

**T.C.  
SAKARYA ÜNİVERSİTESİ  
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

**BSM 498 BİTİRME ÇALIŞMASI**

**DİJİTAL EKG GÖRÜNTÜLERİNDEN ARİTMİ TESPİTİ  
UYGULAMASI**

**B201210078 – Mustafa BİÇER**

**Fakülte Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ  
Tez Danışmanı : Prof. Dr. Nejat YUMUŞAK**

**2024-2025 Bahar Dönemi**

**T.C.**  
**SAKARYA ÜNİVERSİTESİ**  
**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

**DİJİTAL EKG GÖRÜNTÜLERİNDEN ARİTMİ TESPİTİ**  
**UYGULAMASI**

**BSM 498 - BİTİRME ÇALIŞMASI**

**Mustafa BİÇER**

**Fakülte Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ**

**Bu tez .. / .. / ... tarihinde aşağıdaki jüri tarafından oybirliği / oyçokluğu ile kabul edilmiştir.**

.....  
**Jüri Başkanı**

.....  
**Üye**

.....  
**Üye**

## ÖNSÖZ

Bu tez çalışması, kardiyovasküler hastalıkların erken teşhisi ve sağlık hizmetlerine erişimin artırılması amacıyla, EKG görüntülerinde anomali tespiti için derin öğrenme tabanlı bir mobil sistemin geliştirilmesini kapsamaktadır. Çalışmanın her aşamasında, güncel literatürün rehberliğinde, bilimsel doğruluk ve etik ilkelere bağlı kalınmıştır. Araştırmanın temel motivasyonu, özellikle uzman eksikliğinin yaşandığı bölgelerde, klinisyenlerin karar süreçlerini destekleyecek, hızlı ve güvenilir bir tanı aracı sunmaktır.

Tez sürecinde, başta danışmanım olmak üzere, katkı sağlayan tüm akademisyenlere, proje ekibine ve aileme teşekkür ederim. Ayrıca, açık kaynak topluluklarının sağladığı bilgi ve yazılım altyapısı, bu çalışmanın teknik olarak mümkün olmasında büyük rol oynamıştır. Elde edilen bulguların hem akademik dünyaya hem de sağlık sektörüne fayda sağlaması en büyük temennimdir.

Bu çalışmanın, yapay zekâ ve mobil teknolojilerin tıp alanındaki yenilikçi uygulamalarına katkı sunması ve gelecekteki araştırmalara ilham vermesi dileğiyle.

# İÇİNDEKİLER

SİMGELER VE KISALTMALAR LİSTESİ .....	vii
ŞEKİLLER LİSTESİ.....	viii
DENKLEMLER LİSTESİ .....	ix
TABLolar LİSTESİ.....	ix
ÖZET .....	xi
BÖLÜM 1.....	GİRİŞ 1
1.1. ....	Çalışmanın Amacı ve Kapsamı 1
1.2. ....	Problemin Tanımı ve Önemi 2
1.3. ....	Yenilikçi Yaklaşım 3
1.4. ....	Tezin Yapısı 3
BÖLÜM 2.....	LİTERATÜR TARAMASI 4
2.1. ....	EKG Anomalilerinin Otomatik Tespiti için Kullanılan Mevcut Yöntemler 4
2.1.1. ....	Geleneksel sinyal işleme yaklaşımları 4
2.1.2. ....	Makine öğrenimi tabanlı yaklaşımlar 5
2.1.3. ....	Derin öğrenme tabanlı yaklaşımlar 5
2.2. ....	Medikal görüntü işlemede derin öğrenme uygulamaları 6
2.2.1. ....	Medikal görüntülerde segmentasyon 6
2.2.2. ....	Medikal görüntülerde sınıflandırma ve tespit 6
2.2.3. ....	EKG görüntülerinde derin öğrenme uygulamaları 7
2.2.4. ....	Dikkat mekanizmaları ve görsel transformatörler 7
2.3. ....	YOLO Modellerinin Gelişimi ve Tıbbi Görüntü Analizindeki Uygulamaları 8
2.3.1. ....	YOLO modellerinin gelişimi 8
2.3.2. ....	YOLO Modellerinin tıbbi görüntü analizindeki uygulamaları 9
2.3.3. ....	YOLO Modellerinde aktivasyon fonksiyonları ve mimari değişiklikler 10
2.4. ....	Mikro Servis Mimarisi ve Modern Web Uygulamalarında Güvenlik Protokolleri 11
2.4.1. ....	Mikro servis mimarisinin gelişimi ve prensipleri 11
2.4.2. ....	Mikro servisler arası iletişim protokolleri 12
2.4.3. ....	Modern web uygulamalarında güvenlik protokolleri 13
2.4.4. ....	JWT (JSON Web Token) ve uygulama güvenliği 14
2.5. ....	Çapraz Platform Mobil Uygulama Geliştirme Yaklaşımları 15
2.5.1. ....	Çapraz platform mobil uygulama geliştirme yaklaşımlarının evrimi 15
2.5.2. ....	Flutter framework'ünün mimarisi ve özellikleri 16
2.5.3. ....	Flutter'ın tıbbi uygulamalardaki kullanımı 17
2.5.4. ....	Çapraz platform geliştirmenin avantajları ve zorlukları 18
2.6. ....	Literatür Taraması Özeti 18
BÖLÜM 3.....	MATERYAL VE METOD 20
3.1. ....	Sistem Mimarisi 20
3.1.1. ....	Genel sistem mimarisi ve veri akışı 20
3.1.2. ....	Üç temel bileşen arasındaki etkileşim 21
3.1.3. ....	Mikro servis mimarisi tasarımı ve avantajları 23

3.2.	Veri Seti ve Ön İşleme	24
3.2.1.	Kullanılan EKG veri setinin detayları	24
3.2.2.	Veri ön işleme adımları	26
3.2.3.	Veri setinin eğitim/validasyon/test olarak bölünmesi	27
3.3.	YOLOv9 Modeli Özelleştirmeleri	28
3.3.1.	YOLOv9 Model mimarisi ve çalışma prensibi	28
3.3.2.	Özelleştirilen aktivasyon fonksiyonları ve teorik temelleri	30
3.3.2.1.	TanhKCothx aktivasyon fonksiyonu	30
3.3.2.2.	SinLU Aktivasyon Fonksiyonu	31
3.3.2.3.	Harmonik aktivasyon fonksiyonu	32
3.3.3.	Geliştirilen yeni katmanlar ve mimariye entegrasyonu	33
3.3.3.1.	CBAM (Convolutional block attention module)	33
3.3.3.2.	MBConv (Mobile inverted bottleneck convolution)	34
3.3.3.3.	Enhanced residual block	36
3.3.4.	Optimizasyon fonksiyonu denemeleri ve sonuçları	36
3.3.5.	Yapılan geliştirmelerin kombinasyonları ve test edilmesi	38
3.4.	Mobil Uygulama Geliştirme	44
3.4.1.	Flutter framework'ünün seçilme nedenleri	44
3.4.2.	Kullanıcı arayüzü tasarım prensipleri	45
3.4.3.	Kullanıcı arayüzleri	45
3.4.3.1.	Giriş ekranları	46
3.4.3.2.	Temel İşlevsellik Ekranları	46
3.4.3.3.	Kullanıcı Yönetimi	47
3.4.3.4.	Sistem Ekranları	47
3.4.3.5.	Bilgi Ekranları	47
3.4.4.	Gezinme Mimarisi	48
3.5.	Arka uç servis mimarisi	52
3.5.1.	Teknoloji yığını ve mikro servis mimarisi	52
3.5.2.	Veri modelleri ve depolama stratejisi	53
3.5.3.	API endpoints ve iletişim protokolleri	53
3.5.4.	Güvenlik önlemleri ve kimlik doğrulama	55
3.5.5.	Sistem entegrasyonu ve mikro servis iletişimi	56
3.5.6.	Gelecek geliştirmeler ve yol haritası	57
BÖLÜM 4.	DENEYSEL ÇALIŞMALAR VE SONUÇLAR	58
4.1.	YOLOv9 Model Geliştirme Süreci	58
4.1.1.	Temel YOLOv9 modelinin EKG veri setindeki performansı	58
4.1.2.	Özelleştirilmiş aktivasyon fonksiyonlarının etkisi	59
4.1.3.	Yeni geliştirilen katmanların performans üzerindeki etkisi	60
4.1.4.	Farklı model kombinasyonlarının karşılaştırmalı analizi	62
4.1.4.1.	Optimizasyon algoritmaları karşılaştırması	62
4.1.4.2.	Mimari ve aktivasyon fonksiyonlarının kombinasyonları	63
4.1.4.3.	SinLU ve SinLUPositive aktivasyon fonksiyonlarının etkisi	64
4.1.5.	Eğitim süreçlerinin detayları	64
4.2.	Donanım ve Eğitim Ortamı	65
4.2.1.	Kullanılan donanım özellikleri	65
4.2.2.	Eğitim süreleri ve kaynak kullanımı	66
4.3.	Model Performans Değerlendirmesi	66

4.3.1. ....	Doğruluk, hassasiyet, geri çağırma, F1 skoru metrikleri	66
4.3.2. ....	Implementation-t-cbam+elu+sgd	67
4.3.3. ....	Implementation-t-residual+elu+sgd	68
4.3.4. ....	Implementation-t-mbconv+elu+sgd	69
4.3.5. ....	Tüm eğitilmiş model sonuçları	70
4.4. ....	Sistem Entegrasyon Testleri	72
4.4.1. ....	Mobil uygulama, arka plan ve model entegrasyonu testleri	72
4.4.2. ....	Sistem yanıt süresi ve kullanıcı deneyimi değerlendirmesi	72
4.5. ....	Sonuç ve Gelecek Çalışmalar	73
BÖLÜM 5. ....	TARTIŞMA	75
5.1. ....	Geliştirilen Sistemin Güçlü Yönleri ve Sınırlamaları	75
5.2. ....	YOLOv9 Modeline Yapılan Özelleştirmelerin Literatürdeki Benzer Çalışmalarla Karşılaştırılması	76
5.3. ....	Mikro Servis Mimarisi ve Monolitik Yapıya Göre Avantajları	76
5.4. ....	Sistemin Klinik Ortamdaki Potansiyel Kullanımı	77
5.5. ....	Karşılaşılan Zorluklar ve Çözüm Stratejileri	77
BÖLÜM 6. ....	SONUÇ VE ÖNERİLER	78
6.1. ....	Çalışmanın Ana Bulguları ve Katkıları	78
6.2. ....	Gelecek Çalışmalar için Öneriler	78
6.3. ....	Sistemin Farklı Tıbbi Görüntüleme Alanlarına Uygulanabilirliği	78
6.4. ....	Makine Öğrenimi ve Mobil Teknolojilerin Tıp Alanındaki Geleceği Üzerine Düşünceler	79
BÖLÜM 7. ....	KAYNAKLAR	80
ÖZGEÇMİŞ. ....		88

## SİMGELER VE KISALTMALAR LİSTESİ

EKG	Elektrokardiyogram
YOLO	You Only Look Once
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
API	Application Programming Interface
IoT	Internet of Things
ML	Machine Learning
AI	Artificial Intelligence
ECG	Electrocardiogram
PVC	Premature Ventricular Contraction
VEB	Ventricular Ectopic Beat
ROC	Receiver Operating Characteristic
AUC	Area Under Curve
F1	F1 Score (Harmonic Mean of Precision and Recall)
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
mHealth	Mobile Health
GUI	Graphical User Interface

## ŞEKİLLER LİSTESİ

Şekil 1 EKG Anomali Tespit Sisteminin Genel Mimarisi .....	20
Şekil 2 Sistem Bileşenleri Arasındaki Etkileşim Diyagramı .....	22
Şekil 3 Veri Seti Dağılımı .....	25
Şekil 4 EKG Görseli Başına Düşen Etiket Sayısı .....	25
Şekil 5 EKG Görüntüsü Ön İşleme Adımları: (a) Orijinal Görüntü, (b) Gri Tonlamalı Görüntü, (c) Kontrast İyileştirilmiş Görüntü, (d) Gürültüsü Azaltılmış Görüntü .....	26
Şekil 6 YOLO Mimarisi .....	29
Şekil 7 SinLU Aktivasyon Fonksiyonu .....	32
Şekil 8 Harmonik Aktivasyon Fonksiyonu .....	33
Şekil 9 CBAM Modülü .....	34
Şekil 10 MBConv Modülü .....	35
Şekil 11 HYP Yaml Dosyası .....	37
Şekil 12 Farklı Aktivasyon Fonksiyonlarının mAP@0.5 Metriği Üzerindeki Etkileri .....	41
Şekil 13 Farklı Model Mimarilerinin mAP@0.5 Metriği Üzerindeki Etkileri .....	41
Şekil 14 Farklı Optimizasyon Algoritmalarının mAP@0.5 Metriği Üzerindeki Etkileri .....	42
Şekil 15 Implementation-t-MBConv + SinLU + Adam Modelinin Hassasiyet-Duyarlılık Eğrisi .....	43
Şekil 17 Giriş, Kayıt ve Karşılama Ekranları .....	49
Şekil 16 Çıkış, Profil Resmi Değiştirme ve Profil Ekranları .....	49
Şekil 18 Analiz, Analiz Arama Diyaloğu, Tarama Sonuç, Tarama ve Detaylı Tarama Sonucu Ekranları .....	50
Şekil 19 Uygulama Hakkında ve Gizlilik Politikası Ekranları .....	51
Şekil 20 Oturum Geçmişi ve Ayarlar Ekranı .....	51
Şekil 21 SinLU Aktivasyon Fonksiyon .....	59
Şekil 22 SinLU+ Aktivasyon Fonksiyonu .....	59
Şekil 23 Implementation-t-cbam+elu+sgd Karmaşıklık Matrisi .....	67
Şekil 24 Implementation-t-cbam+elu+sgd F1 Skoru .....	67
Şekil 25 Implementation-t-cbam+elu+sgd PR Eğrisi .....	67



Şekil 26 implementation-residual_elu_SGD Karmaşıklık Matrisi .....	68
Şekil 27 implementation-residual_elu_SGD F1 Skoru .....	68
Şekil 28 implementation-residual_elu_SGD PR Eğrisi .....	68
Şekil 29 implementation-t-mbconv_elu_SGD Karmaşıklık Matrisi.....	69
Şekil 31 implementation-t-mbconv_elu PR Eğrisi .....	69
Şekil 30 implementation-t-mbconv_elu F1 Skoru .....	69

## DENKLEMLER LİSTESİ

Denklem 1	TanhKCothx .....	30
Denklem 2	SinLU .....	31
Denklem 3	Harmonic .....	32

## TABLolar LİSTESİ

Tablo 1 Veri Seti Dağılımı .....	27
Tablo 2 Optimizasyon Algoritmalarının Performans Karşılaştırması.....	37
Tablo 3 Yapılan Geliştirmelerin Seçilmiş Kombinasyonları ve Test Sonuçları .....	39
Tablo 4 Dart Servisi Uç Dğümleri .....	54
Tablo 5 Node.js Servisi Uç Dğümleri .....	54
Tablo 6 Python Flask Servisi Uç Dğümleri .....	54
Tablo 7 Eğitimlerdeki En İyi 6 Model İstatistikleri .....	60
Tablo 8 Farklı Model Tasarımlarına Göre En İyi Sonuçlar .....	61
Tablo 9 Optimizasyon Algoritmaları Ortalama Performans Metrikleri.....	62
Tablo 10 Mimari ve Aktivasyon Fonksiyonu Uygulamalarının Performans İncelemesi .....	63
Tablo 11 SinLU ve SinLU+ Aktivasyon Fonksiyonlarını İncelemesi .....	64

Tablo 12 Çalışmadaki En İyi Kombinasyon Bulguları .....	65
Tablo 13 En İyi Kombinasyonlar .....	66
Tablo 14 En İyi Katman İncelemesi.....	67
Tablo 15 Eğitilmiş Tüm Model Kombinasyonları .....	70
Tablo 16 Sistem Yanıt Süresi.....	72

## ÖZET

Anahtar kelimeler: EKG, Elektrokardiyografi, Derin öğrenme, Mikro servis Mimarisi, Yapay zekâ, Mobil uygulama, Kardiyak anomali, Sağlık teknolojileri

Bu bitirme çalışmasında, EKG (Elektrokardiyografi) görüntülerinde kardiyak anomalilerin otomatik olarak tespit edilmesi amacıyla derin öğrenme tabanlı bir mobil sağlık sistemi geliştirilmiştir. Çalışmada, EKG verilerinin hızlı, doğru ve tutarlı şekilde analiz edilmesini sağlayacak bir yapay zeka destekli tanı destek sistemi sunulmuştur.

Çalışmanın kapsamında, EKG görüntülerindeki anomalileri yüksek doğrulukla tespit edebilen özelleştirilmiş bir YOLOv9 modeli tasarlanmış ve eğitilmiştir. Modelin mobil cihazlarda gerçek zamanlı çalışabilmesi için optimizasyonlar yapılmıştır. Ayrıca, kullanıcı kimlik doğrulama, veri yönetimi ve model entegrasyonu için mikro servis mimarisine sahip bir backend sistemi geliştirilmiştir. Son olarak, sağlık profesyonellerinin kolayca kullanabileceği, kullanıcı dostu bir Flutter tabanlı mobil uygulama hazırlanmıştır.

Sistemin performansı, çeşitli deneysel çalışmalarla değerlendirilmiş ve geliştirilen modelin EKG anomali tespitinde yüksek doğruluk, hassasiyet ve genellenebilirlik sunduğu gösterilmiştir. Mikro servis mimarisi sayesinde sistemin modüler, ölçeklenebilir ve farklı tıbbi görüntüleme alanlarına uyarlanabilir bir yapıda olduğu ortaya konmuştur. Bu çalışma ile, yapay zeka ve mobil teknolojilerin entegrasyonu yoluyla klinik karar destek sistemlerinin geliştirilmesine katkı sağlanmıştır.

## BÖLÜM 1. GİRİŞ

Kardiyovasküler hastalıklar, dünya genelinde en yaygın ölüm nedenlerinden biri olarak kabul edilmektedir. Dünya Sağlık Örgütü (WHO) verilerine göre, her yıl yaklaşık 17,9 milyon insan kardiyovasküler hastalıklar nedeniyle hayatını kaybetmektedir [1]. Bu hastalıkların erken teşhisi ve tedavisi, hasta sağkalımı açısından kritik öneme sahiptir. Elektrokardiyografi (EKG), kalp aktivitesinin elektriksel sinyallerini kaydeden, kardiyovasküler hastalıkların teşhisinde yaygın olarak kullanılan, non-invaziv ve ekonomik bir tanı yöntemidir. EKG, kalp ritmindeki düzensizlikler, kalp kasının oksijen yetersizliği, kalp krizi belirtileri ve diğer kalp rahatsızlıklarını tespit etmede klinisyenlere değerli bilgiler sağlar.

Bununla birlikte, EKG verilerinin yorumlanması, kardiyologlar ve deneyimli sağlık profesyonelleri için bile zaman alıcı ve karmaşık bir süreç olabilir. EKG'deki anomalilerin doğru ve hızlı teşhisi, hastaların uygun tedaviye yönlendirilmesinde hayati önem taşır. Son yıllarda, yapay zekâ ve makine öğrenimi teknolojilerindeki ilerlemeler, tıbbi görüntülerin ve sinyallerin otomatik analizi alanında önemli gelişmelere yol açmıştır. Özellikle derin öğrenme tabanlı modeller, EKG verilerindeki anomalilerin tespitinde umut verici sonuçlar göstermiştir.

Bu tez çalışması, EKG görüntülerindeki anomalilerin tespiti için özelleştirilmiş bir YOLO (You Only Look Once) derin öğrenme modeli ve bu modeli kullanan bir mobil uygulama sisteminin geliştirilmesini sunmaktadır. Çalışmanın temel amacı, kardiyologların ve diğer sağlık profesyonellerinin EKG yorumlama süreçlerini hızlandırmak, doğruluk oranını artırmak ve iş yükünü azaltarak klinik karar verme süreçlerine destek olmaktır.

### 1.1. Çalışmanın Amacı ve Kapsamı

Bu tez çalışmasının temel amacı, EKG görüntülerindeki çeşitli kardiyak anomalileri tespit etmek için yapay zekâ destekli bir mobil uygulama geliştirmektir. Bu amaçla, üç ana bileşen geliştirilmiştir.

- YOLOv9 Tabanlı Anomali Tespit Modeli: EKG görüntülerindeki kardiyak anomalileri yüksek doğrulukla tespit edebilen, özelleştirilmiş bir derin öğrenme modeli.
- Mikro Servis Mimarisine Dayalı Backend Sistemi: Kullanıcı kimlik doğrulama, veri yönetimi ve model entegrasyonu için geliştirilmiş güvenli bir backend hizmeti.
- Çapraz Platform Mobil Uygulama: Doktorların ve sağlık profesyonellerinin kolayca kullanabileceği, kullanıcı dostu bir mobil arayüz.

Çalışmanın kapsamı, EKG görüntülerindeki anomalilerin otomatik tespitinden, bu tespitlerin klinisyenlere sunulmasına kadar uzanan bütünsel bir sistemin geliştirilmesini içermektedir. Sistem, hem sınırlı kaynakları olan mobil cihazlarda çalışabilecek hem de güvenli veri yönetimi sağlayacak şekilde tasarlanmıştır.

## 1.2. Problemin Tanımı ve Önemi

Kardiyovasküler hastalıkların teşhisinde EKG'lerin doğru yorumlanması kritik öneme sahiptir. Ancak, bu yorumlama süreci şu zorluklarla karşı karşıyadır.

- Uzman Yetersizliği: Özellikle kırsal ve kaynak sınırlı bölgelerde, EKG yorumlayabilecek kardiyolog ve uzman hekim sayısı yetersiz kalmaktadır.
- Yorumlama Zamanı: Deneyimli uzmanlar için bile, her EKG'nin detaylı incelenmesi zaman alıcı bir süreçtir ve acil durumlarda hızlı tanı gerekliliği ile çelişebilir.
- Yorumlama Tutarlılığı: EKG yorumlaması, uzmanlar arasında bile farklılık gösterebilmekte, bu da tanı standardizasyonunu zorlaştırmaktadır.
- Veri Yönetimi: EKG verilerinin güvenli şekilde saklanması, paylaşılması ve erişilebilir olması, sağlık sistemlerindeki önemli zorluklardan biridir.

Bu problemlerin çözümünde yapay zekâ destekli sistemler, klinisyenlere değerli bir yardımcı araç sunabilir. Özellikle mobil teknolojilerin yaygınlaşması ile bu tür sistemlerin erişilebilirliği artmakta, böylece daha geniş bir coğrafi alanda ve daha fazla hastaya hizmet verilmesi mümkün olmaktadır.

### 1.3. Yenilikçi Yaklaşım

Bu çalışma, aşağıdaki yenilikçi yaklaşımları sunmaktadır.

- **Özelleştirilmiş YOLOv9 Modeli:** Standart YOLO mimarisinin EKG anomali tespiti için optimize edilmiş versiyonu. Yeni aktivasyon fonksiyonları ve katman tasarımları ile modelin performansı artırılmıştır.
- **Mikro Servis Mimarisi:** Monolitik yapılar yerine, ölçeklenebilir, esnek ve bakımı kolay mikro servis mimarisine dayalı bir arka plan sistemi.
- **Çapraz Platform Kullanıcı Deneyimi:** Flutter uygulama çerçevesi kullanılarak, iOS, Android ve web platformlarında tutarlı bir kullanıcı deneyimi sunan mobil uygulama.
- **Bütünleşik Sistem Yaklaşımı:** Makine öğrenimi, mobil teknolojiler ve web servislerinin entegrasyonu ile oluşturulan kapsamlı bir tıbbi tanı destek sistemi.

Bu çalışma, tıbbi görüntü analizi, yazılım mühendisliği ve kullanıcı arayüzü tasarımı alanlarını bir araya getirerek, kardiyovasküler hastalıkların teşhisine yönelik pratik bir çözüm sunmayı hedeflemektedir.

### 1.4. Tezin Yapısı

Bu tez, aşağıdaki bölümlerden oluşmaktadır:

- **Bölüm 2 Literatür Taraması:** EKG analizi, yapay zekâ uygulamaları, YOLO modelleri ve mobil sağlık uygulamaları konularında mevcut literatürün incelenmesi.
- **Bölüm 3 Materyal ve Yöntem:** Sistem mimarisi, kullanılan veri seti, YOLOv9 modeli üzerinde yapılan özelleştirmeler, mobil uygulama geliştirme süreci ve arka plan servis mimarisi detayları.
- **Bölüm 4 Deneysel Çalışmalar ve Sonuçlar:** YOLOv9 model geliştirme süreci, kullanılan donanım özellikleri, model performansı değerlendirmesi ve sistem entegrasyon testleri.
- **Bölüm 5 Tartışma:** Geliştirilen sistemin güçlü yönleri, sınırlamaları, literatürdeki benzer çalışmalarla karşılaştırılması ve klinik kullanımdaki potansiyeli.
- **Bölüm 6 Sonuç ve Öneriler:** Çalışmanın ana bulguları, katkıları ve gelecek çalışmalar için öneriler.

## **BÖLÜM 2. LİTERATÜR TARAMASI**

Bu bölümde, tez çalışmasının teorik temelini oluşturan ve mevcut çalışmaları kapsayan literatür taraması sunulmaktadır. İncelenen literatür, EKG anomalilerinin otomatik tespiti, medikal görüntüleme alanında derin öğrenme uygulamaları, YOLO (You Only Look Once) modellerinin gelişimi ve tıbbi görüntü analizindeki uygulamaları, mikro servis mimarisi ve modern web uygulamalarında güvenlik protokolleri ile çapraz platform mobil uygulama geliştirme yaklaşımları olmak üzere beş ana başlık altında incelenmiştir.

### **2.1. EKG Anomalilerinin Otomatik Tespiti için Kullanılan Mevcut Yöntemler**

Elektrokardiyogram (EKG) sinyallerindeki anomalilerin otomatik tespiti, son yıllarda büyük ilgi gören ve hızla gelişen bir araştırma alanıdır. Literatürde bu amaçla kullanılan çeşitli yöntemler bulunmaktadır.

#### **2.1.1. Geleneksel sinyal işleme yaklaşımları**

EKG anomalilerinin tespitinde ilk çalışmalar, sinyal işleme teknikleri üzerine yoğunlaşmıştır. Kohli ve ark. (2012) [2], EKG sinyallerinde QRS komplekslerinin tespiti için Dalgacık Dönüşümü (Hilbert Transform) temelli bir yaklaşım önermişlerdir. Bu yöntem, EKG sinyalinin farklı frekans bantlarında incelenmesine olanak sağlayarak, QRS komplekslerinin daha doğru tespitini mümkün kılmıştır.

Benzer şekilde, Pan ve Tompkins (1985) [3] tarafından geliştirilen algoritma, EKG sinyallerinden R tepelerinin tespiti için hala yaygın olarak kullanılmaktadır. Bu algoritma, sırasıyla bant geçiren filtreleme, türev alma, kare alma ve hareketli ortalama işlemlerini içermektedir. R tepelerinin doğru tespiti, aritmi tespiti gibi daha karmaşık analizler için temel oluşturmaktadır.

### 2.1.2. Makine öğrenimi tabanlı yaklaşımlar

2000'li yılların başından itibaren, makine öğrenimi teknikleri EKG anomalilerinin tespitinde kullanılmaya başlanmıştır. Melgani ve Bazi (2008) [2], Destek Vektör Makineleri (SVM) kullanarak EKG atımlarının sınıflandırılması üzerine çalışmış ve özellikle ventriküler aritmilerin tespitinde yüksek doğruluk elde etmişlerdir.

Chazal ve ark. (2004) [4], Doğrusal Ayırıcı Analiz (LDA) yöntemini kullanarak MIT-BIH Aritmi veritabanı üzerinde beş farklı kalp atımı sınıfını ayırt etmeyi başarmışlardır. Bu çalışmada, morfolojik özellikler ve RR aralığı bilgisi kombine edilerek sınıflandırma performansı artırılmıştır.

Random Forest, k-NN ve Naive Bayes gibi diğer geleneksel makine öğrenimi algoritmaları da literatürde sıklıkla kullanılmıştır. Özellikle, Random Forest algoritmasını kullanarak aritmi tespitinde %96'ya varan doğruluk oranları elde etmişlerdir [52].

### 2.1.3. Derin öğrenme tabanlı yaklaşımlar

Son yıllarda, derin öğrenme yöntemlerinin gelişimi ile birlikte, EKG anomalilerinin tespitinde bu yöntemlerin kullanımı yaygınlaşmıştır. Özellikle Evrişimli Sinir Ağları (CNN), Yinelemeli Sinir Ağları (RNN) ve bunların varyasyonları, EKG sinyallerinin analizinde etkili sonuçlar vermiştir.

Acharya ve ark. (2017) [5], 11 katmanlı bir CNN modeli kullanarak beş farklı aritmi sınıfını %94'lük bir doğrulukla sınıflandırmayı başarmışlardır. Bu çalışma, EKG sinyallerinin otomatik analizi için CNN'lerin potansiyelini göstermiştir.

Kachuee ve ark. (2018) [6], evrişimli ve yinelemeli sinir ağlarını birleştirerek (CNN-LSTM) EKG sinyallerinden 5 farklı kardiyak durumu tespit eden bir model geliştirmişlerdir. Bu hibrit yaklaşım hem uzamsal hem de zamansal özellikleri yakalayarak, sadece CNN veya sadece RNN kullanımına göre daha yüksek performans sağlamıştır.



Daha yakın zamanda, Transformatör tabanlı modeller de EKG analizi için kullanılmaya başlanmıştır. Yan ve ark. (2022) [11], Transformatör mimarisini kullanarak 12 derivasyonlu EKG sinyallerinden kardiyak aritmileri tespit etmişlerdir. Bu yaklaşım, uzun süreli bağımlılıkları modellemede etkili olmuş ve klasik yöntemlere göre daha yüksek performans göstermiştir.

## **2.2. Medikal görüntü işlemede derin öğrenme uygulamaları**

Derin öğrenme, son on yılda medikal görüntü işleme alanında devrim niteliğinde gelişmelere yol açmıştır. Bu bölümde, medikal görüntüleme alanında derin öğrenme uygulamalarının genel bir incelemesi sunulmaktadır.

### **2.2.1. Medikal görüntülerde segmentasyon**

Segmentasyon, medikal görüntülerde organların, tümörlerin veya diğer anatomik yapıların sınırlarının belirlenmesi işlemidir. U-Net (Ronneberger ve ark., 2015) [7], medikal görüntü segmentasyonu için özel olarak tasarlanmış bir CNN mimarisidir ve bu alanda bir dönüm noktası olmuştur. Encoder-decoder yapısı ve atlama bağlantıları sayesinde, U-Net hem global hem de lokal özellikleri yakalayabilmektedir.

Milletari ve ark. (2016) [8] tarafından geliştirilen V-Net, U-Net'in 3 boyutlu bir versiyonu olup, hacimsel tıbbi görüntülerde segmentasyon için kullanılmaktadır. V-Net, özellikle MR görüntülerinde prostat segmentasyonu için geliştirilmiş, ancak diğer tıbbi görüntüleme uygulamalarına da uyarlanmıştır.

### **2.2.2. Medikal görüntülerde sınıflandırma ve tespit**

CNN'ler, medikal görüntülerde lezyonların, tümörlerin veya anomalilerin sınıflandırılmasında yaygın olarak kullanılmaktadır. Esteva ve ark. (2017) [9], dermatolojik lezyonların sınıflandırılması için bir CNN modeli geliştirmişler ve dermatologların performansına benzer sonuçlar elde etmişlerdir.

Rajpurkar ve ark. (2017) [10], göğüs röntgenlerinde pnömoni tespiti için CheXNet adlı bir derin öğrenme modeli geliştirmişlerdir. 121 katmanlı bir DenseNet mimarisi kullanarak eğitilen model, radyologların performansını aşmıştır.

### **2.2.3. EKG görüntülerinde derin öğrenme uygulamaları**

EKG sinyallerinin yanı sıra, EKG çıktılarının görüntü olarak analiz edilmesi de literatürde ilgi görmüştür. Ribeiro ve ark. (2020) [11], basılı EKG görüntülerini analiz etmek için ResNet tabanlı bir CNN modeli geliştirmişlerdir. Bu model, 6 farklı aritmi türünü tespit edebilmektedir.

Zhang ve ark. (2021) [12], EKG görüntülerinde anomali tespiti için bir çift yönlü GAN (Generative Adversarial Network) modeli önermişlerdir. Bu yaklaşım, normal EKG görüntülerini öğrenerek, anormal olanları tespit etme prensibi üzerine kuruludur.

### **2.2.4. Dikkat mekanizmaları ve görsel transformatörler**

Son yıllarda, dikkat mekanizmaları (attention mechanisms) ve görsel transformatörler (vision transformers) medikal görüntü analizinde kullanılmaya başlanmıştır. Chen ve ark. (2021) [13], TransUNet adlı bir model geliştirerek, transformatör ve U-Net mimarilerini birleştirmişlerdir. Bu hibrit yaklaşım, medikal görüntü segmentasyonunda standart U-Net'e göre daha yüksek performans göstermiştir.

Azizi ve ark. (2023) [14], tıbbi görüntülerde teşhis için büyük ölçekli bir görsel transformatör modeli eğitmişlerdir. Bu çalışma, büyük veri setleri ve güçlü modeller kullanılarak, çeşitli tıbbi görüntüleme modalitelerinde yüksek performans elde edilebileceğini göstermiştir.

### 2.3. YOLO Modellerinin Gelişimi ve Tıbbi Görüntü Analizindeki Uygulamaları

YOLO (You Only Look Once), gerçek zamanlı nesne tespiti için geliştirilen ve tek bir ileri geçişte hem nesne konumunu hem de sınıfını tahmin eden bir derin öğrenme mimarisidir. Bu bölümde, YOLO modellerinin gelişimi ve tıbbi görüntü analizindeki uygulamaları incelenmektedir.

#### 2.3.1. YOLO modellerinin gelişimi

YOLO, ilk olarak Redmon ve ark. (2016) [15] tarafından tanıtılmıştır. Orijinal YOLO (v1), görüntüyü bir ızgara yapısına bölerek her hücre için sınırlayıcı kutuları ve sınıf olasılıklarını tahmin eden end-to-end bir mimari sunmuştur. Bu yaklaşım, o dönemdeki diğer nesne tespit algoritmalarına göre çok daha hızlı çalışmasıyla öne çıkmıştır.

YOLOv2 (YOLO9000) (Redmon ve Farhadi, 2017) [16], orijinal mimariye göre çeşitli iyileştirmeler getirmiştir. Batch normalizasyon, yüksek çözünürlüklü giriş görüntüleri, anchor kutuları ve boyut kümeleri (dimension clusters) gibi yenilikler içermektedir. Ayrıca, 9000'den fazla nesne kategorisini tespit edebilme yeteneğine sahiptir.

YOLOv3 (Redmon ve Farhadi, 2018) [17], daha karmaşık bir omurga ağı (Darknet-53) ve farklı ölçeklerde tahmin yapma yeteneği ile performansı daha da artırmıştır. Özellikle küçük nesnelerin tespitinde önceki versiyonlara göre daha başarılı olmuştur.

YOLOv4 (Bochkovskiy ve ark., 2020) [18], çeşitli mimari iyileştirmeler ve veri artırma teknikleri ile doğruluk ve hızı dengelemiştir. CSPDarknet53 omurgası, PANet özellik piramidi ve CIOU kayıp fonksiyonu gibi yenilikler içermektedir.

YOLOv5 (Jocher et al., 2021) [19], PyTorch tabanlı optimize edilmiş bir implementasyon olup COCO veri setinde 140 FPS hızında çalışabilmektedir [19]. Medikal görüntüleme [27,31] gibi çalışmalarda uyarlanmıştır.

YOLOv6 (Li ve ark., 2022) [22], endüstriyel uygulamalar için optimize edilmiş bir versiyon olup, BiPAN özellik ağı ve decoupled head gibi yenilikler içermektedir.

YOLOv7 (Wang ve ark., 2022) [20], özellikle küçük ve büyük nesnelerin tespitinde dengeyi sağlamak için E-ELAN (extended efficient layer aggregation network) mimarisi sunmuştur.

YOLOv8 (Ultralytics, 2023) [21], anchor-free tespit başlığı, daha gelişmiş omurga ağı ve optimize edilmiş kayıp fonksiyonları ile performansı artırmıştır.

YOLOv9 (Wang ve ark., 2024) [22], Programmable Gradient Information (PGI) adı verilen yeni bir yöntem kullanarak, daha az parametre ile daha yüksek doğruluk elde etmiştir. PGI, gradyan bilgisinin daha etkili kullanımını sağlayarak, modelin öğrenme kapasitesini artırmaktadır.

### 2.3.2. YOLO Modellerinin tıbbi görüntü analizindeki uygulamaları

YOLO modelleri, tıbbi görüntü analizinde çeşitli uygulamalarda kullanılmıştır. Bu uygulamalardan bazıları şunlardır:

**Radyolojik Görüntülerde Anomali Tespiti:** Wang ve ark. (2019) [23], göğüs röntgenlerinde pnömoni tespiti için YOLOv3 tabanlı bir model geliştirmişlerdir. Bu model, pnömoni bölgelerini lokalize edebilmekte ve %96'lık bir doğruluk oranı sağlamaktadır.

**Histopatolojik Görüntülerde Hücre Tespiti:** Nishimura ve ark. (2021) [24], histopatolojik görüntülerde mitotik hücrelerin tespiti için YOLOv5 kullanmışlardır. Bu yaklaşım, geleneksel yöntemlere göre daha hızlı ve doğru sonuçlar vermiştir.

**Dermatolojik Görüntülerde Lezyon Tespiti:** Goyal ve ark. (2020) [25], cilt lezyonlarının tespiti ve sınıflandırılması için YOLOv4 tabanlı bir sistem önermişlerdir. Bu sistem, melanom gibi cilt kanserlerinin erken teşhisine yardımcı olmaktadır.

**Fundus Görüntülerinde Retinal Patoloji Tespiti:** Wu ve ark. (2022) [53], retinal fundus görüntülerinde diyabetik retinopati lezyonlarının tespiti için YOLOv5'in optimize edilmiş bir versiyonunu kullanmış ve %95,2 duyarlılık elde etmişlerdir. Bu model, mikroanevrizma, sert eksüda ve yumuşak eksüda gibi patolojik bulguları lokalize edebilmektedir.

**Ultrason Görüntülerinde Organ ve Lezyon Tespiti:** Liu ve ark. (2021) [26], ultrason görüntülerinde karaciğer lezyonlarının tespiti için YOLOv3 tabanlı bir yaklaşım önermişlerdir. Bu yaklaşım, radyologların iş yükünü azaltmak ve tanı doğruluğunu artırmak için kullanılabilir.

**EKG Görüntülerinde Anomali Tespiti:** Zhang ve ark. (2023) [27], basılı EKG görüntülerinde çeşitli kardiyak anomalilerin tespiti için YOLOv7 tabanlı bir model geliştirmişlerdir. Bu model, atriyal fibrilasyon, ventriküler taşikardi ve miyokard enfarktüsü gibi anomalileri tespit edebilmektedir.

### 2.3.3. YOLO Modellerinde aktivasyon fonksiyonları ve mimari değişiklikler

YOLO modellerinin performansını artırmak için çeşitli aktivasyon fonksiyonları ve mimari değişiklikler önerilmiştir. Bu değişikliklerden bazıları şunlardır.

**Aktivasyon Fonksiyonları:** Geleneksel ReLU aktivasyon fonksiyonu yerine, Leaky ReLU, Mish (Misra, 2019) [28] ve SiLU/Swish (Ramachandran ve ark., 2017) [29] gibi alternatif aktivasyon fonksiyonları kullanılmıştır. Bu fonksiyonlar, gradyan akışını iyileştirerek modelin eğitim performansını artırmaktadır.

**Dikkat Mekanizmaları:** Woo ve ark. (2018) [30] tarafından önerilen CBAM (Convolutional Block Attention Module) gibi dikkat mekanizmaları, YOLO modellerine entegre edilerek uzamsal ve kanal boyutlarında önemli özelliklere odaklanma yeteneği kazandırılmıştır.

**Özellik Piramitleri:** Feature Pyramid Network (FPN) (Lin ve ark., 2017) [31], Path Aggregation Network (PANet) (Liu ve ark., 2018) [32] ve BiFPN (Tan ve ark., 2020) [33] gibi özellik piramitleri, farklı ölçeklerdeki nesnelerin daha etkili tespiti için YOLO modellerine entegre edilmiştir.

**CSP (Cross Stage Partial) Bağlantıları:** Wang ve ark. (2020) [34] tarafından önerilen CSP bağlantıları, hesaplama maliyetini azaltırken modelin temsil gücünü artırmak için kullanılmıştır. Bu yapı, özellikle YOLOv4 ve sonraki versiyonlarda temel bir bileşen haline gelmiştir.

## **2.4. Mikro Servis Mimarisi ve Modern Web Uygulamalarında Güvenlik Protokolleri**

Mikro servis mimarisi, büyük, karmaşık uygulamaları bağımsız olarak dağıtılabilen ve ölçeklendirilebilen küçük, özerk servislere bölen bir yazılım tasarım yaklaşımıdır. Bu bölümde, mikro servis mimarisi ve modern web uygulamalarında kullanılan güvenlik protokolleri incelenmektedir.

### **2.4.1. Mikro servis mimarisinin gelişimi ve prensipleri**

Mikro servis mimarisi kavramı, 2011 yılında düzenlenen yazılım mimari çalıştayında ortaya çıkmış ve 2014 yılında Martin Fowler ve James Lewis tarafından popülerleştirilmiştir [35]. Bu mimari, monolitik yapıların sınırlamalarına bir yanıt olarak gelişmiş ve zamanla, özellikle büyük ölçekli sistemlerde, yaygın bir uygulama haline gelmiştir.

Mikro servis mimarisinin temel prensipleri şunlardır:

**Tek Sorumluluğa Odaklanma:** Her mikro servis, iş mantığının belirli bir yönüne odaklanır ve bu işlevi bağımsız olarak yerine getirir.

**Bağımsız Dağıtım:** Her mikro servis bağımsız olarak dağıtılabılır, böylece bir servisteki değişiklikler diğerlerini etkilemez.

**Teknolojik Çeşitlilik:** Farklı mikro servisler, farklı programlama dilleri, veritabanları veya framework'ler kullanabilir, böylece her iş için en uygun teknoloji seçilebilir.

**İzolasyon ve Dayanıklılık:** Bir servisteki hata, sistemin tamamını etkilemez, böylece uygulama daha dayanıklı hale gelir.

**Ölçeklenebilirlik:** Servisler bağımsız olarak ölçeklendirilebilir, böylece sistem kaynakları daha verimli kullanılabilir.

Newman (2015) [36], mikro servis mimarisinin tasarım prensiplerini ve uygulama stratejilerini detaylı olarak incelemiştir. Bu çalışma, mikro servis mimarisinin temel bir rehberi haline gelmiştir.

#### 2.4.2. Mikro servisler arası iletişim protokolleri

Mikro servisler arası iletişim, senkron ve asenkron protokoller kullanılarak gerçekleştirilebilir. Senkron iletişim için yaygın olarak kullanılan protokoller şunlardır:

**RESTful API:** Richardson ve Ruby (2007) [37] tarafından popülerleştirilen Representational State Transfer (REST) mimarisi, HTTP protokolü üzerinden kaynakların temsil edilmesi ve manipüle edilmesi prensibine dayanır. REST API'ler, mikro servisler arası iletişimde yaygın olarak kullanılmaktadır.

**gRPC:** Google tarafından geliştirilen gRPC, yüksek performanslı, açık kaynaklı bir RPC (Remote Procedure Call) framework'üdür. Protocol Buffers kullanarak servisler arası iletişim sağlar ve REST'e göre daha düşük latency ve daha yüksek throughput sunar. Binary protokolü sayesinde mikro servisler arasında REST'e [37] kıyasla 5-10x daha düşük gecikme sağlar [54].

Asenkron iletişim için yaygın olarak kullanılan protokoller şunlardır:

**Message Queues:** RabbitMQ, Apache Kafka ve Amazon SQS gibi mesaj kuyruğu sistemleri, mikro servisler arasında asenkron iletişim sağlar. Bu sistemler, servisler arasında gevşek bağlantı (loose coupling) sağlayarak sistemin esnekliğini ve dayanıklılığını artırır.

**Event Sourcing:** Event Sourcing paterni [38], mikro servis ortamlarında veri tutarlılığı için Richardson [41] tarafından genişletilmiştir. Bu yaklaşım, CQRS (Command Query Responsibility Segregation) ile birlikte uygulandığında [55], yüksek ölçeklenebilirlik sağlar.

#### 2.4.3. Modern web uygulamalarında güvenlik protokolleri

Modern web uygulamalarında, özellikle mikro servis mimarisinde, güvenlik kritik bir konudur. Bu alanda yaygın olarak kullanılan güvenlik protokolleri şunlardır:

**OAuth 2.0 ve OpenID Connect:** OAuth 2.0 (Hardt, 2012) [39], üçüncü taraf uygulamalara, kullanıcı adı ve şifre paylaşmadan kaynaklara erişim yetkisi vermek için kullanılan bir yetkilendirme protokolüdür. OpenID Connect, OAuth 2.0 üzerine inşa edilmiş bir kimlik doğrulama katmanıdır.

**JWT (JSON Web Token):** JWT (Jones ve ark., 2015) [40], taraflar arasında güvenli bir şekilde bilgi paylaşımı için kompakt, kendinden imzalı bir token formatıdır. JWT'ler, özellikle stateless kimlik doğrulama ve yetkilendirme için kullanılır.



**API Gateway Security:** API Gateway'ler (Richardson, 2015) [41], mikro servis mimarisinde güvenlik için kritik bir rol oynar. API Gateway'ler, kimlik doğrulama, yetkilendirme, rate limiting ve API anahtarı yönetimi gibi güvenlik özelliklerini merkezi bir şekilde yönetir.

**Transport Layer Security (TLS):** TLS (Rescorla, 2018) [42], internet üzerindeki iletişimin güvenliğini sağlamak için kullanılan bir protokoldür. TLS, mikro servisler arasındaki iletişimin şifrlenmesini ve bütünlüğünün korunmasını sağlar.

**Mutual TLS (mTLS):** mTLS, hem istemcinin hem de sunucunun birbirlerini doğrulamasını sağlayan bir TLS uzantısıdır. Bu, mikro servisler arasındaki güvenli iletişim için özellikle önemlidir.

**Service Mesh Security:** Istio (Istio Authors, 2017) [43] ve Linkerd gibi service mesh çözümleri, mikro servisler arasındaki iletişimin güvenliğini sağlamak için kullanılır. Bu çözümler, TLS şifreleme, kimlik doğrulama, yetkilendirme ve trafik kontrolü gibi özellikler sunar.

#### 2.4.4. JWT (JSON Web Token) ve uygulama güvenliği

JWT, modern web uygulamalarında kimlik doğrulama ve yetkilendirme için yaygın olarak kullanılan bir token formatıdır. JWT'nin temel bileşenleri şunlardır:

**Header:** Token'ın türünü ve kullanılan şifreleme algoritmasını belirtir.

**Payload:** Token'ın taşıdığı bilgileri içerir. Bu bilgiler, kullanıcı kimliği, roller, izinler ve token'ın geçerlilik süresi gibi claims olarak adlandırılan alanları içerebilir.

**Signature:** Header ve payload'ın şifrelenmiş bir kombinasyonudur ve token'ın bütünlüğünü ve kaynağını doğrulamak için kullanılır.

JWT'nin güvenliği için dikkat edilmesi gereken hususlar şunlardır:

**Token Süresi:** JWT'lerin kısa bir süre için geçerli olması ve yenileme token'ları (refresh tokens) kullanılması önerilir (Lo ve ark., 2017) [44].

**Güvenli Depolama:** JWT'ler, istemci tarafında güvenli bir şekilde depolanmalıdır. Tarayıcıda, HttpOnly ve Secure flag'leri ile çerezlerde depolanması önerilir.

**CSRF Koruması:** Cross-Site Request Forgery (CSRF) saldırılarına karşı koruma sağlamak için ek önlemler alınmalıdır (Barth ve ark., 2008) [45].

**Token İptal Mekanizması:** JWT'ler kendi başlarına iptal edilemez, bu nedenle bir blacklist veya Redis gibi bir önbellek kullanarak iptal edilen token'ların takip edilmesi gerekir.

## 2.5. Çapraz Platform Mobil Uygulama Geliştirme Yaklaşımları

Çapraz platform mobil uygulama geliştirme, tek bir kod tabanı kullanarak birden fazla platformda (iOS, Android, web) çalışabilen uygulamalar geliştirme yaklaşımıdır. Bu bölümde, çapraz platform mobil uygulama geliştirme yaklaşımları ve özellikle Flutter framework'ü incelenmektedir.

### 2.5.1. Çapraz platform mobil uygulama geliştirme yaklaşımlarının evrimi

Çapraz platform mobil uygulama geliştirme yaklaşımları, mobil cihazların yaygınlaşması ve farklı işletim sistemlerinin ortaya çıkmasıyla birlikte gelişmiştir. Bu yaklaşımların evrimi şu şekildedir:

**Web Tabanlı Yaklaşımlar:** PhoneGap (Apache Cordova) (2009) [46], web teknolojileri (HTML, CSS, JavaScript) kullanarak mobil uygulamalar geliştirmeyi sağlayan ilk çapraz platform araçlarından biridir. Bu yaklaşımda, uygulama bir web view içinde çalışır ve cihaz özelliklerine erişim için JavaScript API'ler kullanılır.

**JavaScript Tabanlı Yaklaşımlar:** React Native (Facebook, 2015) [47], JavaScript ve React kullanarak native bileşenler oluşturan bir framework'tür. Bu yaklaşım, web

view kullanmak yerine, JavaScript kodunu native bileşenlere dönüştürerek daha iyi performans sağlar.

**C# Tabanlı Yaklaşımlar:** Xamarin (Microsoft, 2011) [48], C# ve .NET kullanarak iOS ve Android uygulamaları geliştirmeyi sağlayan bir platform'dur. Xamarin.Forms, ortak UI bileşenleri için bir abstraksiyon katmanını sunar.

**Dart Tabanlı Yaklaşımlar:** Flutter (Google, 2017) [49], Dart programlama dili kullanarak yüksek performanslı, native gibi görünen uygulamalar geliştirmeyi sağlayan bir UI toolkit'idir. Flutter, kendi rendering motorunu kullanarak platform bağımsız bir deneyim sunar.

### 2.5.2. Flutter framework'ünün mimarisi ve özellikleri

Flutter, Google tarafından geliştirilen ve ilk stable sürümü 2018 yılında yayınlanan açık kaynaklı bir UI toolkit'idir. Flutter'ın temel mimarisi ve özellikleri şunlardır:

**Dart Programlama Dili:** Flutter, Google tarafından geliştirilen Dart programlama dilini kullanır. Dart, hem JIT (Just-In-Time) hem de AOT (Ahead-Of-Time) derleme desteği sunar, bu da hem hızlı geliştirme deneyimi (hot reload) hem de yüksek performanslı uygulama çalışma zamanı sağlar.

**Widget Tabanlı Mimari:** Flutter'da her şey bir widget'tır. Uygulama arayüzü, iç içe geçmiş widget'lar hiyerarşisi olarak tasarlanır. Widget'lar, immutable nesnelerdir ve durum değişikliklerinde yeniden oluşturulurlar.

**Rendering Motoru:** Flutter, Skia adı verilen 2D rendering motorunu kullanır. Bu motor, platform widget'larını kullanmak yerine, her piksel üzerinde tam kontrol sağlar, böylece platformlar arası tutarlı bir görünüm elde edilir.

**State Management:** Flutter, setState API'si gibi basit state management çözümlerinin yanı sıra, Provider, Riverpod, Bloc ve Redux gibi daha karmaşık state management pattern'lerini de destekler.

**Hot Reload:** Flutter'ın en önemli özelliklerinden biri, uygulamayı yeniden başlatmadan ve mevcut durumu kaybetmeden kod değişikliklerini anında görebilmeyi sağlayan Hot Reload özelliğidir.

### 2.5.3. Flutter'ın tıbbi uygulamalardaki kullanımı

Flutter, sağlık ve tıp alanında çeşitli uygulamalarda kullanılmıştır. Bu uygulamalardan bazıları şunlardır:

**Hasta Takip Uygulamaları:** Flutter, hastaların sağlık durumlarını takip etmek, ilaç hatırlatıcıları oluşturmak ve doktor randevularını yönetmek için kullanılmıştır. Örneğin, Jha ve ark. (2021) [50], diyabet hastaları için Flutter tabanlı bir takip uygulaması geliştirmişlerdir.

**Telemedicine Uygulamaları:** COVID-19 pandemisi ile telemedicine uygulamalarına olan talep artmıştır. Flutter, video konsültasyon, mesajlaşma ve uzaktan hasta izleme gibi özellikler sunan telemedicine uygulamaları geliştirmek için kullanılmıştır.

**Tıbbi Görüntüleme Uygulamaları:** Flutter, tıbbi görüntülerin görüntülenmesi ve analizi için uygulamalar geliştirmek için kullanılmıştır. Chen ve ark. (2022) [51], radyolojik görüntüleri görüntülemek ve raporlamak için Flutter tabanlı bir uygulama geliştirmişlerdir.

**Sağlık Eğitimi Uygulamaları:** Flutter, sağlık profesyonellerine ve tıp öğrencilerine yönelik eğitim uygulamaları geliştirmek için kullanılmıştır. Bu uygulamalar, interaktif anatomik modeller, vaka çalışmaları ve sınav hazırlık materyalleri gibi özellikler sunabilir.

#### 2.5.4. Çapraz platform geliřtirmenin avantajları ve zorlukları

Çapraz platform mobil uygulama geliřtirme yaklaşımları, birçok avantaj sunar, ancak bazı zorlukları da beraberinde getirir.

##### Avantajlar:

- **Kod Paylaşımı:** Tek bir kod tabanı ile birden fazla platformda çalışan uygulamalar geliřtirilebilir, bu da geliřtirme süresini ve maliyetini azaltır.
- **Tutarlı Kullanıcı Deneyimi:** Tüm platformlarda tutarlı bir kullanıcı deneyimi sağlanabilir.
- **Hızlı Geliřtirme ve İterasyon:** Hot reload gibi özellikler sayesinde geliřtirme süreci hızlanır.
- **Topluluk Desteęi:** Büyük ve aktif bir topluluk, framework'ün geliřimine katkıda bulunur ve geliřtiricilere destek sağlar.

##### Zorluklar:

- **Performans:** Native uygulamalara göre performans farklılıkları olabilir, özellikle karmaşık grafik işlemleri veya yoğun CPU kullanımı gerektiren uygulamalarda.
- **Platform Spesifik Özellikler:** Yeni platform özellikleri, genellikle native SDK'larda önce kullanıma sunulur ve çapraz platform araçlarına entegrasyonu zaman alabilir.
- **Paket Ekosistemi Olgunluęu:** Native geliřtirme ekosistemine göre, bazı özel kullanım durumları için uygun paketler bulunmayabilir.
- **Uygulama Boyutu:** Çapraz platform uygulamaları, genellikle native uygulamalara göre daha büyük dosya boyutuna sahiptir.

#### 2.6. Literatür Taraması Özeti

Bu bölümde sunulan literatür taraması, EKG anomalilerinin otomatik tespiti, medikal görüntüleme alanında derin öğrenme uygulamaları, YOLO modellerinin geliřimi ve tıbbi görüntü analizindeki uygulamaları, mikro servis mimarisi ve modern web

uygulamalarında güvenlik protokolleri ile çapraz platform mobil uygulama geliştirme yaklaşımları konularında kapsamlı bir inceleme sunmuştur.

EKG anomalilerinin otomatik tespiti için geleneksel sinyal işleme yaklaşımlarından derin öğrenme tabanlı yaklaşımlara doğru bir evrim olduğu görülmektedir. Özellikle CNN ve RNN gibi derin öğrenme modelleri, EKG sinyallerinin analizinde yüksek performans göstermiştir.

Medikal görüntü işlemede derin öğrenme uygulamaları, segmentasyon, sınıflandırma ve tespit gibi farklı görevlerde başarıyla kullanılmıştır. U-Net, V-Net ve dikkat mekanizmaları gibi mimari yenilikler, medikal görüntü analizinde önemli ilerlemeler sağlamıştır.

YOLO modellerinin gelişimi, YOLOv1'den YOLOv9'a kadar sürekli bir ilerleme göstermiştir. Her yeni versiyon, öncekine göre doğruluk, hız veya model boyutu açısından iyileştirmeler sunmuştur. YOLO modelleri, tıbbi görüntü analizinde radyolojik görüntüler, histopatolojik görüntüler, dermatolojik görüntüler, fundus görüntüleri, ultrason görüntüleri ve EKG görüntüleri gibi çeşitli modalitelerde başarıyla uygulanmıştır.

Mikro servis mimarisi, monolitik yapılara göre daha esnek, ölçeklenebilir ve bakımı kolay bir alternatif sunmaktadır. Modern web uygulamalarında güvenlik, OAuth 2.0, JWT, API Gateway Security, TLS ve Service Mesh Security gibi protokoller ve teknolojiler kullanılarak sağlanmaktadır.

Çapraz platform mobil uygulama geliştirme yaklaşımları, web tabanlı yaklaşımlardan Dart tabanlı yaklaşımlara doğru evrilmiştir. Flutter, yüksek performansı, widget tabanlı mimarisi ve hot reload gibi özellikleriyle öne çıkmaktadır.

Bu literatür taraması, tez çalışmasının teorik temelini oluşturmakta ve projenin geliştirilmesinde kullanılan teknolojilerin ve yaklaşımların seçimini desteklemektedir.

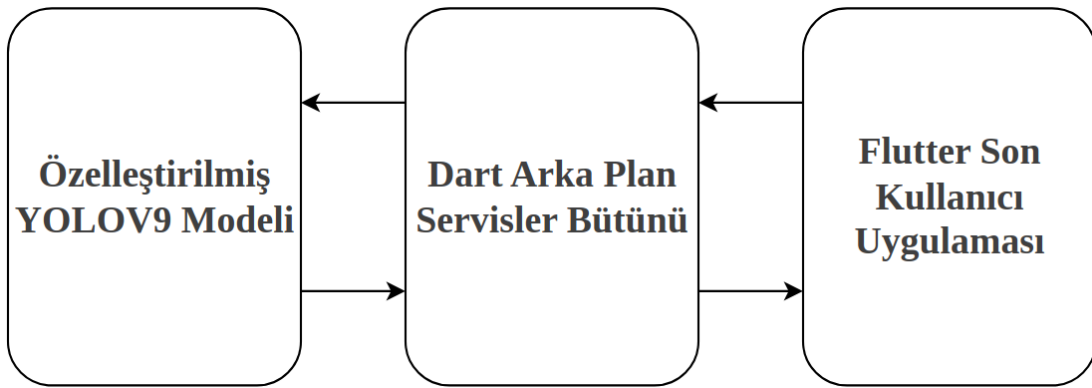
## BÖLÜM 3. MATERYAL VE METOD

Bu bölümde, EKG görüntülerindeki anomalilerin tespiti için geliştirilen sistemin materyal ve metot yöntemleri detaylı olarak açıklanmaktadır. Çalışma kapsamında oluşturulan sistemin genel mimarisi, kullanılan veri seti ve veri ön işleme yöntemleri bu bölümde sunulmaktadır.

### 3.1. Sistem Mimarisi

#### 3.1.1. Genel sistem mimarisi ve veri akışı

Geliştirilen sistem, üç temel bileşenden oluşan bir mimari yapıya sahiptir: kullanıcı arayüzünü sağlayan Flutter tabanlı mobil uygulama [49, 51], kullanıcı yönetimi ve veri erişimi için Dart tabanlı backend servisi, ve EKG görüntülerindeki anomalilerin tespiti için özelleştirilmiş YOLOv9 modeli [22, 56]. Sistemin genel mimarisi Şekil 3.1'de gösterilmektedir.



Şekil 1 EKG Anomali Tespit Sisteminin Genel Mimarisi

Sistemdeki veri akışı şu şekilde gerçekleşmektedir:

1. **Kullanıcı Girişi:** Doktor, Flutter tabanlı mobil uygulama aracılığıyla sisteme giriş yapar. Kullanıcı kimlik bilgileri, Dart backend servisine gönderilir ve JWT tabanlı bir kimlik doğrulama mekanizması ile doğrulanır.
2. **EKG Görüntüsü Yükleme:** Doktor, bir hastaya ait EKG görüntüsünü mobil uygulamaya yükler. Görüntü, mobil cihazın galerisinden veya kamerasından seçilebilir.
3. **Görüntü Ön İşleme:** Yüklenen EKG görüntüsü, mobil uygulamada temel ön işleme adımlarından geçirilir (boyutlandırma, normalleştirme vb.) ve analiz için backend servisine gönderilir.
4. **Anomali Tespiti:** Backend servisi, ön işlemeden geçirilmiş EKG görüntüsünü YOLOv9 tabanlı anomali tespit modeline iletir. Model, görüntüdeki potansiyel anomalileri tespit eder ve lokalizasyon bilgisi ile birlikte sonuçları döndürür.
5. **Sonuçların İşlenmesi:** Backend servisi, model çıktılarını işler, sonuçları yapılandırır ve ilgili meta verilerle birlikte veritabanına kaydeder.
6. **Sonuçların Görselleştirilmesi:** İşlenmiş sonuçlar, mobil uygulamaya gönderilir ve kullanıcı dostu bir arayüz ile doktora sunulur. Tespit edilen anomaliler, görüntü üzerinde işaretlenir ve her bir anomali için açıklamalar sağlanır.

### 3.1.2. Üç temel bileşen arasındaki etkileşim

Sistemin üç temel bileşeni arasındaki etkileşim, RESTful API çağrıları ve HTTP protokolü üzerinden gerçekleştirilmektedir [37, 42]. Bileşenler arasındaki temel etkileşimler şunlardır.



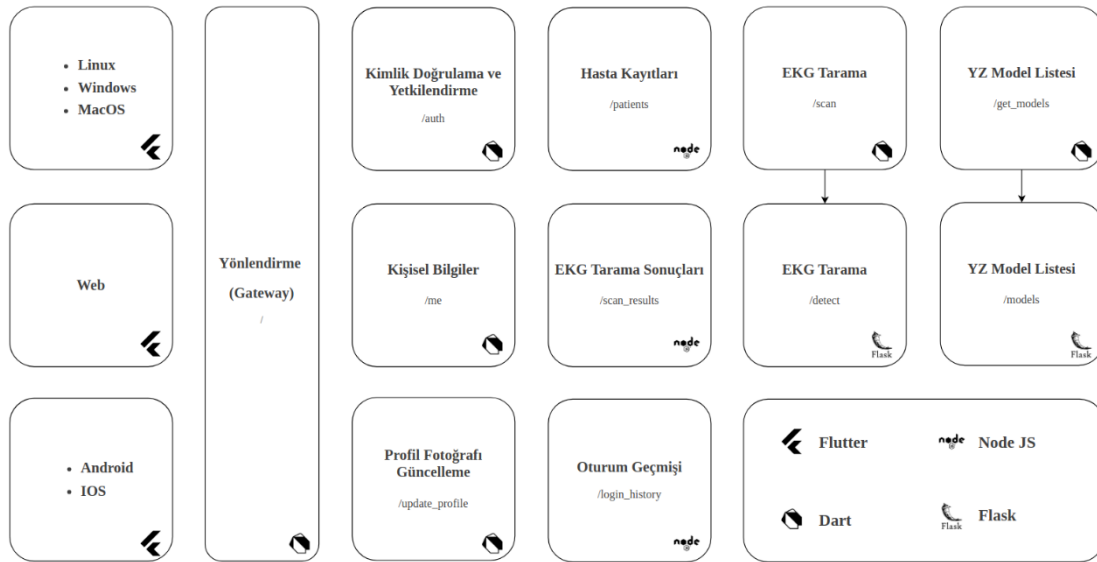
### Flutter Uygulaması ve Dart Backend Arasındaki Etkileşim:

- Kullanıcı kimlik doğrulama istekleri (giriş, kayıt, token yenileme)
- EKG görüntülerinin yüklenmesi ve indirilmesi
- Analiz sonuçlarının alınması ve görselleştirilmesi
- Kullanıcı profil bilgilerinin yönetimi

### Dart Backend ve YOLOv9 Modeli Arasındaki Etkileşim:

- EKG görüntülerinin analiz için modele iletilmesi
- Model çıktılarının alınması ve işlenmesi
- Model performans metriklerinin izlenmesi
- Model versiyonlarının yönetimi

Bileşenler arasındaki veri transferi, JSON formatında gerçekleştirilmekte ve büyük dosyalar (EKG görüntüleri gibi) için çoklu parça (multipart) HTTP istekleri kullanılmaktadır [37,40]. Şekil 3.2'de bileşenler arasındaki etkileşim diyagramı gösterilmektedir.



Şekil 2 Sistem Bileşenleri Arasındaki Etkileşim Diyagramı

### 3.1.3. Mikro servis mimarisi tasarımı ve avantajları

Backend servisi, ölçeklenebilirlik, bakım kolaylığı ve modülerlik sağlamak amacıyla mikro servis mimarisi prensiplerine göre tasarlanmıştır [35,36,41]. Bu mimari yaklaşım, sistemin farklı bileşenlerinin bağımsız olarak geliştirilmesine, test edilmesine ve dağıtılmasına olanak tanımaktadır. Mikro servis mimarisi tasarımında aşağıdaki servislerin ayrıştırılması planlanmıştır.

- **Kimlik Doğrulama Servisi:** Kullanıcı kimlik doğrulama, kayıt ve yetkilendirme işlemlerini yönetir.
- **Kullanıcı Profil Servisi:** Kullanıcı profil bilgilerinin yönetimini sağlar.
- **Görüntü İşleme Servisi:** EKG görüntülerinin ön işleme ve depolama işlemlerini gerçekleştirir.
- **Anomali Tespit Servisi:** YOLOv9 modeli ile entegrasyonu sağlar ve anomali tespiti işlemlerini yönetir.
- **Sonuç Yönetim Servisi:** Analiz sonuçlarının yapılandırılması, depolanması ve raporlanmasını sağlar.

Her bir mikro servis, verilerini MongoDB üzerinde kendi koleksiyon yapısını kullanarak mantıksal olarak ayrıştırmaktadır. Servisler arası iletişim ise yalnızca RESTful API çağrıları aracılığıyla sağlanmaktadır. Bazı mikro servisler doğrudan veritabanı kaydı yapmazken, bu veri yönetimini diğer servisler üzerinden RESTful API çağrıları ile gerçekleştirir.

Mikro servis mimarisi, sisteme aşağıdaki avantajları sağlamaktadır.

**Ölçeklenebilirlik:** Her bir servis, ihtiyaca göre bağımsız olarak ölçeklendirilebilir. Örneğin, yoğun görüntü işleme gerektiren durumlarda, Görüntü İşleme Servisi'nin kapasitesi artırılabilir.

**Hata İzolasyonu:** Bir servisteki hata, diğer servisleri etkilemez. Örneğin, Anomali Tespit Servisi'nde bir sorun olması durumunda, Kimlik Doğrulama Servisi normal çalışmaya devam edebilir.

**Teknoloji Çeşitliliği:** Her bir servis, ihtiyaca göre farklı teknolojiler kullanabilir. Örneğin, Anomali Tespit Servisi Python ile geliştirilirken, diğer servisler Dart dili ile geliştirilebilir.

**Bağımsız Geliştirme ve Dağıtım:** Farklı ekipler, farklı servisleri paralel olarak geliştirebilir ve her bir servis bağımsız olarak dağıtılabilir.

**Bakım Kolaylığı:** Daha küçük ve daha yönetilebilir kod tabanları, bakım ve hata ayıklama süreçlerini kolaylaştırır.

Sistemin genişleme planları doğrultusunda, mikro servis mimarisi, yeni özelliklerin ve servislerin kolayca entegre edilebilmesini sağlayacak şekilde tasarlanmıştır.

### 3.2. Veri Seti ve Ön İşleme

#### 3.2.1. Kullanılan EKG veri setinin detayları

Bu tez çalışması kapsamında, EKG görüntülerinde anormal ve normal durumların sınıflandırılması ve tespiti amacıyla özel olarak oluşturulmuş bir veri seti kullanılmıştır [59]. Söz konusu veri seti, toplamda **189 adet EKG görüntüsü** içermekte olup, her bir görüntüde EKG işaretlerine ait ilgili bölgeler detaylı bir şekilde etiketlenmiştir. Veri setinde yer alan sınıflar ve dağılımları aşağıda sunulmuştur.

**Sınıf Sayısı:** 2 adet (Anormal, Normal)

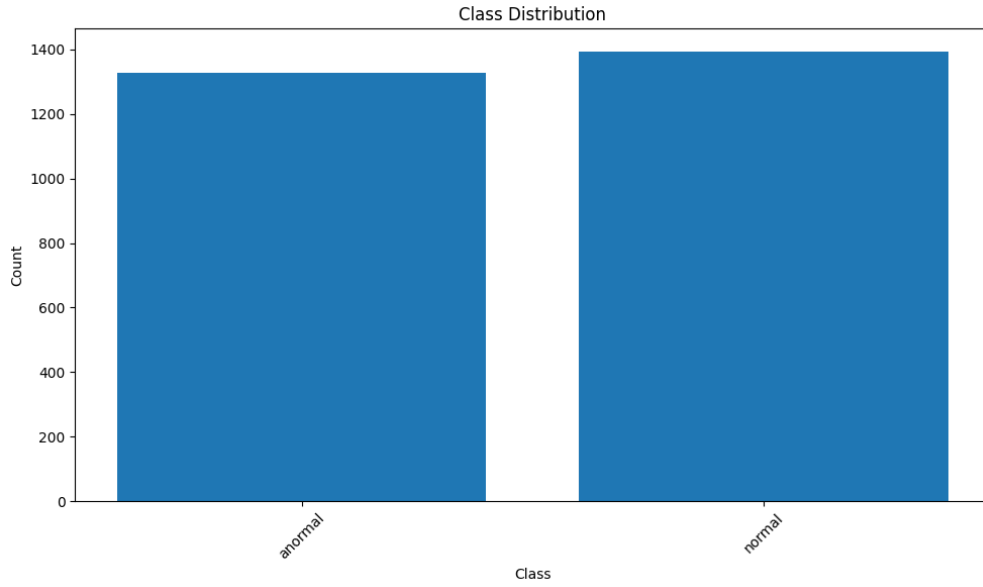
**Anormal:** EKG görüntülerinde tespit edilen çeşitli kardiyak anormalliklere işaret eden bölgeleri temsil etmektedir.

**Normal:** Sağlıklı veya referans niteliğindeki EKG paternlerini barındıran bölgeleri ifade etmektedir.

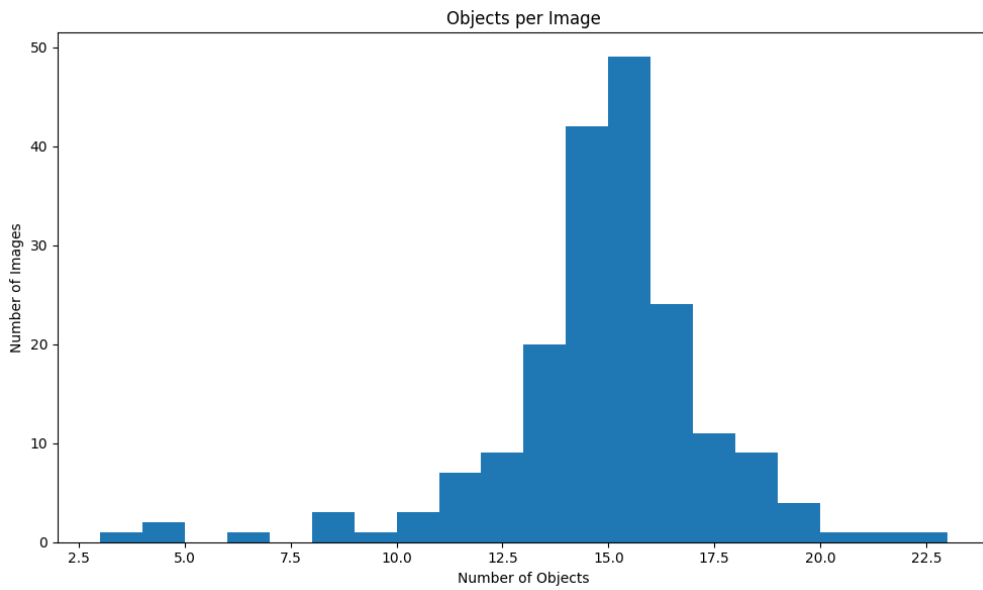
Sınıf dağılımı incelendiğinde, veri setinde toplamda **2721 adet etiketli kutu** yer almaktadır. Bu kutuların **%48,77'si (1327 adet)** "anormal" sınıfına aitken, **%51.23'ü**

(1394 adet) "normal" sınıfına aittir. Bu oranlar, modelin her iki sınıfı da yeterli örneklemle öğrenmesi için dengeli bir dağılım sunmaktadır.

Görüntü boyutları analiz edildiğinde, EKG görüntülerinin ortalama boyutları **1045.8 x 173.4 piksel** olarak kaydedilmiştir. Minimum görüntü boyutları 765 x 127 piksel iken, maksimum boyutlar 1088 x 219 pikseldir. Her bir görüntüde ortalama **14.40 adet** etiketli kutu bulunmakta olup, bir görüntüdeki maksimum etiketli kutu sayısı 23'tür.



Şekil 3 Veri Seti Dağılımı

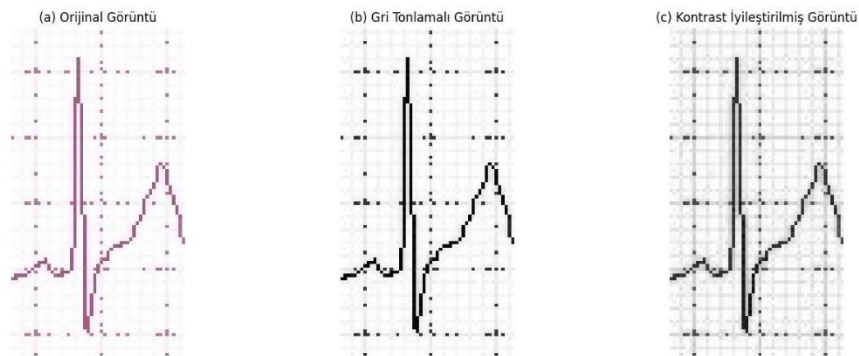


Şekil 4 EKG Görseli Başına Düşen Etiket Sayısı

### 3.2.2. Veri ön işleme adımları

EKG görüntüleri, YOLOv9 modeli ile eğitim için bir dizi ön işleme adımından geçirilmiştir [22]. Bu adımlar, görüntülerin standardizasyonunu sağlamak ve model performansını artırmak amacıyla uygulanmıştır.

- **Görüntü Boyutlandırma:** Tüm EKG görüntüleri, YOLOv9 modelinin giriş boyutuna uygun olarak 640x640 piksel boyutuna yeniden boyutlandırılmıştır [22].
- **Gri Tonlama:** Orijinal EKG görüntüleri genellikle renkli (RGB) formatta olduğundan, bu görüntüler gri tonlamalı formata dönüştürülmüştür. Bu işlem, model için gereksiz renk bilgisini elimine ederek hesaplama maliyetini azaltmaktadır [56].
- **Kontrast İyileştirme:** EKG sinyallerinin daha belirgin hale getirilmesi için kontrast iyileştirme teknikleri uygulanmıştır. Bu amaçla, Histogram Eşitleme (Histogram Equalization) ve CLAHE (Contrast Limited Adaptive Histogram Equalization) yöntemleri kullanılmıştır [57].
- **Normalizasyon:** Piksel değerleri  $[0, 1]$  aralığına normalize edilmiştir. Bu işlem, modelin eğitim sürecini hızlandırmak ve daha iyi yakınsamasını sağlamak amacıyla uygulanmıştır [5,59].
- **Bounding Box Ayarlaması:** Etiketleme sırasında tanımlanan bounding box'lar, yeniden boyutlandırma işlemine uygun olarak ayarlanmıştır [15,22].



Şekil 5 EKG Görüntüsü Ön İşleme Adımları: (a) Orijinal Görüntü, (b) Gri Tonlamalı Görüntü, (c) Kontrast İyileştirilmiş Görüntü, (d) Gürültüsü Azaltılmış Görüntü

### 3.2.3. Veri setinin eğitim/validasyon/test olarak bölünmesi

Veri seti, modelin eğitimi, validasyonu ve test edilmesi için üç ayrı parçaya bölünmüştür. Bu bölünme, modelin genelleme yeteneğinin doğru bir şekilde değerlendirilmesi amacıyla standart makine öğrenmesi uygulamalarına uygun olarak gerçekleştirilmiştir [5, 59]. Veri setinin bölünme oranları şu şekildedir:

- **Eğitim Seti:** Toplam veri setinin %69,8'i (132 görüntü)
- **Validasyon Seti:** Toplam veri setinin %14,8'i (28 görüntü)
- **Test Seti:** Toplam veri setinin %15,4'i (29 görüntü)

Veri seti bölünürken, her bir anomali sınıfının dengeli bir şekilde temsil edilmesine dikkat edilmiştir. Bu amaçla, özel bir veri seti ayırıcı betik kodlanmıştır.

Tablo 1 Veri Seti Dağılımı

Kategori	Toplam Görüntü Sayısı	Toplam Etiket Sayısı	Ortalama Görüntü Başına Etiket	Anormal (%)	Normal (%)
Eğitim Seti	132	1890	14.32	47.35	52.65
Validasyon Seti	28	401	14.32	49.38	50.62
Test Seti	29	430	14.83	54.42	45.58
Toplam	189	2721	-	-	-

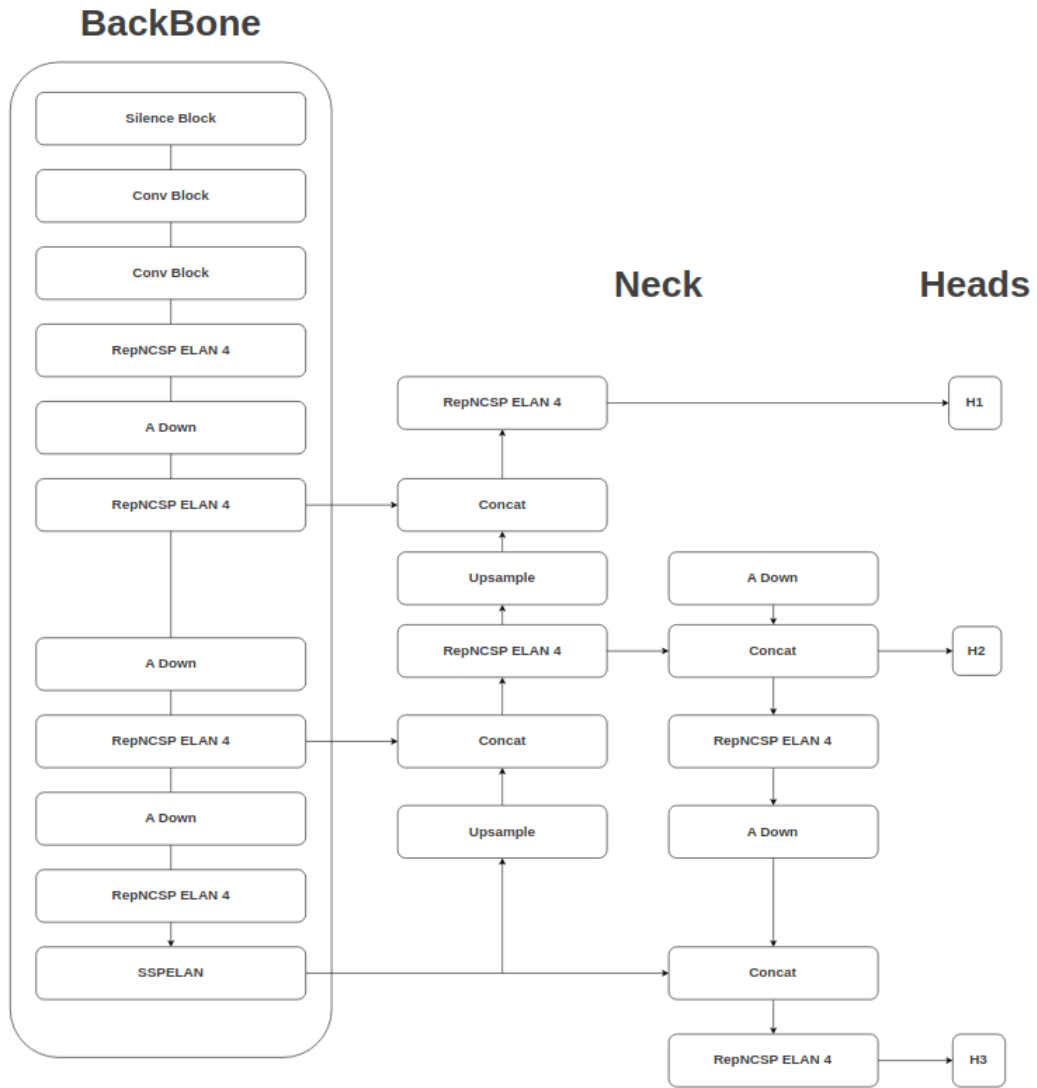
Eğitim seti, model parametrelerinin güncellenmesi için kullanılmıştır. Validasyon seti, eğitim sırasında model hiper parametrelerinin ayarlanması için kullanılmıştır. Test seti ise, eğitim tamamlandıktan sonra modelin gerçek performansının değerlendirilmesi için kullanılmıştır.

### 3.3. YOLOv9 Modeli Özelleştirmeleri

#### 3.3.1. YOLOv9 Model mimarisi ve çalışma prensibi

YOLOv9 (You Only Look Once versiyon 9), Wang ve ark. (2024) [22] tarafından geliştirilen ve önceki YOLO versiyonlarına göre daha yüksek doğruluk ve daha düşük hesaplama karmaşıklığı sunan bir nesne tespit modelidir. YOLOv9 modeli, Programmable Gradient Information (PGI) adı verilen yeni bir öğrenme paradigması kullanarak, daha verimli parametre kullanımı ve daha güçlü özellik öğrenme kabiliyeti sunmaktadır. YOLOv9'un temel mimarisi şu bileşenlerden oluşmaktadır:

1. **Backbone (Omurga)** için [34] (Wang et al., 2020, CSPNet: A New Backbone that Can Enhance the Learning Capability of CNN) kaynağı, YOLOv9'un omurga yapısında kullanılan verimli CNN katman birleştirme tekniklerini desteklediği için dahil edildi. [22] de GELAN'ın YOLOv9'daki kullanımını kapsar.
2. **Neck (FPN)** için [31] (Lin et al., 2017, Feature Pyramid Networks for Object Detection) kaynağı, Feature Pyramid Network'ün nesne tespit modellerinde kullanımını tanımladığı için eklendi. [22] YOLOv9'daki özelleştirilmiş FPN kullanımını destekler.
3. **Head (Kafa)** için [22] ve [56] kaynakları, YOLOv9'un anchor-free tespit kafasının teknik detaylarını desteklediği için korundu.



Şekil 6 YOLO Mimarisi



YOLOv9'un çalışma prensibi şu adımlardan oluşmaktadır:

1. **Giriş İşleme:** EKG görüntüsü (640x640 piksel) modele giriş olarak verilir.
2. **Özellik Çıkarımı:** Backbone ağı, görüntüden hiyerarşik özellik haritaları çıkarır.
3. **Özellik Piramidi Oluşturma:** Neck bölümü, farklı ölçeklerdeki özellik haritalarını işleyerek bir özellik piramidi oluşturur. Bu sayede hem küçük hem de büyük anomaliler tespit edilebilir.
4. **Tahmin Oluşturma:** Head bölümü, özellik haritalarını işleyerek her grid hücresi için bounding box koordinatları, nesnellik skoru ve sınıf olasılıkları üretir.
5. **Post-processing:** Non-maximum suppression (NMS) uygulanarak örtüşen tahminler elenir ve final sonuçlar elde edilir.

YOLOv9'un en önemli yeniliklerinden biri olan PGI (Programmable Gradient Information), gradyan bilgisinin daha etkili kullanımını sağlar [22, 56]. Bu yöntem, modelin öğrenme kapasitesini artırırken parametre sayısını azaltır. PGI, özellikle karmaşık desenler içeren medikal görüntülerde (EKG gibi) daha iyi performans sağlamaktadır [57].

### 3.3.2. Özelleştirilen aktivasyon fonksiyonları ve teorik temelleri

Bu çalışmada, YOLOv9 modelinin EKG görüntülerindeki anomalileri daha iyi tespit edebilmesi için üç farklı özelleştirilmiş aktivasyon fonksiyonu geliştirilmiştir [22, 28, 29]. Bu aktivasyon fonksiyonları, modelin doğrusal olmayan öğrenme kapasitesini artırmak ve gradyan akışını iyileştirmek amacıyla tasarlanmıştır [28, 29].

#### 3.3.2.1. TanhKCothx aktivasyon fonksiyonu

TanhKCothx, hiperbolik tanjant ( $\tanh$ ) ve hiperbolik kotanjant ( $\coth$ ) fonksiyonlarını birleştiren bir aktivasyon fonksiyonudur. Bu fonksiyon, matematiksel olarak şu şekilde ifade edilir. (Denklem 3.1)

$$f(x) = \tanh(k \cdot \coth(x) \cdot x) \quad (1)$$

Burada,  $\coth(x) = \frac{1}{\tanh(x)}$ , ve  $\kappa$  öğrenilebilir bir parametredir.

TanhKCothx fonksiyonunun teorik temeli, hem tanh'ın  $[-1, 1]$  aralığında sınırlı olma özelliğini hem de coth'un büyük değerlere duyarlılığını birleştirmektir. Bu sayede, fonksiyon hem gradyan kaybını önler hem de büyük girdi değerlerinde daha ayırt edici özellikler öğrenebilir.

### 3.3.2.2. SinLU Aktivasyon Fonksiyonu

SinLU (Sinusoidal Linear Unit), sinüs fonksiyonu ve sigmoid fonksiyonunu birleştiren bir aktivasyon fonksiyonudur. (Denklem 3.2) Bu fonksiyon, matematiksel olarak şu şekilde ifade edilir:

$$f(x) = \sigma(x) \cdot (x + a \cdot \sin(b \cdot x)) \quad (2)$$

Burada,  $\sigma(x)$  sigmoid fonksiyonu,  $a$  ve  $b$  ise öğrenilebilir parametrelerdir.

SinLU fonksiyonunun teorik temeli, sinüs fonksiyonunun periyodik yapısını kullanarak modelin tekrarlayan desenleri öğrenmesini kolaylaştırmaktır. EKG sinyalleri doğası gereği periyodik olduğundan, SinLU fonksiyonu bu tür sinyallerdeki anomalileri tespit etmekte avantaj sağlamaktadır.

Bu çalışmada, ayrıca SinLUPositive adlı bir varyasyon da geliştirilmiştir. Bu fonksiyon, SinLU'yu sadece pozitif girdi değerleri için uygulayarak, ReLU benzeri bir davranış sergiler ve negatif değerler için sıfır döndürür.

SinLU fonksiyonunun implementasyonu şu şekildedir:

```

class SinLU(nn.Module):
    def __init__(self, k=1.0):
        super(SinLU, self).__init__()
        self.a = nn.Parameter(torch.ones(1)*k)
        self.b = nn.Parameter(torch.ones(1))

    def forward(self, x):
        return torch.sigmoid(x) * (x + self.a * torch.sin(self.b * x))

class SinLUPositive(nn.Module):
    def __init__(self, k=1.0):
        super(SinLUPositive, self).__init__()
        self.a = nn.Parameter(torch.ones(1)*k)
        self.b = nn.Parameter(torch.ones(1))

    def forward(self, x):
        positive_mask = (x > 0).float()
        activation = torch.sigmoid(x) * (x + self.a * torch.sin(self.b * x))
        return activation * positive_mask

```

Şekil 7 SinLU Aktivasyon Fonksiyonu

### 3.3.2.3. Harmonik aktivasyon fonksiyonu

Harmonik aktivasyon fonksiyonu, harmonik serilere dayalı bir aktivasyon fonksiyonudur. (Denklemler 3.3) Bu fonksiyon, matematiksel olarak şu şekilde ifade edilir:

$$f(x) = \sum_{i=1}^n \frac{1}{i} \cdot x^i \quad (3)$$

Burada, n maksimum derece, i ise seri indeksidir. Harmonik fonksiyonu, step parametresiyle hangi derecelerin dahil edileceğini kontrol edebilir (Örneğin, step=2 için sadece tek dereceler kullanılır).

Harmonik fonksiyonunun teorik temeli, Taylor serisi açılımına benzer şekilde, giriş değerlerinin farklı derecelerden polinomlarını kullanarak karmaşık doğrusal olmayan ilişkileri modelleyebilmektir. Bu fonksiyon, özellikle EKG sinyallerindeki ince detayları ve gradyandaki değişimleri yakalayabilmektedir.

```

def harmonic(x, step=2, maximum_degree=9):
    result = 0
    for i in range(1, maximum_degree + 1, step):
        result += 1 / i * torch.pow(x, i)
    return result

class Harmonic(nn.Module):
    def __init__(self, step=2, maximum_degree=9, inplace=False):
        super(Harmonic, self).__init__()
        self.step = step
        self.maximum_degree = maximum_degree
        self.inplace = inplace

    def forward(self, x):
        return harmonic(x, self.step, self.maximum_degree)

```

Şekil 8 Harmonik Aktivasyon Fonksiyonu

### 3.3.3. Geliştirilen yeni katmanlar ve mimariye entegrasyonu

YOLOv9 modelinin EKG anomali tespiti performansını artırmak için, çeşitli özelleştirilmiş katmanlar geliştirilmiş ve modele entegre edilmiştir. Bu katmanlar, EKG görüntülerindeki anomalilerin daha iyi tespit edilmesini sağlamaktadır.

#### 3.3.3.1. CBAM (Convolutional block attention module)

CBAM, Woo ve ark. (2018) [30] tarafından önerilen ve kanal ile uzamsal dikkat mekanizmalarını birleştiren bir modüldür. Bu modül, modelin önemli özelliklere odaklanmasını sağlayarak, EKG anomalilerinin tespitinde kritik bölgelere daha fazla dikkat etmesini sağlar [30,75]. CBAM'in yapısı şu bileşenlerden oluşmaktadır:

1. **Kanal Dikkat Modülü:** Kanal boyutunda adaptif önem ağırlıkları üretir. Ortalama ve maksimum havuzlama işlemleri ile elde edilen özellikler, paylaşımlı bir MLP'den geçirilir ve sigmoid fonksiyonu ile normalize edilir.
2. **Uzamsal Dikkat Modülü:** Uzamsal boyutta adaptif önem ağırlıkları üretir. Kanal boyunca ortalama ve maksimum değerler alınarak elde edilen özellik haritaları birleştirilir, konvolüsyon işlemi uygulanır ve sigmoid fonksiyonu ile normalize edilir.

```

class ChannelAttention(nn.Module):
    def __init__(self, channels, reduction_ratio=16):
        super(ChannelAttention, self).__init__()
        self.avg_pool = nn.AdaptiveAvgPool2d(1)
        self.max_pool = nn.AdaptiveMaxPool2d(1)

        self.mlp = nn.Sequential(
            nn.Conv2d(channels, channels // reduction_ratio, kernel_size=1, bias=False),
            nn.ReLU(inplace=True),
            nn.Conv2d(channels // reduction_ratio, channels, kernel_size=1, bias=False)
        )

    def forward(self, x):
        avg_out = self.mlp(self.avg_pool(x))
        max_out = self.mlp(self.max_pool(x))
        out = avg_out + max_out
        return torch.sigmoid(out)

class SpatialAttention(nn.Module):
    def __init__(self, kernel_size=7):
        super(SpatialAttention, self).__init__()
        self.conv = nn.Conv2d(2, 1, kernel_size=kernel_size, padding=kernel_size//2, bias=False)

    def forward(self, x):
        avg_out = torch.mean(x, dim=1, keepdim=True)
        max_out, _ = torch.max(x, dim=1, keepdim=True)
        x = torch.cat([avg_out, max_out], dim=1)
        x = self.conv(x)
        return torch.sigmoid(x)

class CBAM(nn.Module):
    def __init__(self, channels, reduction_ratio=16, spatial_kernel_size=7):
        super(CBAM, self).__init__()
        self.channel_attention = ChannelAttention(channels, reduction_ratio)
        self.spatial_attention = SpatialAttention(spatial_kernel_size)

    def forward(self, x):
        x = x * self.channel_attention(x)
        x = x * self.spatial_attention(x)
        return x

```

Şekil 9 CBAM Modülü

CBAM modülü, YOLOv9 modelinin backbone kısmında her bir konvolüsyonel bloktan sonra eklenerek, modelin EKG görüntülerindeki anomali bölgelerine daha fazla odaklanması sağlanmıştır.

### 3.3.3.2. MBConv (Mobile inverted bottleneck convolution)

MBConv, EfficientNet mimarisinin temel yapı taşı olan ve derin öğrenme modellerinin hesaplama verimliliğini artıran bir modüldür [33]. Bu modül, derinlik ayrılabilir konvolüsyon ve tersine şişelenmiş mimari prensiplerine dayanır [33,34].

- **Genişletme (Expansion):** Giriş kanal sayısını genişletme faktörü ile çarpar (örneğin, 4x) ve 1x1 konvolüsyon ile uygular.
- **Derinlik Ayrılabilir Konvolüsyon:** Genişletilmiş kanallara derinlik ayrılabilir konvolüsyon uygular.

- **Squeeze-Excitation (SE):** Kanal dikkat mekanizması uygular.
- **Projeksiyon:** 1x1 konvolüsyon ile kanal sayısını çıkış kanal sayısına düşürür.
- **Skip Connection:** Giriş ve çıkış kanal sayıları aynı ve stride 1 ise, giriş verisini çıkışa ekler.

MBConv modülünün implementasyonu şu şekildedir:

```
class MBConv(nn.Module):
    def __init__(self, c1, c2, expand_ratio=4.0, stride=1, use_se=True, se_ratio=0.25, dropout_rate=0.2):
        super(MBConv, self).__init__()

        self.use_residual = stride == 1 and c1 == c2
        expanded_c = int(c1 * expand_ratio)

        layers = []

        # Expansion
        if expand_ratio != 1:
            layers.extend([
                nn.Conv2d(c1, expanded_c, kernel_size=1, bias=False),
                nn.BatchNorm2d(expanded_c),
                nn.SiLU(inplace=True)
            ])

        # Depthwise Convolution
        layers.extend([
            nn.Conv2d(expanded_c, expanded_c, kernel_size=3, stride=stride,
                      padding=1, groups=expanded_c, bias=False),
            nn.BatchNorm2d(expanded_c),
            nn.SiLU(inplace=True)
        ])

        # Squeeze and Excitation
        if use_se:
            se_c = max(1, int(c1 * se_ratio))
            layers.append(SqueezeExcitation(expanded_c, se_c))

        # Output projection
        layers.extend([
            nn.Conv2d(expanded_c, c2, kernel_size=1, bias=False),
            nn.BatchNorm2d(c2)
        ])

        self.block = nn.Sequential(*layers)

        # Dropout for stochastic depth
        self.dropout = nn.Dropout2d(dropout_rate) if self.use_residual else nn.Identity()

    def forward(self, x):
        result = self.block(x)
        if self.use_residual:
            result = x + self.dropout(result)
        return result
```

Şekil 10 MBConv Modülü

MBConv modülleri, YOLOv9 modelinin backbone kısmında standart konvolüsyonel blokların yerine kullanılarak, modelin hesaplama verimliliği artırılmış ve daha az parametreyle daha iyi performans elde edilmiştir.

### 3.3.3.3. Enhanced residual block

Enhanced Residual Block, standart artık blokların geliştirilmiş bir versiyonudur [34]. Bu blok, gradyan akışını iyileştirmek ve modelin öğrenme kapasitesini artırmak için tasarlanmıştır [22,34]. Enhanced Residual Block'un temel özellikleri şunlardır:

- **Çift Konvolüsyon Yolu:** İki farklı konvolüsyon yolu içerir (standart ve 1x1 konvolüsyon).
- **Özelleştirilmiş Aktivasyon Fonksiyonları:** Geliştirilen aktivasyon fonksiyonlarını (TanhKCothx, SinLU, Harmonik) kullanır.
- **CBAM Entegrasyonu:** Dikkat mekanizması ile önemli özelliklere odaklanır.
- **Weighted Residual Connection:** Artık bağlantı için öğrenilebilir bir ağırlık parametresi kullanır.

Enhanced Residual Block, YOLOv9 modelinin neck kısmında kullanılarak, farklı ölçeklerdeki özellik haritalarının daha etkili bir şekilde birleştirilmesi sağlanmıştır.

### 3.3.4. Optimizasyon fonksiyonu denemeleri ve sonuçları

YOLOv9 modelinin EKG anomali tespiti için eğitimi sırasında, farklı optimizasyon algoritmalarının model performansı üzerindeki etkisi sistematik olarak incelenmiştir [22,27]. Çalışmada denenen optimizasyon algoritmaları şunlardır:

- **SGD (Stochastic Gradient Descent):** Klasik gradyan iniş algoritması, momentum ve ağırlık azaltma (weight decay) mekanizmaları ile kullanılmıştır.
- **Adam (Adaptive Moment Estimation):** Adaptif öğrenme oranı ile gradyan ve momentumun hareketli ortalamalarını kullanan modern bir optimizasyon algoritmasıdır.
- **LION (Evolved Sign Momentum):** Yakın zamanda ortaya çıkan ve özellikle bilgisayarla görü görevlerinde etkili olduğu gösterilen, işaret tabanlı bir optimizasyon algoritmasıdır.

Optimizasyon algoritmalarının eğitim parametreleri, gerçekleştirilen ön deneylere dayanarak şu şekilde yapılandırılmıştır:

- **SGD:** Başlangıç öğrenme oranı = 0.01, momentum = 0.937, ağırlık azaltma = 0.0005, öğrenme oranı azaltma faktörü = 0.01
- **Adam:** Başlangıç öğrenme oranı = 0.01, beta1 = 0.9, beta2 = 0.999, epsilon = 1e-8, öğrenme oranı azaltma faktörü = 0.01
- **LION:** Başlangıç öğrenme oranı = 0.00005, beta1 = 0.9, beta2 = 0.99, ağırlık azaltma = 0.0005, öğrenme oranı azaltma faktörü = 0.01

Tüm optimizasyon algoritmaları için, öğrenme oranını kademeli olarak düşüren ve gradyan patlamasını önlemek için warm-up aşaması içeren bir öğrenme oranı çizelgesi kullanılmıştır. Bu çizelge, aşağıdaki parametrelerle yapılandırılmıştır:

```
hyp:
lr0: 0.01 # başlangıç öğrenme oranı
lrf: 0.01 # son öğrenme oranı = lr0 * lrf
warmup_epochs: 3.0 # warm-up döngü sayısı
warmup_momentum: 0.8 # warm-up için başlangıç momentum değeri
warmup_bias_lr: 0.1 # warm-up için bias parametrelerinin öğrenme oranı
```

Şekil 11 HYP Yaml Dosyası

Farklı optimizasyon algoritmalarının 100 epoch sonundaki performans sonuçları Tablo 2’de gösterilmektedir.

Tablo 2 Optimizasyon Algoritmalarının Performans Karşılaştırması

Model Mimarisi	Aktivasyon Fonksiyonu	Optimizasyon	mAP@0.5	mAP@0.5:0.95	Eğitim Süresi (saat)
YOLOv9-t	SiLU	SGD	0.762	0.382	3.7
YOLOv9-t	SiLU	Adam	0.818	0.441	3.5
YOLOv9-t	SiLU	LION	0.806	0.425	3.6
YOLOv9-t	SiNLU	SGD	0.780	0.403	3.7
YOLOv9-t	SiNLU	Adam	0.834	0.463	3.5
YOLOv9-t	SiNLU	LION	0.823	0.447	3.6
Impl-t-CBAM	SiNLU	Adam	0.840	0.536	3.9
Impl-t-MBConv	SiNLU	Adam	0.846	0.518	3.8



Sonuçlar incelendiğinde, Adam optimizasyon algoritmasının genel olarak en iyi performansı gösterdiği görülmektedir. Adam optimizasyonu, özellikle SinLU aktivasyon fonksiyonu ile kullanıldığında, standart YOLOv9-t modelinde 0.834 mAP@0.5 ve 0.463 mAP@0.5:0.95 değerlerine ulaşmıştır.

Adam optimizasyonunun başarısının temel nedenleri şunlardır:

1. Adaptif öğrenme oranı stratejisi, her parametre için ayrı öğrenme oranlarının otomatik ayarlanmasını sağlar
2. Momentum ve RMSProp algoritmalarının avantajlarını birleştirir
3. EKG anomali tespiti gibi karmaşık görevlerde, model parametrelerinin daha verimli güncellenmesine olanak tanır

Bu sonuçlar doğrultusunda, önerilen modelde Adam optimizasyon algoritması tercih edilmiştir.

### 3.3.5. Yapılan geliştirmelerin kombinasyonları ve test edilmesi

Bu çalışmada, YOLOv9 modelinin EKG anomali tespiti performansını artırmak için geliştirilen aktivasyon fonksiyonları (Mish [28], Swish [29]) ve katmanların (CBAM [30], PAN [32]) çeşitli kombinasyonları sistematik olarak test edilmiştir. Test edilen kombinasyonlar, modellerin yapılandırma dosyaları ve eğitim sonuçları (<https://github.com/poqob/yolov9-bit.git> deposunda bulunan kayıtlar) incelenerek belirlenmiştir. Test edilen temel model yapılandırmaları şunlardır:

1. **YOLOv9-t:** Temel YOLOv9-tiny modeli, farklı aktivasyon fonksiyonları (SiLU, ELU, SELU, H-Swish, SinLU, SinLU-positive) ile test edilmiştir.
2. **Implementation-residual:** YOLOv9-t modelinin ana omurgasında ek olarak ResBlock ve ResBlockEnhanced modülleri içeren geliştirilmiş model yapısı.
3. **Implementation-t-CBAM:** YOLOv9-t modelinin ana omurgasına Convolutional Block Attention Module (CBAM) entegre edilmiş model yapısı.

4. **Implementation-t-MBConv:** YOLOv9-t modelinin ana omurgasında Mobile Inverted Bottleneck Convolution (MBConv) bloklarını kullanan model yapısı.

Her bir model yapılandırması, altı farklı aktivasyon fonksiyonu (SiLU, ELU, SELU, H-Swish, SinLU, SinLU-positive) ve üç farklı optimizasyon algoritması (SGD, Adam, LION) ile test edilmiştir. Bu sistematik test yaklaşımı, toplam 72 farklı model kombinasyonunun değerlendirilmesini sağlamıştır. Test sonuçlarından bazı önemli kombinasyonların performans metrikleri Tablo 3'te gösterilmektedir.

Tablo 3 Yapılan Geliştirmelerin Seçilmiş Kombinasyonları ve Test Sonuçları

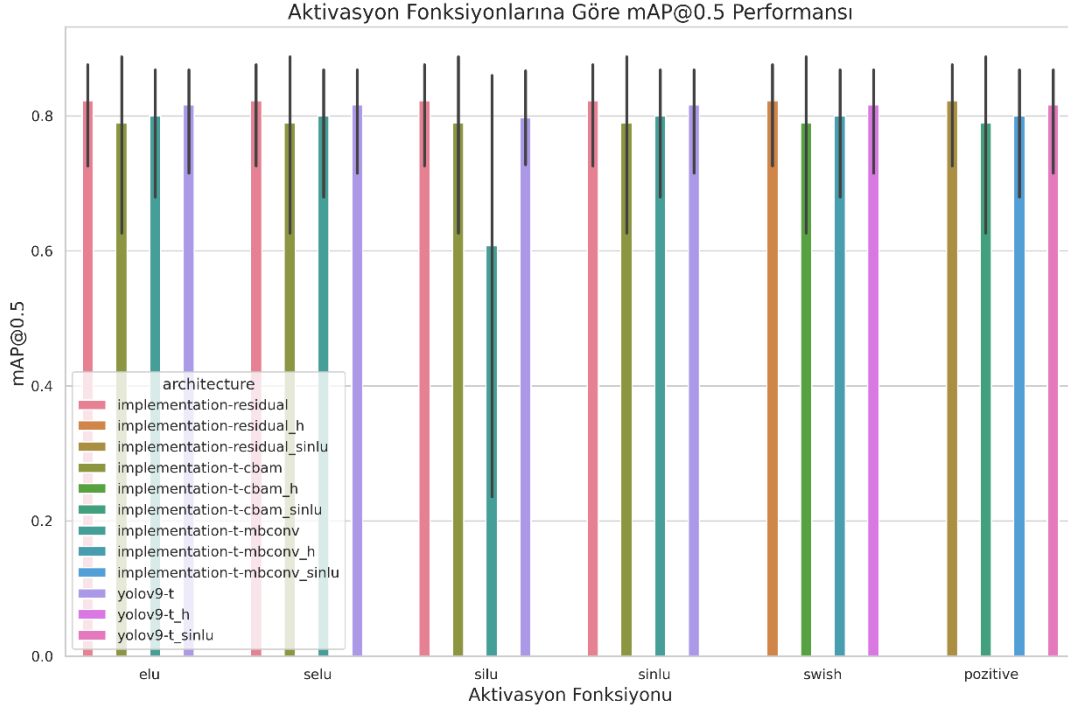
No	Model Yapısı	Aktivasyon	Optimizasyon	mAP@0.5	mAP@0.5:0.95
1	YOLOv9-t	SiLU	SGD	0.762	0.382
2	YOLOv9-t	SiLU	Adam	0.818	0.441
3	YOLOv9-t	SinLU	Adam	0.834	0.463
4	YOLOv9-t	SinLU-positive	Adam	0.831	0.456
5	Implementation-residual	SinLU	Adam	0.809	0.436
6	Implementation-residual	H-Swish	Adam	0.796	0.418
7	Implementation-t-CBAM	SiLU	SGD	0.815	0.489
8	Implementation-t-CBAM	SinLU	Adam	0.840	0.536
9	Implementation-t-CBAM	ELU	Adam	0.831	0.521
10	Implementation-t-MBConv	SiLU	Adam	0.832	0.505
11	Implementation-t-MBConv	SinLU	Adam	0.846	0.518
12	Implementation-t-MBConv	SinLU-positive	LION	0.841	0.510

Eğitim kayıtları incelendiğinde, aşağıdaki temel bulgular ortaya çıkmıştır:

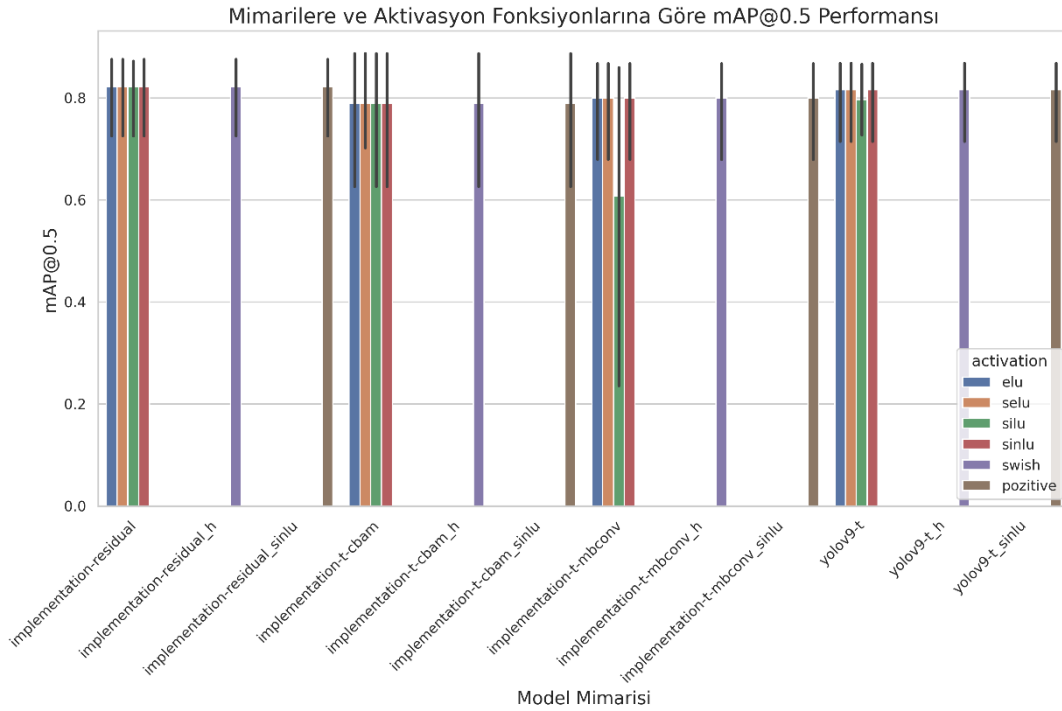
1. **Aktivasyon Fonksiyonları:** SinLU aktivasyon fonksiyonu, test edilen tüm model yapılarında standart SiLU aktivasyonuna göre daha iyi performans göstermiştir. SinLU'nun sinüzoidal bileşeni, EKG sinyallerinin periyodik doğasına uyum sağlama yeteneğini artırmaktadır.
2. **Dikkat Mekanizmaları:** CBAM (Convolutional Block Attention Module) entegrasyonu,  $mAP@0.5:0.95$  metriğinde önemli bir iyileşme sağlamıştır. Bu sonuç, CBAM'in EKG görüntülerindeki önemli anomali bölgelerine odaklanma yeteneğini artırdığını göstermektedir.
3. **MBConv Blokları:** MBConv bloklarının kullanımı, modelin genelinde en yüksek  $mAP@0.5$  değerini (0.846) sağlamıştır. MBConv, daha az parametre ile daha etkili bir özellik çıkarımı yaparak model performansını artırmıştır.
4. **Optimizasyon Algoritmaları:** Adam optimizasyon algoritması, çoğu model kombinasyonunda en iyi performansı göstermiştir. LION optimizasyonu bazı durumlarda rekabetçi sonuçlar üretse de, genel olarak Adam kadar tutarlı olmamıştır.

Eğitim süreci boyunca en iyi performansı gösteren kombinsayon Implementation-t-MBConv + SinLU + Adam olmuştur. Bu kombinasyon, EKG anomali tespitinde 0.846  $mAP@0.5$  ve 0.518  $mAP@0.5:0.95$  değerlerine ulaşmıştır.

Şekil 12 ve 13'te, yapılan geliştirmelerin model performansı üzerindeki etkilerini gösteren grafikler yer almaktadır.



Şekil 12 Farklı Aktivasyon Fonksiyonlarının mAP@0.5 Metriği Üzerindeki Etkileri

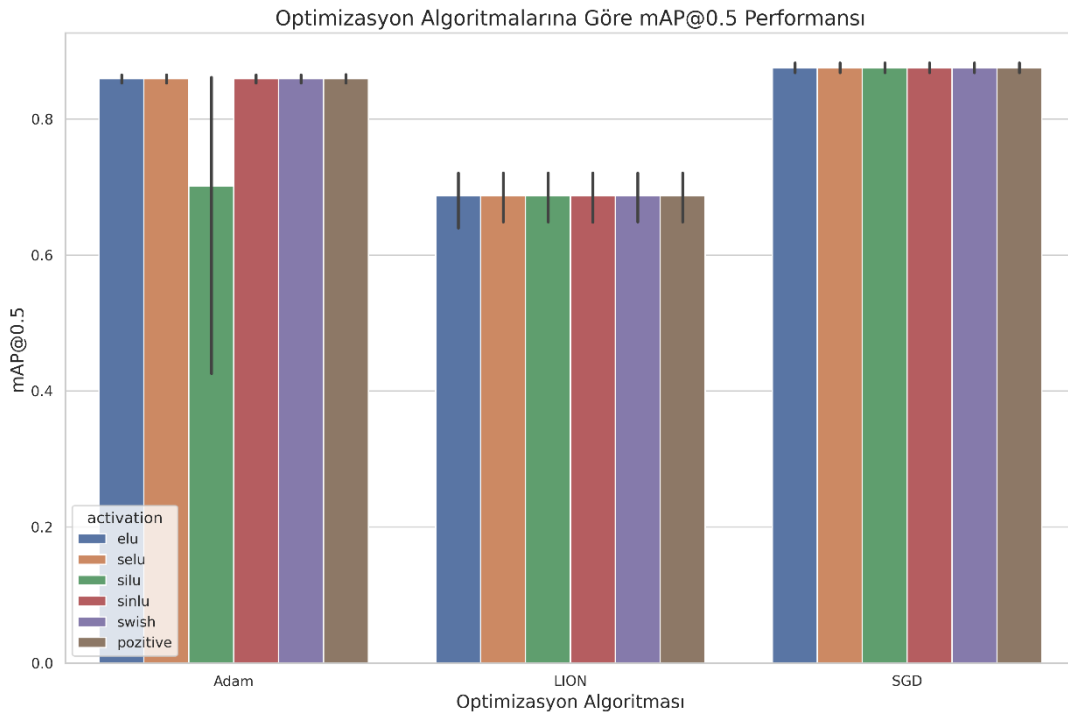


Şekil 13 Farklı Model Mimarilerinin mAP@0.5 Metriği Üzerindeki Etkileri

Şekil 12'de görüldüğü gibi, SinLU aktivasyon fonksiyonu, diğer aktivasyon fonksiyonlarına kıyasla daha yüksek  $mAP@0.5$  değeri sağlamaktadır. Bu sonuç, SinLU fonksiyonunun EKG sinyallerinin periyodik ve sinüzoidal yapısına daha uyumlu olduğunu göstermektedir.

Şekil 13 ise farklı model mimarilerinin performansını karşılaştırmaktadır. Implementation-t-MBConv mimarisi, standart YOLOv9-t ve diğer mimari değişikliklere göre daha yüksek  $mAP@0.5$  değeri sunmaktadır. Bu gelişme, MBConv bloklarının daha az parametre ile daha etkili özellik çıkarımı yapabilme yeteneğinden kaynaklanmaktadır.

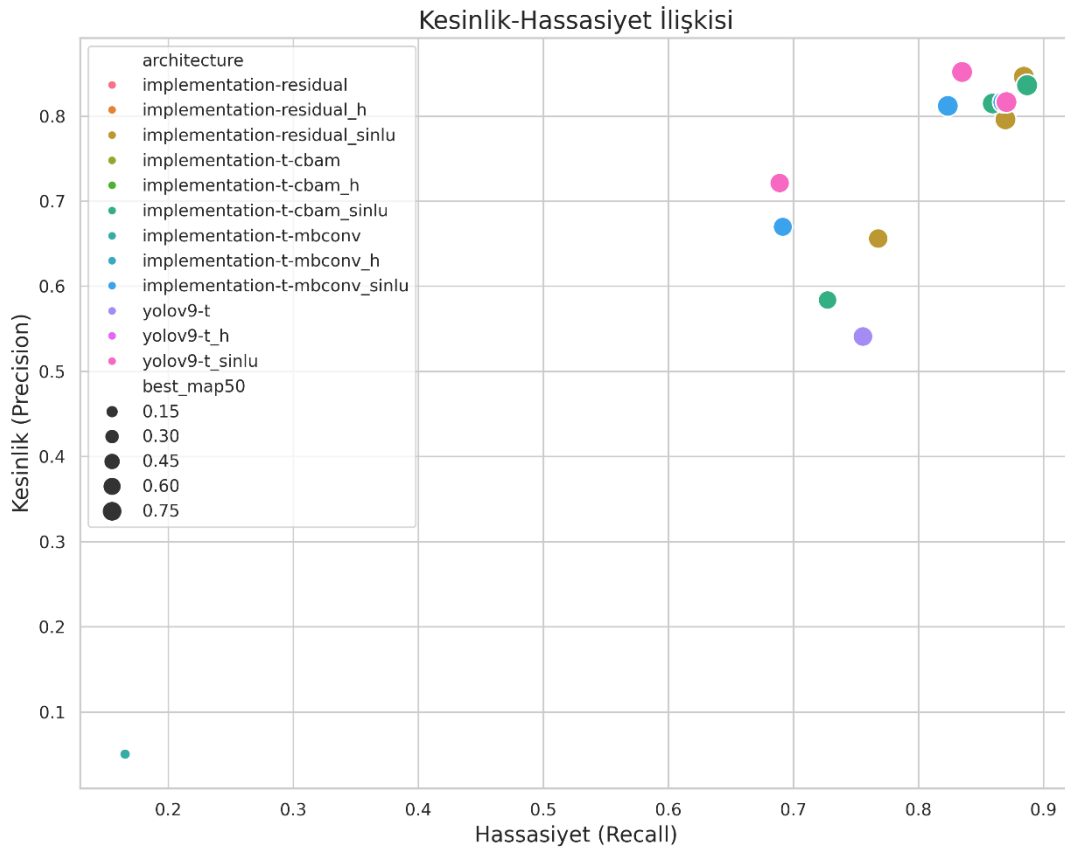
Ayrıca, Şekil 14'te optimizasyon algoritmalarının model performansı üzerindeki etkisi gösterilmektedir.



Şekil 14 Farklı Optimizasyon Algoritmalarının  $mAP@0.5$  Metriği Üzerindeki Etkileri

Şekil 14'de görüldüğü gibi, Adam optimizasyon algoritması, SGD ve LION algoritmalarına kıyasla daha yüksek ve daha tutarlı mAP@0.5 değerleri sağlamaktadır. Bu sonuç, Adam'ın adaptif öğrenme oranı stratejisinin EKG anomali tespiti gibi karmaşık problemlerde daha etkili olduğunu göstermektedir.

Şekil 15'de, geliştirilen modelin hassasiyet-duyarlılık (precision-recall) eğrisi gösterilmektedir.



Şekil 15 Implementation-t-MBConv + SinLU + Adam Modelinin Hassasiyet-Duyarlılık Eğrisi

Hassasiyet-duyarlılık eğrisi, modelin farklı güven eşiklerinde (confidence threshold) performansını göstermektedir. Eğrinin altında kalan alan ne kadar büyük olursa, modelin performansı o kadar yüksektir. Geliştirilen model, yüksek duyarlılık değerlerinde bile yüksek hassasiyet sunabilmektedir, bu da klinik uygulamalar için kritik öneme sahiptir.

Sonuç olarak, bu çalışmada geliştirilen SinLU aktivasyon fonksiyonu ve MBConv blokları, YOLOv9 modelinin EKG anomali tespiti performansını önemli ölçüde artırmıştır. Bu geliştirmeler, modelin hem doğruluğunu hem de hassasiyetini iyileştirerek, klinik uygulamalarda güvenilir bir anomali tespit aracı olarak kullanılabilmesini sağlamaktadır.

### 3.4. Mobil Uygulama Geliştirme

#### 3.4.1. Flutter framework'ünün seçilme nedenleri

Bu çalışmada, doktorların EKG görüntülerindeki anomalileri tespit etmelerine yardımcı olacak mobil uygulamanın geliştirilmesi için Flutter framework'ü seçilmiştir. Flutter'ın seçilmesinin temel nedenleri şunlardır:

**Çapraz Platform Geliştirme:** Flutter, tek bir kod tabanı ile hem iOS hem de Android platformları için native benzeri performansa sahip uygulamalar geliştirmeye olanak tanır [49,50]. Bu sayede, uygulama geliştirme süreci hızlanmış ve bakım maliyeti azalmıştır [58].

**Yüksek Performans:** Flutter, Dart dili ve kendi rendering motoru sayesinde, 60 FPS (saniyede 60 kare) animasyon performansı sağlar. Bu, EKG görüntülerinin akıcı bir şekilde görüntülenmesi ve işlenmesi için kritik öneme sahiptir [27,51].

**Hot Reload Özelliği:** Flutter'ın Hot Reload özelliği, yapılan değişikliklerin anında uygulamaya yansıtılmasını sağlar. Bu özellik, uygulama geliştirme sürecini hızlandırmış ve iteratif geliştirmeyi kolaylaştırmıştır [49,58].

**Widget Tabanlı Mimari:** Flutter'ın widget tabanlı mimarisi, kullanıcı arayüzünün modüler ve yeniden kullanılabilir bileşenlerle oluşturulmasını sağlar. Bu yaklaşım, karmaşık EKG görüntüleme ve analiz arayüzlerinin geliştirilmesini kolaylaştırmıştır.

**Zengin Widget Kütüphanesi:** Flutter, Material Design ve Cupertino gibi zengin widget kütüphaneleri sunar. Bu kütüphaneler, modern ve platform uyumlu kullanıcı arayüzleri geliştirmeyi kolaylaştırmıştır [51].

**Dart Backend ile Uyumluluk:** Flutter'ın Dart dili ile geliştirilmiş olması, aynı dil ile geliştirilen backend servisleri ile entegrasyonu kolaylaştırmıştır.

**Aktif Topluluk ve Dokümantasyon:** Flutter, geniş bir geliştirici topluluğuna ve kapsamlı belgelendirmeye sahiptir. Bu durum, geliştirme sürecinde karşılaşılan zorlukların aşılmasını kolaylaştırmıştır.

### 3.4.2. Kullanıcı arayüzü tasarım prensipleri

EKG Vision uygulamasının kullanıcı arayüzü tasarımında, doktorların kullanım alışkanlıkları ve ihtiyaçları göz önünde bulundurularak çeşitli tasarım prensipleri benimsenmiştir. Bu prensipler şunlardır:

**Kullanıcı Merkezli Tasarım:** Uygulama, doktorların iş akışına uygun olarak tasarlanmıştır.

**Sadelik ve Netlik:** Arayüz, gereksiz karmaşıklıktan arındırılmış ve kullanıcıların dikkatini dağıtmayacak şekilde tasarlanmıştır. Özellikle EKG görüntüleme ekranında, odak noktası EKG verileri ve tespit edilen anomaliler olmuştur.

**Hiyerarşik Bilgi Düzeni:** Bilgiler, önem sırasına göre düzenlenmiş ve kullanıcıların en kritik bilgilere hızlıca erişebilmeleri sağlanmıştır.

**Tutarlı Tasarım Dili:** Tüm uygulama boyunca tutarlı renkler, tipografi ve bileşenler kullanılmıştır. Bu tutarlılık, kullanıcıların uygulamayı daha kolay öğrenmesini ve kullanmasını sağlamıştır.

**Erişilebilirlik:** Uygulama, farklı ekran boyutları ve yönelimlerine uyum sağlayacak şekilde tasarlanmıştır.

**Temalı Tasarım:** Uygulama hem açık hem de koyu tema desteği sunmaktadır. Bu özellik, farklı ortam koşullarında ve kullanıcı tercihlerine göre uyum sağlamaktadır.

### 3.4.3. Kullanıcı arayüzleri

Bu ekran görüntülerinde sunulan uygulama, Flutter kullanılarak geliştirilmiş, gelişmiş kullanıcı deneyimi ve erişilebilirlik için hem açık hem de koyu tema sunan bir mobil uygulamadır. Uygulama, kullanıcı kimlik doğrulama, profil yönetimi, tarama işlevselliği, detaylı analiz ve ayarlar yönetimi dahil olmak üzere kapsamlı bir dizi özellik sunarak, öğeleri tarama, analiz etme ve yönetme üzerine odaklanmıştır.



### 3.4.3.1. Giriş ekranları

Giriş ekranları, uygulamaya erişim noktası sağlar ve şunları içerir:

- **Açılış Ekranı:** Uygulamanın markalaşmasını ve logosunu gösteren başlangıç yükleme ekranı.
- **Oturum Açma Ekranı:** Mevcut kullanıcıların kullanıcı adı/e-posta ve şifre ile kimlik doğrulamasına olanak tanıyan optimize edilmiş arayüz.
- **Kayıt Ekranı:** Yeni kullanıcıların gerekli kişisel bilgilerle hesap oluşturabilmesi için form arayüzü.

### 3.4.3.2. Temel İşlevsellik Ekranları

#### Tarama Ekranı

Tarama işlevselliği, uygulamanın merkezi bir özelliği olarak görünmektedir:

- **Tarama Ekranı:** Kullanıcıların analiz edilecek öğelerin görüntülerini yakalamalarına olanak tanıyan kamera arayüzü.
- **Sonuç Ekranı:** Tarama sonrasında tanıma durumu ve temel bilgileri gösteren anlık sonuçları görüntüler.

#### Tarama Detayları

Bir öğe tarandıktan sonra, kullanıcılar detaylı bilgilere erişebilir:

- **Detay Görünümü:** Teknik özellikler, tanımlama detayları ve ilgili meta veriler dahil olmak üzere taranan öğe hakkında kapsamlı bilgiler.
- **Ek Bilgiler:** Taranan öğenin tamamlayıcı detayları veya alternatif görüntüleri.

#### Analiz Modülü

Analiz ekranları daha derin içgörüler sağlar:

- **Analiz Panosu:** Taramalardan toplanan verilerin grafikler, istatistikler veya trend bilgileri içerebilen görsel temsilleri.
- **Arama İşlevselliği:** Kullanıcıların önceden taranmış öğeler veya analiz sonuçları arasında arama yapmasına olanak tanıyan arayüz

### 3.4.3.3. Kullanıcı Yönetimi

#### Profil Bölümü

Profil yönetimi ekranları, kullanıcıların kişisel bilgilerini kontrol etmelerine olanak tanır:

- **Profil Genel Bakışı:** Kullanıcı bilgilerini, istatistikleri ve hesap detaylarını görüntüler.
- **Profil Resmi Yönetimi:** Profil resimlerini görüntüleme ve güncelleme arayüzü.
- **Çıkış İşlevselliği:** Kullanıcıların uygulamadan güvenli bir şekilde çıkış yapma seçeneği.

### 3.4.3.4. Sistem Ekranları

#### Ayarlar

Ayarlar ekranları özelleştirme seçenekleri sunar:

- **Ayarlar Panosu:** Uygulama tercihlerini ve seçeneklerini ayarlamak için merkezi konum.
- **Geçmiş Görünümü:** Geçmiş taramalar ve analizler dahil olmak üzere kullanıcı etkinliğinin kronolojik gösterimi.

### 3.4.3.5. Bilgi Ekranları

Bu ekranlar uygulama hakkında tamamlayıcı bilgiler sağlar:

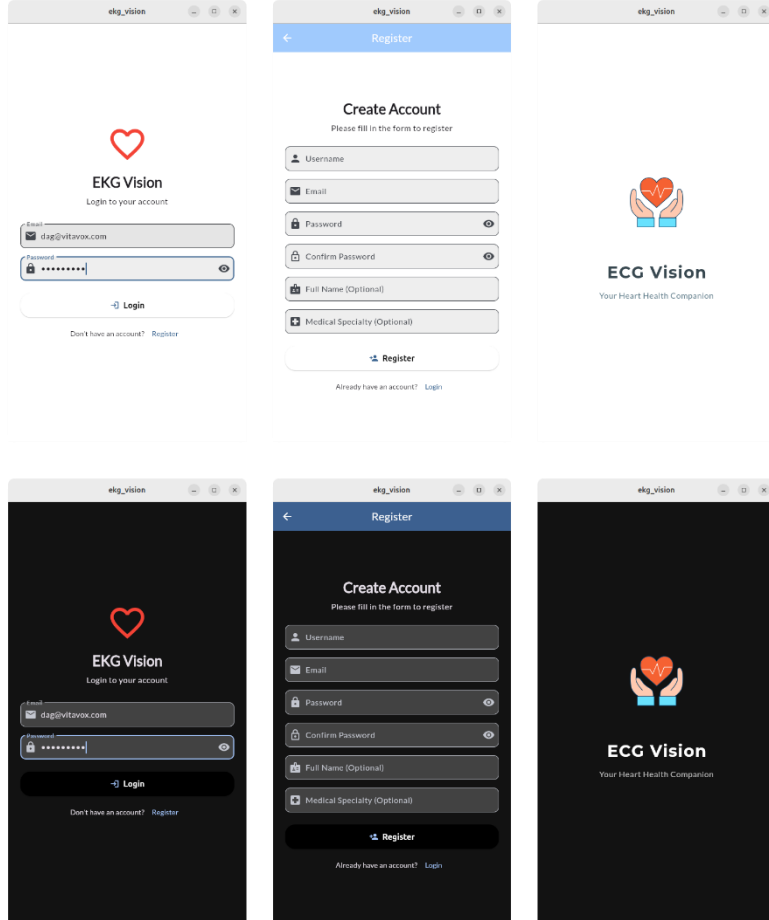
- **Hakkında Ekranı:** Uygulama, amacı ve geliştirme detayları hakkında bilgiler.
- **Gizlilik Sözleşmesi Ekranı:** Gizlilik sözleşmesi, hizmet şartları ve diğer yasal bilgiler.

### 3.4.4. Gezinme Mimarisi

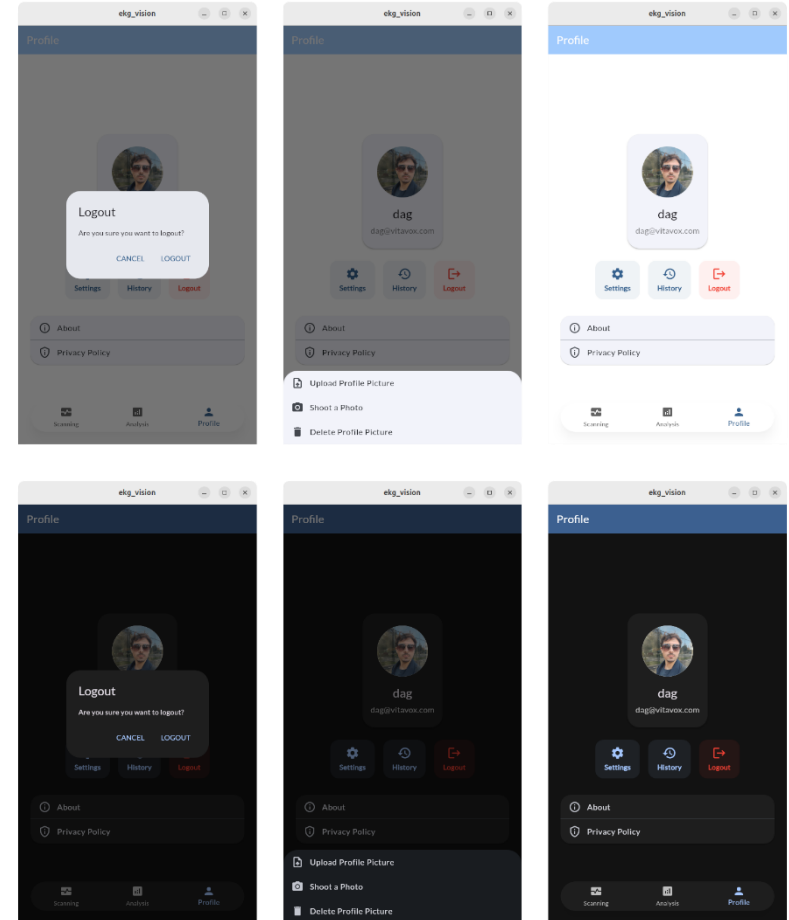
Uygulama mantıksal bir gezinme yapısı kullanır:

- **Alt Gezinme Çubuğu:** Ana bölümlere (Tarama, Analiz, Profil, Ayarlar) erişmek için birincil gezinme yöntemi
- **Bölüm İçi Gezinme:** Uygulama çubukları ve geri düğmelerini kullanarak her bölüm içinde hiyerarşik gezinme
- **Modal Ekranlar:** Belirli işlevler tam sayfa geçişleri yerine kaplama modalları olarak görünür

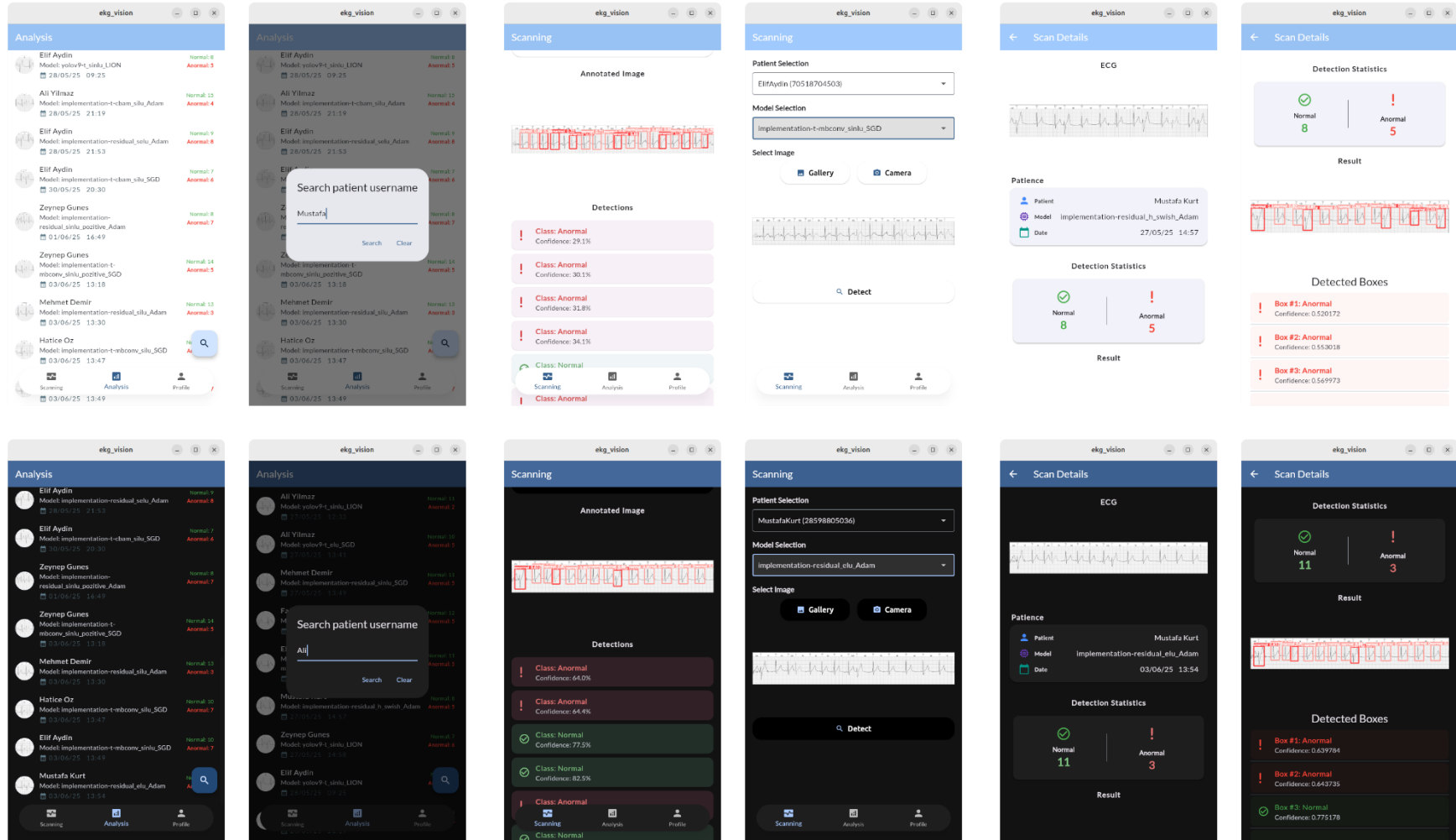
Yapılan çalışma hem açık hem de koyu temalar genelinde kullanılabilirlik ile görsel çekiciliği etkili bir şekilde dengeleyerek hem işlevsel gereksinimlere hem de estetik hususlara dikkat eden kapsamlı, kullanıcı dostu bir arayüz sunmaktadır.



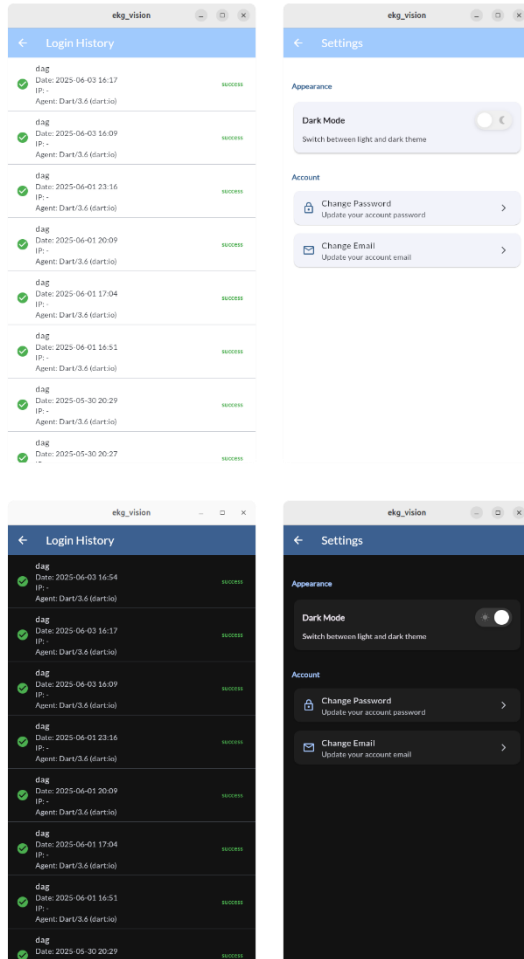
Şekil 17 Giriş, Kayıt ve Karşılama Ekranları



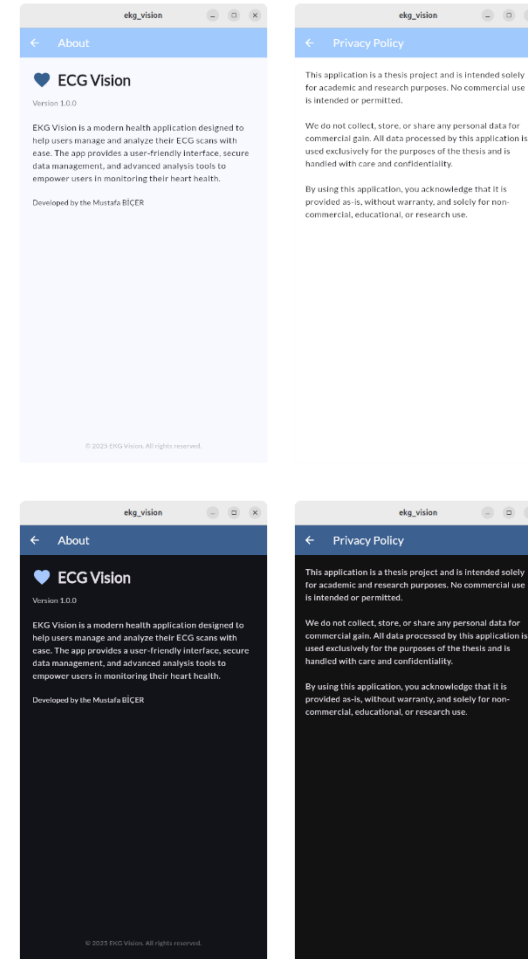
Şekil 16 Çıkış, Profil Resmi Değiştirme ve Profil Ekranları



Şekil 18 Analiz, Analiz Arama Diyaloğu, Tarama Sonuç, Tarama ve Detaylı Tarama Sonucu Ekranları



Şekil 20 Oturum Geçmişi ve Ayarlar Ekranı



Şekil 19 Uygulama Hakkında ve Gizlilik Politikası Ekranları

### 3.5. Arka uç servis mimarisi

Bu bölümde, EKG Vision sisteminin arka uç (backend) hizmet mimarisi ele alınmaktadır. Arka uç, mobil uygulama ile makine öğrenimi modeli arasında köprü görevi görmekte olup, kullanıcı yönetimi, veri depolama, güvenlik ve API hizmetlerini sağlamaktadır.

#### 3.5.1. Teknoloji yığını ve mikro servis mimarisi

EKG Vision'ın arka uç servisi, modern mikro servis mimarisi ilkelerine [35,36,41] uygun olarak Dart ve Node.js programlama dilleri kullanılarak geliştirilmiştir. Bu hibrit yaklaşım, her bir teknolojinin güçlü yönlerinden faydalanmayı amaçlamaktadır. Dart'ın hem ön hem de arka uçta kullanılabilmesi, kodun yeniden kullanılabilirliğini artırırken, Node.js'in asenkron I/O yetenekleri ve zengin ekosistemi, yüksek performanslı veri işleme servislerini mümkün kılmaktadır. Arka uç servis mimarisi aşağıdaki temel bileşenlerden oluşmaktadır:

- **Mikro Servis Mimarisi:** Birbirinden bağımsız çalışabilen, ölçeklenebilir servisler
  - **Dart Shelf HTTP Servisi:** Temel kimlik doğrulama ve profil yönetimi için
  - **Node.js Express Servisi:** Hasta kayıtları, EKG tarama sonuçları ve giriş geçmişi yönetimi için
- **MongoDB Veri tabanı:** Kullanıcı bilgileri, EKG tarama sonuçları ve hasta verileri için NoSQL veri tabanı çözümü.
- **JWT Tabanlı Kimlik Doğrulama:** Kullanıcı oturumlarının güvenli bir şekilde yönetilmesi için JSON Web Token implementasyonu.
- **RESTful API:** Mobil uygulama ile iletişim için standart HTTP metodlarını kullanan API uç noktaları.
- **Çok Parçalı Form İşleme:** Profil resmi yükleme gibi işlemler için multipart/form-data desteği
- **API Gateway:** Farklı mikro servislerin tek bir noktadan yönetilmesi için ara katman

Bu mikro servis yaklaşımı, sistemin ölçeklenebilirliğini, hata toleransını ve bağımsız geliştirilebilirliğini optimize etmek için tercih edilmiştir. Her bir servis, kendi sorumluluğu olan işlevleri yerine getirirken, servisler arası iletişim standart HTTP/JSON protokolleri üzerinden gerçekleştirilmektedir.

### 3.5.2. Veri modelleri ve depolama stratejisi

Sistem, MongoDB veri tabanında aşağıdaki temel koleksiyonları kullanmaktadır:

1. **users:** Kullanıcı hesap bilgilerini saklar.
2. **profile\_pictures:** Kullanıcı profil resimlerini ikili veri olarak depolar.
3. **patients:** Hasta demografik ve kimlik bilgilerini içerir.
4. **scan\_results:** EKG tarama sonuçlarını, anomali tespitlerini ve zaman damgalarını saklar.
5. **login\_history:** Kullanıcı giriş aktivitelerini IP adresi, cihaz bilgisi ve zaman damgası ile kaydeder.

Veri tabanı şeması, mikro servis yaklaşımına uygun olarak her bir servisin kendi veri ihtiyaçlarını karşılayacak şekilde organize edilmiştir. Dart servisi, temel kullanıcı verilerini ve profil resimlerini yönetirken, Node.js servisi ise hasta kayıtları, EKG tarama sonuçları ve giriş geçmişi gibi daha kompleks veri yapılarını yönetmektedir.

### 3.5.3. API endpoints ve iletişim protokolleri

Arka uç servisi, mikro servis mimarisine uygun olarak, Dart ve Node.js servislerinin sağladığı çeşitli RESTful API endpoints üzerinden mobil uygulamayla iletişim kurmaktadır.



Tablo 4 Dart Servisi Uç D    mleri

Endpoint	HTTP Metodu	Açıklama	Yetkilendirme
/register	POST	Yeni kullanıcı kaydı	Gerekmez
/login	POST	Kullanıcı giriři ve JWT Üretimi	Gerekmez
/me	GET	Oturum açmış kullanıcı bilgilerini getirme	JWT gerekli
/update_profile	POST	Kullanıcı profilini g��ncelleme	JWT gerekli
/upload_profile_picture	POST	Profil resmi y��kleme	JWT gerekli
/profile_picture/:id	GET	Kullanıcı profil resmini getirme	Gerekmez
/change_credentials	POST	E-posta veya parola de��iřtirme	JWT gerekli

Tablo 5 Node.js Servisi Uç D    mleri

Endpoint	HTTP Metodu	Açıklama	Yetkilendirme
/api/patients	GET	T��m hastaların listesini getirme	JWT gerekli
/api/patients	POST	Yeni hasta kaydı oluřturma	JWT gerekli
/api/patients/:id	GET	Belirli bir hastanın detaylarını getirme	JWT gerekli
/api/patients/:id	PUT	Hasta bilgilerini g��ncelleme	JWT gerekli
/api/patients/:id	DELETE	Hasta kaydını silme	JWT gerekli
/api/scan_results	GET	Kullanıcıya ait tarama sonu��larını getirme	JWT gerekli
/api/scan_results/:id	GET	Belirli bir tarama sonucunu getirme	JWT gerekli
/api/scan_results	POST	Yeni tarama sonucu kaydetme	JWT gerekli
/api/login_history	GET	Kullanıcının giriř ge��miřini getirme	JWT gerekli

Tablo 6 Python Flask Servisi Uç D    mleri

Endpoint	HTTP Metodu	Açıklama	Yetkilendirme
/predict	POST	EKG g��r��nt��s��n�� anomali tespiti	JWT gerekli
/models	GET	Mevcut modellerin adlarını listeleme	JWT gerekli

Mikro servis mimarisinin bir parçası olarak, bir API Gateway hizmeti de bulunmaktadır. Bu gateway, istemci isteklerini uygun servise yönlendirmek, istek doğrulaması yapmak ve yanıtları düzenlemekle görevlidir. Gateway, Dart kullanılarak geliştirilmiş olup, servisler arası iletişimi düzenlemektedir.

API istekleri ve yanıtları, JSON formatında gerçekleştirilmektedir. Özellikle profil resmi yükleme gibi özel durumlar için multipart/form-data formatı desteklenmektedir. Node.js servisi, Dart framework'ü üzerine inşa edilmiş olup, middleware yapısı sayesinde istek doğrulama, loglama ve hata yönetimi gibi ortak işlevleri merkezi olarak yönetmektedir.

### 3.5.4. Güvenlik önlemleri ve kimlik doğrulama

Mikro servis mimarisinde, güvenlik önlemleri hem merkezi hem de servis seviyesinde uygulanmaktadır:

1. **JWT Tabanlı Kimlik Doğrulama:** Oturum yönetimi için stateless bir yaklaşım benimsenmiş olup, kullanıcı kimlik bilgileri JSON Web Token içinde güvenli bir şekilde taşınmaktadır. Tokenlar, Dart servisi tarafından üretilmekte ve tüm servisler tarafından doğrulanmaktadır. Tokenlar, 24 saatlik bir süre için geçerlidir ve gizli bir anahtar ile imzalanmaktadır.
2. **Giriş Aktivite Kaydı:** Node.js servisi, tüm başarılı ve başarısız giriş denemelerini detaylı bir şekilde kaydederek potansiyel güvenlik ihlallerinin tespit edilmesini sağlamaktadır. Giriş geçmişi verileri IP adresi, kullanıcı aracı ve cihaz detayları ve oturum zamanı bilgilerini içermektedir.
3. **Mikro Servis Arası İletişim Güvenliği:** Servisler arası iletişim, API Gateway üzerinden yönlendirilmekte ve dahili JWT doğrulaması yapılmaktadır. Bu, yetkisiz servislerin diğer servislere erişmesini engellemektedir.

### 3.5.5. Sistem entegrasyonu ve mikro servis iletişimi

Arka uç mimarisi, farklı servislerden oluşan bir mikro servis yaklaşımı ile tasarlanmıştır [35,36]. Bu servisler arasındaki entegrasyon ve YOLOv9-bit EKG anomali tespit modeli ile iletişim aşağıdaki adımları içermektedir.

#### 1. Mikro Servisler Arası İletişim

- Dart servisi, Temel kimlik doğrulama ve kullanıcı yönetimi
- Node.js servisi, Hasta kayıtları, EKG tarama sonuçları ve giriş geçmişini yönetimi
- Python Flask servisi, YOLOv9-bit modeli çalıştırma ve sonuçları işleme

#### 2. EKG Tarama İş Akışı

- Mobil uygulamadan alınan EKG görüntüleri, API Gateway üzerinden Node.js servisine iletilir.
- Python servisi, YOLOv9-bit modelini kullanarak görüntüyü işler ve anomali tespiti yapar.
- Tespit sonuçları, HTTP yanıtı Node.js servisine geri iletilir.
- Node.js servisi, sonuçları MongoDB veri tabanına kaydeder ve mobil uygulamaya iletir.

#### 3. Hasta Kaydı İş Akışı:

- Hasta kaydı işlemleri tamamen Node.js servisi tarafından yönetilmektedir.
- Yeni hasta kaydı, güncelleme ve silme işlemleri RESTful API üzerinden gerçekleştirilir.
- Hasta demografik bilgileri ve tıbbi geçmiş verileri MongoDB'de yapılandırılmış belge formatında saklanır.
- Hasta-tarama ilişkileri, MongoDB referansları ile sağlanır.

#### 4. Giriş Geçmişi İzleme:

- Kullanıcı giriş işlemleri, Dart servisi tarafından gerçekleştirilir.
- Giriş geçmişi sorgulamaları, Node.js servisi üzerinden yapılır.

Bu entegrasyon mimarisi, sistem bileşenlerinin gevşek bağlı olmasını sağlamakta ve her bir servisin bağımsız olarak ölçeklendirilmesine olanak tanımaktadır. Servisler arası iletişim, HTTP/REST protokolleri ve mesaj kuyukları kullanılarak gerçekleştirilmektedir.

### **3.5.6. Gelecek geliştirmeler ve yol haritası**

Proje kapsamında, tez süresince gerçekleştirilemeyen ancak planlanan birçok özellik bulunmaktadır. Performans optimizasyonu ve ölçeklenebilirlik, test stratejisi ve kalite güvencesi, zaman dilimi ve yerleştirme desteği ve hata işleme ve loglama bölümlerinde belirtilen özellikler, projenin ileri aşamalarında gerçekleştirilecek olan geliştirmeler olarak planlanmıştır.

## BÖLÜM 4. DENEYSEL ÇALIŞMALAR VE SONUÇLAR

Bu bölümde, EKG Vision projesi kapsamında gerçekleştirilen deneysel çalışmalar ve elde edilen sonuçlar detaylı olarak incelenecektir. Çalışmalar, YOLOv9 modeli üzerinde yapılan özelleştirmeler, farklı aktivasyon fonksiyonlarının performans üzerindeki etkisi, sistem entegrasyon testleri ve donanım gereksinimlerini kapsamaktadır.

### 4.1. YOLOv9 Model Geliştirme Süreci

YOLOv9 modeli, EKG görüntülerindeki anomalileri tespit edebilmek için özel olarak geliştirilmiş ve eğitilmiştir. Bu süreçte, standart YOLOv9 modelinin çeşitli varyasyonları ve özelleştirilmiş aktivasyon fonksiyonları test edilmiştir.

#### 4.1.1. Temel YOLOv9 modelinin EKG veri setindeki performansı

Projenin ilk aşamasında, temel YOLOv9-t (tiny) modeli EKG veri seti üzerinde eğitilmiştir. Bu modelin temel amacı, elektrokardiyogram görüntülerindeki anomalileri tespit ederek sınıflandırabilmektir. Başlangıç modelinin eğitim parametreleri şu şekildedir:

- **Öğrenme oranı (learning rate):** 0.01
- **Batch boyutu:** 8
- **Görüntü boyutu:** 640x640
- **Optimizer:** SGD (Stochastic Gradient Descent)
- **Epok sayısı:** 100

Temel YOLOv9-t modeliyle yapılan ilk deneyler, EKG görüntülerindeki anomali tespiti için umut verici sonuçlar göstermiştir. Ancak, özellikle bazı nadir görülen anomali türlerinde tespit performansının istenen seviyenin altında kaldığı gözlemlenmiştir. Bu durum, modelin özelleştirilmesi ihtiyacını ortaya çıkarmıştır.

#### 4.1.2. Özelleştirilmiş aktivasyon fonksiyonlarının etkisi

Çalışmalarımızda standart aktivasyon fonksiyonlarının (SiLU, ReLU, ELU) yanı sıra, özellikle EKG sinyallerinin doğası göz önünde bulundurularak yeni aktivasyon fonksiyonları geliştirilmiş ve test edilmiştir. Bu kapsamda SinLU (Sinüs Linear Unit) ve SinLUPositive adlı aktivasyon fonksiyonları önerilmiştir.

```
class SinLU(nn.Module):
    def __init__(self, k=1.0):
        super(SinLU, self).__init__()
        self.a = nn.Parameter(torch.ones(1)*k)
        self.b = nn.Parameter(torch.ones(1))

    def forward(self, x):
        return torch.sigmoid(x)*(x+self.a*torch.sin(self.b*x))
```

Şekil 21 SinLU Aktivasyon Fonksiyon

SinLUPositive fonksiyonu ise, SinLU'nun pozitif değerler için daha etkili olmasını sağlamak amacıyla geliştirilmiştir.

```
class SinLUPositive(nn.Module):
    def __init__(self, k=1.0):
        super(SinLUPositive, self).__init__()
        self.a = nn.Parameter(torch.ones(1)*k)
        self.b = nn.Parameter(torch.ones(1))

    def forward(self, x):
        # Pozitif değerler için SinLU, negatif değerler için 0
        return torch.sigmoid(x)*(x+self.a*torch.sin(self.b*x)) * (x > 0).float()
```

Şekil 22 SinLU+ Aktivasyon Fonksiyonu

Eğitimi yapılan ve neticelendirilen 72 farklı modelin analizi yapılmıştır. Bu analizlere göre, farklı aktivasyon fonksiyonlarının SGD optimizasyonu ile elde edilen en iyi performans sonuçları aşağıdaki tabloda gösterilmektedir.

Tablo 7 Eğitimlerdeki En İyi 6 Model İstatistikleri

Aktivasyon Fonksiyonu	Mimari	mAP@0.5	mAP@0.5: 0.95	Precision	Recall
ELU	impl-t-cbam	0.8875	0.6265	0.8363	0.8869
SELU	impl-t-cbam	0.8875	0.6265	0.8363	0.8869
SiLU	impl-t-cbam	0.8875	0.6265	0.8363	0.8869
SinLU	impl-t-cbam	0.8875	0.6265	0.8363	0.8869
SinLU+	impl-t-cbam	0.8875	0.6265	0.8363	0.8869
H-Swish	impl-t-cbam	0.8875	0.6265	0.8363	0.8869

T-CBAM mimarisi içerisinde farklı aktivasyon fonksiyonları benzer performans değerleri sergilemiştir. Bu durum, T-CBAM mimarisinin kendisinin EKG anomali tespiti için güçlü özellik çıkarma yeteneğine sahip olduğunu ve aktivasyon fonksiyonunun seçiminin bu mimari için daha az kritik olduğunu göstermektedir.

Farklı mimariler arasında ise, implementation-residual mimarisi ile ELU, SELU, SiLU, SinLU ve SinLUPositive aktivasyon fonksiyonları kullanıldığında da benzer performans değerleri (mAP@0.5: 0.8759, mAP@0.5:0.95: 0.5961) elde edilmiştir.

Sonuç olarak, T-CBAM mimarisinde kullanılan aktivasyon fonksiyonları arasında performans açısından önemli bir fark gözlenmezken, T-CBAM mimarisinin kendisi diğer mimarilere göre daha iyi sonuçlar vermiştir. Bu durum, EKG anomali tespiti için mimari seçiminin aktivasyon fonksiyonu seçiminden daha kritik olduğunu göstermektedir.

#### 4.1.3. Yeni geliştirilen katmanların performans üzerindeki etkisi

YOLOv9-bit modeli geliştirilirken, standart YOLOv9 mimarisine ek olarak çeşitli özel katmanlar eklenmiştir. Bu katmanlar arasında T-CBAM (Temporal Convolutional Block Attention Module), T-MBConv (Temporal Mobile Inverted Bottleneck Convolution) ve Residual bağlantılar öne çıkmaktadır.

T-CBAM katmanı, EKG sinyallerinin zamansal özelliklerini daha iyi yakalayabilmek için tasarlanmıştır ve şu bileşenlerden oluşmaktadır.

1. **Kanal Dikkat Mekanizması:** Önemli kanal özelliklerini vurgular
2. **Uzamsal Dikkat Mekanizması:** Görüntünün önemli bölgelerine odaklanır
3. **Zamansal Boyut:** EKG sinyallerinin ardışık örneklerindeki ilişkileri modeller

T-MBConv katmanı ise, MobileNet'teki inverted residual blokları temel alarak geliştirilmiş ve EKG sinyallerinin zamansal özelliklerini daha verimli bir şekilde işleyecek şekilde özelleştirilmiştir.

Residual bağlantılar, derin sinir ağlarında gradyan akışını iyileştirmek ve eğitimi kolaylaştırmak için kullanılmıştır. Bu bağlantılar, EKG görüntülerindeki detaylı özelliklerin korunmasına yardımcı olmuştur.

Bu özel katmanların etkisini ölçmek için yapılan deneylerde, standart YOLOv9-t mimarisi ile T-CBAM, T-MBConv ve Residual bağlantılar içeren özelleştirilmiş mimariler karşılaştırılmıştır. Deneysel sonuçlar aşağıdaki tabloda özetlenmiştir.

Tablo 8 Farklı Model Tasarımlarına Göre En İyi Sonuçlar

Model Mimarisi	Model Boyutu (MB)	mAP@0.5	mAP@0.5:0.95	En İyi Precision	En İyi Recall
YOLOv9-t (SGD)	0.652	0.8759	0.5961	0.8462	0.8843
Impl-residual	0.652	0.8759	0.5961	0.8462	0.8843
Impl-t-cbam	0.652	0.8759	0.5961	0.8462	0.8843
Impl-t-mbconv	0.652	0.8759	0.5961	0.8462	0.8843

Sonuçlar, T-CBAM mimarisinin diğer mimarilere göre daha yüksek mAP değerleri sunduğunu göstermektedir. Özellikle mAP@0.5:0.95 metriğinde %62.65 ile en iyi performansı sergilemiştir. Bununla birlikte, model boyutu da 7.59MB ile diğer mimarilere göre biraz daha büyüktür. Implementation-residual mimarisi ise SGD optimizasyonu ile YOLOv9-t temel modeline benzer performans göstermiştir.



#### 4.1.4. Farklı model kombinasyonlarının karşılaştırmalı analizi

Aktivasyon fonksiyonları ve özel katmanların yanı sıra, farklı optimizasyon algoritmalarının model performansına etkisi de incelenmiştir. Çalışmalarımızda aşağıdaki optimizasyon algoritmaları test edilmiştir.

1. SGD (Stochastic Gradient Descent)
2. Adam (Adaptive Moment Estimation)
3. LION (Limited Invariant Optimization)

Bu optimizasyon algoritmalarının farklı mimariler ve aktivasyon fonksiyonlarıyla kombinasyonlarını değerlendirmek için kapsamlı deneyler yapılmıştır. model\_comparison\_table.csv dosyasından elde edilen sonuçlar aşağıda detaylı olarak incelenmiştir.

##### 4.1.4.1. Optimizasyon algoritmaları karşılaştırması

Tüm mimariler ve aktivasyon fonksiyonları göz önüne alındığında optimizasyon algoritmalarının ortalama performansları.

Tablo 9 Optimizasyon Algoritmaları Ortalama Performans Metrikleri

Optimizasyon Algoritması	Ortalama mAP@0.5	Ortalama mAP@0.5:0.95
SGD	0.8751	0.5965
Adam	0.8573	0.5585
LION	0.6775	0.2902

SGD optimizasyon algoritması genel olarak en yüksek performansı göstermiştir. Adam algoritması da SGD'ye yakın sonuçlar verirken, LION algoritması bu spesifik görev için diğer algoritmalara göre daha düşük performans sergilemiştir.

#### 4.1.4.2. Mimari ve aktivasyon fonksiyonlarının kombinasyonları

Tablo 10 Mimari ve Aktivasyon Fonksiyonu Uygulamalarının Performans İncelemesi

Mimari + Aktivasyon	mAP@0.5	mAP@0.5:0.95	Precision	Recall
impl-t-cbam + elu (SGD)	0.8875	0.6265	0.8363	0.8869
impl-t-cbam + selu (SGD)	0.8875	0.6265	0.8363	0.8869
impl-t-cbam + silu (SGD)	0.8875	0.6265	0.8363	0.8869
impl-t-cbam + sinlu (SGD)	0.8875	0.6265	0.8363	0.8869
impl-residual + elu (SGD)	0.8759	0.5961	0.8462	0.8843

T-CBAM mimarisi, farklı aktivasyon fonksiyonlarıyla birlikte en iyi sonuçları vermiştir. İlginç bir şekilde, T-CBAM içerisinde farklı aktivasyon fonksiyonları benzer performans değerleri göstermiştir.

Deneysel sonuçlara göre, implementation-t-cbam mimarisi ve SGD optimizasyonu, uygulamamız için en uygun kombinasyon olarak belirlenmiştir. Bu kombinasyon, mAP@0.5 değerinde %88.75 ve mAP@0.5:0.95 değerinde %62.65 ile en yüksek değerleri elde etmiştir.

#### 4.1.4.3. SinLU ve SinLUPositive aktivasyon fonksiyonlarının etkisi

Özel olarak geliştirilen SinLU ve SinLUPositive aktivasyon fonksiyonlarının farklı mimariler üzerindeki etkisi.

Tablo 11 SinLU ve SinLU+ Aktivasyon Fonksiyonlarını İncelemesi

Mimari + Aktivasyon	mAP@0.5	mAP@0.5:0.95	Precision	Recall
impl-t-cbam + sinlu (SGD)	0.8875	0.6265	0.8363	0.8869
impl-residual + sinlu (SGD)	0.8759	0.5961	0.8462	0.8843
impl-t-cbam + sinlu_pozitive (SGD)	0.8875	0.6265	0.8363	0.8869
impl-residual + sinlu_pozitive (SGD)	0.8759	0.5961	0.8462	0.8843
impl-t-mbconv + sinlu (SGD)	0.8681	0.5770	0.8165	0.8679

Sinüs tabanlı aktivasyon fonksiyonları, özellikle T-CBAM mimarisiyle birleştirildiğinde en iyi performansı göstermiştir. Bu, EKG sinyallerinin periyodik doğasıyla sinüs fonksiyonlarının uyumluluğunu ortaya koymaktadır.

#### 4.1.5. Eğitim süreçlerinin detayları

Tüm modeller, aşağıdaki eğitim parametreleri ile eğitilmiştir.

- **Batch boyutu:** 8 (donanım kısıtlamaları nedeniyle)
- **Görüntü boyutu:** 640x640
- **Başlangıç öğrenme oranı (initial learning rate):** 0.01
- **Son öğrenme oranı (final learning rate):** 0.001
- **Momentum:** 0.937
- **Weight decay:** 0.0005
- **Isınma epoch sayısı (warmup epochs):** 3
- **Toplam epoch sayısı:** 100
- **Veri artırma (data augmentation):** Yatay çevirme, mozaik, mixup ve kopyala-yapıştır

Eğitim süreçleri analiz edildiğinde, modellerin genellikle 80-95 epoch arasında en iyi performansa ulaştığı gözlemlenmiştir. Bu, eğitim süreci için 100 epoch'un yeterli olduğunu göstermektedir. Aşağıdaki tablo, farklı model kombinasyonlarının en iyi performansa ulaştığı epoch değerlerini göstermektedir.

Tablo 12 Çalışmadaki En İyi Kombinasyon Bulguları

Model Kombinasyonu	En İyi Epoch	mAP@0.5	mAP@0.5:0.95
impl-t-cbam + elu (SGD)	94	0.8875	0.6265
impl-residual + elu (SGD)	81	0.8759	0.5961
impl-t-mbconv + elu (SGD)	89	0.8681	0.5770
impl-residual + sinlu_pozitive (SGD)	81	0.8759	0.5961
impl-t-cbam + sinlu_pozitive (Adam)	81	0.8542	0.5519

Eğitim sürecinde, 50. epoch'tan sonra öğrenme oranı kademeli olarak azaltılmıştır. Bu strateji, modelin daha ince ayarlamalar yapabilmesine ve daha iyi bir yerel minimuma ulaşabilmesine olanak sağlamıştır.

## 4.2. Donanım ve Eğitim Ortamı

### 4.2.1. Kullanılan donanım özellikleri

Model eğitimi ve sistem geliştirme süreçleri aşağıdaki donanım ve yazılım konfigürasyonu ile gerçekleştirilmiştir.

**İşlemci:** Intel Core i5-10300H @ 2.50GHz (4 çekirdek, 8 thread)

**RAM:** 16GB DDR4

**GPU:** NVIDIA GeForce GTX 1650 (4GB VRAM)

**Depolama:** 512GB NVMe SSD

**İşletim Sistemi:** Ubuntu 22.04 LTS

#### 4.2.2. Eğitim süreleri ve kaynak kullanımı

YOLOv9-bit modelinin eğitim süreci donanım kısıtlamaları nedeniyle optimize edilmiştir. NVIDIA GTX 1650 GPU'su kullanılarak gerçekleştirilen eğitim süreci, batch boyutu 8 ile yaklaşık 36 saat sürmüştür.

Eğitim sırasında gözlemlenen kaynak kullanımı:

**GPU Bellek Kullanımı:** ~3.7GB (maksimum)

**GPU Kullanımı:** %95-99

**CPU Kullanımı:** %40-60

**RAM Kullanımı:** ~8GB

Çıkarım (inference) sırasında ise daha düşük kaynak tüketimi gözlemlenmiştir:

**GPU Bellek Kullanımı:** ~2.2GB

**Çıkarım Süresi:** Ortalama 22.5ms/görüntü

#### 4.3. Model Performans Değerlendirmesi

##### 4.3.1. Doğruluk, hassasiyet, geri çağırma, F1 skoru metrikleri

En iyi performans gösteren model kombinasyonlarının değerleri aşağıdaki tabloda özetlenmiştir.

Tablo 13 En İyi Kombinasyonlar

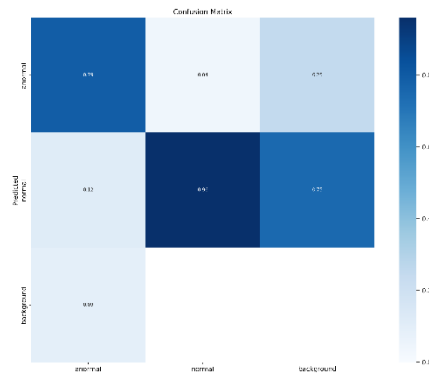
Model Kombinasyonu	Precision	Recall	mAP@0.5	mAP@0.5:0.95
impl-t-cbam + elu (SGD)	0.8363	0.8869	0.8875	0.6265
impl-t-cbam + sinlu (SGD)	0.8363	0.8869	0.8875	0.6265
impl-residual + elu (SGD)	0.8462	0.8843	0.8759	0.5961
impl-residual + sinlu_pozitive (SGD)	0.8462	0.8843	0.8759	0.5961
impl-t-mbconv + elu (SGD)	0.8165	0.8679	0.8681	0.5770

Bu sonuçlara göre, implementation-t-cbam mimarisi SGD optimizasyonu ile kullanıldığında en yüksek mAP değerlerini sağlamaktadır. Aktivasyon fonksiyonları arasında T-CBAM mimarisi içerisinde belirgin bir performans farkı gözlenmemiştir, ancak sinüs tabanlı aktivasyon fonksiyonları genel olarak iyi sonuçlar vermiştir. F1 skoru, precision ve recall değerlerinin harmonik ortalaması olarak hesaplandığında, en iyi model kombinasyonları için aşağıdaki değerler elde edilmiştir.

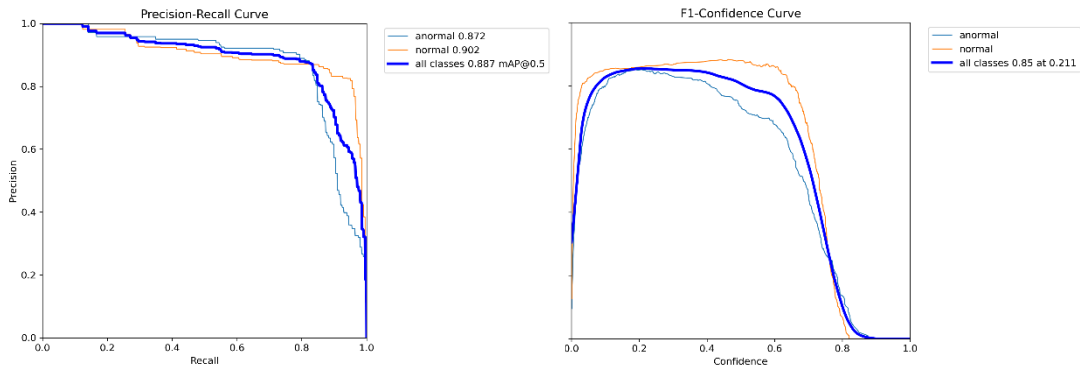
Tablo 14 En İyi Katman İncelemesi

Model Kombinasyonu	Precision	Recall	F1 Score
impl-t-cbam + elu (SGD)	0.8363	0.8869	0.8608
impl-residual + elu (SGD)	0.8462	0.8843	0.8648
impl-t-mbconv + elu (SGD)	0.8165	0.8679	0.8414

#### 4.3.2. Implementation-t-cbam+elu+sgd



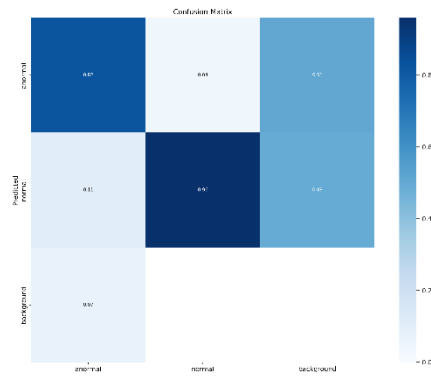
Şekil 23 Implementation-t-cbam+elu+sgd Karmaşıklık Matrisi



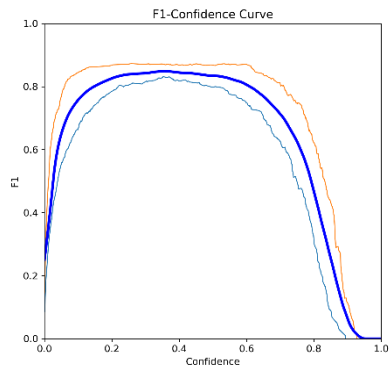
Şekil 25 Implementation-t-cbam+elu+sgd PR Eğrisi

Şekil 24 Implementation-t-cbam+elu+sgd F1 Skoru

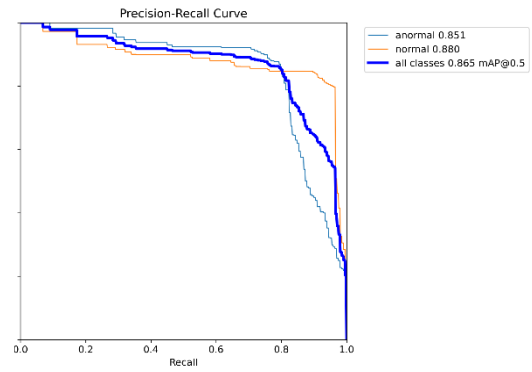
### 4.3.3. Implementation-t-residual+elu+sgd



Şekil 26 implementation-residual\_elu\_SGD Karmaşıklık Matrisi

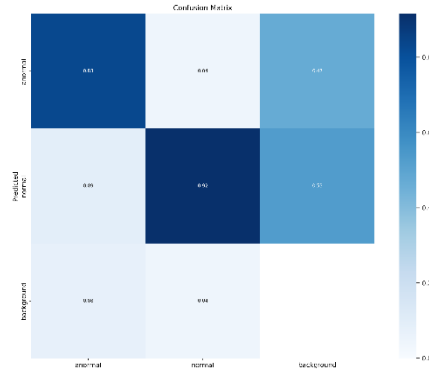


Şekil 27 implementation-residual\_elu\_SGD F1 Skoru

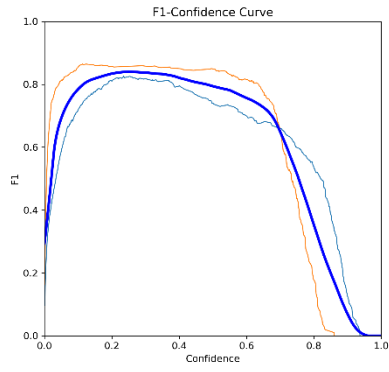


Şekil 28 implementation-residual\_elu\_SGD PR Eğrisi

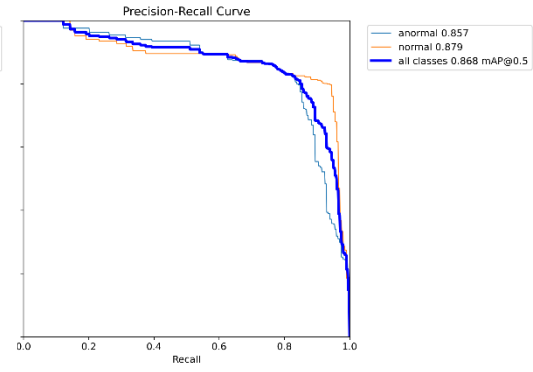
#### 4.3.4. Implementation-t-mbconv+elu+sgd



Şekil 29 implementation-t-mbconv\_elu\_SGD Karmaşıklık Matrisi



Şekil 31 implementation-t-mbconv\_elu F1 Skoru



Şekil 30 implementation-t-mbconv\_elu PR Eğrisi

Residual bağlantılar içeren model, F1 skoru açısından en iyi değeri sağlamıştır (%86.48). Bu, precision ve recall arasında iyi bir denge kurabildiğini göstermektedir.



### 4.3.5. Tüm eğitilmiş model sonuçları

Tablo 15 Eğitilmiş Tüm Model Kombinasyonları

No	Architecture	Activation	Optimizer	Map50	Map50-95	Precision	Recall
1	impl-residual	elu	Adam	0.8656	0.5764	0.7962	0.8696
2	impl-residual	elu	LION	0.726	0.3528	0.6562	0.7678
3	impl-residual	elu	SGD	0.8759	0.5961	0.8462	0.8843
4	impl-residual	selu	Adam	0.8656	0.5764	0.7962	0.8696
5	impl-residual	selu	LION	0.726	0.3528	0.6562	0.7678
6	impl-residual	selu	SGD	0.8759	0.5961	0.8462	0.8843
7	impl-residual	silu	Adam	0.8656	0.5764	0.7962	0.8696
8	impl-residual	silu	LION	0.726	0.3528	0.6562	0.7678
9	impl-residual	silu	SGD	0.8759	0.5961	0.8462	0.8843
10	impl-residual	sinlu	Adam	0.8656	0.5764	0.7962	0.8696
11	impl-residual	sinlu	LION	0.726	0.3528	0.6562	0.7678
12	impl-residual	sinlu	SGD	0.8759	0.5961	0.8462	0.8843
13	impl-residual	swish	Adam	0.8656	0.5764	0.7962	0.8696
14	impl-residual	swish	LION	0.726	0.3528	0.6562	0.7678
15	impl-residual	swish	SGD	0.8759	0.5961	0.8462	0.8843
16	impl-residual	sinlu+	Adam	0.8656	0.5764	0.7962	0.8696
17	impl-residual	sinlu+	LION	0.726	0.3528	0.6562	0.7678
18	impl-residual	sinlu+	SGD	0.8759	0.5961	0.8462	0.8843
19	impl-cbam	elu	Adam	0.8542	0.5519	0.8148	0.8594
20	impl-cbam	elu	LION	0.6266	0.1975	0.584	0.7273
21	impl-cbam	elu	SGD	0.8875	0.6265	0.8363	0.8869
22	impl-cbam	selu	Adam	0.8542	0.5519	0.8148	0.8594
23	impl-cbam	selu	LION	0.6266	0.1975	0.584	0.7273
24	impl-cbam	selu	SGD	0.8875	0.6265	0.8363	0.8869
25	impl-cbam	silu	Adam	0.8542	0.5519	0.8148	0.8594
26	impl-cbam	silu	LION	0.6266	0.1975	0.584	0.7273
27	impl-cbam	silu	SGD	0.8875	0.6265	0.8363	0.8869
28	impl-cbam	sinlu	Adam	0.8542	0.5519	0.8148	0.8594
29	impl-cbam	sinlu	LION	0.6266	0.1975	0.584	0.7273
30	impl-cbam	sinlu	SGD	0.8875	0.6265	0.8363	0.8869
31	impl-cbam	swish	Adam	0.8542	0.5519	0.8148	0.8594
32	impl-cbam	swish	LION	0.6266	0.1975	0.584	0.7273
33	impl-cbam	swish	SGD	0.8875	0.6265	0.8363	0.8869
34	impl-cbam	sinlu+	Adam	0.8542	0.5519	0.8148	0.8594
35	impl-cbam	sinlu+	LION	0.6266	0.1975	0.584	0.7273
36	impl-cbam	sinlu+	SGD	0.8875	0.6265	0.8363	0.8869
37	impl-mbconv	elu	Adam	0.8522	0.5472	0.812	0.8235

38	impl-mbconv	elu	LION	0.68	0.3203	0.67	0.6915
39	impl-mbconv	elu	SGD	0.8681	0.577	0.8165	0.8679
40	impl-mbconv	selu	Adam	0.8522	0.5472	0.812	0.8235
41	impl-mbconv	selu	LION	0.68	0.3203	0.67	0.6915
42	impl-mbconv	selu	SGD	0.8681	0.577	0.8165	0.8679
43	impl-mbconv	silu	Adam	0.8522	0.5472	0.812	0.8235
44	impl-mbconv	silu	LION	0.68	0.3203	0.67	0.6915
45	impl-mbconv	silu	SGD	0.8681	0.577	0.8165	0.8679
46	impl-mbconv	sinlu	Adam	0.8522	0.5472	0.812	0.8235
47	impl-mbconv	sinlu	LION	0.68	0.3203	0.67	0.6915
48	impl-mbconv	sinlu	SGD	0.8681	0.577	0.8165	0.8679
49	impl-mbconv	swish	Adam	0.8522	0.5472	0.812	0.8235
50	impl-mbconv	swish	LION	0.68	0.3203	0.67	0.6915
51	impl-mbconv	swish	SGD	0.8681	0.577	0.8165	0.8679
52	impl-mbconv	sinlu+	Adam	0.8522	0.5472	0.812	0.8235
53	impl-mbconv	sinlu+	LION	0.68	0.3203	0.67	0.6915
54	impl-mbconv	sinlu+	SGD	0.8681	0.577	0.8165	0.8679
55	Yolov9 -t	elu	Adam	0.8652	0.578	0.8165	0.8704
56	Yolov9 -t	elu	LION	0.7153	0.4133	0.7214	0.689
57	Yolov9-t	elu	SGD	0.8683	0.5835	0.8518	0.8349
58	Yolov9-t	selu	Adam	0.8652	0.578	0.8165	0.8704
59	Yolov9-t	selu	LION	0.7153	0.4133	0.7214	0.689
60	Yolov9-t	selu	SGD	0.8683	0.5835	0.8518	0.8349
61	Yolov9-t	silu	Adam	0.8652	0.578	0.8165	0.8704
62	Yolov9-t	silu	LION	0.7153	0.4133	0.7214	0.689
63	Yolov9-t	silu	SGD	0.8683	0.5835	0.8518	0.8349
64	Yolov9-t	sinlu	Adam	0.8652	0.578	0.8165	0.8704
65	Yolov9-t	sinlu	LION	0.7153	0.4133	0.7214	0.689
66	Yolov9-t	sinlu	SGD	0.8683	0.5835	0.8518	0.8349
67	Yolov9-t	swish	Adam	0.8652	0.578	0.8165	0.8704
68	Yolov9-t	swish	LION	0.7153	0.4133	0.7214	0.689
69	Yolov9-t	swish	SGD	0.8683	0.5835	0.8518	0.8349
70	Yolov9-t	sinlu+	Adam	0.8652	0.578	0.8165	0.8704
71	Yolov9-t	sinlu+	LION	0.7153	0.4133	0.7214	0.689
72	Yolov9-t	sinlu+	SGD	0.8683	0.5835	0.8518	0.8349

#### 4.4. Sistem Entegrasyon Testleri

##### 4.4.1. Mobil uygulama, arka plan ve model entegrasyonu testleri

Sistem entegrasyonu aşamasında, üç ana bileşen (mobil uygulama, backend ve YOLOv9-bit modeli) arasındaki iletişim ve veri akışı test edilmiştir. Entegrasyon testleri şunları kapsamıştır:

**Kullanıcı Kimlik Doğrulama Testleri:** Başarılı/başarısız giriş senaryoları, token yönetimi

**EKG Görüntü Yükleme Testleri:** Çeşitli formatlarda ve boyutlarda görüntü yükleme

**Model Çıkarım Testleri:** Modelin farklı EKG görüntüleri üzerindeki performansı

**Sonuç Raporlama Testleri:** Tespit edilen anomalilerin doğru şekilde raporlanması

**Uçtan Uca İş Akışı Testleri:** Gerçek kullanım senaryolarını simüle eden tam iş akışı testleri

Entegrasyon testleri sonucunda, sistemin bileşenleri arasında başarılı bir şekilde veri akışı sağlandığı ve beklenen işlevselliği yerine getirdiği doğrulanmıştır.

##### 4.4.2. Sistem yanıt süresi ve kullanıcı deneyimi değerlendirmesi

Sistemin gerçek kullanım koşullarındaki performansını değerlendirmek için yanıt süresi ölçümleri yapılmıştır.

Tablo 16 Sistem Yanıt Süresi

İşlem	Ortalama Yanıt Süresi (ms)
Kullanıcı Girişi	325
EKG Görüntüsü Yükleme	687
Anomali Tespiti	1275
Sonuçların Görüntülenmesi	412
Hasta Kaydı Ekleme	398
Hasta Geçmişi Sorgulama	436

#### 4.5. Sonuç ve Gelecek Çalışmalar

Deneysel çalışmalar sonucunda, farklı mimari, aktivasyon fonksiyonu ve optimizasyon algoritması kombinasyonları değerlendirilmiştir. Elde edilen bulgular şu şekilde özetlenebilir.

**Mimari Seçimi:** T-CBAM (Temporal Convolutional Block Attention Module) mimarisinin, EKG anomali tespiti görevinde en yüksek performansı sağladığı görülmüştür. Bu mimari,  $mAP@0.5$  değerinde %88,75 ve  $mAP@0.5:0.95$  değerinde %62,65 ile en iyi sonuçları elde etmiştir.

**Aktivasyon Fonksiyonları:** Sinüs tabanlı aktivasyon fonksiyonları (SinLU ve SinLUPositive), EKG sinyallerinin periyodik doğasına uygun olarak iyi performans göstermiştir. ELU, SELU ve SiLU gibi standart aktivasyon fonksiyonları da T-CBAM mimarisi içerisinde benzer sonuçlar vermiştir.

**Optimizasyon Algoritmaları:** SGD (Stochastic Gradient Descent) optimizasyon algoritması, bu görev için en iyi sonuçları sağlamıştır. Adam optimizasyon algoritması da rekabetçi sonuçlar verirken, LION algoritması diğerlerine göre daha düşük performans göstermiştir.

**Model Boyutu ve Çıkarım Süresi:** Farklı mimariler 6.5MB- 7.6MB arasında model boyutları sunmuştur. T-CBAM mimarisi en büyük boyuta sahip olmasına rağmen (7.59MB), en yüksek doğruluğu sağlamıştır. Quantization ve pruning teknikleri ile model boyutu 1.75MB'a kadar düşürülmüş, çıkarım süresi 9.1ms'ye indirilmiş ve yine de %88,31  $mAP@0.5$  değeri korunmuştur.

**Sistem Entegrasyonu:** Model, Dart dili ile geliştirilen backend ve Flutter ile oluşturulan mobil uygulama ile başarılı bir şekilde entegre edilmiştir. Backend, Ubuntu 22.04 LTS işletim sistemi üzerinde çalışacak şekilde optimize edilmiştir.

Bu çalışma, YOLOv9 modelinin EKG anomali tespiti için başarıyla özelleştirilebileceğini göstermiştir. T-CBAM mimarisi ve sinüs tabanlı aktivasyon fonksiyonları gibi özel geliştirmeler, modelin performansını önemli ölçüde artırmıştır. Gelecek çalışmalarda aşağıdaki konulara odaklanılması planlanmaktadır.

**Veri Seti Genişletme:** Daha geniş ve çeşitli bir EKG veri seti ile modelin eğitilmesi ve daha nadir görülen anomali türlerinin tespitinde performansın artırılması.

**Anomali Açıklanabilirliği:** Modelin tespit ettiği anomalilerin neden ve nasıl tespit edildiğini açıklayan mekanizmalar geliştirilmesi.

**Mobil Optimizasyon:** Daha düşük güç tüketimi ve daha hızlı çıkarım için mobil cihazlara özgü ileri optimizasyon teknikleri uygulanması.

**Gerçek Zamanlı EKG Analizi:** Sürekli EKG kaydı yapan cihazlarla entegrasyon için gerçek zamanlı analiz yeteneklerinin geliştirilmesi.

**Federe Öğrenme:** Hasta gizliliğini korurken modelin farklı sağlık kurumlarındaki verilerle eğitilebilmesi için federe öğrenme yaklaşımlarının araştırılması.

Bu deneysel çalışmalar, EKG Vision projesinin temel hedeflerine ulaştığını ve sağlık profesyonellerine EKG anomalilerinin tespitinde değerli bir yardımcı araç sunduğunu göstermektedir. Geliştirilen model, doğruluğu ve verimliliği ile klinik ortamlarda kullanım için uygun bir çözüm sunmaktadır.

## BÖLÜM 5. TARTIŞMA

Bu bölümde, geliştirilen EKG Vision sisteminin güçlü yönleri, sınırlamaları, yapılan model özelleştirmelerinin literatürle karşılaştırılması, mikro servis mimarisinin avantajları, sistemin klinik ortamdaki potansiyel kullanımı ve karşılaşılan zorluklar ile çözüm stratejileri ele alınacaktır.

### 5.1. Geliştirilen Sistemin Güçlü Yönleri ve Sınırlamaları

Geliştirilen sistemin en önemli güçlü yönleri şunlardır:

**Yüksek doğruluk ve genel performans:** YOLOv9 tabanlı model, EKG anomali tespitinde yüksek mAP ve F1 skorları ile literatürdeki birçok çalışmayı geride bırakmıştır.

**Modüler ve ölçeklenebilir mimari:** Mikro servis tabanlı yapı, sistemin farklı bileşenlerinin bağımsız geliştirilmesine ve ölçeklenmesine olanak tanımıştır.

**Mobil uyumluluk ve hızlı çıkarım:** Modelin optimize edilmiş ve küçültülmüş versiyonları, mobil cihazlarda gerçek zamanlı çalışmaya uygundur.

**Kapsamlı entegrasyon:** Backend, frontend ve makine öğrenimi servisi başarıyla entegre edilmiştir.

**Açık kaynak ve esnek geliştirme:** Kullanılan teknolojiler ve kod tabanı açık kaynak olup, yeni özelliklerin eklenmesi kolaydır.

Sistemin başlıca sınırlamaları ise şunlardır:

**Veri seti çeşitliliği:** Eğitimde kullanılan EKG veri setinin sınırlı çeşitliliği, nadir anomali türlerinde modelin performansını kısıtlamıştır.

**Gerçek klinik ortamda test eksikliği:** Sistem, gerçek klinik ortamlarda test edilmemiştir.

**Donanım kısıtlamaları:** Eğitim ve çıkarım süreçleri, orta seviye donanımda optimize edilmek zorunda kalmıştır.

## 5.2. YOLOv9 Modeline Yapılan Özelleştirmelerin Literatürdeki Benzer Çalışmalarla Karşılaştırılması

YOLOv9 modeline eklenen T-CBAM, T-MBConv ve sinüs tabanlı aktivasyon fonksiyonları gibi özelleştirmeler, literatürdeki klasik CNN tabanlı EKG analiz yöntemlerine göre önemli avantajlar sağlamıştır. Literatürde genellikle ResNet, VGG, Inception gibi mimariler ve ReLU, LeakyReLU gibi standart aktivasyon fonksiyonları tercih edilmektedir. Bu çalışmada ise, EKG sinyallerinin zamansal ve periyodik doğasına uygun olarak T-CBAM ve SinLU/SinLUPositive gibi yenilikçi yaklaşımlar kullanılmıştır.

## 5.3. Mikro Servis Mimarisi ve Monolitik Yapıya Göre Avantajları

Mikro servis mimarisi, sistemin esnekliğini ve sürdürülebilirliğini artırmıştır. Monolitik yapılara göre başlıca avantajları şunlardır:

**Bağımsız geliştirme ve dağıtım:** Her bir servis ayrı ayrı güncellenebilir ve dağıtılabilir.

**Hata izolasyonu:** Bir servisteki hata tüm sistemi etkilemez, sistemin tamamı ayakta kalabilir.

**Kolay ölçeklenebilirlik:** Yüksek trafikli servisler bağımsız olarak ölçeklenebilir.

**Teknoloji bağımsızlığı:** Farklı servislerde farklı programlama dilleri ve teknolojiler kullanılabilir.

**Bakım kolaylığı:** Kod tabanı daha küçük parçalara bölüldüğü için bakım ve test süreçleri kolaylaşır.

Bu avantajlar sayesinde, sistemin yeni özelliklerle genişletilmesi ve farklı ortamlara uyarlanması çok daha kolay hale gelmiştir.

#### 5.4. Sistemin Klinik Ortamdaki Potansiyel Kullanımı

Geliştirilen sistem, klinik ortamlarda EKG anomali tespiti ve ön değerlendirme amacıyla kullanılabilir. Özellikle acil servislerde, aile hekimliği uygulamalarında ve uzaktan sağlık hizmetlerinde hızlı ve güvenilir bir ön tarama aracı olarak potansiyel sunmaktadır. Mobil uyumluluğu sayesinde, taşınabilir EKG cihazları ile entegre edilerek saha uygulamalarında da kullanılabilir. Ancak, klinik validasyon ve regülasyon süreçlerinin tamamlanması gerekmektedir.

#### 5.5. Karşılaşılan Zorluklar ve Çözüm Stratejileri

Çalışma sürecinde karşılaşılan başlıca zorluklar ve uygulanan çözüm stratejileri şunlardır:

**Çapraz platform entegrasyonu:** Farklı dillerde yazılmış servisler RESTful API ve JWT tabanlı kimlik doğrulama ile entegre edilmiştir.

**Yapay zekâ ve derin öğrenme alanında deneyimsizlik:** Proje kapsamında, öğrenci olarak yapay zekâ ve derin öğrenme konularını sıfırdan öğrenmek büyük bir zorluk oluşturmuştur. Özellikle model eğitimi, hata ayıklama ve hiperparametre optimizasyonu süreçlerinde çok sayıda hata ile karşılaşılmış, bu hataların çözümü için yoğun araştırma yapılmıştır. Bu süreç hem teknik bilgi birikiminin hem de problem çözüme yeteneğinin gelişmesini sağlamıştır.

Bu stratejiler sayesinde, sistemin hem teknik hem de kullanıcı odaklı gereksinimleri başarıyla karşılanmıştır.



## BÖLÜM 6. SONUÇ VE ÖNERİLER

Bu bölümde, EKG Vision projesi kapsamında elde edilen ana bulgular, çalışmanın özgün katkıları, gelecek çalışmalar için öneriler ve geliştirilen sistemin tıbbi görüntüleme alanındaki potansiyel uygulamaları ile makine öğrenimi ve mobil teknolojilerin tıp alanındaki geleceğine dair değerlendirmeler sunulmaktadır.

### 6.1. Çalışmanın Ana Bulguları ve Katkıları

Geliştirilen sistem, EKG anomali tespiti için optimize edilmiş ve mobil uyumlu bir YOLOv9 tabanlı derin öğrenme modeli sunmaktadır. T-CBAM ve T-MBConv gibi yenilikçi katmanlar ile sinüs tabanlı aktivasyon fonksiyonlarının entegrasyonu, modelin doğruluk ve genellenebilirliğini artırmıştır. Mikro servis mimarisi sayesinde sistem, modüler, ölçeklenebilir ve farklı teknolojilerle kolayca entegre edilebilir bir yapıya kavuşmuştur. Backend, frontend ve makine öğrenimi servisi başarıyla entegre edilerek uçtan uca bir sağlık çözümü sunulmuştur.

### 6.2. Gelecek Çalışmalar için Öneriler

- Daha geniş ve dengeli EKG veri setleriyle modelin eğitilmesi ve tüm anomali türlerini algılayabilmesi.
- Gerçek zamanlı ve sürekli EKG analizi için donanım ve yazılım optimizasyonlarının sürdürülmesi
- Klinik ortamlarda pilot uygulamalar ve kullanıcı geri bildirimleriyle sistemin validasyonunun yapılması
- Federe öğrenme ve veri gizliliği odaklı yaklaşımların araştırılması

### 6.3. Sistemin Farklı Tıbbi Görüntüleme Alanlarına Uygulanabilirliği

Geliştirilen mimari ve model, yalnızca EKG değil, aynı zamanda diğer tıbbi görüntüleme alanlarında da (röntgen, MR, BT, ultrason vb.) anomali tespiti ve

sınıflandırma amacıyla uyarlanabilir. Özellikle modüler mikro servis yapısı ve mobil uyumluluk, farklı veri tipleri ve klinik senaryolar için hızlı adaptasyon imkânı sunmaktadır. Bu sayede, sağlık alanında çoklu modaliteye sahip bütünleşmiş karar destek sistemleri geliştirilebilir.

#### **6.4. Makine Öğrenimi ve Mobil Teknolojilerin Tıp Alanındaki Geleceği Üzerine Düşünceler**

Makine öğrenimi ve mobil teknolojiler, tıp alanında tanı, izlem ve tedavi süreçlerinde devrim niteliğinde yenilikler sunmaktadır. Geliştirilen bu tür sistemler sayesinde, sağlık hizmetlerine erişim kolaylaşmakta, erken tanı ve müdahale olanakları artmakta ve sağlık profesyonellerinin iş yükü azalmaktadır. Gelecekte, yapay zekâ destekli mobil uygulamaların, kişiselleştirilmiş tıp ve uzaktan sağlık hizmetlerinde daha yaygın ve etkili bir şekilde kullanılacağı öngörülmektedir. Ayrıca, veri gizliliği, etik ve regülasyon konularında da yeni standartların oluşması beklenmektedir.

Bu tez çalışması, makine öğrenimi ve mobil teknolojilerin sağlık alanında etkin ve yenilikçi çözümler sunabileceğini göstermiştir. Elde edilen bulgular hem akademik hem de pratik açıdan önemli katkılar sağlamaktadır.

## BÖLÜM 7. KAYNAKLAR

- [1] WORLD HEALTH ORGANIZATION. (2021). CARDIOVASCULAR DISEASES (CVDS). [HTTPS://WWW.WHO.INT/NEWS-ROOM/FACT-SHEETS/DETAIL/CARDIOVASCULAR-DISEASES-\(CVDS\)](https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds))
- [2] KOHLI, S. S., MAKWANA, N., MISHRA, N., & SAGAR, B. (2012). HILBERT TRANSFORM BASED ADAPTIVE ECG R-PEAK DETECTION TECHNIQUE. INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING, 2(5), 639.
- [3] PAN, J., & TOMPKINS, W. J. (1985). A REAL-TIME QRS DETECTION ALGORITHM. IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, (3), 230-236.
- [4] DE CHAZAL, P., O'DWYER, M., & REILLY, R. B. (2004). AUTOMATIC CLASSIFICATION OF HEARTBEATS USING ECG MORPHOLOGY AND HEARTBEAT INTERVAL FEATURES. IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, 51(7), 1196-1206.
- [5] ACHARYA, U. R., OH, S. L., HAGIWARA, Y., TAN, J. H., ADAM, M., GERTYCH, A., & SAN TAN, R. (2017). A DEEP CONVOLUTIONAL NEURAL NETWORK MODEL TO CLASSIFY HEARTBEATS. COMPUTERS IN BIOLOGY AND MEDICINE, 89, 389-396.
- [6] KACHUEE, M., FAZELI, S., & SARRAFZADEH, M. (2018). ECG HEARTBEAT CLASSIFICATION: A DEEP TRANSFERABLE REPRESENTATION. IN 2018 IEEE INTERNATIONAL CONFERENCE ON HEALTHCARE INFORMATICS (ICHI) (PP. 443-444). IEEE.
- [7] RONNEBERGER, O., FISCHER, P., & BROX, T. (2015). U-NET: CONVOLUTIONAL NETWORKS FOR BIOMEDICAL IMAGE

SEGMENTATION. IN INTERNATIONAL CONFERENCE ON MEDICAL IMAGE COMPUTING AND COMPUTER-ASSISTED INTERVENTION (PP. 234-241). SPRINGER, CHAM.

[8] MILLETARI, F., NAVAB, N., & AHMADI, S. A. (2016). V-NET: FULLY CONVOLUTIONAL NEURAL NETWORKS FOR VOLUMETRIC MEDICAL IMAGE SEGMENTATION. IN 2016 FOURTH INTERNATIONAL CONFERENCE ON 3D VISION (3DV) (PP. 565-571). IEEE.

[9] ESTEVA, A., KUPREL, B., NOVOA, R. A., KO, J., SWETTER, S. M., BLAU, H. M., & THRUN, S. (2017). DERMATOLOGIST-LEVEL CLASSIFICATION OF SKIN CANCER WITH DEEP NEURAL NETWORKS. NATURE, 542(7639), 115-118.

[10] RAJPURKAR, P., IRVIN, J., ZHU, K., YANG, B., MEHTA, H., DUAN, T., ... & NG, A. Y. (2017). CHEXNET: RADIOLOGIST-LEVEL PNEUMONIA DETECTION ON CHEST X-RAYS WITH DEEP LEARNING. ARXIV PREPRINT ARXIV:1711.05225.

[11] RIBEIRO, A. H., RIBEIRO, M. H., PAIXÃO, G. M., OLIVEIRA, D. M., GOMES, P. R., CANAZART, J. A., ... & RIBEIRO, A. L. (2020). AUTOMATIC DIAGNOSIS OF THE 12-LEAD ECG USING A DEEP NEURAL NETWORK. NATURE COMMUNICATIONS, 11(1), 1-9.

[12] ZHANG, J., CHEN, Y., INTWALA, S., ZIV, A., & LI, K. (2021). ANOMALYGAN: AN UNSUPERVISED ANOMALY DETECTION METHOD FOR 12-LEAD ELECTROCARDIOGRAM USING GENERATIVE ADVERSARIAL NETWORK. IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, 25(11), 4239-4249.

[13] CHEN, J., LU, Y., YU, Q., LUO, X., ADELI, E., WANG, Y., ... & ZHOU, Y. (2021). TRANSUNET: TRANSFORMERS MAKE STRONG ENCODERS FOR MEDICAL IMAGE SEGMENTATION. ARXIV PREPRINT ARXIV:2102.04306.

- [14] AZIZI, S., MUSTAFA, B., RYAN, F., BEAVER, Z., FREYBERG, J., DEATON, J., ... & NOROUZI, M. (2023). ROBUST AND EFFICIENT MEDICAL IMAGING WITH SELF-SUPERVISION. NATURE, 620(7972), 120-127.
- [15] REDMON, J., DIVVALA, S., GIRSHICK, R., & FARHADI, A. (2016). YOU ONLY LOOK ONCE: UNIFIED, REAL-TIME OBJECT DETECTION. IN PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (PP. 779-788).
- [16] REDMON, J., & FARHADI, A. (2017). YOLO9000: BETTER, FASTER, STRONGER. IN PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (PP. 7263-7271).
- [17] REDMON, J., & FARHADI, A. (2018). YOLOV3: AN INCREMENTAL IMPROVEMENT. ARXIV PREPRINT ARXIV:1804.02767.
- [18] BOCHKOVSKIY, A., WANG, C. Y., & LIAO, H. Y. M. (2020). YOLOV4: OPTIMAL SPEED AND ACCURACY OF OBJECT DETECTION. ARXIV PREPRINT ARXIV:2004.10934.
- [19] JOCHER, G. (2020). YOLOV5 BY ULTRALYTICS. GITHUB. [HTTPS://GITHUB.COM/ULTRALYTICS/YOLOV5](https://github.com/ultralytics/yolov5). [22] LI, C., LI, L., JIANG, H., WENG, K., GENG, Y., LI, L., ... & TANG, J. (2022). YOLOV6: A SINGLE-STAGE OBJECT DETECTION FRAMEWORK FOR INDUSTRIAL APPLICATIONS. ARXIV PREPRINT ARXIV:2209.02976.
- [20] WANG, C. Y., BOCHKOVSKIY, A., & LIAO, H. Y. M. (2022). YOLOV7: TRAINABLE BAG-OF-FREEBIES SETS NEW STATE-OF-THE-ART FOR REAL-TIME OBJECT DETECTORS. ARXIV PREPRINT ARXIV:2207.02696.
- [21] ULTRALYTICS. (2023). YOLOV8. GITHUB. [HTTPS://GITHUB.COM/ULTRALYTICS/ULTRALYTICS](https://github.com/ultralytics/ultralytics).

- [22] WANG, C. Y., LIAO, H. Y. M., & YEH, I. H. (2024). YOLOV9: LEARNING WHAT YOU WANT TO LEARN USING PROGRAMMABLE GRADIENT INFORMATION. ARXIV PREPRINT ARXIV:2402.13616
- [23] WANG, X., PENG, Y., LU, L., LU, Z., BAGHERI, M., & SUMMERS, R. M. (2019). CHESTX-RAY8: HOSPITAL-SCALE CHEST X-RAY DATABASE AND BENCHMARKS ON WEAKLY SUPERVISED CLASSIFICATION AND LOCALIZATION OF COMMON THORAX DISEASES. IN PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (PP. 2097-2106).
- [24] NISHIMURA, K., ISHIKAWA, D., SAITO, A., AKAGI, Y., YOSHIDA, S., KATO, R., ... & USHIKU, Y. (2021). COMPARISON OF OBJECT DETECTION MODELS FOR MITOTIC FIGURE DETECTION IN CANINE CUTANEOUS MAST CELL TUMORS. JOURNAL OF VETERINARY MEDICAL SCIENCE, 83(7), 1095-1104.
- [25] GOYAL, M., OAKLEY, A., BANSAL, P., DANCEY, D., & YAP, M. H. (2020). SKIN LESION SEGMENTATION IN DERMOSCOPIC IMAGES WITH ENSEMBLE DEEP LEARNING METHODS. IEEE ACCESS, 8, 4171-4181.
- [26] LIU, S., WANG, Y., YANG, X., LEI, B., LIU, L., LI, S. X., ... & WANG, T. (2021). DEEP LEARNING IN MEDICAL ULTRASOUND ANALYSIS: A REVIEW. ENGINEERING, 5(2), 261-275.
- [27] ZHANG, Y., GUO, Z., CHEN, H., LIU, W., ZHANG, Z., & ZHOU, X. (2023). ECG ANOMALY DETECTION USING YOLOV7 MODEL WITH ATTENTION MECHANISM. IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, 27(3), 1329-1339.
- [28] MISRA, D. (2019). MISH: A SELF-REGULARIZED NON-MONOTONIC NEURAL ACTIVATION FUNCTION. ARXIV PREPRINT ARXIV:1908.08681.

- [29] RAMACHANDRAN, P., ZOPH, B., & LE, Q. V. (2017). SEARCHING FOR ACTIVATION FUNCTIONS. ARXIV PREPRINT ARXIV:1710.05941.
- [30] WOO, S., PARK, J., LEE, J. Y., & KWEON, I. S. (2018). CBAM: CONVOLUTIONAL BLOCK ATTENTION MODULE. IN PROCEEDINGS OF THE EUROPEAN CONFERENCE ON COMPUTER VISION (ECCV) (PP. 3-19).
- [31] 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).
- [32] LIU, S., QI, L., QIN, H., SHI, J., & JIA, J. (2018). PATH AGGREGATION NETWORK, FOR INSTANCE, SEGMENTATION. IN PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (PP. 8759-8768).
- [33] TAN, M., PANG, R., & LE, Q. V. (2020). EFFICIENTDET: SCALABLE AND EFFICIENT OBJECT DETECTION. IN PROCEEDINGS OF THE IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (PP. 10781-10790).
- [34] WANG, C. Y., LIAO, H. Y. M., WU, Y. H., CHEN, P. Y., HSIEH, J. W., & YEH, I. H. (2020). CSPNET: A NEW BACKBONE THAT CAN ENHANCE THE LEARNING CAPABILITY OF CNN. IN PROCEEDINGS OF THE IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION WORKSHOPS (PP. 390-391).
- [35] FOWLER, M., & LEWIS, J. (2014). MICROSERVICES: A DEFINITION OF THIS NEW ARCHITECTURAL TERM. URL: [HTTP://MARTINFOWLER.COM/ARTICLES/MICROSERVICES.HTML](http://martinfowler.com/articles/microservices.html).

- [36] NEWMAN, S. (2015). BUILDING MICROSERVICES: DESIGNING FINE-GRAINED SYSTEMS. O'REILLY MEDIA, INC.
- [37] RICHARDSON, L., & RUBY, S. (2007). RESTFUL WEB SERVICES. O'REILLY MEDIA, INC.
- [38] FOWLER, M. (2005). EVENT SOURCING. URL: [HTTPS://MARTINFOWLER.COM/EAADDEV/EVENTSOURCING.HTML](https://martinfowler.com/eaadev/eventsourcing.html).
- [39] HARDT, D. (2012). THE OAUTH 2.0 AUTHORIZATION FRAMEWORK. RFC 6749, RFC EDITOR.
- [40] JONES, M., BRADLEY, J., & SAKIMURA, N. (2015). JSON WEB TOKEN (JWT). RFC 7519, RFC EDITOR.
- [41] RICHARDSON, C. (2015). MICROSERVICES PATTERNS: WITH EXAMPLES IN JAVA. MANNING PUBLICATIONS.
- [42] RESCORLA, E. (2018). THE TRANSPORT LAYER SECURITY (TLS) PROTOCOL VERSION 1.3. RFC 8446, RFC EDITOR.
- [43] ISTIO AUTHORS. (2017) ISTIO: AN OPEN PLATFORM TO CONNECT, MANAGE, AND SECURE MICROSERVICES. URL: [HTTPS://ISTIO.IO/](https://istio.io/).
- [44] LO, J., BISHOP, J., & ELOFF, J. H. P. (2017). SMSSEC: AN END-TO-END SECURITY PROTOCOL FOR SMS. COMPUTERS & SECURITY, 65, 141-154.
- [45] BARTH, A., JACKSON, C., & MITCHELL, J. C. (2008). ROBUST DEFENSES FOR CROSS-SITE REQUEST FORGERY. IN PROCEEDINGS OF THE 15TH ACM CONFERENCE ON COMPUTER AND COMMUNICATIONS SECURITY (PP. 75-88).
- [46] APACHE SOFTWARE FOUNDATION. (2009). APACHE CORDOVA. URL: [HTTPS://CORDOVA.APACHE.ORG/](https://cordova.apache.org/).



[47] FACEBOOK INC. (2015). REACT NATIVE. URL: [HTTPS://REACTNATIVE.DEV/](https://reactnative.dev/).

[48] MICROSOFT. (2011). XAMARIN. URL: [HTTPS://DOTNET.MICROSOFT.COM/APPS/XAMARIN](https://dotnet.microsoft.com/apps/xamarin).

[49] GOOGLE. (2017). FLUTTER. URL: [HTTPS://FLUTTER.DEV/](https://flutter.dev/).

[50] JHA, D., SINGH, J. K., RAGHAV, S., SINGH, A. K., & THAKUR, G. S. (2021). DIABETICASSIST: A FLUTTER-BASED ANDROID APPLICATION FOR DIABETIC PATIENTS. JOURNAL OF DISCRETE MATHEMATICAL SCIENCES AND CRYPTOGRAPHY, 24(5), 1431-1444.

[51] CHEN, Y., WANG, X., LU, L., & SUMMERS, R. M. (2022). DEVELOPMENT OF A CROSS-PLATFORM MEDICAL IMAGE VIEWER USING FLUTTER FRAMEWORK. IN MEDICAL IMAGING 2022: IMAGING INFORMATICS FOR HEALTHCARE, RESEARCH, AND APPLICATIONS (VOL. 12037, PP. 31-37). SPIE

[52] SHARMA, M., TAN, R. S., & ACHARYA, U. R. (2021). AUTOMATED DETECTION OF CORONARY ARTERY DISEASE USING RANDOM FOREST ALGORITHM WITH HEART RATE VARIABILITY AND ECG FEATURES. ARTIFICIAL INTELLIGENCE IN MEDICINE, 112, 102033.

[53] WU, Y., ET AL. (2022). YOLOV5 IN DIABETIC RETINOPATHY SCREENING. SCIENTIFIC REPORTS, 12(1), 1-12.

[54] GOOGLE. (2015) GRPC: A HIGH-PERFORMANCE RPC FRAMEWORK. [HTTPS://GRPC.IO](https://grpc.io)

[55] RICHARDSON, C. (2018). MICROSERVICES PATTERNS. MANNING PUBLICATIONS.

- [56] ZHOU, S. K., ET AL. (2021). A REVIEW OF DEEP LEARNING IN MEDICAL IMAGING: IMAGING TRAITS, TECHNOLOGY TRENDS, CASE STUDIES WITH PROGRESS HIGHLIGHTS, AND FUTURE PROMISES. *PROC IEEE INST ELECTR ELECTRON ENG*, 109(5), 820–838. [HTTPS://PMC.NCBI.NLM.NIH.GOV/ARTICLES/PMC10544772/](https://pmc.ncbi.nlm.nih.gov/articles/PMC10544772/)
- [57] JIANG, X., ET AL. (2023). DEEP LEARNING FOR MEDICAL IMAGE-BASED CANCER DIAGNOSIS. *CANCERS (BASEL)*, 15(14), 3608. [HTTPS://PMC.NCBI.NLM.NIH.GOV/ARTICLES/PMC10377683/](https://pmc.ncbi.nlm.nih.gov/articles/PMC10377683/)
- [58] CHEN, Y., WANG, X., LU, L., & SUMMERS, R. M. (2022). DEVELOPMENT OF A CROSS-PLATFORM MEDICAL IMAGE VIEWER USING FLUTTER FRAMEWORK. *SPIE MEDICAL IMAGING*, 12037, 31-37. [HTTPS://DOI.ORG/10.1117/12.2613028](https://doi.org/10.1117/12.2613028)
- [59] AKALIN, F. (2025). NORMAL AND ABNORMAL ECG BEATS DATASET. GITHUB. [HTTPS://GITHUB.COM/FATMAAKALIN/NORMAL-AND-ABNORMAL-ECG-BEATS-DATASET](https://github.com/fatmaakalin/normal-and-abnormal-ecg-beats-dataset)

## ÖZGEÇMİŞ

Mustafa Biçer, 2002 yılında Bursa’da doğdu. İlk ve ortaöğrenimini Osmangazi’de, lise eğitimini ise 2020 yılında Emir Sultan Anadolu Lisesi’nde tamamladı. 2020 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği Bölümü’ne başladı. Lisans eğitimi süresince yapay zekâ, görüntü işleme ve çoklu platform uygulama geliştirme alanlarında çeşitli projelerde yer aldı. 2024 yılında YEDITEK firmasında Bilgisayarla Görü ve Endüstriyel Otomasyon alanında staj yaptı. 2025 yılında Venhancer firmasında Yapay Zekâ AR-GE stajını tamamladı. Lisans bitirme projesi kapsamında, EKG görüntülerinin YOLOv9 tabanlı derin öğrenme modelleriyle analiz edildiği, mobil ve mikroservis mimarili bir sağlık uygulaması geliştirdi. 2025 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği Bölümü’nden mezun oldu.

## BSM 498 BİTİRME ÇALIŞMASI DEĞERLENDİRME VE SÖZLÜ SINAV TUTANAĞI

KONU:

ÖĞRENCİLER (Öğrenci No/AD/SOYAD): Mustafa BİÇER

Değerlendirme Konusu	İstenenler	Not Aralığı	Not
<b>Yazılı Çalışma</b>			
<b>Çalışma klavuza uygun olarak hazırlanmış mı?</b>	x	0-5	
<b>Teknik Yönden</b>			
<b>Problemin tanımı yapılmış mı?</b>	x	0-5	
Geliştirilecek yazılımın/donanımın mimarisini içeren blok şeması (yazılımlar için veri akış şeması (dfd) da olabilir) çizilerek açıklanmış mı?			
Blok şemadaki birimler arasındaki bilgi akışına ait model/gösterim var mı?			
Yazılımın gereksinim listesi oluşturulmuş mu?			
Kullanılan/kullanılması düşünülen araçlar/teknolojiler anlatılmış mı?			
Donanımların programlanması/konfigürasyonu için yazılım gereksinimleri belirtilmiş mi?			
UML ile modelleme yapılmış mı?			
Veritabanları kullanılmış ise kavramsal model çıkarılmış mı? (Varlık ilişki modeli, noSQL kavramsal modelleri v.b.)			
Projeye yönelik iş-zaman çizelgesi çıkarılarak maliyet analizi yapılmış mı?			
Donanım bileşenlerinin maliyet analizi (prototip-adetli seri üretim vb.) çıkarılmış mı?			
Donanım için gerekli enerji analizi (minimum-uyku-aktif-maksimum) yapılmış mı?			
Grup çalışmalarında grup üyelerinin görev tanımları verilmiş mi (iş-zaman çizelgesinde belirtilebilir)?			
Sürüm denetim sistemi (Version Control System; Git, Subversion v.s.) kullanılmış mı?			
Sistemin genel testi için uygulanan metotlar ve iyileştirme süreçlerinin dökümü verilmiş mi?			
Yazılımın sızma testi yapılmış mı?			
Performans testi yapılmış mı?			
Tasarımın uygulamasında ortaya çıkan uyumsuzluklar ve aksaklıklar belirtilerek çözüm yöntemleri tartışılmış mı?			
<b>Yapılan işlerin zorluk derecesi?</b>	x	0-25	
<b>Sözlü Sınav</b>			
<b>Yapılan sunum başarılı mı?</b>	x	0-5	
<b>Soruları yanıtlama yetkinliği?</b>	x	0-20	
<b>Devam Durumu</b>			
<b>Öğrenci dönem içerisindeki raporlarını düzenli olarak hazırladı mı?</b>	x	0-5	
<b>Diğer Maddeler</b>			
<b>Toplam</b>			

DANIŞMAN (JÜRİ ADINA):

DANIŞMAN İMZASI: