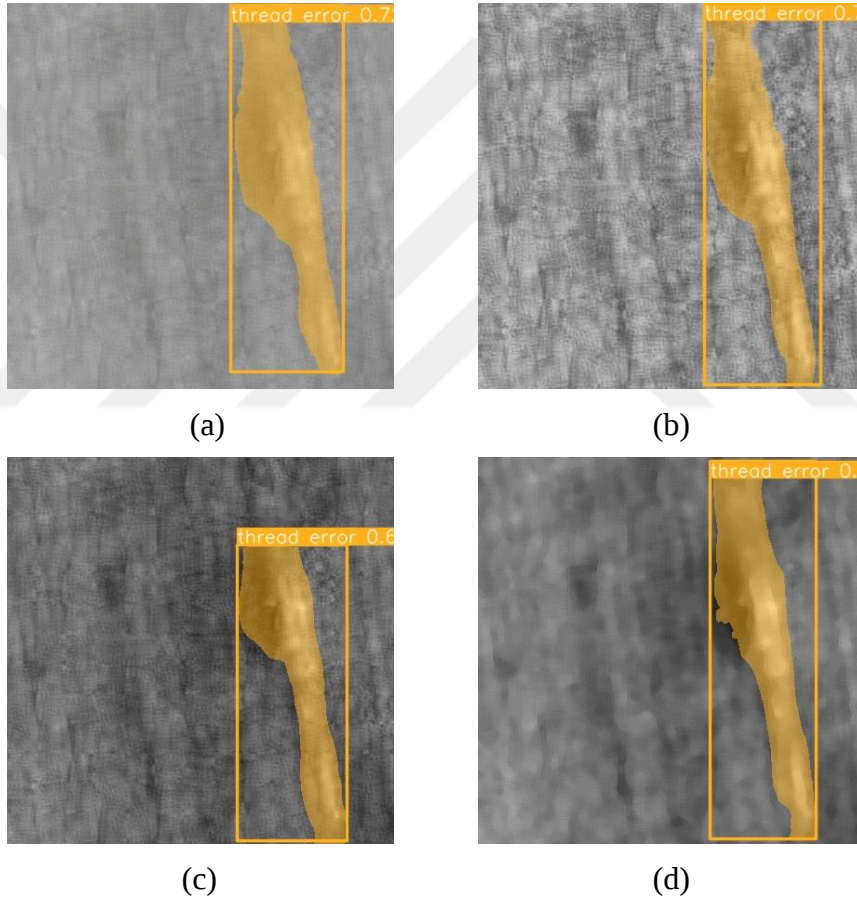
Şekil 6.3 : (a) Orijinal görüntüler ile (b) CLAHE ile elde edilen görüntüler ile (c) görüntü doğrusal kontrast germe ile elde edilen görüntüler ile ve (d) sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe ile elde edilen görüntüler ile eğitilen modellerin yağ lekesine karşı sonuçları.

Şekil 6.3.a'da orijinal görüntüler ile modelin eğitilmesi sonucu elde edilen sonuçlara yağ lekesi hatası üzerinden rastgele bir örnek görülmektedir. Bu örnekteki tespit edilen yağ lekesinin sınıf doğruluk olasılığı %69'dur. Şekil 6.3.b'de CLAHE yönteminden geçirilerek elde edilen görüntüler ile eğitilen modelin sonucu görülmektedir. Aynı görsel için bu modelden elde edilen sonuçta yağ lekesi sınıfı doğruluk olasılığı %69'dur. Şekil 6.3.c'de görüntü doğrusal kontrast germe yönteminden geçirilen görüntüler ile eğitilen modelin sonucu görülmektedir. Aynı görsel için bu modelden elde edilen sonuç, yağ lekesi sınıfı için %69'luk bir doğruluk

olasılığıyla gösterilmektedir. Son olarak Şekil 6.3.d’de ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe yöntemlerinin hibrit kullanımından elde edilen görseller ile eğitilen model görülmektedir. Aynı görsel için bu modelin sonucunda yağ lekesi sınıfı için %47’lik bir doğruluk olasılığı bulunmaktadır. Ek olarak şekil 6.3.d’de yabancı obje bularak, yanlış bir tahmin yapmaktadır.

6.2.3 İplik hatası

Segmentasyon işlemi için kumaş yüzeylerinde görülen iplik hatalarının test işlemi 3 farklı görüntü işleme yöntemi ve orijinal görüntü ile Şekil 6.4’te bir örnek ile sunulmuştur.



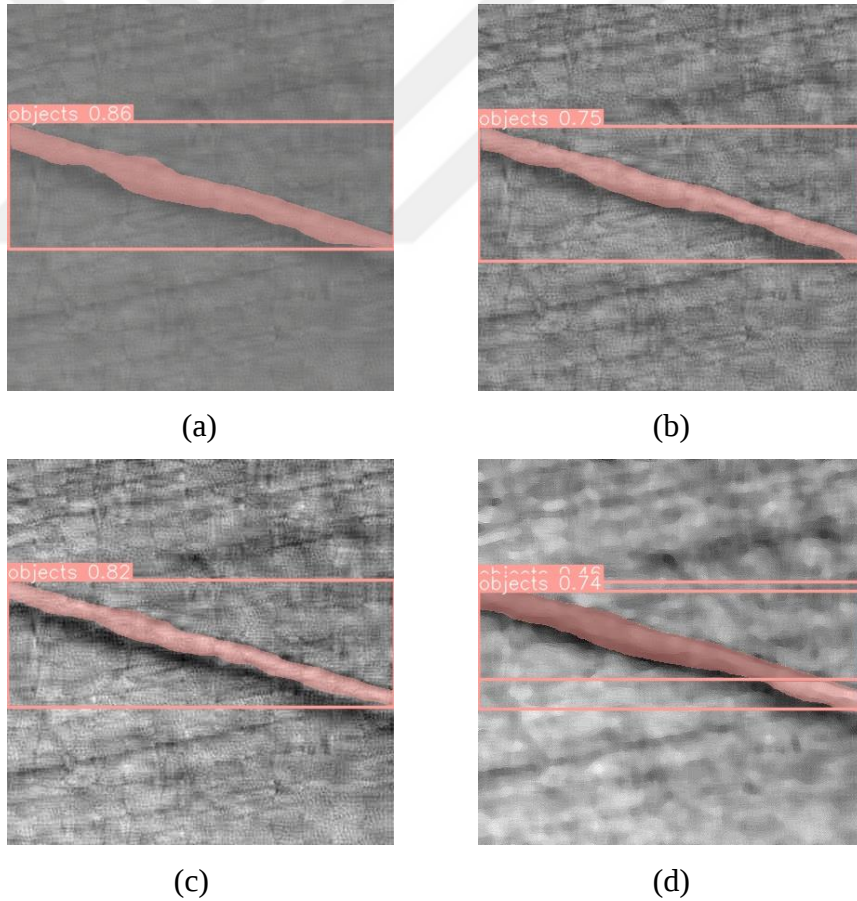
Şekil 6.4 : (a) Orijinal görüntüler ile (b) CLAHE ile elde edilen görüntüler ile (c) görüntü doğrusal kontrast germe ile elde edilen görüntüler ile ve (d) sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe ile elde edilen görüntüler ile eğitilen modellerin iplik hatasına karşı sonuçları.

Şekil 6.4.a’da orijinal görüntüler ile modelin eğitilmesi sonucu elde edilen sonuçlara iplik hatası üzerinden rastgele bir örnek görülmektedir. Bu örnekte tespit edilen iplik hatası sınıf doğruluk olasılığı %70’tir. Şekil 6.4.b’de CLAHE yönteminden geçirilerek elde edilen görüntüler ile eğitilen modelin sonucu görülmektedir. Bu

modelden elde edilen sonuca göre, aynı görsel için iplik hatası sınıf doğruluk olasılığı %70'tir. Şekil 6.4.c'de görüntü doğrusal kontrast germe yönteminden geçirilen görüntüler ile eğitilen modelin sonucu görülmektedir. Bu modelin aynı görsel için elde ettiği iplik hatası sınıfı doğruluk olasılığı %60'tır. Son olarak Şekil 6.4.d'de ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe yöntemlerinin hibrit kullanımından elde edilen görseller ile eğitilen model görülmektedir. Bu modelin sonucuna göre, aynı görsel için iplik hatası sınıf doğruluk olasılığı %70'tir.

6.2.4 Yabancı obje hatası

Segmentasyon işlemi için kumaş yüzeylerinde görülen yabancı obje hatalarının test işlemi 3 farklı görüntü işleme yöntemi ve orijinal görüntü ile Şekil 6.4'te bir örnek ile sunulmuştur.



Şekil 6.5 : (a) Orijinal görüntüler ile (b) CLAHE ile elde edilen görüntüler ile (c) görüntü doğrusal kontrast germe ile elde edilen görüntüler ile ve (d) sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe ile elde edilen görüntüler ile eğitilen modellerin yabancı obje hatasına karşı sonuçları.

Şekil 6.5.a'da orijinal görüntüler ile modelin eğitilmesi sonucu elde edilen sonuçlara yabancı obje hatası üzerinden rastgele bir örnek görülmektedir. Bu örnekteki tespit edilen yabancı obje sınıf doğruluk olasılığı %86'dır. Şekil 6.5.b'de CLAHE yönteminden geçirilerek elde edilen görüntüler ile eğitilen modelin sonucu görülmektedir. Aynı görsel için bu modelin çıktısında yabancı obje sınıfı için doğruluk olasılığı %75'tir. Şekil 6.5.c'de görüntü doğrusal kontrast germe yönteminden geçirilen görüntüler ile eğitilen modelin sonucu görülmektedir. Bu modelin aynı görsel için elde ettiği yabancı obje sınıfı doğruluk olasılığı sonucu %82'dir. Son olarak şekil 6.5.d'de sırası ile ortalama filtre, keskinleştirme ve görüntü doğrusal kontrast germe yöntemlerinin hibrit kullanımından elde edilen görseller ile eğitilen model görülmektedir. Bu modelin sonucunda, aynı görsel için yabancı obje sınıfı doğruluk olasılığı %74'tür. Şekil 6.5.d'de iki adet yabancı obje tespit edildiği gözlemlenmektedir ki bu istenmeyen bir durumdur.

6.3 Uç Cihazlarda Gerçek Zamanlı Çalışma

Karşılaştırmalı analizlerin tekil bir hesaplama ortamıyla sınırlı kalmaması önemlidir. Kapsamlı bir inceleme, özellikle Raspberry Pi 4B ve NVIDIA GeForce GTX 1050 (3072MiB GPU) ile donatılmış bireysel bir bilgisayarda ve Nvidia 3090 GPU'ya sahip bir bilgisayarın kullanıldığı Yüksek Performanslı Hesaplama (HPC) laboratuvarında olmak üzere birden fazla hesaplama platformunda gerçekleştirilmektedir. Hesaplama bağlamlarının bu şekilde çeşitlendirilmesi, modelin farklı donanım özellikleri altındaki performansının sağlam bir şekilde değerlendirilmesini sağlamakta ve böylece çalışmanın bulgularının genellenebilirliğini ve güvenilirliğini artırmaktadır. Bu karşılaştırmalı analizlerin hem algoritmik çerçeveler hem de hesaplama ortamları açısından titizlikle düzenlenmesi, modelin farklı koşullar altındaki davranışının bütünsel bir şekilde anlaşılmasına katkıda bulunmaktadır. Bu çok yönlü araştırma, yalnızca görüntü işlemenin segmentasyon ve tespit üzerindeki etkisinin inceliklerini ortaya çıkarmakla kalmayıp, aynı zamanda önerilen yöntemlerin farklı hesaplama mimarileri genelinde potansiyel ölçeklenebilirliği ve uyarlanabilirliği hakkında değerli bilgiler sağlamaktadır.

Çizelge 6.1 : Uç cihazda üretici modele göre hız ve parametre performansı.

Cihaz	Bilgi	ESRGAN Üretici Tahmin Zamanı	Üretici Toplam Parametre Sayısı	Üretici Tahmin Edilen Boyut
Raspberry Pi 4B	RRDB 1 Blok	40,3548 sn	114,083	219.73 mb
	RRDB 2 Blok	53,9412 sn	218,003	284.82 mb
	RRDB 4 Blok	80,1538 sn	425,843	415.01 mb
	RRDB 8 Blok	135,8344 sn	841,523	675.37 mb
	Orijinal Blok	-	16,697,987	4435.38 mb
Kişisel Bilgisayar (NVIDIA GeForce GTX 1050 3072MiB GPU)	RRDB 1 Blok	0,0038 sn	114,083	219.73 mb
	RRDB 2 Blok	0,0059 sn	218,003	284.82 mb
	RRDB 4 Blok	0,0097 sn	425,843	415.01 mb
	RRDB 8 Blok	0,0149 sn	841,523	675.37 mb
	Orijinal Blok	0,038 sn	16,697,987	4435.38 mb

Çizelge 6.1’de ESRGAN modelinin ağırlıklarını, parametre sayılarını ve çizelge 6.2’de belirli bir girdideki tüm segmentlere ayrılmış görüntülerin tahmini sırasındaki hızının performansına ilişkin kapsamlı bir fikir vermektedir. ESRGAN’ın üreticisi model kısmı yapay görüntüyü üretmektedir. Üretilen bu görüntünün uç cihazda ne kadar süre içerisinde sonuç alındığı önem arz etmektedir. Raspberry pi gibi cihazlarda büyük modeller sonuç vermemektedir. Model parametrelerinde azalmaya gidilmesi gerekmektedir.

ESRGAN modeli tarafından çözünürlük ve kalitede iyileştirilmeye giden görüntü 416x416 piksel çözünürlükte 16 parçaya sıralı olarak bölünmesini içermektedir. Yamalara ayrılmış görüntülerin belirlenen hatalı durumları ve yeniden yapılandırma sürecinin zamansal süresi de dâhil olmak üzere bu yamalara ayırma sonuçları titizlikle belgelenmiştir. Orijinal görüntüyü 16 küçük parçadan oluşan bir ızgaraya bölen bu yamalara ayırma stratejisi, tespit sürecinin ayrıntı düzeyini artırmaya hizmet etmektedir.

Çizelge 6.2 : Uç cihazların YOLOv8 modeline göre hız performansı.

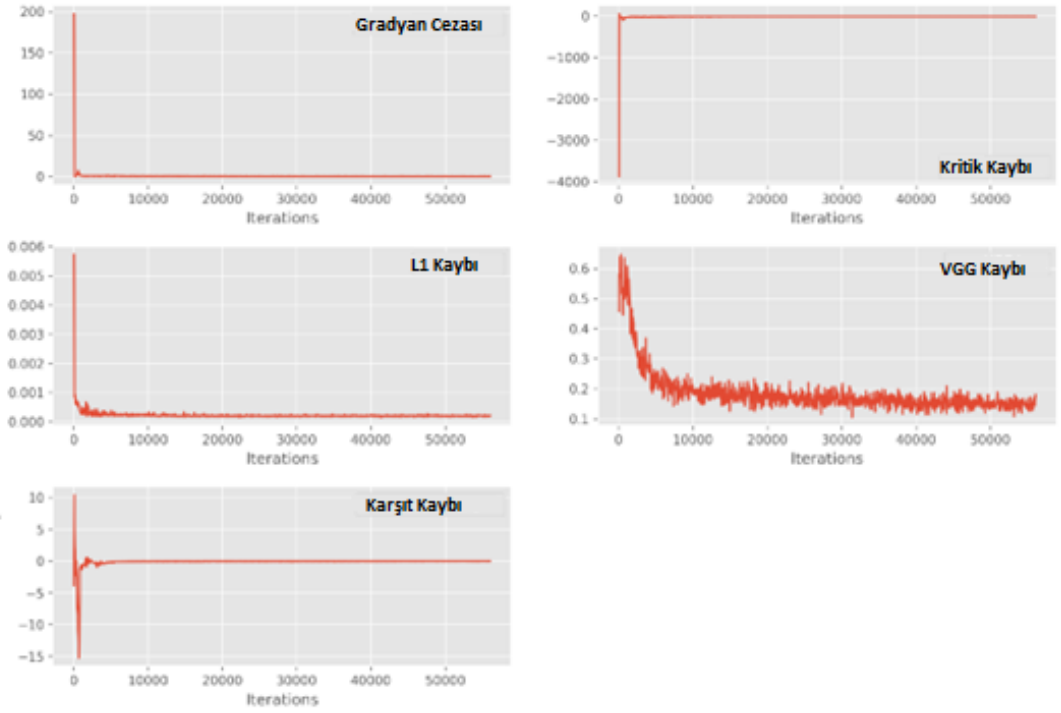
Uç Cihaz	YOLOv8
Raspberry pi 4b	15,1835 sn
Laptop	0,4391 sn
HPCS	0,0597 sn

Çizelge 6.2’de lokal bölgelere ayrılmış 16 parçadan bir adet görüntü parçasının uç cihazlardaki test süreci karşılaştırılmıştır. Her 416x416 piksel segment bağımsız analize tabi tutularak tespit edilen özelliklerin mekânsal dağılımının daha iyi anlaşılmasına katkıda bulunmuştur. YOLOv8 modeli daha sonra bu bölümlere

ayrılmış görüntülerin her biri için tahminler üreterek, kumaş veya ilgili bölge içindeki hatalı durumların tanımlanmasıyla ilgili bölgeleri belirlemiştir. En önemlisi, sonraki yeniden yapılandırma aşaması, ayrı ayrı tahmin edilen 16 adet 416x416 görüntünün titiz bir şekilde birleştirilmesiyle karakterize edilmesidir. Bu yeniden yapılandırma işlemi sayesinde, bölümlere ayrılmış görüntüler 1664x1664 piksel boyutlarında kapsamlı bir görüntüye dönüştürülmektedir. Bu yeniden yapılandırma, yalnızca bölümlere ayrılmış görüntülerden elde edilen tahminleri birleştirmekle kalmayıp, aynı zamanda tüm görüntünün yüksek bir çözünürlükte bütünsel olarak anlaşılmasını da kolaylaştırmaktadır. Yeniden yapılandırma sürecinin zamansal süresinin dâhil edilmesi, değerlendirmeye zamansal bir boyut katarak, modelin bölümlere ayrılmış tahminleri yeniden yapılandırma ve sentezlemedeki hesaplama verimliliğine ilişkin içgörüler sunmaktadır. Segmentasyon, yeniden yapılandırma ve zamansal hususları kapsayan bu bütünsel analiz, YOLOv8 ve ESRGAN modelinin görüntü analizi ve tespit görevleri bağlamındaki etkinliğinin kapsamlı bir şekilde değerlendirilmesine katkıda bulunmaktadır.

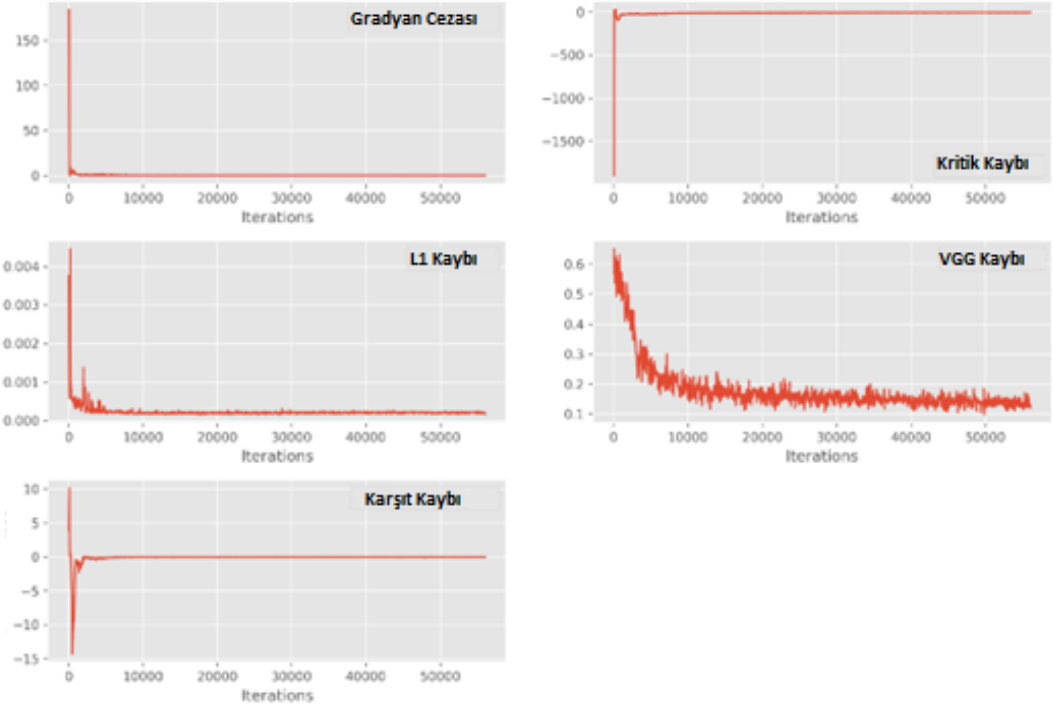
6.4 ESRGAN Boyut Azaltma

ESRGAN eğitim işlemlerini kontrol ettiğimizde CNN kanal sayılarının ve RRDB blok sayısının azaltılması model eğitim süresini de azaltmaktadır. Eğitim işlemi boyunca RRDB blokları orijinal modelden (23 RRDB blok), 1 RRDB bloğa kadar düşürülmüştür. 1000 epokta ve toplamda 50000'e yakın iterasyon döngüsü ile eğitilmiş RRDB ve kanal sayısı azaltılmış üretici modellerin ilgili kayıp fonksiyonları aşağıda verilmiştir.



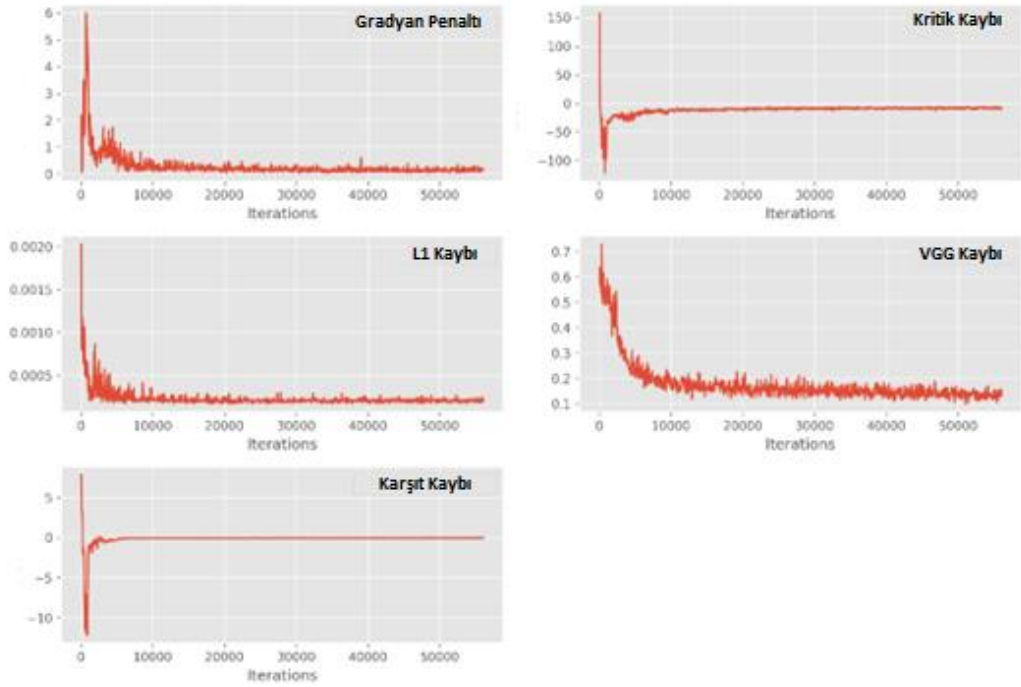
Şekil 6.6 : 1 RRDB bloğuna sahip üreticinin toplam iterasyon sayısına karşılık gelen kayıp fonksiyonu grafiği

Şekil 6.6’da 1 RRDB bloğuna sahip üretici modelin kayıp fonksiyonlarına bakıldığında VGG kayıp fonksiyonu hariç diğer tüm kayıp fonksiyonları başlangıçta bir salınım yaptıktan sonra hızlıca optimuma ulaştığı görülebilmektedir. VGG kayıp fonksiyonu ise iterasyonlar ilerledikçe istenilen optimum sonuca vardığı gözlenmektedir. Diğer kayıp fonksiyonlarının aksine VGG kayıp fonksiyonunun eğimi aşağı doğru olduğundan dolayı iterasyon sayısı arttırılsa öğrenime devam edeceği çıkarılmaktadır. Şekil 6.6’ya bakıldığında Gradyan cezası, Kritik kaybı, L1 kaybı, Karşıt kaybı ve VGG kaybı başlangıç değerlerinin çok yüksek bir başlangıç kaybı ile başladığı görülmektedir.



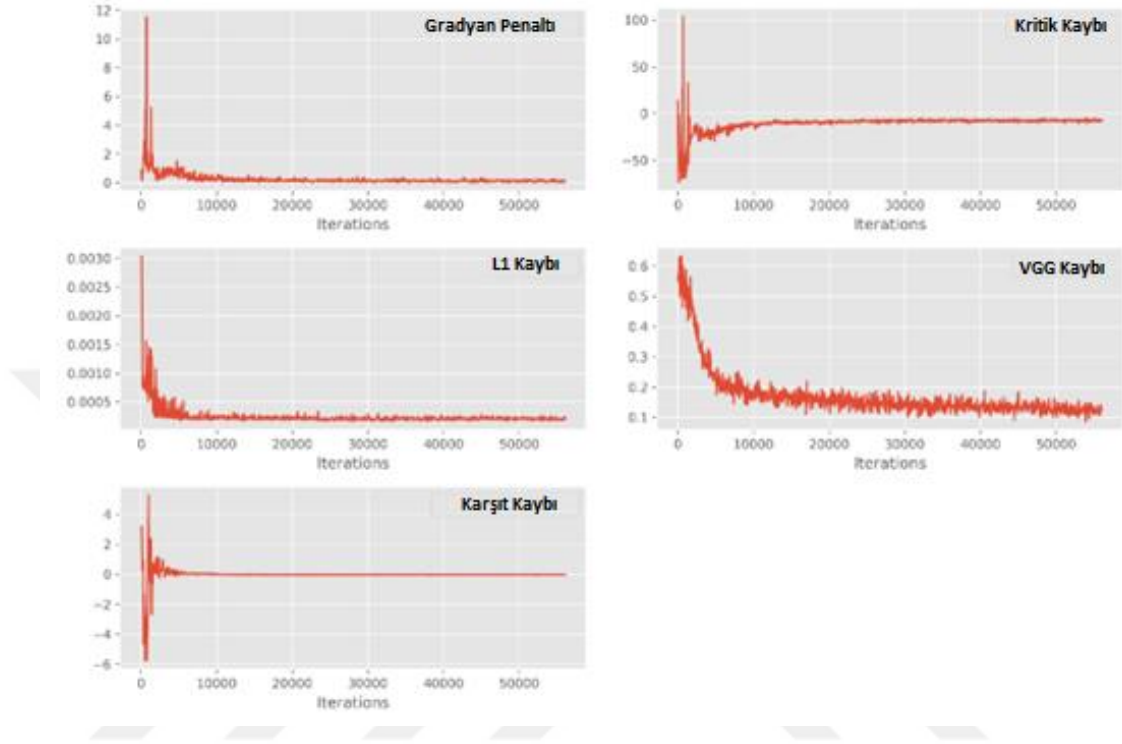
Şekil 6.7 : 2 RRDB bloğuna sahip üreticinin kayıp fonksiyonu

Şekil 6.7’de Gradyan cezası kaybı, L1 kaybı ve Kritik kayıp salınımı daha iyi bir noktadan başlamaktadır. Bu da daha kısa sürede optimum noktaya ulaşacağını göstermektedir. Diğer kayıp fonksiyonları şekil 6.6’deki kayıp fonksiyonlarına benzerlik göstermektedir.



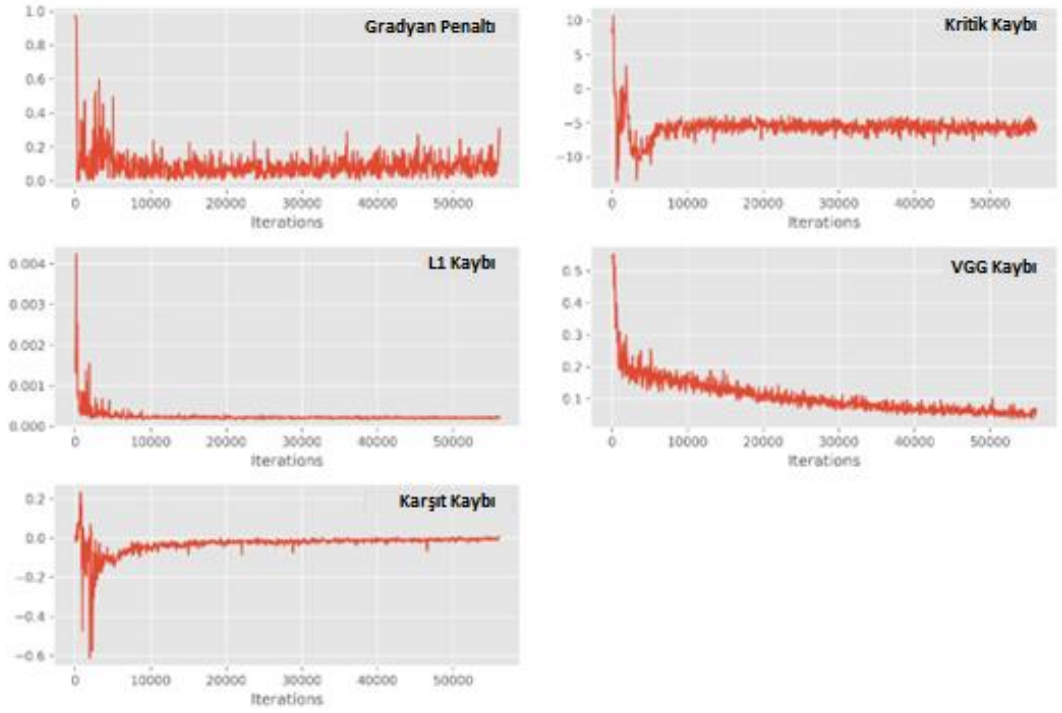
Şekil 6.8 : 4 RRDB bloğuna sahip üreticinin kayıp fonksiyonu

Şekil 6.8'deki 4 adet RRDB bloğuna sahip üretici model, VGG kaybı haricinde tüm kayıp fonksiyonlarında salınma daha iyi bir noktadan başlamaktadır. Önceki üretici modelin aksine bu model daha kısa sürede optimum noktaya ulaşılacağını göstermektedir.



Şekil 6.9 : 8 RRDB bloğuna sahip üreticinin kayıp fonksiyonu

Şekil 6.9'deki 8 adet RRDB bloğuna sahip üretici model Kritik kaybı, VGG kaybı ve Karşıt kaybının salınma daha iyi bir noktadan başlamaktadır. Şekil 6.8'e bakıldığında Gradyan cezası 6'dan başlarken şekil 6.9'da ise 12'den başlamaktadır. Önceki üretici modelin aksine bu modelin eğitiminin daha kısa sürede optimum noktaya ulaşılacağını göstermektedir. L1 kaybının şekil 6.8'den daha kötü olması ise yüksek çözünürlüklü görüntülerin ince ayrıntıları yeniden yapılandırma daha kötü hale geldiğini göstermektedir. Gradyan kaybının şekil 6.8'ye göre kötü olması ise 8 RRDB bloğuna sahip üretici modelin daha kötü bir eğitim istikrarına sahip olduğunu göstermektedir.



Şekil 6.10 : Orijinal RRDB bloğuna sahip üreticinin kayıp fonksiyonu

Şekil 6.10'daki orijinal RRDB bloğuna sahip üretici modelin bir önceki modellere göre başlangıçta çok daha iyi yakınsadığı ve en optimum noktaya vardığını görmekteyiz. Bunun başlıca nedenleri CNN'lerin kanal sayısının fazla olması ve RRDB bloklarının yeteri kadar öznetelik öğrenmesinden kaynaklanmaktadır. Yapılan bu işlemlerin kaliteyi arttırdığı gibi eğitim ve test süresini de arttırdığı gözlemlenmektedir. Artı ve eksi yönleri değerlendirildikten sonra optimum bir üretici modelin seçilmesi gerekmektedir.

Tüm üretici modellerin kayıp fonksiyonlarına baktığımızda L1, VGG, Karşıt kayıp eğitim iterasyonları boyunca azalıyor gibi görünmektedir, bu da iyi bir işarettir. Bu, modelin gerçekçi ve doğru yüksek çözünürlüklü görüntüler üretme yeteneğini giderek geliştirdiğini göstermektedir. Karşıt kayıp 0 civarındadır. Bu genellikle GAN eğitiminde arzu edilen bir durumdur çünkü üretici sürekli olarak gelişmeye teşvik eder. Gradyan cezası eğitim boyunca nispeten istikrarlı görünmektedir, bu da eğitim sürecinin istikrarını korumaya etkili bir şekilde yardımcı olduğunu göstermektedir. L1 kaybı, süper çözünürlük gibi görüntü yeniden yapılandırma görevleri için tipik olan bir şekilde giderek azalıyor gibi görünmektedir. Bu, modelin yüksek çözünürlüklü görüntülerin ince ayrıntılarını yeniden yapılandırmada daha iyi hale geldiğini gösterir. VGG kaybı da sabit bir düşüş gösterir, bu da modelin önceden

eğitilmiş VGG ağı tarafından öğrenilenlere benzer özellikleri öğrendiğini gösterir. Bu, modelin görsel olarak gerçekçi görüntüler oluşturmaya yardımcı olabilir.

Üretilen modellerin kalitesini kontrol ederken de birtakım metriklere ihtiyaç duyulmaktadır. PSNR piksel düzeyinde doğruluğa, SSIM ve LPIPS algısal benzerliğe ve VIF bilgi doğruluğuna odaklanır. PSNR basit ve verimliken, SSIM, LPIPS ve VIF gibi ölçütler daha karmaşık ve hesaplama açısından maliyetli olabilir. Yani, bu ölçütlerin hesaplanması daha fazla işlem gücü, bellek veya zaman gerektirebilir. PSNR'nin yorumlanması kolayken SSIM, LPIPS ve VIF'in altında yatan ilkelerin daha iyi anlaşılması gerekebilir. Piksel düzeyinde doğruluk çok önemliyse PSNR uygun olabilmektedir. Görsel benzerlik ve insan algısı önemliyse SSIM veya LPIPS'i uygun olabilmektedir. Hem kalite hem de doğruluk için kapsamlı bir ölçüme ihtiyacınız varsa VIF uygun olmaktadır. Çizelge 6.3' de bahsedilen tüm metriklerin karşılaştırmalı sonuçları bulunmaktadır.

Çizelge 6.3 : Üretici kısımdaki RRDB blok sayısının azaltılması ve performansı.

	Metrik	1 RRDB Blok vs Orijinal	2 RRDB Blok vs Orijinal	4 RRDB Blok vs Orijinal	8 RRDB Blok vs Orijinal
Delik Hatası	PSNR ↑	32.45924 dB	34.14448 dB	34.26174 dB	34.14357 dB
	SSIM ↑	0.82801	0.83882	0.85240	0.82853
	LPIPS ↓	0.22182	0.22005	0.20543	0.21639
	VIF ↑	0.65335	0.69677	0.78226	0.78236
Yağ Lekesi Hatası	PSNR ↑	29.47268 dB	30.06158 dB	30.35300 dB	30.14099 dB
	SSIM ↑	0.74858	0.73102	0.74005	0.75968
	LPIPS ↓	0.19617	0.30164	0.24772	0.19325
	VIF ↑	0.63504	0.63501	0.58555	0.62797
İplik Hatası	PSNR ↑	28.88531 dB	29.75024 dB	30.26740 dB	29.26663 dB
	SSIM ↑	0.67455	0.71727	0.76575	0.72448
	LPIPS ↓	0.28651	0.28135	0.23066	0.26096
	VIF ↑	0.39905	0.54387	0.70018	0.5716
Obje Hatası	PSNR ↑	33.15888 dB	33.89889 dB	32.26240 dB	35.05957 dB
	SSIM ↑	0.86443	0.87527	0.87052	0.87644
	LPIPS ↓	0.21871	0.21437	0.20101	0.19897
	VIF ↑	0.62281	0.63091	0.65574	0.63642

Çizelge 6.3'te 300 epok kadar eğitilmiş orijinal ESRGAN ve üretici kısmındaki CNN kanallarının ve RRDB bloklarının azaltılması ile elde edilen sonuçların karşılaştırmalı hali sunulmaktadır. Her bir hata sınıfı için 4 farklı kalite ölçme metriği sonuçları gözlenmiştir ve her bir hata sınıfı için en başarılı metrik siyah renkle belirtilmiştir. 5 kat çapraz doğrulama yapıldıktan sonra elde edilen sonuçlara bakıldığında 4 RRDB bloğu ve 8 RRDB bloğu arasında kalite metrik sonuçları

arasında önemli bir fark olmadığı gözlemlenmekle beraber 4 RRDB bloğu ile elde edilen sonuçların nispeten daha iyi olduğu gözlemlenmiştir. Ayrıca şekil 6.8'deki kayıp fonksiyonu da bu iddiayı destekleyecek niteliktedir. Şekil 6.8'deki L1 ve Gradyan kaybının 8 adet RRDB bloğuna sahip olan modelin, 4 adet RRDB üretici modelden daha kötü olduğu gözlenmektedir. Bunun başlıca nedenlerinden biri ise RRDB blokları artmış olsa bile CNN kanal sayısının azalması ile bir yerden sonra model yeteri kadar bilgiyi tutamamaktadır. Bilgi tutamamadaki bu farklılık Raspberry Pi gibi düşük işlem gücüne sahip cihazlarda süre ve hesaplama açısından fark yaratmaktadır.

6.5 TILDA Veri Seti ile Sonuçlar

Sağlanan ölçümler, bir modelin dört farklı sınıfa göre hata algılama ve segmentasyondaki performansını farklı koşullar altında değerlendirir. Giriş görüntülerinin değiştirilmemiş durumunu temsil eden orijinal görüntü, temel çizgi olarak durur. Herhangi bir ek ön işleme gerek kalmadan modelin hata algılama ve bölümlendirme konusundaki asıl yeteneğini değerlendirmek için bir referans noktası görevi görür.

Çizelge 6.4 : Görüntü işleme metotları ile her hatanın sınırlayıcı kutu sonuçları.

Metot	Sınıf	Kesinlik	Sınırlayıcı Kutu			
			Duyarlılık	F1 Skoru	mAP50	mAP50-95
Orijinal	Delik Hatası	0.866	0.76	0,809	0.861	0.572
	Obje Hatası	0.841	0.771	0,804	0.849	0.576
	Yağ Lekesi	0.863	0.812	0,836	0.871	0.542
	İplik Hatası	0.713	0.618	0,662	0.659	0.359
CLAHE	Delik Hatası	0.709	0.471	0,565	0.612	0.308
	Obje Hatası	0.679	0.765	0.719	0.848	0.624
	Yağ Lekesi	0.797	0.833	0.814	0.842	0.578
	İplik Hatası	0.478	0.548	0.510	0.437	0.221
Doğrusal Kontrast Germe	Delik Hatası	0.951	0.846	0.895	0.907	0.581
	Obje Hatası	0.739	0.725	0.731	0.768	0.605
	Yağ Lekesi	0.818	0.585	0.682	0.74	0.414
	İplik Hatası	0.79	0.609	0.687	0.705	0.4
Birleşik Metotlar	Delik Hatası	0.962	0.7	0.810	0.744	0.586
	Obje Hatası	0.579	0.621	0.599	0.663	0.518
	Yağ Lekesi	0.705	0.649	0.675	0.71	0.39
	İplik Hatası	0.611	0.543	0.574	0.582	0.281

Çizelge 6.4'te öncesinde kumaşa görüntü işleme metotları uygulanan her bir hata sınıfı veri setinin YOLOv8 modeli ile sınırlayıcı kutu sonuçları gösterilmiştir. Siyah renk ile vurgulanan her bir değer o sınıf hatasındaki en başarılı sınırlayıcı kutu sonuçlarını göstermektedir.

Çizelge 6.5 : Görüntü işleme metotları ile her hatanın sınırlayıcı maske sonuçları.

Metot	Sınıf	Kesinlik	Sınırlayıcı Maske			
			Duyarlılık	F1 Skoru	mAP50	mAP50-95
Orijinal	Delik Hatası	0.81	0.706	0.754	0.829	0.41
	Obje Hatası	0.752	0.686	0.717	0.657	0.335
	Yağ Lekesi	0.844	0.784	0.812	0.84	0.437
	İplik Hatası	0.751	0.647	0.695	0.649	0.234
CLAHE	Delik Hatası	0.809	0.499	0.617	0.617	0.249
	Obje Hatası	0.746	0.706	0.725	0.755	0.391
	Yağ Lekesi	0.868	0.712	0.782	0.815	0.41
	İplik Hatası	0.531	0.516	0.523	0.452	0.135
Doğrusal Kontrast Germe	Delik Hatası	0.951	0.846	0.895	0.866	0.476
	Obje Hatası	0.715	0.702	0.708	0.723	0.436
	Yağ Lekesi	0.838	0.6	0.699	0.738	0.35
	İplik Hatası	0.79	0.609	0.687	0.695	0.254
Birleşik Metotlar	Delik Hatası	0.991	0.7	0.820	0.722	0.412
	Obje Hatası	0.563	0.578	0.570	0.595	0.308
	Yağ Lekesi	0.727	0.635	0.677	0.673	0.301
	İplik Hatası	0.589	0.486	0.532	0.483	0.172

Çizelge 6.5'te öncesinde kumaşa görüntü işleme metotları uygulanan her bir hata sınıfı verisinin YOLOv8 modeli ile sınırlayıcı maske sonuçları gösterilmiştir. Siyah renk ile vurgulanan her bir değer o sınıf hatasındaki en başarılı sınırlayıcı maske sonuçlarını göstermektedir. Çizelge 6.4'te ve çizelge 6.5'te gösterilen sonuçlarda, orijinal görüntünün incelenmesi üzerine, objeleri ve yağ lekelerini algılamak için sınırlayıcı kutu tahmininde fark edilebilir bir model ortaya çıkıyor. Ayrıca orijinal görüntü, özellikle yağ lekeleri hataları bağlamında maske tahmininde dikkate değer bir etkinlik sergiliyor. CLAHE maske tahmini sırasında obje hatalarını belirleme konusunda kayda değer bir yeterlilik sergiliyor ve sınırlayıcı kutu tahmininde oldukça tatmin edici sonuçlar veriyor. CLAHE metoduna göre görüntü doğrusal kontrast germenin uygulanmasında çok daha iyi sonuçlar gözlemlenmektedir. Bu yöntem hem kutu hem de maske tahmininde, özellikle ip hataları ve delik hataları alanlarında, çok olumlu sonuçlar vermektedir. Bu hata sınıflarındaki performans oldukça ortalamanın üstünde olup, bu belirli hata türleri için görüntü doğrusal

kontrast germenin etkinliğindeki potansiyeli ortaya çıkarmaktadır. Ortalama filtre, keskinleştirme ve doğrusal kontrast germeyi birleştiren birleşik metot ise yeteri kadar performans ortaya koyamamıştır. Yalnızca delik hatalarının inceliklerini hem maske hem de kutu tahminlerinde yakalama konusunda dengeli bir yeterlilik sergiliyor. "İplik Hatası" hata türü için tüm işlemler ve metodolojiler düşük performans göstermiştir, bu da algoritmanın bu özel hata tipini tespit etmede zorluk çektiğini gösterir. Tüm metodolojilerin uygulandığı durumda ise "Delik" hata türü için en yüksek kutu mAP50 değeri elde edilmiş, ancak "İplik Hatası" için en düşük maske mAP50-95 değeri görülmüştür. Bu ayrıntılı analiz, çeşitli görüntü işleme yöntemlerinin hata algılama sınırlayıcısı içindeki farklı hata sınıfları üzerindeki incelikli etkisinin altını çizmektedir.

Çizelge 6.6 : Görüntü işleme ile toplam hataların sınırlayıcı kutu sonuçları.

Metot	Sınıf	Kesinlik	Duyarlılık	F1 Skoru	mAP50	mAP50-95
Orijinal	Hepsi	0.821	0.74	0.778	0.81	0.512
CLAHE	Hepsi	0.666	0.654	0.660	0.685	0.433
Doğrusal						
Kontrast	Hepsi	0.824	0.691	0.751	0.78	0.5
Germe						
Birleşik						
Metotlar	Hepsi	0.714	0.628	0.668	0.675	0.444

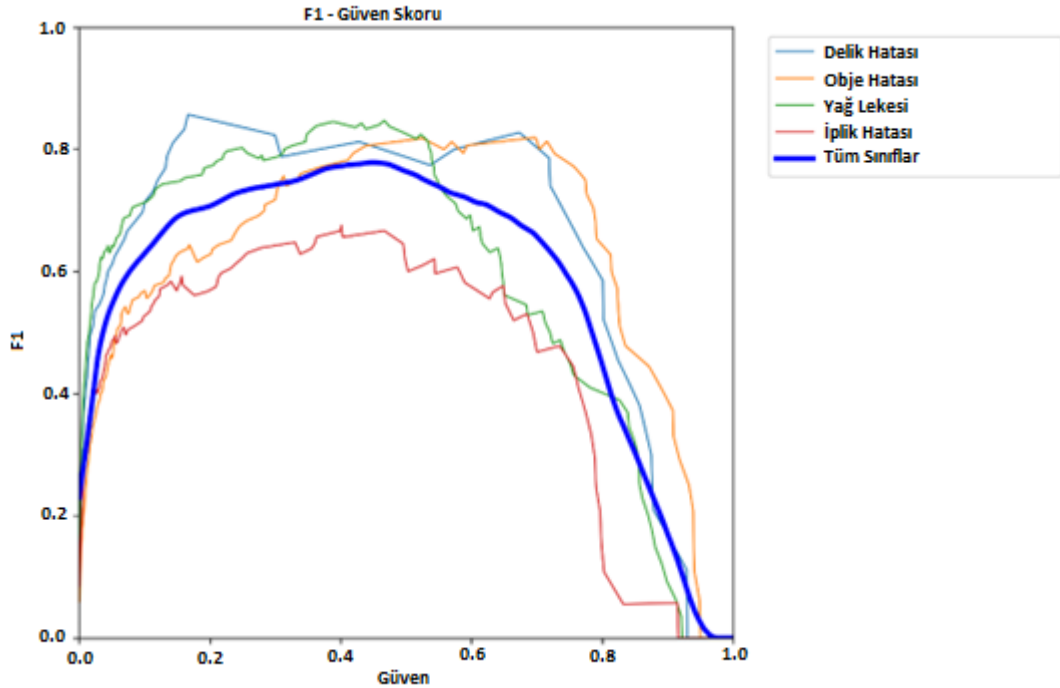
Çizelge 6.6’da öncesinde kumaşa görüntü işleme metotları uygulanan toplam hata sınıfı verisinin YOLOv8 modeli ile sınırlayıcı kutu sonuçları elde edilmektedir. Siyah renk ile vurgulanan her bir değer o sınıf hatasındaki en başarılı sınırlayıcı kutu sonuçlarını göstermektedir.

Çizelge 6.7 : Görüntü işleme ile toplam hataların sınırlayıcı maske sonuçları.

Metot	Sınıf	Kesinlik	Duyarlılık	F1 Skoru	mAP50	mAP50-95
Orijinal	Hepsi	0.789	0.706	0.745	0.744	0.354
CLAHE	Hepsi	0.738	0.608	0.667	0.66	0.296
Doğrusal						
Kontrast	Hepsi	0.824	0.689	0.750	0.755	0.379
Germe						
Birleşik						
Metotlar	Hepsi	0.717	0.6	0.653	0.618	0.298

Çizelge 6.7’de öncesinde kumaşa görüntü işleme metotları uygulanan toplam hata sınıfı verisinin YOLOv8 modeli ile sınırlayıcı maske sonuçları elde edilmektedir.

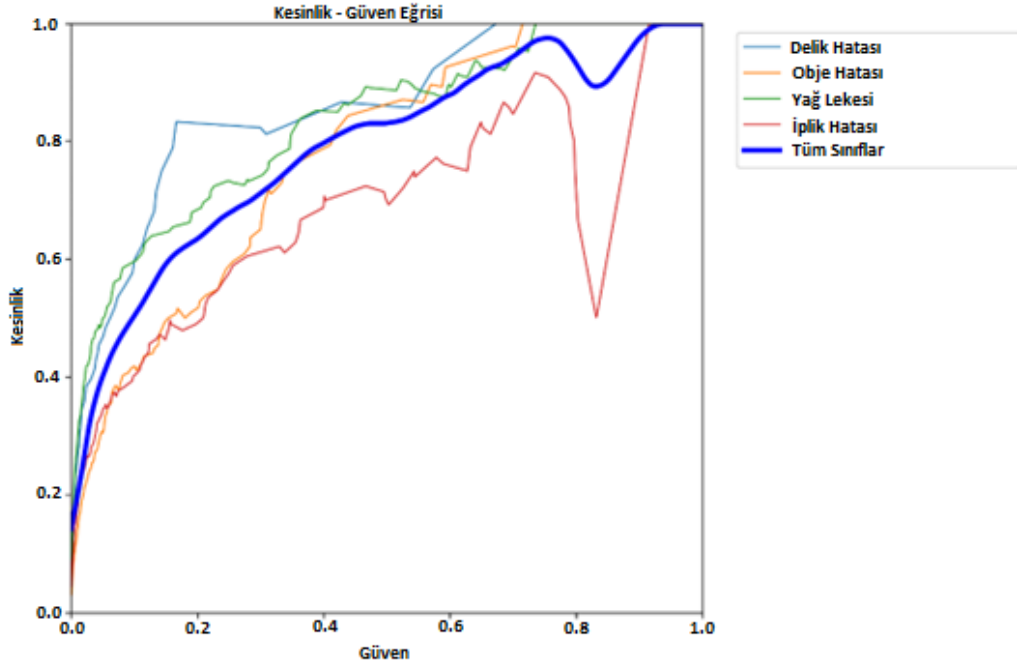
Siyah renk ile vurgulanan her bir değer o sınıf hatasındaki en başarılı sınırlayıcı maske sonuçlarını göstermektedir. Çizelge 6.6'da ve çizelge 6.7'de gösterilen, sınırlayıcı maske ve kutu sonuçlarına bakıldığında orijinal görüntü ile doğrusal kontrast germe metodu metrik başarımları arasında bariz bir fark olmadığı görülmektedir. Çizelge 6.6'ya göre sınırlayıcı kutu performansında orijinal görüntüler üstün gelirken, çizelge 6.7'ye bakıldığında ise sınırlayıcı maske performansında doğrusal kontrast germe metodu üstün gelmektedir. Orijinal görüntü birçok noktada istenilen performansı verdiği ve yeni bir metodun hesaplama maliyeti getireceği için bu noktadan sonra tüm işlemler ve sonuçlar görüntü işleme metotları uygulanmadan elde edilmektedir. Görüntü işleme metotları uygulanmadan kullanılan orijinal görüntüler ile YOLOv8 modelinin sonuçları detaylı bir şekilde aşağıda incelenmektedir.



Şekil 6.11 : Sınırlayıcı kutunun F1 skoru ve güven düzeyleri arasındaki ilişkisini gösteren grafik.

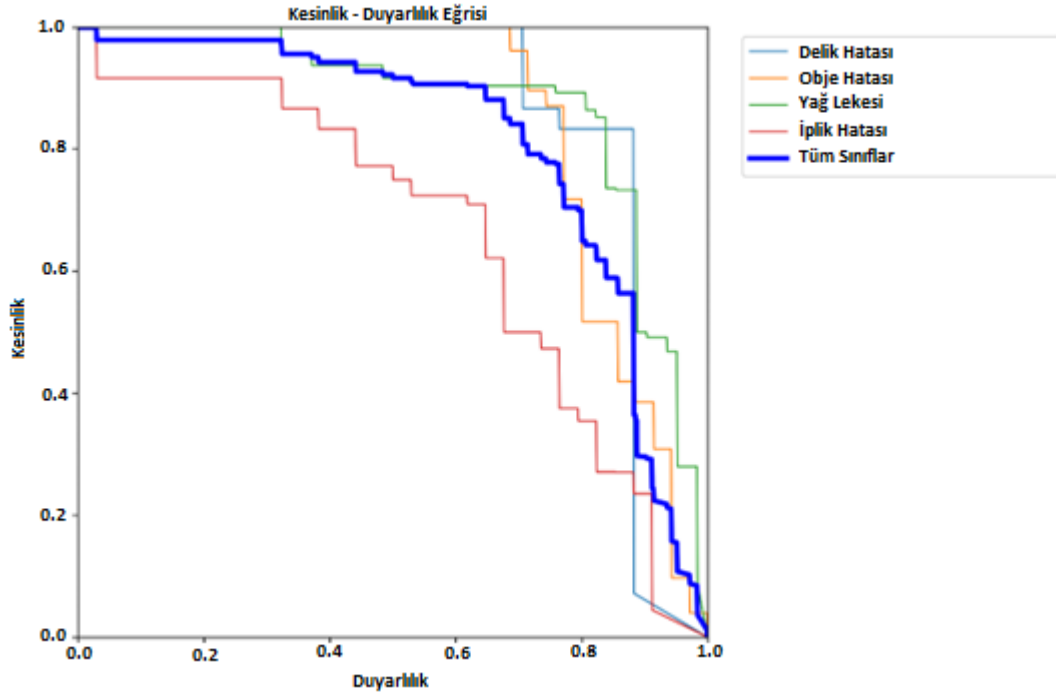
Şekil 6.11, bir makine öğrenimi modelinin farklı sınırlayıcı kutu sınıfları için F1 skoru ve güven düzeyleri arasındaki ilişkiyi gösteren bir F1-Güven eğrisidir. F1 skoru, bir modelin sınıflandırma performansının bir ölçüsüdür ve hem kesinlik hem de duyarlılık metriklerinin harmonik ortalamasını alarak hesaplanmaktadır. Bu, modelin hem kesinlik hem de duyarlılık açısından dengeli bir performans gösterip göstermediğini değerlendirmek için yararlı bir metriktir. Grafik, her bir sınıf için

(“Delik Hatası”, “Obje Hatası”, “Yağ Lekesi”, “İplik Hatası”) ve tüm sınıflar için (“Tüm Sınıflar”) F1 skorlarının modelin tahminlerine duyduğu güven düzeyine bağlı olarak nasıl değiştiğini göstermektedir. Güven düzeyi, modelin kendi tahminlerinin doğruluğuna ne kadar emin olduğunun bir göstergesidir ve genellikle 0 ile 1 arasında değişen bir olasılık değeri olmaktadır. Grafikte görüldüğü üzere, tüm sınıflar için F1 skoru güven düzeyi arttıkça önce artmakta, belirli bir noktaya ulaştıktan sonra azalmaktadır. F1 skorunun en yüksek olduğu nokta, modelin tahminlerinin optimal dengesini temsil etmektedir. “Tüm Sınıflar” eğrisi, tüm sınıfların birleşik F1 skorunu gösterir ve en yüksek F1 skoru olan 0.78’i, 0.449 güven düzeyinde elde etmektedir. Her sınırlayıcı kutu sınıfının eğrisi, o sınıfa özel F1 skorunun güven düzeyine bağlı değişimini yansıtmaktadır. “Yağ Lekesi” sınıfının eğrisi en yüksek zirveye sahipken, “İplik Hatası” sınıfı daha düşük bir zirveye sahip olmaktadır. Bu, “Yağ Lekesi” sınıfının model tarafından “İplik Hatası” sınıfından daha iyi sınıflandırıldığını göstermektedir. Bu grafik, modelin farklı sınıflar arasındaki performansını ve sınıflandırma eşiğinin nasıl ayarlanabileceğini değerlendirmede kullanışlı olmaktadır. F1 skorunun maksimum olduğu güven eşiği, modelin pratikte kullanımı için en uygun eşik olarak seçilebilmektedir. Ancak, bu eşiğin seçimi, yanlış pozitiflerin ve yanlış negatiflerin kabul edilebilir oranına bağlı olarak değişmektedir.



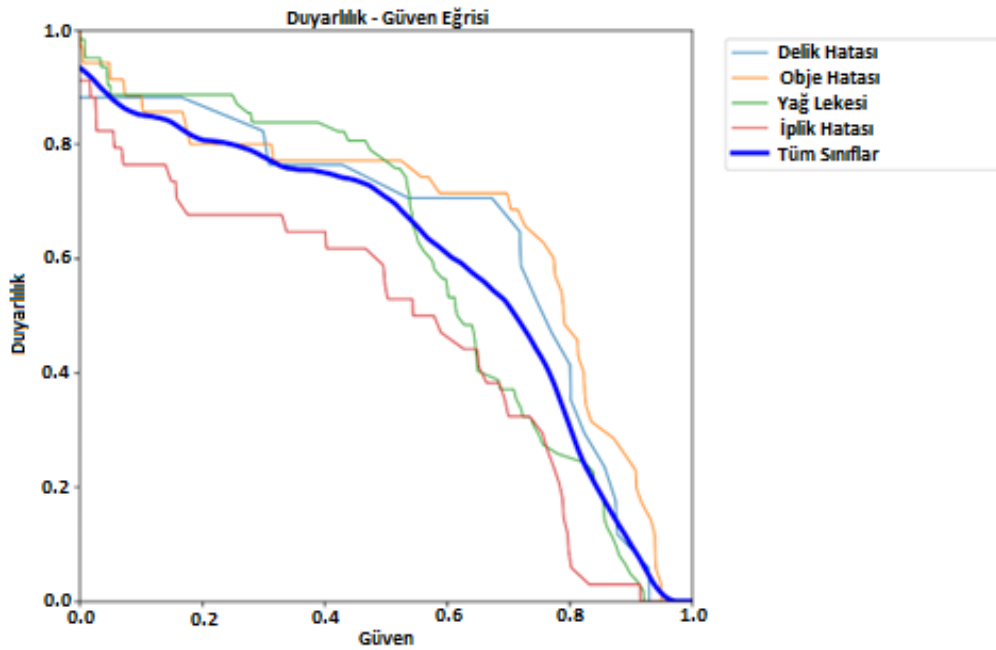
Şekil 6.12 : Sınırlayıcı kutunun kesinlik ve güven arasındaki ilişkisini gösteren grafik.

Şekil 6.12, bir makine öğrenimi modelinin farklı sınırlayıcı kutu sınıfları için kesinlik ve güven arasındaki ilişkiyi gösteren bir Kesinlik-Güven eğrisidir. Kesinlik, modelin pozitif olarak tahmin ettiği örneklerin ne kadarının gerçekten pozitif olduğunu ölçmektedir. Güven düzeyi, modelin kendi tahminlerinin doğruluğuna ne kadar emin olduğunu ifade eden bir değer olmaktadır. Grafikte dört farklı sınıfın (“Delik Hatası”, “Obje Hatası”, “Yağ Lekesi”, “İplik Hatası”) doğruluk değerlerinin güven düzeyine bağlı olarak nasıl değiştiği görülmektedir. Ayrıca, “Tüm Sınıflar” eğrisi tüm sınıfların birleşik performansını göstermektedir. Görüldüğü gibi, tüm sınırlayıcı kutu sınıflarının, kesinlik değerleri başlangıçta düşük güven düzeylerinde artmaktadır ve belirli bir güven düzeyine ulaştığında azalmaktadır. Bu, genellikle modelin çok emin olmadığı tahminlerde daha yüksek bir kesinlik oranı yakalayabildiğini, ancak güven düzeyi çok yüksek olduğunda yanlış pozitiflerin (modelin yanlış bir şekilde pozitif olarak sınıflandırdığı örnekler) arttığını göstermektedir. Eğrilerin en üst noktası, her sınıf için optimal güven eşiğini temsil etmektedir. “Tüm Sınıflar” için bu değer, 0.941 güven düzeyinde mükemmel bir doğruluk oranı olan 1.00'a ulaşmıştır. “İplik Hatası” sınıfının eğrisi diğerlerine göre daha düşük bir zirve yapmış ve daha hızlı bir şekilde düşmüştür, bu da modelin bu sınıftaki örnekleri diğerlerine göre daha az doğrulukla sınıflandırdığını göstermektedir.



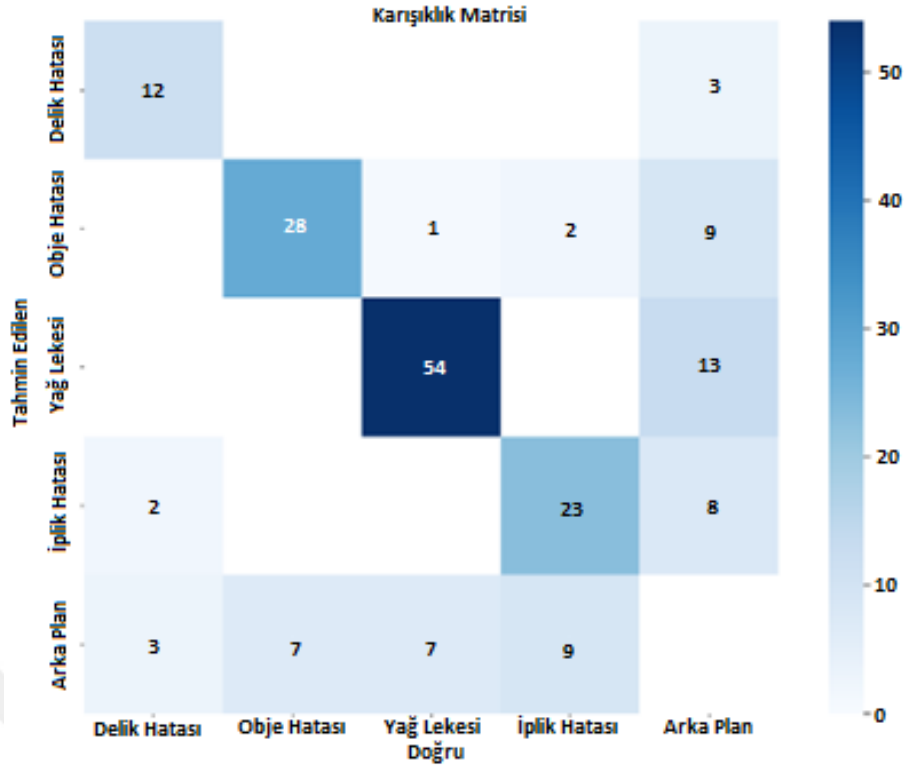
Şekil 6.13 : Sınırlayıcı kutunun kesinlik ve duyarlılık arasındaki ilişkisini gösteren grafik.

Şekil 6.13'te, bir makine öğrenimi modelinin farklı sınırlayıcı kutu sınıfları için hesaplanmış Kesinlik-Duyarlılık eğrisini göstermektedir. Kesinlik-Duyarlılık eğrisi, bir sınıflandırıcı modelin performansını değerlendirmek için kullanılan bir araçtır; burada kesinlik, modelin pozitif olarak sınıflandırdığı örneklerden gerçekten pozitif olanların oranını, duyarlılık ise gerçekte pozitif olan örneklerin ne kadarının model tarafından pozitif olarak tespit edildiğini ölçmektedir. Her bir renkli eğri, bir sınıfın kesinlik ve duyarlılık değerlerinin ilişkisini göstermektedir. Eğrilerin altındaki alan, modelin ilgili sınıf için performansını yansıtır; eğri ne kadar yüksek ve sağa doğru uzanırsa, model o sınıf için o kadar iyi performans göstermiş demektir. Mavi çizgi ile belirtilen "Tüm Sınıflar" eğrisi, tüm sınıfların birleşik performansını gösterir ve $mAP@0.5$ (ortalama kesinlik ortalaması, IoU eşiği 0.5'te) değeri 0.744 olarak belirtilmiştir. $mAP@0.5$ modelin farklı sınıflar arasındaki genel performansını ve tüm tahmin eşiklerindeki ortalama kesinlik değerini göstermektedir. “Yağ Lekesi” sınıfı en yüksek kesinlik ve duyarlılık değerlerine sahip olup model bu sınıfta en iyi performansı göstermiştir (0.840). “Delik Hatası” sınıfı da yüksek bir kesinlik değerine sahiptir (0.829) ve modelin bu sınıfı doğru bir şekilde tespit etme yeteneğinin iyi olduğunu göstermektedir. “Obje Hatası” ve “İplik Hatası” sınıfları daha düşük kesinlik değerlerine sahiptir, bu da modelin bu sınıflardaki örnekleri diğerlerine göre daha az doğrulukla sınıflandırdığını göstermektedir.



Şekil 6.14 : Sınırlayıcı kutunun duyarlılık ve güven arasındaki ilişkisini gösteren grafik.

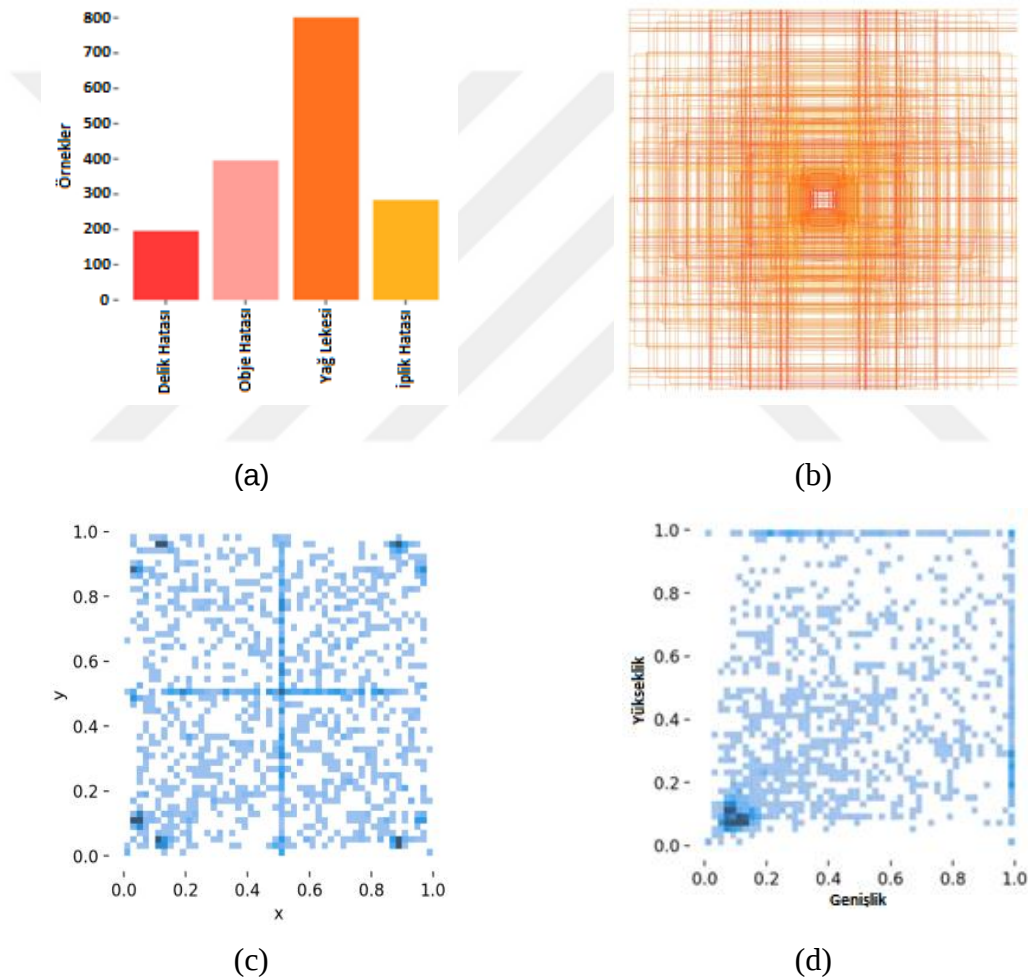
Şekil 6.14'te, bir makine öğrenimi modelinin hata tespit görevi için Duyarlılık-Güven eğrisini göstermektedir. Duyarlılık, modelin tüm pozitif vakaları ne kadar iyi tespit edebildiğini gösteren bir metriktir; yani, gerçekte pozitif olan örneklerin ne kadarının model tarafından doğru olarak tespit edildiğini ölçmektedir. Güven ise modelin tahminlerinin güvenilirliğini, yani her bir tahminin doğru olma olasılığını ifade etmektedir. Grafikte, dört farklı sınıfın ve tüm sınıfların birleşik performansının güven düzeyleri arttıkça duyarlılık değerlerinde nasıl değişim gösterdiği izlenmektedir. İdeal olarak, yüksek bir duyarlılık değeri istenirken, modelin yüksek güven seviyelerinde de yüksek duyarlılık değerlerine sahip olması tercih edilmektedir. Görselde, "İplik Hatası" sınıfının diğer sınıflara göre daha düşük bir duyarlılık değerine sahip olduğu görülür, bu da modelin bu sınıftaki pozitif vakaların daha azını tespit edebildiğini göstermektedir. Diğer taraftan, "Delik Hatası", "Obje Hatası" ve "Yağ Lekesi" sınıfları için eğriler daha yüksek duyarlılık değerleri göstermekte ve güven düzeyi arttıkça daha düşük oranda azalmaktadır. Bu durum, bu sınıflarda modelin genel olarak daha iyi performans gösterdiğini ve güven düzeyi yüksek tahminlerde dahi pozitif vakaları tespit edebildiğini göstermektedir. Grafiğin sağ alt kısmında, tüm sınıflar için duyarlılık değerinin 0.93 olduğu, ancak güvenin 0.000 olduğu belirtilmiştir. Bu, modelin çok düşük güven seviyelerinde dahi yüksek bir duyarlılık değerine sahip olduğunu, yani birçok pozitif örneği doğru olarak tespit edebildiğini göstermektedir. Ancak, bu durum aynı zamanda modelin çok fazla yanlış pozitif (yani gerçekte negatif ama model tarafından pozitif olarak tespit edilen) sonuçlar üretmiş olabileceğine de işaret edebilmektedir.



Şekil 6.15 : Karışıklık matrisini gösteren grafik.

Şekil 6.15'te, bir makine öğrenimi modelinin sınıflandırma performansını gösteren bir karışıklık matrisi olarak yorumlanmaktadır. Karışıklık matrisi, bir sınıflandırma modelinin tahminlerinin doğruluğunu ve hatalarını görselleştiren bir araç olmaktadır. Matrisin satırları modelin yaptığı tahminleri, sütunları ise gerçek etiketleri (True) temsil etmektedir. Bu karışıklık matrisinde beş sınıf bulunmaktadır: "Delik Hatası", "Objek Hatası", "Yağ Lekesi", "İplik Hatası", ve "Arka Plan". Matrisin ana köşegeni (üst sol köşeden alt sağ köşeye), modelin her sınıf için doğru tahmin sayısını gösterir. Örneğin, model 12 kez "Delik Hatası" sınıfını, 28 kez "Objek Hatası" sınıfını, 54 kez "Yağ Lekesi" sınıfını, 23 kez "İplik Hatası" sınıfını ve 9 kez "Arka Plan" sınıfını doğru olarak tahmin etmektedir. Ana köşegen dışındaki hücreler, yanlış sınıflandırmaları temsil etmektedir. Örneğin, "Delik Hatası" sınıfı için model 3 kez yanlışlıkla "Arka Plan" olarak tahminde bulunmaktadır. Aynı şekilde "İplik Hatası" sınıfı için 13 kez "Yağ Lekesi" olarak yanlış tahmin yapılmıştır. Bu hataların sayısı ve dağılımı, modelin belirli sınıfları diğerlerine göre daha iyi veya daha kötü tanıyıp tanımadığını belirlemek için kullanılmaktadır. Genel olarak, modelin "Yağ Lekesi" sınıfını diğer sınıflara göre daha iyi tanıdığı gözlemlenmektedir, çünkü bu sınıfta en yüksek doğru tahmin sayısı (54) bulunmaktadır ve yanlış tahmin sayıları diğer sınıflarla karşılaştırıldığında nispeten daha az bulunmaktadır. "İplik Hatası" sınıfı ise

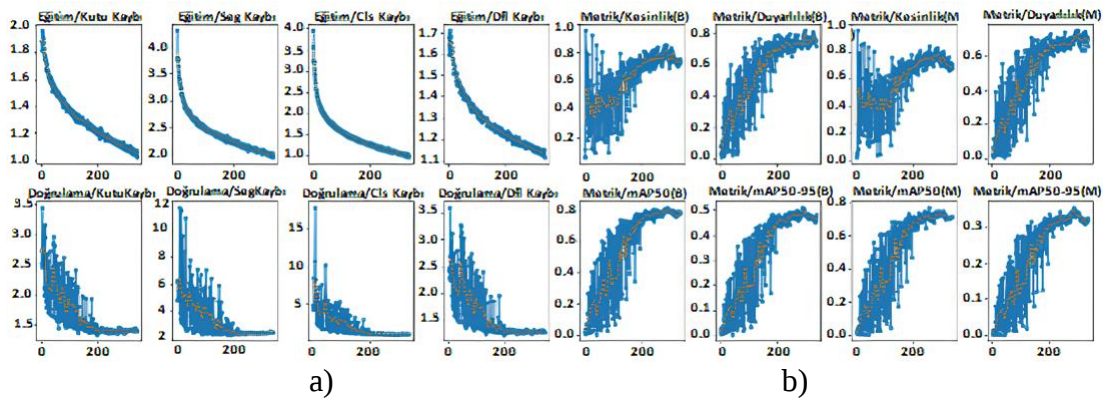
model tarafından en zor tanınan sınıf gibi görünmektedir, çünkü bu sınıfta hem yanlış negatif (modelin "İplik Hatası" olmasına rağmen başka bir sınıf olarak tahmin ettiği vakalar) hem de yanlış pozitif (modelin "İplik Hatası" olmayan bir durumu "İplik Hatası" olarak tahmin ettiği vakalar) tahminlerin sayısı diğer sınıflara göre daha yüksek bulunmaktadır. Karışıklık matrisi, modelin güçlü ve zayıf yönlerini detaylı bir şekilde anlamak için değerli bir araçtır ve modelin performansını iyileştirmek için odaklanılması gereken alanları belirlemeye yardımcı olmaktadır. Bu matrise göre, model özellikle "İplik Hatası" sınıfında yanlış sınıflandırmaları azaltacak şekilde iyileştirilmeye ihtiyaç duymaktadır.



Şekil 6.16 : YOLOv8 modelini veri dağılımı sonuçlarına ait grafikleri (a) bar grafik, (b) ısı haritası, (c) ve (d) dağılım grafikleri.

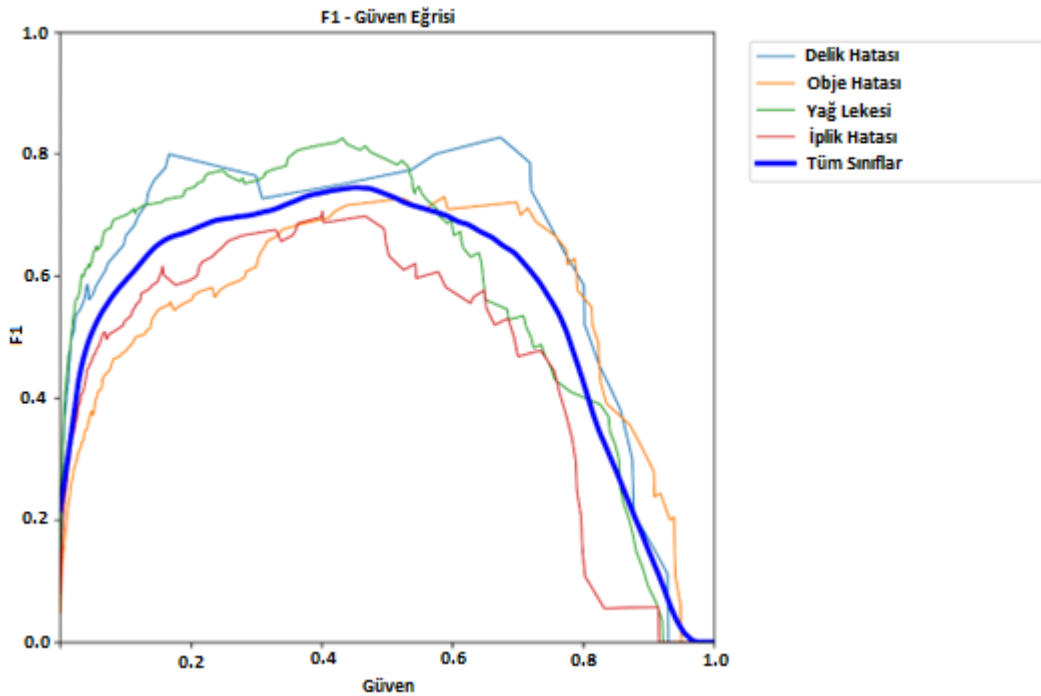
Şekil 6.16’da dört ayrı grafik bulunmaktadır. Her biri, bir makine öğrenimi modelinin hata tespit görevi için çeşitli veri analizlerini yansıtmaktadır. Şekil 6.16.a’da bulunan grafik, veri setindeki dört farklı sınıfın örnek sayılarını göstermektedir. “Yağ Lekesi” sınıfının diğerlerine göre daha fazla örneğe sahip

olduğunu, “İplik Hatası” ve “Obje Hatası” sınıflarının orta düzeyde ve “Delik Hatası” sınıfının en az örneğe sahip olduğunu göstermektedir. Örnek sayısının dağılımı, modelin eğitim sürecinde bu sınıflar üzerindeki performansını etkileyebilir; örneğin, daha fazla örneğe sahip sınıflar genellikle daha iyi tanınmaktadır. Şekil 6.16.b’deki grafik, hata sınırlayıcı kutularının (bounding boxes) yoğunluğunu bir ısı haritası olarak göstermektedir. Yoğunluk, belirli bir alanda ne kadar çok tespit çerçevesi olduğunu göstermektedir. Burada, çerçevelerin büyük çoğunluğu orta kısma yoğunlaşmış gibi görünmektedir. Bu durum, hataların görüntülerde genellikle merkezi bir konuma sahip olabileceğini veya belirli bir alanda daha sık rastlandığını göstermektedir. Şekil 6.16.c ve şekil 6.16.d’deki iki grafik, tespit edilen hataların göreceli konumlarını ve boyutlarını göstermektedir. Sol taraftaki grafik, hataların x ve y koordinatlarına göre konum dağılımını göstermektedir. Bu grafikte, hataların genel olarak görüntünün ortasına doğru yoğunlaştığı gözlemlenmektedir. Sağ taraftaki grafik ise, hataların göreceli genişlik (width) ve yükseklik (height) değerlerinin dağılımını göstermektedir. Çoğu hatanın genişlik ve yükseklik oranlarının küçük olduğu görülürken, özellikle yükseklikte belirgin bir yoğunlaşma var gibi görünmektedir, bu da hataların göreceli olarak daha dar ve uzun olabileceğini düşündürmektedir. Bu tür veri analizleri, modelin eğitiminde ve hata tespiti yaparken karşılaşılabileceği zorlukları belirlemek için önemli olmaktadır. Örneğin, eğer "hole" sınıfı için çok az örnek varsa, model bu sınıfı tanıma konusunda yetersiz kalabilir veya hataların yoğunlukla belirli bir konumda bulunması, modelin bu alanda daha fazla yanlış pozitif tahminler yapma olasılığını artırabilmektedir. Bu gözlemler, veri setinin dengelenmesi, veri artırma ve modelin öğrenme algoritmasının ayarlanması gibi konularda bilgi sağlayabilmektedir.



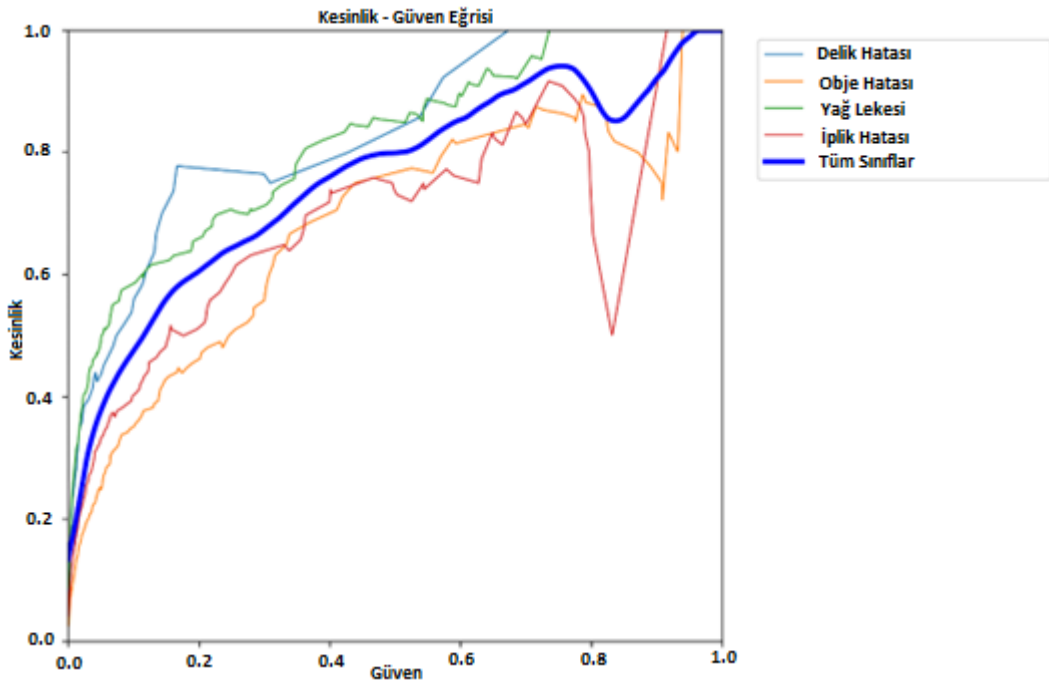
Şekil 6.17 : YOLOv8 sonuçlarına ait epok sayısına karşılık a) kayıp fonksiyonu değişimi b) başarımlar değişimi.

Şekil 6.17’de, modelin eğitim sürecindeki kayıp (loss) ve performans metriklerinin izlenmesini göstermektedir. “Eğitim/Kutu Kaybı” ve “Doğrulama/KutuKaybı” grafikleri, modelin hataların sınırlayıcı kutularını tahmin etmedeki başarısını ölçen hata miktarlarını göstermektedir; eğitim ve doğrulama aşamalarında kayıpların azalması, modelin bu konuda geliştiğini göstermiştir. “Eğitim/Seg Kaybı” ve “Doğrulama/SegKaybı” ise modelin hataların piksel düzeyindeki sınırlarını ne kadar doğru tahmin ettiğini yansıtmaktadır; buradaki dalgalanmalar eğitim ve doğrulama veri kümelerinin tutarlılığı göstermektedir. “Eğitim/Cls Kaybı” ve “Doğrulama/Cls Kaybı” grafikleri sınıflandırma hatasını ifade etmektedir ve burada gözlenen düşüş eğilimi, modelin sınıflandırma doğruluğunun arttığını işaret etmektedir. Son olarak, “Kesinlik”, “Duyarlılık”, “mAP50”, ve “mAP50-95” grafikleri modelin hata algılama performansını değerlendirir; buradaki artış eğilimi, modelin hataların doğru bir şekilde algılama yeteneğinin geliştiğini göstermektedir. Genel olarak, bu metriklerdeki iyileşme, modelin ilgili görevlerde daha iyi performans göstermeye başladığının bir göstergesidir. Ancak, bu iyileşmenin gerçek dünya verileri üzerinde nasıl bir performansa işaret ettiğini değerlendirmek için, eğitim sürecinin bağlamını ve veri setinin niteliklerini de göz önünde bulundurmak gerekmektedir.



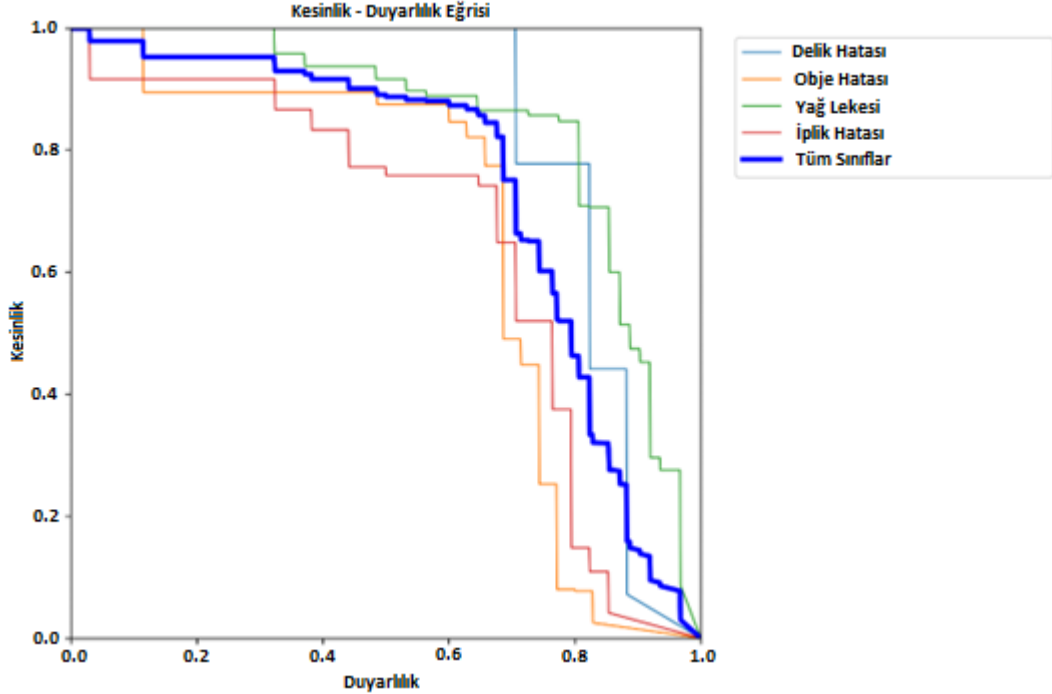
Şekil 6.18 : Sınırlayıcı maskenin F1 skoru ve güven düzeyleri arasındaki ilişkisini gösteren grafik.

Şekil 6.18’de, bir makine öğrenimi modelinin farklı sınırlayıcı maske sınıfları için F1 skoru ve güven düzeyleri arasındaki ilişkiyi gösteren bir F1-Güven eğrisidir. Grafik, her bir sınırlayıcı maske sınıfı için ("Delik Hatası", "Obje Hatası", "Yağ Lekesi", "İplik Hatası") ve tüm sınıflar için ("Tüm Sınıflar") F1 skorlarının modelin tahminlerine duyduğu güven düzeyine bağlı olarak nasıl değiştiğini göstermektedir. Güven düzeyi, modelin kendi tahminlerinin doğruluğuna ne kadar emin olduğunun bir göstergesidir ve genellikle 0 ile 1 arasında değişen bir olasılık değerler olmaktadır. Grafikte görüldüğü üzere, tüm sınıflar için F1 skoru güven düzeyi arttıkça önce artmakta, belirli bir noktaya ulaştıktan sonra azalmaktadır. F1 skorunun en yüksek olduğu nokta, modelin tahminlerinin optimal dengesini temsil etmektedir. "Tüm Sınıflar" eğrisi, tüm sınırlayıcı maske sınıflarının birleşik F1 skorunu gösterir ve en yüksek F1 skoru olan 0.75'i, 0.452 güven düzeyinde elde etmektedir. Her sınırlayıcı maske sınıfının eğrisi, o sınıfa özel F1 skorunun güven düzeyine bağlı değişimini yansıtmaktadır. "Yağ Lekesi" sınıfının eğrisi en yüksek zirveye sahipken, "İplik Hatası" sınıfı daha düşük bir zirveye sahip olmaktadır. Bu, "Yağ Lekesi" sınıfının model tarafından "İplik Hatası" sınıfından daha iyi sınıflandırıldığını göstermektedir.



Şekil 6.19 : Sınırlayıcı maskenin kesinlik ve güven düzeyleri arasındaki ilişkisini gösteren grafik.

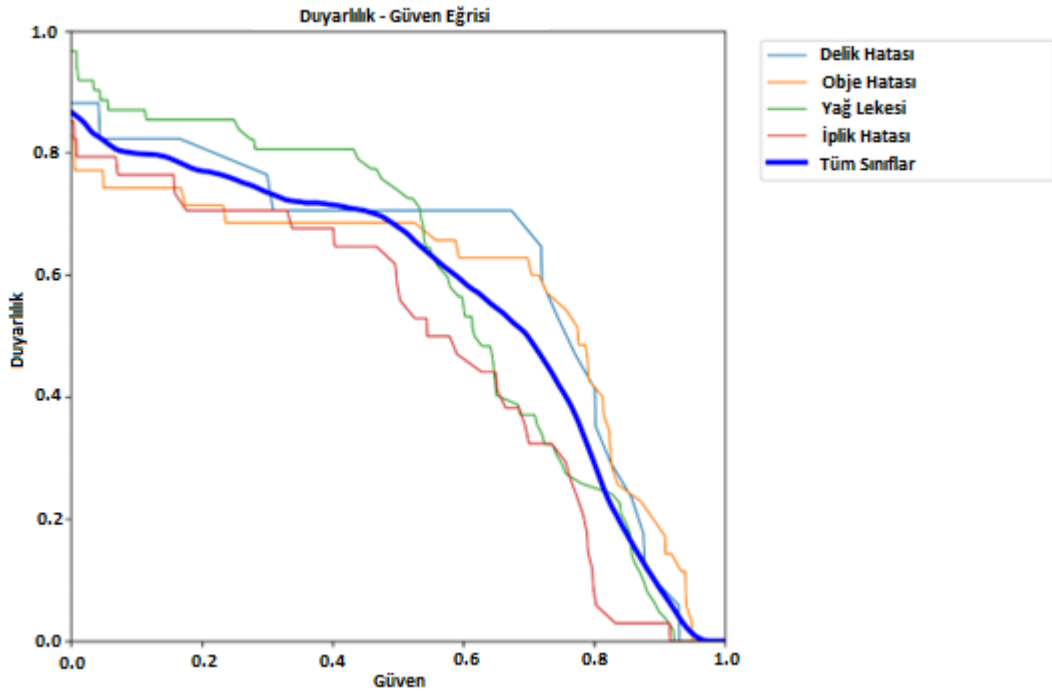
Şekil 6.19'te, bir makine öğrenimi modelinin farklı sınırlayıcı maske sınıfları için doğruluk ve güven arasındaki ilişkiyi gösteren bir Kesinlik-Güven eğrisidir. Grafikte dört farklı sınıfın kesinlik değerlerinin güven düzeyine bağlı olarak nasıl değiştiği görülmektedir. Ayrıca, "Tüm Sınıflar" eğrisi tüm sınıfların birleşik performansını göstermektedir. Görüldüğü gibi, tüm sınıfların kesinlik değerleri başlangıçta düşük güven düzeylerinde artmaktadır ve belirli bir güven düzeyine ulaştığında azalmaktadır. Bu, genellikle modelin çok emin olmadığı tahminlerde daha yüksek bir kesinlik oranı yakalayabildiğini, ancak güven düzeyi çok yüksek olduğunda yanlış pozitiflerin (modelin yanlış bir şekilde pozitif olarak sınıflandırdığı örnekler) arttığını gösterir. Eğrilerin en üst noktası, her sınıf için optimal güven eşliğini temsil eder. "Tüm Sınıflar" için bu değer, 0.965 güven düzeyinde mükemmel bir kesinlik oranı olan 1.00'a ulaşmıştır. Ancak, bu değer tüm sınıfların birleşik bir ölçümü olduğundan, her sınıfın kendine has eğrisine bakmak daha ayrıntılı bilgi sağlayacaktır. "İplik Hatası" sınıfının eğrisi diğerlerine göre daha düşük bir zirve yapmış ve daha hızlı bir şekilde düşmektedir, bu da modelin bu sınıftaki örnekleri diğerlerine göre daha az doğrulukla sınıflandırdığını göstermiştir.



Şekil 6.20 : Sınırlayıcı maskenin kesinlik ve duyarlılık arasındaki ilişkisini gösteren grafik.

Şekil 6.20'te, bir makine öğrenimi modelinin farklı sınırlayıcı maske sınıfları için hesaplanmış Kesinlik-Duyarlılık eğrisini göstermektedir. Her bir renkli eğri, bir

sınırlayıcı maske sınıfının, kesinlik ve duyarlılık değerlerinin ilişkisini göstermektedir. Eğrilerin altındaki alan, modelin ilgili sınıf için performansını yansıtmaktadır; eğri ne kadar yüksek ve sağa doğru uzanırsa, model o sınıf için o kadar iyi performans göstermiş demektir. Mavi çizgi ile belirtilen “Tüm Sınıf” eğrisi, tüm sınıfların birleşik performansını gösterir ve $mAP@0.5$ (ortalama kesinlik ortalaması, IoU eşiği 0.5'te) değeri 0.744 olarak belirtilmektedir. $mAP@0.5$, modelin farklı sınıflar arasındaki genel performansını ve tüm tahmin eşiklerindeki ortalama kesinlik değerini göstermektedir. “Yağ Lekesi” sınıfı en yüksek kesinlik ve duyarlılık değerlerine sahip olup model bu sınıfta en iyi performansı göstermiştir (0.840). “Delik Hatası” sınıfı da yüksek bir kesinlik değerine sahiptir (0.829) ve modelin bu sınıfı doğru bir şekilde tespit etme yeteneğinin iyi olduğunu göstermektedir. “Obje Hatası” ve “İplik Hatası” sınıfları daha düşük kesinlik değerlerine sahiptir, bu da modelin bu sınıflardaki örnekleri diğerlerine göre daha az doğrulukla sınıflandırdığını göstermektedir.



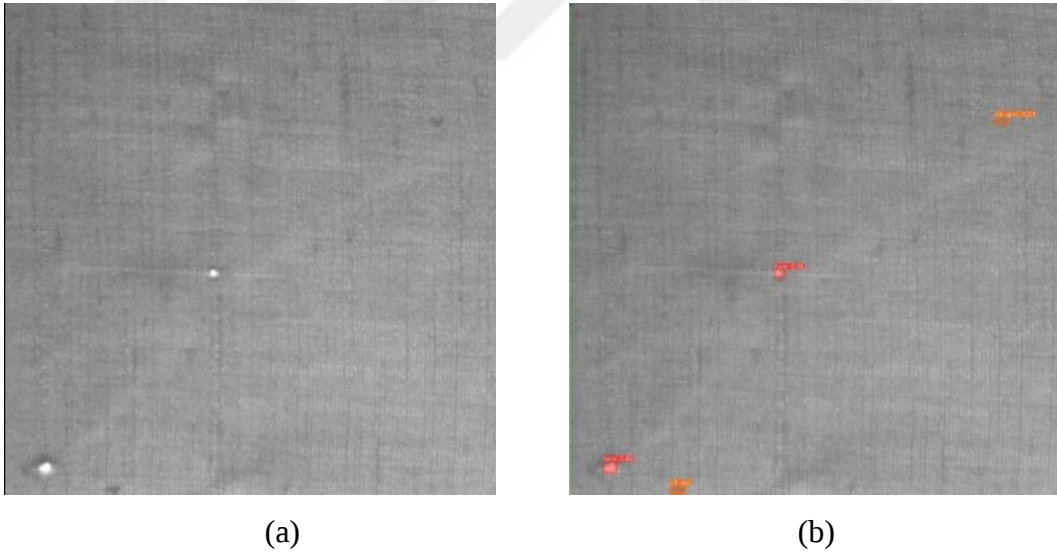
Şekil 6.21 : Sınırlayıcı maskenin duyarlılık ve güven düzeyi arasındaki ilişkisini gösteren grafik.

Şekil 6.21’te bir makine öğrenimi modelinin farklı sınıfların sınırlayıcı maskeleri için duyarlılık ve güven düzeyi arasındaki ilişkiyi gösteren bir Duyarlılık-Güven eğrisidir. Grafikte dört farklı sınıf için duyarlılık değerleri ve ayrıca tüm sınıfların birleşik performansı için bir eğri (“Tüm Sınıflar”) gösterilmektedir. Her bir renkli

eğri, ilgili sınıfın duyarlılık değerinin modelin tahminlerine olan güven düzeyine bağlı olarak nasıl değiştiğini göstermektedir. Güven düzeyi arttıkça, genellikle duyarlılık değerleri düşmektedir. Bu, modelin daha yüksek güven düzeylerinde daha az örnek tespit ettiği anlamına gelebilir, çünkü daha seçici davranmaktadır. “Tüm Sınıflar” için duyarlılık değeri, 0 güven düzeyinde 0,87 olarak belirtilmiştir. Bu, modelin en düşük güven eşiğinde bile oldukça yüksek bir tespit kapasitesine sahip olduğunu göstermektedir. “Yağ Lekesi” sınıfı, diğer sınıflara kıyasla daha yüksek bir duyarlılık değerine sahiptir, bu da modelin bu sınıftaki örnekleri diğer sınıflara göre daha iyi tespit edebildiğini göstermektedir. “Obje Hatası” ve “İplik Hatası” sınıfları, daha düşük duyarlılık değerlerine sahiptir, bu da modelin bu sınıflardaki örnekleri diğerlerine göre daha az tespit ettiğini göstermektedir.

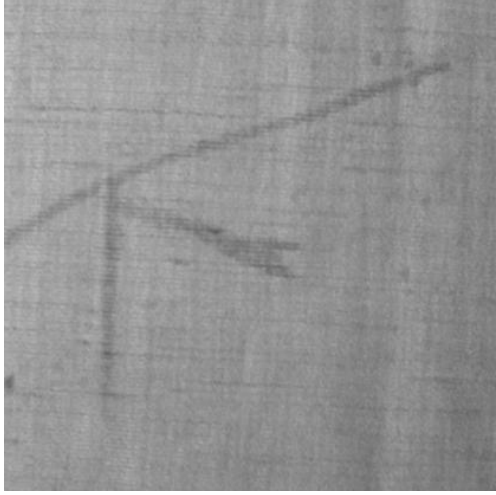
6.5.1 TILDA yeniden yapılandırılan görsel sonuçları

TILDA veri setini ESRGAN modeli ile çözünürlüğü yükselttikten sonra yamalara bölünen görüntüler ile eğitilen YOLOv8 modelinin yamaları yeniden yapılandırılmış sonuç görüntüsü sunulmaktadır.

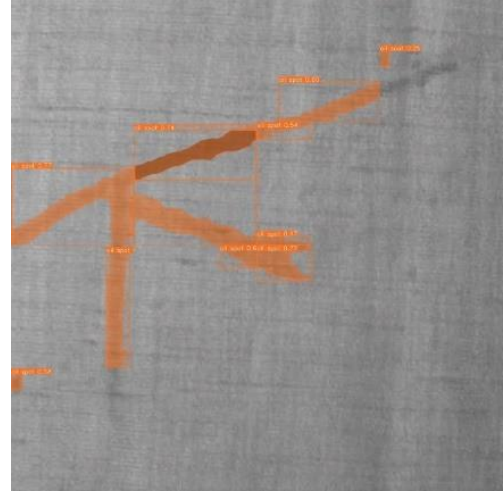


Şekil 6.22 : (a) TILDA verisinde bulunan orijinal delik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen delik hatası görüntüsü.

Şekil 6.22’de TILDA verisine ait görüntüler gösterilmektedir. 6.22.a’daki görüntü orijinal görüntüyü temsil etmektedir. 6.22.b’deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış delik hatası görüntüsüdür.



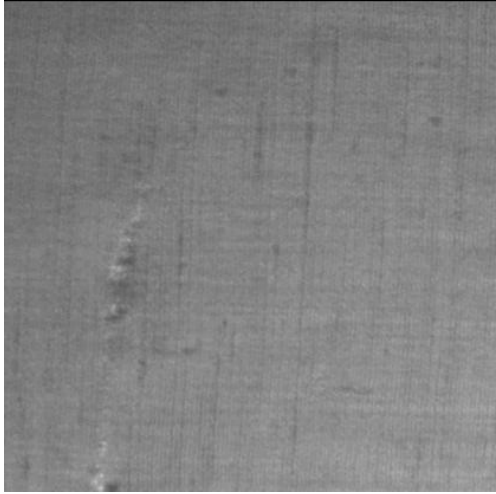
(a)



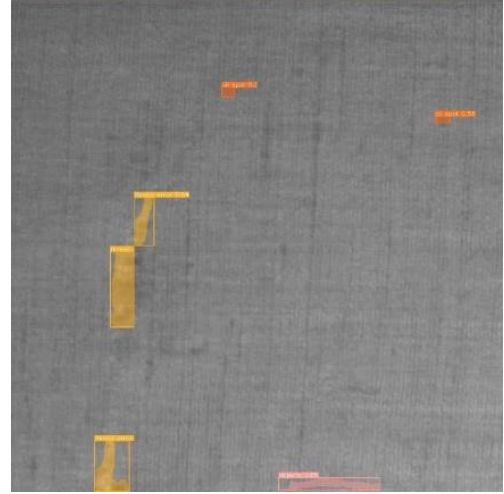
(b)

Şekil 6.23 : (a) TILDA verisinde bulunan orijinal yağ lekesi hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen yağ lekesi hatası görüntüsü.

Şekil 6.23'te TILDA verisine ait görüntüler gösterilmektedir. 6.23.a'daki görüntü orijinal görüntüyü temsil etmektedir. 6.23.b'deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, yağ lekesi hatasının görüntüsünü temsil etmektedir.



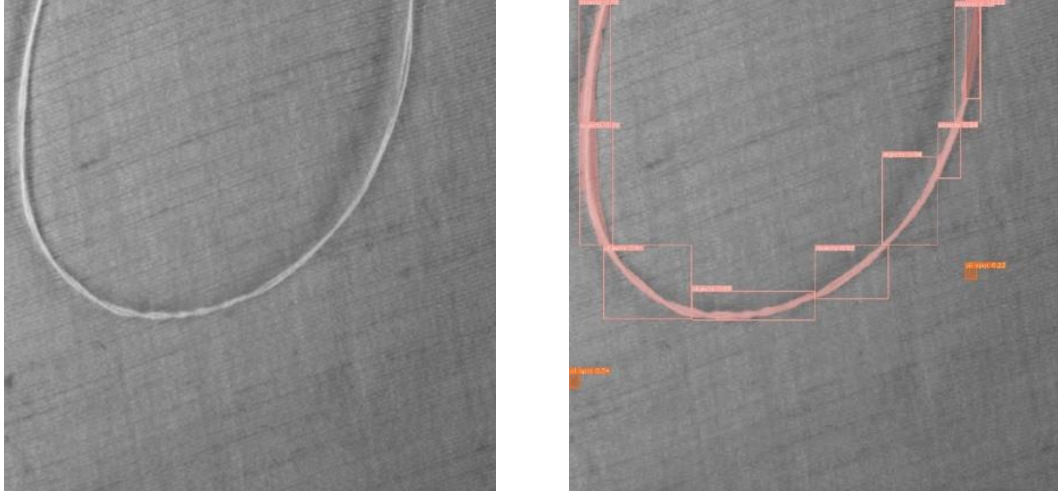
(a)



(b)

Şekil 6.24 : (a) TILDA verisinde bulunan orijinal iplik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen iplik hatası görüntüsü.

Şekil 6.24'te TILDA verisine ait görüntüler gösterilmektedir. 6.24.a'daki görüntü orijinal görüntüyü temsil etmektedir. 6.24.b'deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, iplik hatası görüntüsünü temsil etmektedir.



(a)

(b)

Şekil 6.25 : (a) TILDA verisinde bulunan orijinal obje hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen obje hatası görüntüsü.

Şekil 6.25'te TILDA verisine ait görüntüler gösterilmektedir. 6.25.a'daki görüntü orijinal görüntüyü temsil etmektedir. 6.25.b'deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, obje hatasını görüntüsünü temsil etmektedir.

Özellikle yağ damlası lekeleri üzerinde çok daha başarılı bir sonuç alındığı görsel sonuçlar üzerinde gözlemlenmektedir. Şekil 6.24'te ise fazladan bir obje hatası bulunduğunu gözlemlemekteyiz.

6.6 Yeni Veri Seti ile Sonuçlar

Aşağıda verilen çizelgelerde, üretilen veri seti ile bilgisayarlı görü alanında kullanılan bir hata tespiti modelinin, farklı veri setleri üzerinde nasıl performans gösterdiğini gösteren ölçümleri içermektedir. "Açık renk Kumaş" ve "Koyu renk Kumaş" isimleriyle belirtilen veri setleri üzerinde, sınıflar genelinde ortalama değerler göz önüne alınarak değerlendirme yapılmaktadır. Modelin sınırlayıcı kutu (bounding box) ve maske tahminleri için kesinlik ve duyarlılık oranları, ayrıca ortalama kesinlik değerleri (mAP) sunuluyor. mAP değerleri, %50'den başlayarak %95 IoU'ya kadar olan farklı eşikler için hesaplanarak sunulmaktadır. Bu durum, modelin hataların konum ve şekil tahminindeki hassasiyetini ve tüm doğru tahminleri ne kadar iyi yakaladığını gösteren bir metrik olarak değerlendirilmemektedir. Kısıtlı boyuttaki veri seti ile eğitilen YOLOv8 modeli sonuçları aşağıdaki çizelge 6.4 ve 6.5'te verilmiştir.

Çizelge 6.8 : Üretilen veri setleri ile her hata sınıfının sınırlayıcı kutu tahminleri.

Veri Seti	Sınıf	Sınırlayıcı Kutu				
		Kesinlik	Duyarlılık	F1 Skoru	mAP50	mAP50-95
Koyu Renk Kumaş	Delik	0.917	1	0.956	0.995	0.995
	Objekt	1	0.63	0.773	0.945	0.711
	Yağ Lekesi	0.946	1	0.972	0.995	0.809
	İplik Hatası	0.728	1	0.842	0.945	0.724
Açık Renk Kumaş	Delik	0.634	0.75	0.687	0.565	0.233
	Objekt	0.935	0.75	0.832	0.87	0.761
	Yağ Lekesi	0.451	1	0.621	0.995	0.648
	İplik Hatası	1	0.911	0.953	0.995	0.378

Çizelge 6.8'daki sınırlayıcı kutu metrikleri, koyu renkli veri kümelerindeki nesneleri tespit etmede açık renkli veri kümelerine göre daha iyi performans göstermektedir. Aynı zamanda bu metriklere göre delik ve kumaş yağ lekeleri, koyu kumaş veri setinde üstün bir performans sağlarken, iplik hatası da açık kumaş veri setinde daha iyi bir performans elde etmektedir.

Çizelge 6.9 : Üretilen veri setleri ile her hata sınıfının maske kutu tahminleri.

Veri Seti	Sınıf	Sınırlayıcı Maske				
		Kesinlik	Duyarlılık	F1 Skoru	mAP50	mAP50-95
Koyu Renk Kumaş	Delik	0.917	1	0.956	0.995	0.846
	Objekt	1	0.63	0.773	0.63	0.578
	Yağ Lekesi	0.946	1	0.972	0.995	0.681
	İplik Hatası	0.728	1	0.842	0.945	0.505
Açık Renk Kumaş	Delik	0.467	0.5	0.482	0.266	0.208
	Objekt	0.993	0.75	0.854	0.87	0.702
	Yağ Lekesi	0.477	1	0.645	0.995	0.673
	İplik Hatası	1	0.649	0.787	0.695	0.673

Çizelge 6.9'de bakıldığında koyu kumaş veri setindeki sınırlayıcı maskelerin genel performansı, açık kumaş veri setine göre daha iyi olduğu gözlemlenmektedir. Açık kumaş veri setinde objekt hatası üstün bir şekilde daha iyi bir sınırlayıcı maske tahmini yapmaktadır. Koyu kumaş veri setinde ise delik ve yağ lekeleri hatası diğer veri setine göre bariz bir şekilde üstün bir sonuç sunmaktadır. TILDA veri setindeki kaydedilmiş ağırlıklar ile yeni veri setindeki görüntülerin tekrardan YOLOv8 modelinde eğitilmesi ile eğitim işlemlerini gerçekleştirildi. TILDA ağırlıkları ile eğitime başlayan modelin sonuçları aşağıdaki çizelgelerde gösterilmektedir.

Çizelge 6.10 : TILDA ağırlıkları ile eğitilen YOLOv8 sınırlayıcı kutu sonuçları.

Veri Seti	Sınıf	Kesinlik	Duyarlılık	Sınırlayıcı Kutu		
				F1 Skoru	mAP50	mAP50-95
Koyu Kumaş	Delik	0.986	1	0.992	0.995	0.921
	Objekt	0.92	1	0.958	0.995	0.706
	Yağ Lekesi	0.96	1	0.979	0.995	0.898
	İplik Hatası	1	0.847	0.917	0.995	0.703
Açık Kumaş	Delik	0.732	0.75	0.740	0.596	0.257
	Objekt	1	0.773	0.871	0.995	0.772
	Yağ Lekesi	0.882	1	0.937	0.995	0.726
	İplik Hatası	0.975	1	0.987	0.995	0.516

Çizelge 6.10'ya baktığımızda önceden eğitilmiş model ağırlıkları kullanıldığında çizelge 6.8'e göre çok daha üstün bir sınırlayıcı kutu performansı elde ettiğimiz görülmektedir. Her kumaş ve hata türünde çok daha iyi sınırlayıcı kutu performansı elde edilmektedir.

Çizelge 6.11 : TILDA ağırlıkları ile eğitilen YOLOv8 sınırlayıcı maske sonuçları.

Veri Seti	Sınıf	Kesinlik	Duyarlılık	Sınırlayıcı Maske		
				F1 Skoru	mAP50	mAP50-95
Koyu Kumaş	Delik	0.986	1	0.992	0.995	0.86
	Objekt	0.92	1	0.958	0.995	0.674
	Yağ Lekesi	0.96	1	0.979	0.995	0.712
	İplik Hatası	1	0.847	0.917	0.995	0.393
Açık Kumaş	Delik	0.755	0.777	0.765	0.945	0.317
	Objekt	1	0.769	0.869	0.995	0.771
	Yağ Lekesi	0.889	1	0.941	0.995	0.699
	İplik Hatası	0.652	0.667	0.659	0.654	0.402

Çizelge 6.11'ya baktığımızda önceden eğitilmiş model ağırlıkları kullanıldığında çizelge 6.9'e göre çok daha üstün bir sınırlayıcı kutu performansı elde ettiğimiz görülmektedir. Açık Kumaş veri seti, "Objekt" hatası sınıfında, kutu ve maske tahmini için mükemmel yakın kesinlik ve duyarlılık oranları sergilemiş, bu da modelin objeleri tespit etmede hatasız olduğunu ima etmektedir. Bununla birlikte, açık kumaş veri seti, özellikle "İplik Hatası" ve "Delik" sınıfında daha düşük performans göstermiştir ki bu da modelin bu tür hataları tespit etme konusunda bazı zorluklar yaşadığını ortaya koymaktadır. Özellikle Koyu Kumaş veri seti için modelin sınırlayıcı kutu ve maske tahminlerinde çok yüksek kesinlik ve duyarlılık oranları gösterdiği görülmektedir. Bu modelin renkli kumaşlar üzerinde daha iyi çalıştığını düşündürmektedir. Ancak, tüm veri setleri için daha yüksek IoU eşikleri kullanıldığında (Box mAP50-95 ve Mask mAP50-95), modelin performansında bir

düşüş gözlemlenmektedir. Bu, modelin daha kesin ve dar tanımlı alanlarda hataları tespit etmede zorlandığını göstermektedir. Maske tabanlı tahminlerin, genellikle kutu tabanlı tahminlere göre daha benzer performans gösterdiği dikkat çekmektedir. Bu, modelin hataların sadece konumunu değil, aynı zamanda şeklini de anlamada da etkili olduğunu ve maske tabanlı segmentasyonun bu tür veri setleri için daha uygun olabileceğini göstermektedir. Sonuçlar, modelin farklı arka plan ve doku koşullarına adaptasyon kabiliyetinin, veri setinin niteliğine göre değişkenlik gösterebileceğini işaret etmektedir.

Bu çalışmada, TILDA görüntüsü ile eğitilmiş bir YOLOv8 modeli, bu çalışmada oluşturulan açık ve koyu renk kumaş veri setleri üzerinde test edilmiştir. Amaç, benzer hatalara sahip veri setlerinde modelin renk ve desen farklılıklarına karşı ne kadar başarılı olduğunu belirlemektir. Eğitim aşamasında modele yeni veri seti dahil edilmediğinden, modelin yeni kumaş türlerine uyum sağlama yeteneği de gözlemlenmiştir.

Çizelge 6.12 : TILDA ağırlıkları ile yeni kumaş sınırlayıcı kutu sonuçları.

Veri Seti	Sınıf	Kesinlik	Duyarlılık	Sınırlayıcı Kutu		
				F1 Skoru	mAP50	mAP50-95
Koyu Kumaş	Delik	0.316	0.467	0.376	0.283	0.243
	Obje	0.12	0.25	0.162	0.298	0.258
	Yağ Lekesi	0.825	0.75	0.785	0.912	0.696
	İplik Hatası	1	0	0	0.118	0.103
Açık Kumaş	Delik	0.657	0.486	0.558	0.637	0.255
	Obje	0.0984	0.5	0.164	0.285	0.254
	Yağ Lekesi	0.357	0.75	0.483	0.352	0.183
	İplik Hatası	1	0	0	0.249	0.128

Çizelge 6.12'ye göre koyu kumaş veri setinde, yağ lekeleri hataları diğer hatalara kıyasla daha üstün bir şekilde tespit edilebilmektedir. Açık kumaş veri setinde ise hem yağ lekeleri hem de delik hataları nispeten daha iyi sonuçlar vermektedir. Her iki kumaş türünde de iplik hatası sınırlayıcı kutu tespiti yapılamamıştır.

Çizelge 6.13 : TILDA ağırlıkları ile yeni kumaş sınırlayıcı maske sonuçları.

Veri Seti	Sınıf	Sınırlayıcı Maske				
		Kesinlik	Duyarlılık	F1 Skoru	mAP50	mAP50-95
Koyu Kumaş	Delik	0.316	0.467	0.376	0.283	0.255
	Obje	0.12	0.25	0.162	0.298	0.217
	Yağ Lekesi	0.825	0.75	0.785	0.912	0.578
	İplik Hatası	1	0	0	0.0184	0.011
Açık Kumaş	Delik	0.35	0.25	0.291	0.413	0.219
	Obje	0.0995	0.5	0.165	0.285	0.0387
	Yağ Lekesi	0.361	0.75	0.487	0.321	0.135
	İplik Hatası	1	0	0	0.249	0.0781

Çizelge 6.13'te görüldüğü üzere sınırlayıcı kutu tespitinde olduğu gibi, sınırlayıcı maske tespiti de yeterince başarılı olmamıştır. Koyu kumaş türünde yağ lekeleri başarılı bir şekilde tespit edilebilmektedir. Bu durum, modelin kumaş renginden bağımsız olarak yağ lekesi hata tespiti yapabildiğini göstermektedir. Fakat diğer kumaş hatalarında aynı başarı elde edilememiştir. Model, hiç görmediği yeni kumaş türlerinde, renk ve desen farklılıklarında istenilen performansı yakalayamamıştır.

Çizelge 6.14 : Farklı kumaşlarda toplam hataların sınırlayıcı kutu sonuçları.

Veri Seti	Özellik	Sınıf	Sınırlayıcı Kutu				
			Kesinlik	Duyarlılık	F1-Skor	mAP50	mAP50-95
TILDA	İpek Kumaş	Hepsi	0.821	0.74	0.778	0.81	0.512
Koyu Kumaş Verisi	Yüksek Çözünürlüklü Kamera	Hepsi	0.923	0.971	0.946	0.986	0.824
	ESRGAN Modeli	Hepsi	0.897	0.907	0.902	0.97	0.764
	Düşük Çözünürlüklü Kamera	Hepsi	0.796	0.658	0.720	0.778	0.524
Açık Kumaş Verisi	Yüksek Çözünürlüklü Kamera	Hepsi	0.922	0.931	0.926	0.957	0.696
	ESRGAN Modeli	Hepsi	0.755	0.853	0.801	0.856	0.505
	Düşük Çözünürlüklü Kamera	Hepsi	0.57	0.75	0.648	0.793	0.549

Çizelge 6.14’te farklı kumaş türleri için tüm hata sınıflarının toplam sonuçları gösterilmektedir. TILDA veri seti, düşük ve yüksek çözünürlüklü kamera ile elde edilen yeni kumaş veri setlerinin sınırlayıcı kutu başarımları gösterilmektedir. Yüksek çözünürlüklü kameradan elde edilen görüntülerin tekrardan boyutlandırılarak 1664x1664 çözünürlük seviyesine getirilmektedir. Elde edilen koyu ve açık kumaş verileri 416x416 bölütlere ayrılarak YOLOv8 modeli ile sonuç alınmaktadır. Buna karşılık düşük çözünürlüklü kameradan elde edilen verilerde ise 640x480 çözünürlükten tekrardan 416x416’ya boyutlandırılmaktadır. Sonuçlara bakıldığında yüksek çözünürlüklü ve yamalara ayrılmış veri seti yüksek sınırlayıcı kutu sonuçları elde etmektedir. Önerilen ESRGAN yöntemi ise düşük çözünürlüklü kameraya göre çok daha iyi sınırlayıcı kutu sonucu vermektedir.

Çizelge 6.15 : Farklı kumaşlarda toplam hataların sınırlayıcı maske sonuçları.

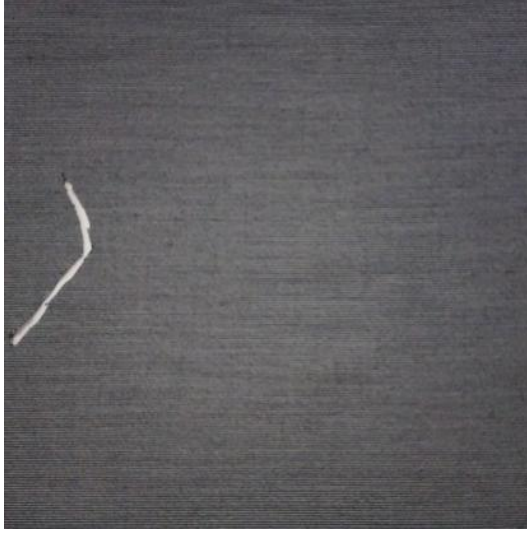
Veri Seti	Özellik	Sınıf	Kesinlik	Duyarlılık	Sınırlayıcı Maske		
					F1-Skor	mAP50	mAP50-95
TILDA	İpek Kumaş	Hepsi	0.789	0.706	0.745	0.744	0.354
Koyu Kumaş Verisi	Yüksek Çözünürlüklü Kamera	Hepsi	0.976	0.929	0.952	0.982	0.7
	ESRGAN Modeli	Hepsi	0.897	0.897	0.897	0.922	0.652
	Düşük Çözünürlüklü Kamera	Hepsi	0.796	0.658	0.720	0.769	0.427
Açık Kumaş Verisi	Yüksek Çözünürlüklü Kamera	Hepsi	0.958	0.966	0.962	0.995	0.7
	ESRGAN Modeli	Hepsi	0.734	0.725	0.729	0.707	0.506
	Düşük Çözünürlüklü Kamera	Hepsi	0.57	0.75	0.648	0.755	0.457

Çizelge 6.15’te farklı kumaş türleri için tüm hata sonuçlarının toplamı gösterilmektedir. TILDA veri seti, düşük ve yüksek çözünürlüklü kamera ile elde edilen yeni kumaş veri setlerinin sınırlayıcı maske başarımları gösterilmektedir. Çizelge 6.14 ve çizelge 6.15’te sunulan veriler, bilgisayarlı görme

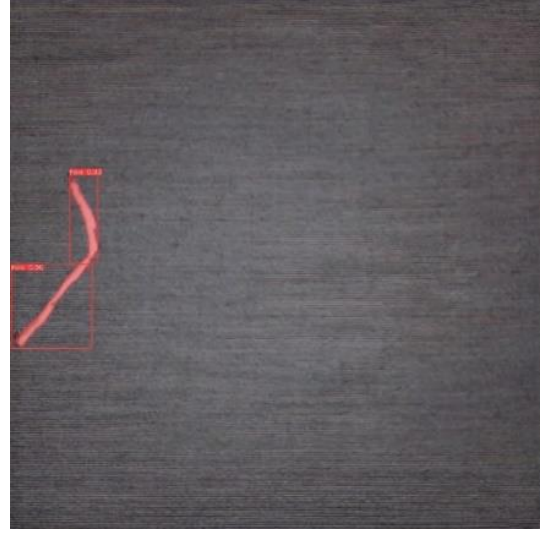
alanında özel bir kumaş hatası verisi olan TILDA'nın ve kendi veri setlerimiz olan Açık Kumaş ve Koyu Kumaş adı veri seti bulunmaktadır. Üç ayrı veri seti ve iki farklı kamera üzerinde gerçekleştirdiğimiz, sınırlayıcı kutu ve maske tahminlerinin doğruluğunu ve kapsamını ölçen bir dizi metrik kullanılmıştır. Sonuçlara bakıldığında yüksek çözünürlüklü ve yamalara ayrılmış veri seti yüksek sınırlayıcı maske sonuçları elde etmektedir. Önerilen ESRGAN yöntemi ise düşük çözünürlüklü kameraya göre çok daha iyi sınırlayıcı maske sonucu vermektedir. Koyu Kumaş veri setinin sonuçları özellikle dikkate değerdir; neredeyse tüm sınıflar ve metriklerde başarılı bir skorlar elde edilmiştir. Bu, modelin koyu renkli zeminler üzerindeki hataları algılama konusunda aşırı derecede yetkin olduğunu göstermektedir. Bu yüksek performans, özellikle hata algılama modellerinin zorlu renk koşullarına uyum sağlama yeteneği açısından önem taşımaktadır. Genel olarak, modelin mAP50-95 değerleri, tüm veri setleri ve sınıflar için daha düşük çıkmıştır, bu da modelin yüksek IoU eşliğinde tespit performansının azaldığını göstermektedir. Ayrıca, kutu tabanlı tahminler genellikle maske tabanlı tahminlerden daha yüksek skorlar elde etmiştir, bu da modelin şekil ve doku gibi daha detaylı özellikleri anlama konusunda daha başarısız olduğunu düşündürmektedir. Sonuç olarak, bu değerlendirme, hata algılama modelinin farklı veri setlerinde nasıl farklılaşan performanslar sergilediğini göstermektedir. Kamera çözünürlük ve bütçe farklılıkları önerilen yöntem ile nasıl kapatılacağı gösterilmektedir. Bu farklılıklar, modelin gerçek dünya senaryolarında nasıl uygulanabileceğine dair önemli bilgiler sağlamaktadır ve bu da geliştiriciler için uygulama geliştirme sürecinde dikkate alınması gereken önemli bir husustur.

6.6.1 Koyu kumaş yeniden yapılandırılan görsel sonuçları

Öncesinde TILDA veri seti ile eğitilmiş YOLOv8 modelinin ağırlıklarını düşük çözünürlüğe sahip kamera ile elde edilen koyu kumaş veri setine uygulanarak, tekrardan YOLOv8 modeli üzerinde eğitim işlemlerini uygulamaktayız. Yamalara ayrılarak tespit edilen bölütlere ayrılmış hata görüntülerini tekrar yapılandırdıktan sonra sonuç görüntüsü verilmektedir.



(a)



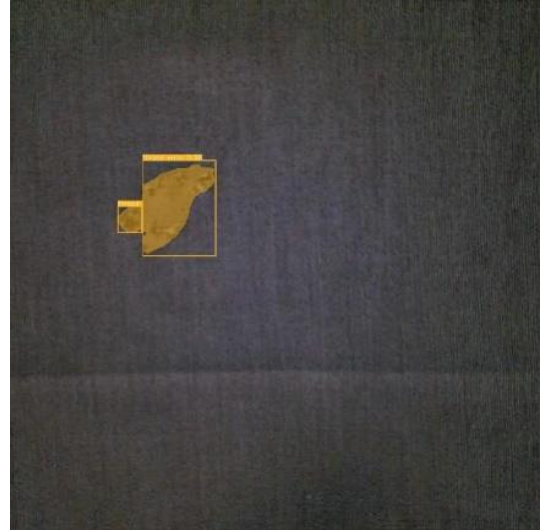
(b)

Şekil 6.26 : (a) Koyu Kumaş verisinde bulunan orijinal delik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen delik hatası görüntüsü.

Şekil 6.26’de düşük çözünürlüğe sahip kamera ile elde edilen koyu kumaş verisine ait görüntüler gösterilmektedir. 6.26.a’daki görüntü orijinal görüntüyü temsil etmektedir. 6.26.b’deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, delik hatası görüntüsünü temsil etmektedir.



(a)



(b)

Şekil 6.27 : (a) Koyu Kumaş verisinde bulunan orijinal iplik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen iplik hatası görüntüsü.

Şekil 6.27’de TILDA verisine ait görüntüler gösterilmektedir. 6.27.a’daki görüntü orijinal görüntüyü temsil etmektedir. 6.27.b’deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, iplik hatası görüntüsünü temsil etmektedir.



(a)



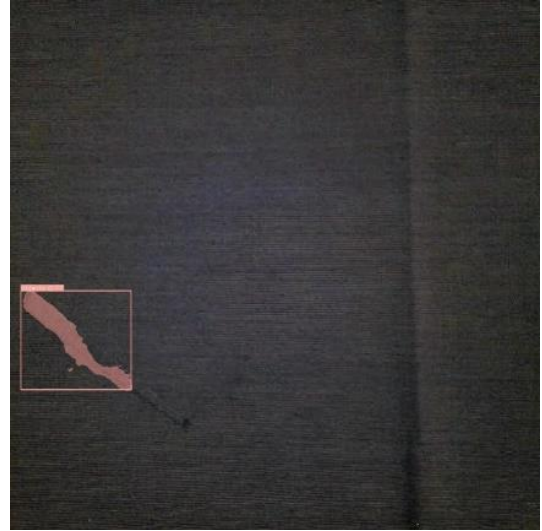
(b)

Şekil 6.28 : (a) Koyu Kumaş verisinde bulunan orijinal yağ lekesi hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen iplik hatası görüntüsü.

Şekil 6.28’de TILDA verisine ait görüntüler gösterilmektedir. 6.28.a’daki görüntü orijinal görüntüyü temsil etmektedir. 6.28.b’deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, yağ lekesi hatasının görüntüsünü temsil etmektedir.



(a)



(b)

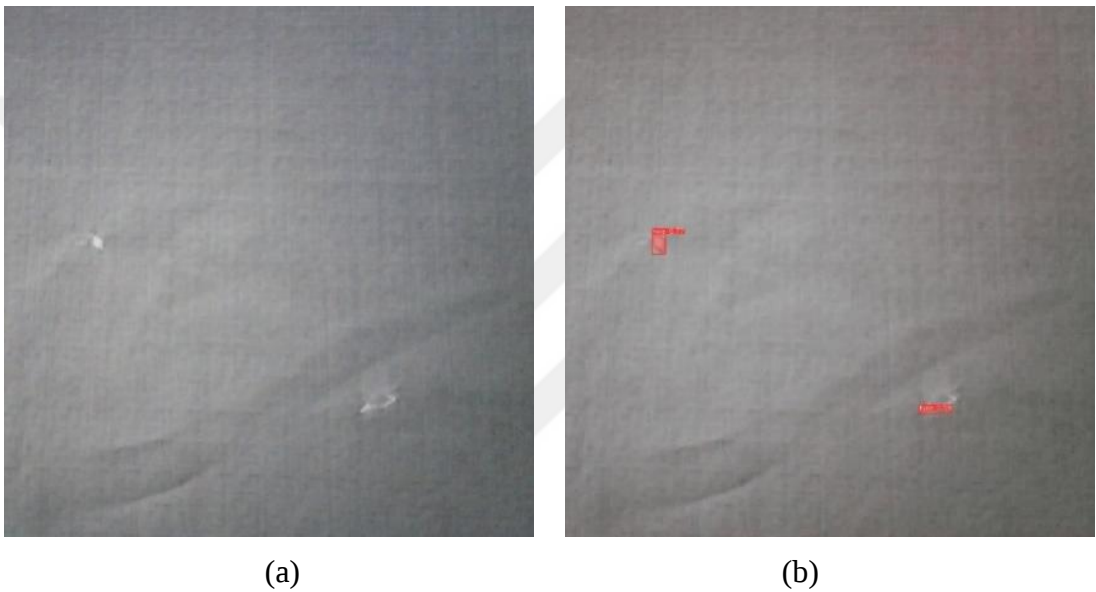
Şekil 6.29 : (a) Koyu Kumaş verisinde bulunan orijinal obje hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen obje hatası görüntüsü.

Şekil 6.29’te TILDA verisine ait görüntüler gösterilmektedir. 6.29.a’daki görüntü orijinal görüntüyü temsil etmektedir. 6.29.b’deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, obje hatasının görüntüsünü temsil etmektedir. Özellikle koyu kumaş üzerindeki hatalarda çok daha başarılı bir

sonuç alındığı görsel sonuçlar üzerinde gözlemlenmektedir. Şekil 6.29'te ise obje hatasının bir kısmını tahmin edemediği gözlemlenmektedir.

6.6.2 Açık kumaş yeniden yapılandırılan görsel sonuçları

Öncesinde TILDA veri seti ile eğitilmiş YOLOv8 modelinin ağırlıklarını düşük çözünürlüğe sahip kamera ile elde edilen açık kumaş veri setine uygulanarak, tekrardan YOLOv8 modeli üzerinde eğitim işlemlerini uygulamaktayız. Yamalara ayrılarak tespit edilen bölütlere ayrılmış hata görüntülerini tekrar yapılandırdıktan sonra sonuç görüntüsü verilmektedir.

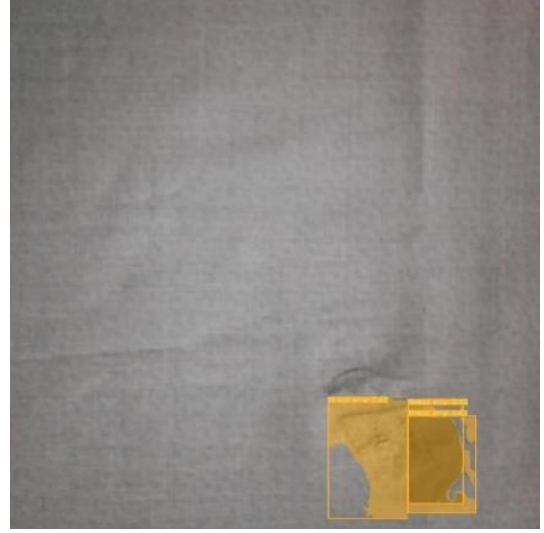


Şekil 6.30 : (a) Açık Kumaş verisinde bulunan orijinal delik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen delik hatası görüntüsü.

Şekil 6.30'te Açık Kumaş verisine ait görüntüler gösterilmektedir. 6.30.a'daki görüntü orijinal görüntüyü temsil etmektedir. 6.30.b'deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, delik lekeli hatasının görüntüsünü temsil etmektedir.



(a)



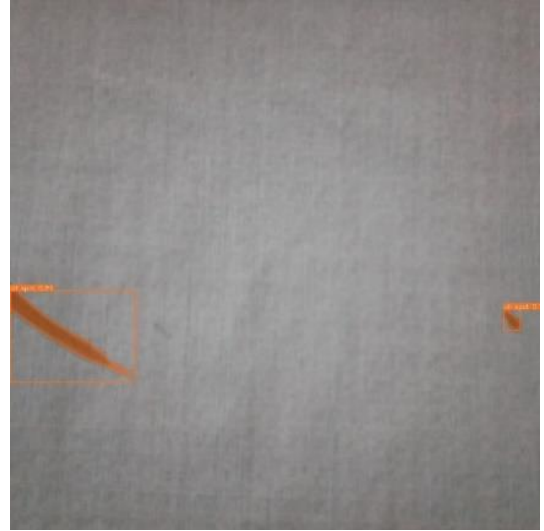
(b)

Şekil 6.31 : (a) Açık Kumaş verisinde bulunan orijinal iplik hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen iplik hatası görüntüsü.

Şekil 6.31’te Açık Kumaş verisine ait görüntüler gösterilmektedir. 6.31.a’daki görüntü orijinal görüntüyü temsil etmektedir. 6.31.b’deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, iplik hatasının görüntüsünü temsil etmektedir.



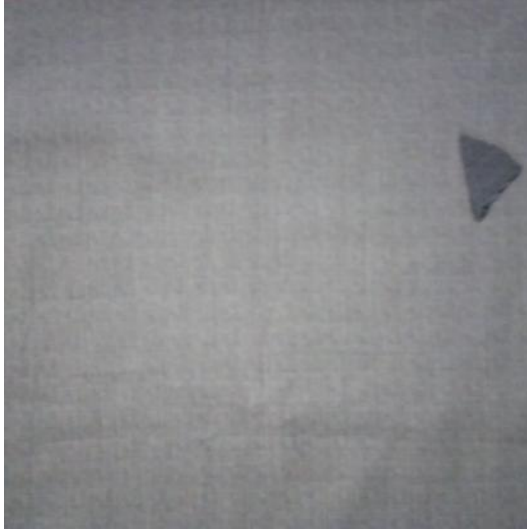
(a)



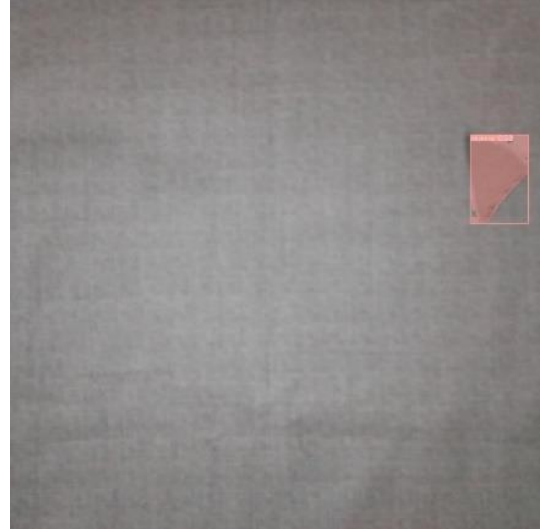
(b)

Şekil 6.32 : (a) Açık Kumaş verisinde bulunan orijinal yağ lekesi hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen yağ lekesi hatası görüntüsü.

Şekil 6.32’de Açık Kumaş verisine ait görüntüler gösterilmektedir. 6.32.a’daki görüntü orijinal görüntüyü temsil etmektedir. 6.32.b’deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, yağ lekesi hatasının görüntüsünü temsil etmektedir.



(a)



(b)

Şekil 6.33 : (a) Açık Kumaş verisinde bulunan orijinal obje hatası görüntüsü, (b) YOLOv8 sonrası yeniden yapılandırılmış üretilen obje hatası görüntüsü.

Şekil 6.33'te Açık Kumaş verisine ait görüntüler gösterilmektedir. 6.33.a'daki görüntü orijinal görüntüyü temsil etmektedir. 6.33.b'deki görüntü ise YOLOv8 modelinden sonuçlar elde edildikten sonra yeniden yapılandırılmış, obje hatası görüntüsünü temsil etmektedir. Özellikle açık kumaş üzerindeki delik ve iplik hatalarında koyu kumaştaki başarıyı yakalayamadığını gözlemlemekteyiz. Şekil 6.30'teki örnekte delik hatasının bir kısmını neredeyse bulamadığını gözlemlemekteyiz.

7. SONUÇLAR VE ÖNERİLER

Bu tezde, TILDA ve YVS veri setleri kullanılarak, önerilen yöntem ile kumaş kusuru tespiti işlemi gerçekleştirilmiştir. Ayrıca bu çalışma özelinde YVS veri seti iki farklı özellik içerecek şekilde üretilmiştir. Bir kısmı açık bir renk tonuna sahipken, diğerleri sadece koyu renk tonludur. Özellikle, düşük maliyetli ve düşük çözünürlüklü kameraların kullanımıyla yüksek çözünürlüklü kameralara kıyasla benzer performans elde edilebilmesi, endüstriyel uygulamalar için önemli bir adımdır. ESRGAN modelinin kullanımıyla düşük çözünürlüklü görüntülerin kalitesinin artırılması, endüstriyel kalite kontrolünde yeni ve maliyet etkin bir yöntemin belirlenmesini sağlamıştır. Eğitim başlangıç aşamasında görüntülerin boyutları 416x416'dan 104x104 piksele düşüren stratejik bir yeniden boyutlandırma uygulanmaktadır.

Önerilen ESRGAN modeli sayesinde, eğitim aşamasında görüntülerin çözünürlüklerinin 4 kat büyütülmesi öğretilmiştir. Eğitim aşamasında 104x104 boyuttan 416x416 piksele yeniden boyutlandırılan görüntüler, sayesinde elde edilen model ağırlıkları kullanılarak test aşamasında 416x416 boyutunda olan görüntüler 4 kat daha büyütülerek 1664x1664 boyutlarında yen, görüntüler elde edilmiştir. Bu 4 kat yükseltmenin görüntü kalitesinden ödün vermemesi ve görsel doğruluğun her iki kumaş verisinde de oldukça tutarlı kalması dikkat çekici olmaktadır.

Çözünürlük iyileştirmesinin ardından, görüntülere bölütleme teknikleri kullanılmıştır. Yapılan bu işlem gözle görülmeyecek kadar küçük bölgesel kusur hatalarının YOLOv8 modeli tarafından incelenmesini kolaylaştırmaktadır. YOLOv8 modeli ve ESRGAN modelinin başarıyla entegre edilerek kumaş kusurlarının tespitinde önemli bir ilerleme kaydedilmiştir. Ön değerlendirmeler görüntülerde küçük yağ lekelerinin bulunmadığını göstermektedir; sonuçlara bakıldığında ise bu kusurların fark edilebilirliğinde belirgin bir iyileşme olduğunu ortaya çıkmaktadır.

Kusur tespit mekanizmalarının etkinliğini artırmak için bir dizi görüntü işleme algoritması uygulandı. Görüntü işleme metotlarına rağmen hiçbir yaklaşım tüm kusur

kategorilerinde orijinal görüntülerden kesin olarak daha iyi performans göstermemektedir. Özellikle, orijinal görüntüler, obje ve yağ lekesi kusurlarının tespitinde üstün performans ölçümleri sergilemektedir. Buna karşılık, görüntülerin doğrusal kontrast germesi, delik ve iplik anormalliklerini belirlemede daha başarılı sonuçlar vermektedir. Üstün bir performans farkı olmadığı için kendi veri setlerimizde orijinal görüntülerin kullanıldığı modellerin ağırlıklarını kullanmayı tercih ettik. TILDA veri seti ile %81'lik sınırlayıcı kutu ve %74'lük maskeleme mAP50 değerine ulaşan model, kaydedilen bu ağırlıklar ile yapılan bu ağırlık kullanım işlemi sonuçları etkin bir şekilde iyileştirmiştir. Koyu renkli kumaş veri kümesinin %99'lık bir sınırlayıcı kutu ve maske tanımlama mAP50 değeri bulduğu, açık renkli kumaş veri kümesinin ise %90'lık bir sınırlayıcı kutu ve maske tanımlama mAP50 değeri sağladığı görülmüştür.

Düşük çözünürlüklü kameraların tercih edilmesinin ardındaki ana motivasyon ise yüksek çözünürlüklü kamera ve lens ile düşük çözünürlüklü kameralar arasındaki maliyet farkının 5 katına yakın olmasıdır. Özellikle, ESRGAN modeli gibi görüntü iyileştirme tekniklerinin kullanılmasıyla, düşük çözünürlüklü kameradan elde edilen görüntülerin kalitesinin artırılması ve yüksek çözünürlüklü kameraların sağladığı performansla benzer sonuçlar elde edilmesi mümkün hale gelmektedir.

Gerçek zamanlı çalışmada ise Raspberry Pi 4b gibi donanımların yeterli gelmediği gözlenmektedir. Bu tarz çalışmalarda model boyutlarının fazla olması CPU ile çalışan donanımsal cihazlarda uzun bekleme sürelerine neden olmaktadır. Bu nedenle ESRGAN modelinin üretici kısmındaki RRDB blokları ve kanal sayısında azaltmaya gidilerek, orijinal modelin performansına yakın bir başarımla elde edilmiştir. Kanal sayısındaki azalması ile RRDB bloklarının ve üretici modelin öğrenme oranları da düşmektedir. Bu sebep ile RRDB blok sayısında optimizasyon yapılarak, istenilen model hız ve başarımla performansı elde edilmiştir.

Model, renk farklılaştırma konusunda yeterlilik gösterirken, özellikle kumaş tasarımlarında desen tanıma konusunda fark edilebilir iyileştirme fırsatları devam etmektedir. Modelin daha genelleştirilebilmesi için gelecekte farklı kumaş türlerini, farklı aydınlatma koşullarını ve çeşitli kusur tiplerini içeren daha geniş ve çeşitli bir veri seti oluşturulabileceği düşünülmektedir. Ek olarak GPU'ya sahip uç cihazların kullanılmasıyla birlikte gelecekteki çalışmalarda yüksek parametre sayısına sahip

modellerin optimize edilerek, gerçek zamanlı çalışmaya çok daha iyi uyum sağlanacağı düşünülmektedir.

RRDB bloklarının sayısının belirlenmesi, genellikle deneysel olarak yapılmakta ve deneme yanılma yoluyla optimize edilmektedir. Ancak, gelecek çalışmalarda bu süreci iyileştirmek ve otomatikleştirmek için RRDB blok sayılarının sezgisel algoritmalar ile belirlenmesi için çeşitli yöntemler geliştirilmesi düşünülmektedir.

Modelin hesaplama maliyetini azaltmak için çeşitli stratejiler kullanılabilir. Örneğin, modeldeki bağlantı sayısını azaltarak veya Pruning teknikleri kullanarak, modeldeki gereksiz veya düşük katkı sağlayan parametreleri kaldırarak hesaplama maliyeti azaltılabilir. Bunun yanı sıra, Kuantizasyon teknikleri uygulanarak, modeldeki parametreler daha az bellek kullanacak şekilde düşük hassasiyetli sayılara dönüştürülebilir ve böylece hesaplama maliyeti azaltılabilir. Ayrıca, Depthwise separable convolution gibi teknikler, geleneksel convolution operasyonlarına kıyasla daha hafif ve daha az hesaplama maliyetine sahip olduğundan, bu tür teknikler modelin hesaplama maliyetini azaltmak için kullanılabilir.

Optimize edilmiş ESRGAN modelinin çıktı görüntüleri arasındaki kalite farklılıklarını etkili bir şekilde değerlendirmek için yeni metodolojiler geliştirilmesi planlanmaktadır. Bu yeni metodolojilerin odak noktası, görüntüler arasındaki kalite farklarını belirleme yeteneğini artırmak ve bu farklılıkları insan görsel algısıyla daha yakın şekilde eşleştirmektir. Bu amaçla, iki ana yaklaşım öne çıkmaktadır: İlk olarak, frekans alanında karşılaştırma metriklerinin geliştirilmesi planlanmaktadır. Bu metrikler, görüntüler arasındaki frekans bileşenlerini karşılaştırarak, görüntülerin dokusu ve çözünürlük farklılıklarını belirleme amacını taşımaktadır. Bu sayede, model çıktıları arasındaki kalite farklılıkları daha hassas bir şekilde tanımlanabilir. İkinci olarak, LIPS benzeri derin öğrenme tabanlı karşılaştırma metriklerinin geliştirilmesi hedeflenmektedir. Bu metrikler, derin öğrenme tekniklerini kullanarak görüntüler arasındaki algısal benzerliği değerlendirmeyi amaçlamaktadır. Bu sayede, insan görsel algısına daha yakın sonuçlar elde edilebilir ve model çıktıları arasındaki kalite farklılıkları daha doğru bir şekilde ölçülebilir. Bu yeni metodolojilerin geliştirilmesi, optimize edilmiş ESRGAN modelinin kalite değerlendirme sürecini iyileştirecek ve endüstriyel uygulamalarda daha etkili bir kalite kontrol sağlayacaktır.

KAYNAKLAR

- Andersen, I.** (2021, October). *TILDA Fabric Dataset*. *Roboflow Universe*. Retrieved from <https://universe.roboflow.com/irvin-andersen/tilda-fabric>
- Ashiba, H. I., & Ashiba, M. I.** (2023). Novel proposed technique for automatic fabric defect detection. *Multimedia Tools and Applications*, 1-24.
- Bai, Z., & Jing, J.** (2023). Mobile-Deeplab: a lightweight pixel segmentation-based method for fabric defect detection. *Journal of Intelligent Manufacturing*, 1-16.
- Cui, Y., Zhou, F., Wang, J., Liu, X., Lin, Y., & Belongie, S.** (2017). Kernel pooling for convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2921-2930).
- Deotale, N. T., & Sarode, T. K.** (2019). Fabric defect detection adopting combined GLCM, Gabor wavelet features and random decision forest. *3D Research*, 10, 1-13.
- Dubey, A. K., & Jain, V.** (2019). Comparative study of convolution neural network's relu and leaky-relu activation functions. In *Applications of Computing, Automation and Wireless Systems in Electrical Engineering: Proceedings of MARC 2018* (pp. 873-880). Springer Singapore.
- Dunne, R. A., & Campbell, N. A.** (1997, June). On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function. In *Proc. 8th Aust. Conf. on the Neural Networks*, Melbourne (Vol. 181, p. 185). Citeseer.
- Géron, A.** (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. "O'Reilly Media, Inc."
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y.** (2020). *Generative adversarial networks*. *Communications of the ACM*, 63(11), 139-144.
- Han, J., & Moraga, C.** (1995, June). The influence of the sigmoid function parameters on the speed of backpropagation learning. In *International workshop on artificial neural networks* (pp. 195-201). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Hastie, T., Tibshirani, R., & Friedman, J.** (2009). Unsupervised learning. In *The elements of statistical learning* (pp. 485-585). Springer, New York, NY.
- Hess, A.** (2018). Exploring feature reuse in DenseNet architectures. *arXiv preprint arXiv:1806.01935*.

- Hosang, J., Benenson, R., & Schiele, B.** (2017). Learning non-maximum suppression. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4507-4515).
- Hsiao, T. Y., Chang, Y. C., Chou, H. H., & Chiu, C. T.** (2019). Filter-based deep-compression with global average pooling for convolutional networks. *Journal of Systems Architecture*, 95, 9-18.
- Hu, G., Huang, J., Wang, Q., Li, J., Xu, Z., & Huang, X.** (2020). Unsupervised fabric defect detection based on a deep convolutional generative adversarial network. *Textile Research Journal*, 90(3-4), 247-270.
- Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q.** (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700-4708).
- Huang, Y., & Xiang, Z.** (2022). RPDNet: Automatic fabric defect detection based on a convolutional neural network and repeated pattern analysis. *Sensors*, 22(16), 6226.
- Johnson, J., Alahi, A., & Fei-Fei, L.** (2016). Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14* (pp. 694-711). Springer International Publishing.
- Jing, J. F., Ma, H., & Zhang, H. H.** (2019). Automatic fabric defect detection using a deep convolutional neural network. *Coloration Technology*, 135(3), 213-223.
- Jing, J., Wang, Z., Rättsch, M., & Zhang, H.** (2022). Mobile-Unet: An efficient convolutional neural network for fabric defect detection. *Textile Research Journal*, 92(1-2), 30-42.
- Jing, J., Zhuo, D., Zhang, H., Liang, Y., & Zheng, M.** (2020). Fabric defect detection using the improved YOLOv3 model. *Journal of engineered fibers and fabrics*, 15, 1558925020908268.
- Kumar, A.** (2008). Computer-vision-based fabric defect detection: A survey. *IEEE transactions on industrial electronics*, 55(1), 348-363.
- LeCun, Y., Bengio, Y., & Hinton, G.** (2015). Deep learning. *nature*, 521(7553), 436-444.
- Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., ... & Shi, W.** (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4681-4690).
- Li, H., Li, J., Guan, X., Liang, B., Lai, Y., & Luo, X.** (2019, December). Research on overfitting of deep learning. In *2019 15th international conference on computational intelligence and security (CIS)* (pp. 78-81). IEEE.
- Li, Y., & Yuan, Y.** (2017). Convergence analysis of two-layer neural networks with relu activation. *Advances in neural information processing systems*, 30.

- Li, Y., Zhang, D., & Lee, D. J.** (2019). Automatic fabric defect detection with a wide-and-compact network. *Neurocomputing*, 329, 329-338.
- Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S.** (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117-2125).
- Liu, J., Wang, C., Su, H., Du, B., & Tao, D.** (2019). Multistage GAN for fabric defect detection. *IEEE Transactions on Image Processing*, 29, 3388-3400.
- Liu, Q., Wang, C., Li, Y., Gao, M., & Li, J.** (2022). A fabric defect detection method based on deep learning. *IEEE access*, 10, 4284-4296.
- Liu, Z., Zhang, C., Li, C., Ding, S., Dong, Y., & Huang, Y.** (2019). Fabric defect recognition using optimized neural networks. *Journal of Engineered Fibers and Fabrics*, 14, 1558925019897396.
- Mahajan, P. M., Kolhe, S. R., & Patil, P. M.** (2009). A review of automatic fabric defect detection techniques. *Advances in Computational Research*, 1(2), 18-29.
- Mei, S., Wang, Y., & Wen, G.** (2018). Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model. *Sensors*, 18(4), 1064.
- Ngan, H. Y., Pang, G. K., & Yung, N. H.** (2011). Automated fabric defect detection—A review. *Image and vision computing*, 29(7), 442-458.
- O'Shea, K., & Nash, R.** (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.
- Peng, P., Wang, Y., Hao, C., Zhu, Z., Liu, T., & Zhou, W.** (2020). Automatic fabric defect detection method using PRAN-net. *Applied Sciences*, 10(23), 8434.
- Priya, S., Kumar, T. A., & Paul, V.** (2011, July). A novel approach to fabric defect detection using digital image processing. In *2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies* (pp. 228-232). IEEE.
- Raheja, J. L., Kumar, S., & Chaudhary, A.** (2013). Fabric defect detection based on GLCM and Gabor filter: A comparison. *Optik*, 124(23), 6469-6474.
- Rasheed, A., Zafar, B., Rasheed, A., Ali, N., Sajid, M., Dar, S. H., ... & Mahmood, M. T.** (2020). Fabric defect detection using computer vision techniques: a comprehensive review. *Mathematical Problems in Engineering*, 2020, 1-24.
- Rebhi, A., Abid, S., & Fnaiech, F.** (2016, November). Fabric defect detection using local homogeneity and morphological image processing. In *2016 International Image Processing, Applications and Systems (IPAS)* (pp. 1-5). IEEE.

- Reza, A. M.** (2004). Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. *Journal of VLSI signal processing systems for signal, image and video technology*, 38, 35-44.
- Rong-qiang, L., Ming-hui, L., Jia-chen, S., & Yi-bin, L.** (2021, June). Fabric defect detection method based on improved u-net. In *Journal of Physics: Conference Series* (Vol. 1948, No. 1, p. 012160). IOP Publishing.
- Santurkar, S., Tsipras, D., Ilyas, A., & Madry, A.** (2018). How does batch normalization help optimization?. *Advances in neural information processing systems*, 31.
- Sheikh, H. R., & Bovik, A. C.** (2005, January). A visual information fidelity approach to video quality assessment. In *The first international workshop on video processing and quality metrics for consumer electronics* (Vol. 7, No. 2, pp. 2117-2128). sn.
- Shi, M., Fu, R., Guo, Y., Bai, S., & Xu, B.** (2011). Fabric defect detection using local contrast deviations. *Multimedia tools and applications*, 52, 147-157.
- Shrestha, A., & Mahmood, A.** (2019). Review of deep learning algorithms and architectures. *IEEE access*, 7, 53040-53065.
- Singh, N., Tewari, V. K., Biswas, P. K., Pareek, C. M., & Dhruw, L. K.** (2021). Image processing algorithms for in-field cotton boll detection in natural lighting conditions. *Artificial Intelligence in Agriculture*, 5, 142-156.
- Song, S., Jing, J., Huang, Y., & Shi, M.** (2021). EfficientDet for fabric defect detection based on edge computing. *Journal of Engineered Fibers and Fabrics*, 16, 15589250211008346.
- Şeker, A., Peker, K. A., Yürksek, A. G., & Delibaş, E.** (2016, May). Fabric defect detection using deep learning. In *2016 24th Signal Processing and Communication Application Conference (SIU)* (pp. 1437-1440). IEEE.
- Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M.** (2021). Scaled-yolov4: Scaling cross stage partial network. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition* (pp. 13029-13038).
- Wang, K., Gou, C., Duan, Y., Lin, Y., Zheng, X., & Wang, F. Y.** (2017). Generative adversarial networks: introduction and outlook. *IEEE/CAA Journal of Automatica Sinica*, 4(4), 588-598.
- Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., ... & Change Loy, C.** (2018). Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops* (pp. 0-0).
- Wang, Y. E., Wei, G. Y., & Brooks, D.** (2019). Benchmarking TPU, GPU, and CPU platforms for deep learning. *arXiv preprint arXiv:1907.10701*.

- Wei, W., Deng, D., Zeng, L., & Zhang, C.** (2021). Real-time implementation of fabric defect detection based on variational automatic encoder with structure similarity. *Journal of Real-Time Image Processing*, 18, 807-823.
- Wu, J., Le, J., Xiao, Z., Zhang, F., Geng, L., Liu, Y., & Wang, W.** (2021). Automatic fabric defect detection using a wide-and-light network. *Applied Intelligence*, 51(7), 4945-4961.
- Xie, H., & Wu, Z.** (2020). A robust fabric defect detection method based on improved RefineDet. *Sensors*, 20(15), 4260.
- Xu, J., Zhang, Z., Friedman, T., Liang, Y., & Broeck, G.** (2018, July). A semantic loss function for deep learning with symbolic knowledge. In *International conference on machine learning* (pp. 5502-5511). PMLR.
- Yang, J., Shi, Y., & Qi, Z.** (2020). Dfr: Deep feature reconstruction for unsupervised anomaly segmentation. *arXiv preprint arXiv:2012.07122*.
- Yue, X., Wang, Q., He, L., Li, Y., & Tang, D.** (2022). Research on tiny target detection technology of fabric defects based on improved Yolo. *Applied Sciences*, 12(13), 6823.
- Zhang, H. W., Zhang, L. J., Li, P. F., & Gu, D.** (2018, May). Yarn-dyed fabric defect detection with YOLOV2 based on deep convolution neural networks. In *2018 IEEE 7th data driven control and learning systems conference (DDCLS)* (pp. 170-174). IEEE.
- Zhang, X., Wang, Y., Zhang, N., Xu, D., & Chen, B.** (2019). Research on scene classification method of high-resolution remote sensing images based on RFPNet. *Applied Sciences*, 9(10), 2028.
- Zheng, Y., Yang, C., & Merkulov, A.** (2018, May). Breast cancer screening using convolutional neural network and follow-up digital mammography. In *Computational Imaging III* (Vol. 10669, p. 1066905). SPIE.

ÖZGEÇMİŞ

TARANMIŞ
VESİKALIK
FOTOĞRAF

Ad-Soyad : Ahmet METİN

Doğum Tarihi ve Yeri :

E-posta :

ÖĞRENİM DURUMU:

- **Lisans** : 2019, Manisa Celal Bayar Üniversitesi, Hasan Ferdi Turgutlu Teknoloji Fakültesi, Mekatronik Mühendisliği

MESLEKİ DENEYİM VE ÖDÜLLER:

- Devam ediyor, Bursa Teknik Üniversitesi, Bilgisayar Mühendisliği, Araştırma Görevlisi.

YAYINLAR:

Metin, A., Kasif, A., & Catal, C. (2023). Temporal fusion transformer-based prediction in aquaponics. *The Journal of Supercomputing*, 79(17), 19934-19958.