

TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



AKADEMİK YOKLAMA SİSTEMİ

20011038 – Büşra Medine GÜRAL
20011059 – Emir OĞUZ

BİLGİSAYAR PROJESİ

Danışman
Doç. Dr. Mehmet Amaç GÜVENSAN

Haziran, 2024

TEŞEKKÜR

Projemiz boyunca bilgi ve deneyimlerini bizimle paylaştılarak bizlere rehberlik eden ve eksikliklerimizi gidermemiz konusunda yardımcı olan kıymetli danışman hocamız Doç. Dr. Mehmet Amaç GÜVENSAN'a en içten teşekkürlerimizi sunarız. Aynı zamanda Yıldız Teknik Üniversitesi Bilgisayar Mühendisliği bölümündeki tüm akademisyenlerimizin gelişimimiz süresince verdikleri emekler için ayrı ayrı teşekkür ederiz.

Büşra Medine GÜRAL
Emir OĞUZ

İÇİNDEKİLER

SİMGE LİSTESİ	v
KISALTMA LİSTESİ	vi
ŞEKİL LİSTESİ	vii
TABLO LİSTESİ	x
ÖZET	xi
ABSTRACT	xii
1 Giriş	1
2 Ön İnceleme	3
3 Fizibilite	5
3.1 Teknik Fizibilite	5
3.1.1 Yazılım Fizibilitesi	5
3.1.2 Donanım Fizibilitesi	6
3.1.3 Önerilen Yazılımlar ve Donanımlar	6
3.2 İş Gücü ve Zaman Planlaması	7
3.3 Yasal Fizibilite	7
3.4 Ekonomik Fizibilite	8
4 Sistem Analizi	9
4.1 Kullanım Senaryosu Diyagramı	9
4.2 Kullanım Senaryosu Dokümantasyonu	12
4.3 Kavramsal Sınıf Diyagramı	17
5 Sistem Tasarımı	18
5.1 Mobil Uygulama İçin Yazılım Tasarımı	18
5.1.1 Sınıf Diyagramı	18
5.1.2 Ardışıl Diyagram	19
5.2 Görüntü İşleme İçin Tasarım	20

5.2.1	Veri Seti Tasarımı	20
5.2.2	Yazılım Tasarımı	22
5.3	Yoklama Şablonu Tasarımı	30
5.3.1	Yoklama Şablonu Oluşturma Süreci	30
5.3.2	Yazılım Tasarımı	32
5.4	Veri Tabanı Tasarımı	48
5.5	Girdi Çıktı Tasarımı	48
6	Uygulama	49
7	Deneysel Sonuçlar	63
7.1	Model Eğitimi	63
7.2	Model Performansı	63
7.3	Model Tahmin Sonucu Oluşan Çıktıların Değerlendirilmesi	67
7.4	Örnek Şablon Çıktısı	70
8	Performans Analizi	71
8.1	Veri İşleme Süresi ve Sistem Kullanımı	71
8.2	Model Eğitimi Sırasında Sistem Kullanımı	72
8.3	Yoklama Şablonu İşleme Süresi	73
9	Sonuç	74
Referanslar		75
Özgeçmiş		77

SİMGE LİSTESİ

- | | |
|----------|-----------------------------|
| α | Benzerlik Durumu İçin Sabit |
| β | Farklılık Durumu İçin Sabit |

KISALTMA LİSTESİ

CNN	Convolutional Neural Network
CEDAR	The Center of Excellence for Document Analysis and Recognition
CSV	Comma Separated Values
FN	False Negative
FP	False Positive
FHK	Fikri Haklar Kanunu
FK	Foreign Key
HD	High Definition
KVKK	Kişisel Verilerin Korunması Kanunu
OCR	Optical Character Recognition
PK	Primary Key
QR	Quick Response (Çabuk Tepki)
RAM	Random Access Memory
RFID	Radio Frequency Identification
SNN	Siamese Neural Network
TN	True Negative
TP	True Positive

ŞEKİL LİSTESİ

Şekil 3.1 Gantt Şeması	7
Şekil 4.1 Temel Ders İşlemleri	9
Şekil 4.2 Yoklama İşlemleri	10
Şekil 4.3 İmza Karşılaştırma ve Bildirim	11
Şekil 4.4 İstatistik İşlemleri	12
Şekil 4.5 Kavramsal Sınıf Diyagramı	17
Şekil 5.1 Sınıf Diyagramı	18
Şekil 5.2 Ardişil Diyagram	19
Şekil 5.3 CEDAR İmza Veri Seti - Örnek İmza	21
Şekil 5.4 Kaggle İmza Veri Seti - Örnek İmza	21
Şekil 5.5 CSV Dosyasının İçeriği	21
Şekil 5.6 Gri Tonlama [7]	22
Şekil 5.7 Boyutlandırma	23
Şekil 5.8 Gauss Bulanıklaştırma İşlemi	23
Şekil 5.9 Adaptif Eşikleme İşlemi	24
Şekil 5.10 Temel Alınan CNN Mimarisi	25
Şekil 5.11 Temel CNN Mimarisinin Alt Ağ Olarak Kullanıldığı SNN Tasarımı	26
Şekil 5.12 Karışıklık Matrisi (Confusion Matrix) [13]	28
Şekil 5.13 Yoklama Şablonu 1	30
Şekil 5.14 Yoklama Şablonu 2	31
Şekil 5.15 Örnek Görüntü	32
Şekil 5.16 Gri Tonlama	33
Şekil 5.17 Bulanıklaştırma	34
Şekil 5.18 Adaptif Eşikleme	35
Şekil 5.19 Kenar Tespit	36
Şekil 5.20 Çizgi Tespit	37
Şekil 5.21 Çizgi Tamamlama	38
Şekil 5.22 Tespit Edilen Tüm Konturlar	40
Şekil 5.23 Tespit Edilen Öğrenci Numaraları	41
Şekil 5.24 Tespit Edilen Bölge - Örnek 1: 10010001	42
Şekil 5.25 Tespit Edilen Bölge - Örnek 2: 10010015	42

Şekil 5.26 Easy OCR Öğrenci Numarası Tespiti Sonuçları	42
Şekil 5.27 Tespit Edilen İmza Bölgeleri	43
Şekil 5.28 Devamsızlık Görüntüsü Örnek 1 - Ortalama Değer: 0.984	44
Şekil 5.29 Devamsızlık Görüntüsü Örnek 2 - Ortalama Değer: 0.978	44
Şekil 5.30 Öğrencilerin Yoklama Durumu Bilgisi	45
Şekil 5.31 Yoklama Durum Bilgisi İle Karşılaştırmak İçin Örnek Görüntü	45
Şekil 5.32 İmza Karşılaştırma ve Gruplandırma Süreci	46
Şekil 5.33 Varlık-İlişki Diyagramı	48
Şekil 6.1 Kayıt Olma	49
Şekil 6.2 Giriş Ekranı	49
Şekil 6.3 Uygulamaya İlk Giriş	50
Şekil 6.4 Ders Ekleme	50
Şekil 6.5 Eklenen Ders	51
Şekil 6.6 Ders Seçenekleri	51
Şekil 6.7 Ders Silme	52
Şekil 6.8 Sınır Güncelleme	52
Şekil 6.9 Haftalar	53
Şekil 6.10 Katılım Lisesi	53
Şekil 6.11 Fotoğraf Ekleme	54
Şekil 6.12 Galeriden Seçme	54
Şekil 6.13 Görsel Uyarısı	55
Şekil 6.14 Dönem Uyarısı	55
Şekil 6.15 Yoklama Gönderme	56
Şekil 6.16 Güncellenmiş Haftalar	56
Şekil 6.17 Bildirim	57
Şekil 6.18 Haftalık Katılım	57
Şekil 6.19 Devamsızlıklar	58
Şekil 6.20 Seçenekler	58
Şekil 6.21 Mail Gönderme	59
Şekil 6.22 Gelen Mail	59
Şekil 6.23 Haftalık İstatistikler	60
Şekil 6.24 Öğrenci Devamsızlıkları	60
Şekil 6.25 İmza Benzerliği-1	61
Şekil 6.26 İmza Benzerliği-2	61
Şekil 6.27 Öğrenci Listesi	62
Şekil 6.28 Çıkış	62
Şekil 7.1 Model Doğruluk Grafiği	64
Şekil 7.2 Model Kayıp Grafiği	64
Şekil 7.3 Model F1-Skoru Grafiği	65

Şekil 7.4 Eğitim (Training) Seti Karışıklık Matrisi	65
Şekil 7.5 Doğrulama (Validation) Seti Karışıklık Matrisi	66
Şekil 7.6 Test Seti Karışıklık Matrisi	66
Şekil 7.7 Doğruluk (Accuracy) vs. Eşik (Threshold)	67
Şekil 7.8 Karşılaştırma 1 - İmza Görüntüleri	68
Şekil 7.9 Karşılaştırma 1 - Çıktı	68
Şekil 7.10 Karşılaştırma 2 - İmza Görüntüleri	68
Şekil 7.11 Karşılaştırma 2 - Çıktı	68
Şekil 7.12 Karşılaştırma 3 - İmza Görüntüleri	69
Şekil 7.13 Karşılaştırma 3 - Çıktı	69
Şekil 7.14 Yıklama Şablonu - Örnek Çıktı	70
Şekil 8.1 Veri İşleme Sürecinde Sistem Kullanımı	71
Şekil 8.2 Veri İşleme Süresi Kod Çıktısı	72
Şekil 8.3 Model Eğitimi Sırasında Sistem Kullanımı	72

TABLO LİSTESİ

Tablo 3.1 Kullanılan Donanımlar	6
Tablo 3.2 Ekonomik Fizibilite Tablosu	8
Tablo 4.1 Kayıt Olma	12
Tablo 4.2 Sisteme Giriş	13
Tablo 4.3 İlgili Olduğu Dersleri Görüntüleme ve Ekleme	13
Tablo 4.4 İlgili Olduğu Dersleri Silme	13
Tablo 4.5 Bir Dersin Devamsızlık Sınırını Değiştirme	14
Tablo 4.6 Öğrenci Listesini Görüntüleme	14
Tablo 4.7 Yoklama Kağıdının Fotoğrafını Ekleme	14
Tablo 4.8 İmza Karşlaştırılması ve Kaydedilmesi	15
Tablo 4.9 Bir Dersin Haftalık Katılımını Görüntüleme	15
Tablo 4.10 Derslerin Haftalık Katılım İstatistiklerini Görüntüleme	15
Tablo 4.11 Devamsızlık ile İlgili Bildirim Gönderme	16
Tablo 4.12 Devamsızlığı Olan Öğrencileri Bilgilendirme	16
Tablo 4.13 İmza Detaylarını Görüntüleme	16
Tablo 5.1 Veri Seti Kaynakları ve İmza Sayıları	21
Tablo 5.2 CSV Veri Ayrımları	22
Tablo 5.3 Model İçin Görüntü İşleme Adımları - Özeti Tablo	24
Tablo 5.4 Yoklama Şablonu İçin Görüntü İşleme Adımları - Özeti Tablo	39
Tablo 5.5 OCR Teknolojilerinin Başarı Oranları	41
Tablo 5.6 Algoritmanın Karmaşıklık Analizi	47
Tablo 7.1 Model Karşılaştırma Tablosu	63
Tablo 8.1 Veri İşleme Süresi	72
Tablo 8.2 Yoklama Şablonu İşleme Süresi	73

ÖZET

AKADEMİK YOKLAMA SİSTEMİ

Büşra Medine GÜRAL

Emir OĞUZ

Bilgisayar Mühendisliği Bölümü

Bilgisayar Projesi

Danışman: Doç. Dr. Mehmet Amaç GÜVENSAN

Teknolojinin ve yapay zekanın eğitim sektöründe yer edinmeye başlamasının ardından bu alandaki gelişmeler hızla ilerlemektedir. İlgili proje ile bu gelişmelere bir yenisi eklenerek imza tabanlı yoklama sisteme bir alternatif sunulmaktadır. Akademisyenlerin yoklama bilgilerini sisteme girmek için fazlaıyla zaman harcadığı bu dönemde, zamanı daha verimli kullanmalarına olanak tanımak ve öğrenci devamsızlıklarının doğru bir şekilde kayıt altına alınmasını sağlamak temel motivasyon kaynakları olarak belirlenmiştir. Proje, görüntü işleme teknikleri kullanarak yoklama kağıdı üzerinden imzaları tespit etmekte ve benzerlik analizi ile sonuçları bir mobil uygulama üzerinden akademisyene sunmaktadır. Bu işlemler sonucunda yoklama süreçleri hızlandırılarak akademisyenlerin iş yükünün azaltılması hedeflenmektedir.

Anahtar Kelimeler: Yoklama yönetim sistemi, eğitim teknolojileri, dijital dönüşüm, görüntü işleme, mobil uygulama, imza benzerliği, evrişimsel sinir ağı, siyam sinir ağları

ABSTRACT

ACADEMIC ATTENDANCE SYSTEM

Büşra Medine GÜRAL
Emir OĞUZ

Department of Computer Engineering
Computer Project

Advisor: Assoc. Prof. Mehmet Amaç GÜVENSAN

After technology and artificial intelligence started to gain a place in the education sector, developments in this field are progressing rapidly. With the related project, an alternative to the signature-based attendance system is offered by adding a new one to these developments. In this period when academicians spend a lot of time and effort to enter the attendance information into the system, the main motivations are to enable them to use their time more efficiently and to ensure that student absences are recorded accurately. The project detects signatures on the attendance sheet using image processing techniques and presents the results to the academician via a mobile application with similarity analysis. As a result of these processes, it is aimed to speed up the attendance processes and reduce the workload of academics.

Keywords: Attendance management system, education technologies, digital transformation, image processing, mobile application, signature similarity, convolutional neural network, siamese neural network

1

Giriş

Eğitim süreçlerinin dijital bir dönüşümden geçtiği günümüzde, öğrenciler için olduğu gibi akademisyenler için de fayda sağlayacak çeşitli çalışmalar正在被制作中。 Bu çalışmalar sayesinde akademisyenin yönetmesi gereken işlemler, teknoloji desteğiyle daha kısa sürede ve pratik bir şekilde gerçekleştirilebilmektedir. Bu proje ise akademisyenlerin yoklama işlemlerini daha kolay bir şekilde yönetmelerini hedefleyerek, geleneksel yoklama sistemlerine alternatif bir çözüm sunmaktadır.

Temelde, Türkiye'deki üniversitelerde eğitim veren akademisyenlerin çok büyük bir kısmı, akademik başarının ölçülmesi ve sürdürülmesi konusunda düzenli olarak yoklama almakta ve bunun için bir imza kağıdı kullanmaktadır. Yoklama kağıdındaki imza sayısına göre sisteme verileri işlerken kontrol sağlamak ve gerekli hesaplamaları her bir ders ve öğrenci için ayrı ayrı yapmaktadır. Bu durum öğretim üyelerinin verimliliğini düşürmekte ve öğrenci devamsızlığının doğru bir şekilde kaydedilememeye riskini doğurmaktadır. Bu proje akademisyenlerin yoklama listelerini manuel olarak girmelerine gerek kalmadan, çekilen yoklama kağıdı fotoğrafları üzerinden dijitalleştirerek sisteme kaydetmelerine olanak sağlamaktadır. Bu süreç, yalnızca veri dijitalleştirimeyle kalmayıp, aynı zamanda görüntü işleme tekniklerini kullanarak öğrencilerin imzalarının kontrollerini gerçekleştirmeyi içermektedir. Yoklama bilgileri sisteme girilmeden önce, öğrencinin yerine başka birinin imza atmış olma ihtimaline karşı imza benzerlikleri detaylıca incelenmektedir. İnceleme sonuçları ise ilgili akademisyenlerle şeffaf bir şekilde paylaşılırak devamsızlık takibi daha güvenilir kılmaktadır.

Projenin bir başka hedefi ise uyarı modülüdür. Akademisyenin ders ekleme aşamasında veya sonrasında belirlediği katılım oranına göre, öğrencilerin devam takibi ile ilgili uyarı alabilecekleri bir bildirim sistemi bulunmaktadır. Devamsızlık sınırındaki veya sınırı aşmış kişilerin listelendiği bir ekran sayesinde devamsızlık yönetimi gerçekleştirilebilecektir. Ek olarak, bu kişileri bilgilendirmek için mail atabilecekleri bir modül de kullanılabilecektir.

Projede amaçlanan sonuçlar arasında, geleneksel yoklama yöntemlerine bir seçenek olarak akademisyenlere kolaylık sağlamak; zaman ve emek tasarrufu yapabilmelerine imkan tanımak bulunmaktadır. Ayrıca imza benzerliklerini tespit etmek ve mobil uygulama aracılığıyla herhangi bir cihaz üzerinden bu bilgilere erişebilmelerini sağlamak da sonuçlar arasında yer almaktadır.

2 Ön İnceleme

Bu bölümde, proje teması üzerine yapılmış çalışmaların literatür taraması ve konu üzerine genel açıklamalar bulunmaktadır. Proje genel bir başlık altında incelendiğinde, akademisyenler için geliştirilmiş çeşitli yoklama yöntemleri ve sistemleri bulunmaktadır. Ancak her bir projenin odaklandığı farklı temel yaklaşımalar vardır. Bu yaklaşımalar aşağıda listelenmiştir.

Geleneksel yöntemde yoklamalar, öğrenci listesinin bulunduğu bir kağıt üzerindeki imzalar ile alınmaktadır. Akademisyen, ders saatinde öğrenciler arasında bir yoklama kağıdı dolaştırarak bulundukları hafta için imzalarını almakta ve bunu tüm dönem boyunca uygulamaktadır. Ardından bu imzaları inceleyerek ve sayarak ilgili öğrencinin ne kadar devamsızlık yaptığını sisteme kaydetmektedir. Akademisyenin her bir öğrenci için bu işlemi gerçekleştirmesi oldukça zaman alıcıdır. Ek bir durum olarak, akademisyenin imza farklılıklarını detaylı bir şekilde tespit etmesi ise gözle görülür bir fark olmadıkça mümkün değildir. Bu sebeple farklı yöntemlere ihtiyaç duyulmaktadır.

Yoklama yönetiminin yapılabilmesi adına önerilen bir model [1], RFID (Radio Frequency Identification) tabanlı yüz tanıma teknolojisi ile öğrencilerin derse katılımlarının tespit edilmesine odaklanmaktadır. Bu sistemde, sınıfa giriş/çıkış yapan her bir öğrencinin kaydı tutulmakta ve sisteme işlenmektedir. Bu yöntem sayesinde, öğrencilerin sınıfta hangi zaman aralıklarında bulundukları hakkında güvenilir veriler elde edilmektedir. Ancak çeşitli dezavantajlar bulunmaktadır. RFID sistemler yüksek kurulum ve bakım maliyetlerine sahiptir. Her sınıf için sistem gerekliliği nedeniyle ciddi bütçelere ihtiyaç olacaktır. Ayrıca ışıklandırmalar ve yüz değişiklikleri (gözlük, sakal vb.) gibi etmenler, sistemin hatalı çalışmasına yol açabilmektedir.

Bir başka sistemde [2] bir öncekinde olduğu gibi biyometrik bir veri olan iris kullanılmaktadır. Öncelikle öğrencilerin irisleri bir şablon olarak veri tabanına kaydedilmektedir. Ardından bu kişilerin sınıfa girişlerinde göz görüntüleri yakalanarak veri tabanındaki irisleri ile karşılaştırma yapılmaktadır. Tasarlanan

prototip web tabanlı çalışmaktadır. Bu sistemin en büyük dezavantajı hassas bir biyometrik veri olan irisin toplanması, saklanması ve işlenmesidir. Verilerin yetkisiz erişimlerden korunması için yüksek güvenlik önlemleri gerektirmektedir. Ayrıca bir önceki sistemde olduğu gibi, dış faktörler nedeniyle tanıma başarısızlıklarını yaşaması mümkündür.

Yoklama sistemlerini mobil uygulama ile birleştiren SEAts Mobile App [3], QR kodu ve Bluetooth teknolojilerini kullanmaktadır. İlk seçenek olan QR kodda, akademisyenler bir QR kod üretmekte ve öğrenciler bu QR kodu taratmaktadır. Bu şekilde yoklama alınmaktadır. Ancak teknolojik yetersizlik bu sistemde büyük bir problem oluşturmaktadır. QR kodun taratılabileceği bir cihaza erişimi olmayan öğrencilerin bulunduğu veya QR kodun yansıtılabileceği bir ortamın sağlanmadığı sınıflarda, bu sistem kullanıssız olacaktır. Diğer seçenekte ise sınıflara Bluetooth işaretleri konumlandırılmaktadır ve öğrencilerin fiziksel konumları bu şekilde algılanmaktadır. Ancak eski model cihazların yeni Bluetooth sürümleriyle uyumsuz olması ve çok sayıda Bluetooth sinyalinin etkin olması sistemin doğruluk oranını düşürebilmektedir.

Geleneksel yöntem olan imzalı yoklama sistemlerinde kolaylık sağlanması amacıyla yapılan bir proje ise yapay sinir ağlarının kullanıldığı imza karşılaştırmalı bir çalışmadır [4]. Bu çalışma dört adımdan oluşmaktadır: görüntü elde etme, görüntü ön işleme, kayıt ve doğrulama. Çalışmada öğrenci imzaları bir veri tabanında saklanmaktadır ve yeni imzalar, veri tabanındaki imzalar ile karşılaştırılmaktadır. İmza gibi hassas bir verinin saklanması olması güvenlik açısından büyük bir sorun oluşturmaktadır. Ayrıca çalışmada kullanılan imza verilerinin yetersiz olması nedeniyle daha fazla veri kullanılarak modelin iyileştirilmesi gerekmektedir.

Projemiz, diğer projeler ile karşılaştırıldığında basitlik, maliyet ve güvenlik açısından en optimum yolu sunmaktadır. Diğer projelerden en büyük farkı, literatürde incelenen projelerde bulunmayan, imza karşılaştırma sistemine sahip bir mobil arayüz içermesidir. Bu arayüz sayesinde yoklama kağıdının fotoğrafının eklenmesi ve imza benzerlik sonuçlarının görünmesi gibi işlemler pratik bir şekilde gerçekleştirilebilmektedir. Ayrıca QR kod, Bluetooth veya RFID gibi ek yapılara ihtiyaç duyulmadan yalnızca bir mobil arayüz ile yoklama verilerinin dijitalleştirilmesi, ortalama bir maliyet ile mümkün kılınmaktadır. İmza verilerin yalnızca karşılaştırma aşamasında kullanılıp herhangi bir ortamda saklanmaması ise güvenliği üst düzeye çıkarmaktadır.

3

Fizibilite

Bu bölümde, projenin uygulanabilirliği hakkında yapılan fizibilite çalışmalarına dair bilgiler sunulmaktadır.

3.1 Teknik Fizibilite

3.1.1 Yazılım Fizibilitesi

Proje kapsamında geliştirilen mobil uygulama için React Native çerçevesi tercih edilmiştir. Bu çerçevenin cross-platform çalışabilmesi nedeniyle iOS ve Android işletim sistemlerinde uyumlu çalışma avantajına sahiptir. Ayrıca Javascript tabanlı olmasıyla performans ve taşınabilirlik açısından proje gereksinimlerine uyum sağlamaktadır. Ek olarak ekibin React Native üzerinde deneyiminin bulunması da bu çerçevenin seçilmesinde rol oynamıştır.

React Native'in iOS ve Android üzerinde çalışabilmesi için gerekli konfigürasyonları efektif bir şekilde yapan Expo Go adlı yazılım kullanılmıştır. Bu yazılımlar ile Visual Studio Code idesinde uygulama geliştirilmiştir.

Projenin derin öğrenme modeli ile imza karşılaştırma kısmı Python dilinde geliştirilmiştir. Bu seçimin sebebi; Python dilinin yapay zeka ve derin öğrenme alanlarında kapsamlı araçlara ve kütüphanelere sahip olmasıdır. Ayrıca Python, anlaşılır bir sözdizimine ve esnek bir yapıya sahiptir. Bu özellikleri sayesinde, model tasarıımı sürecinde hızlı örnekleme ve prototip oluşturma gibi kolaylıklar da sağlanmaktadır.

Geliştirme sürecinde; görüntü işleme için OpenCV, sayısal işlemler için NumPy, veri analizi için Pandas, veri görselleştirme ve grafik çizimi için Matplotlib ve Seaborn, derin öğrenme için Tensorflow ve Keras kütüphaneleri kullanılmıştır. Projenin düzgün ve sorunsuz çalışması için bu kütüphanelerin yüklü ve uygun sürümlere sahip olması gereklidir.

Veri tabanı yönetim sistemi olarak MongoDB kullanılmıştır. Belge tabanlı ve ilişkisel olmayan bir veri tabanı olmasıyla JSON benzeri belgeleri depolayarak veri modellemeyi ve uygulama geliştirmeyi daha esnek bir hale getirmektedir. Yoklama verilerinin artması sebebiyle büyüyen veri hacimlerine en uyum sağlayan veri tabanı olduğundan proje için ideal bir seçenekdir.

Projenin analiz ve tasarım aşamalarında geliştirilen kullanım senaryoları ve veri-ilişki diyagramları gibi şemalar, draw.io üzerinden yapılmıştır. Basit ve kullanıcı dostu bir arayüz sunması, tercih edilmesinin başlıca sebeplerindendir.

3.1.2 Donanım Fizibilitesi

Proje, farklı donanımlara ve işletim sistemlerine sahip iki farklı bilgisayarda geliştirilmiş, mobil uygulamanın test edilmesi için iki ayrı işletim sistemine sahip telefon kullanılmıştır. Kullanılan cihazlara ait detaylar Tablo 3.1'de verilmiştir.

Tablo 3.1 Kullanılan Donanımlar

Marka/Model	İşletim Sistemi	CPU	GPU	RAM	Depolama
Macbook Pro M3	MacOS	Apple M3 8 çekirdek	Apple GPU 10 çekirdek	16 GB	512 GB SSD
Casper Excalibur	Windows 10 Pro	Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz	NVIDIA GeForce GTX 2060	32 GB	1 TB NVMe SSD
iPhone 11	iOS	A13 Dual-Core 2.65 GHz 6 çekirdek	Apple GPU 4 çekirdek	4 GB	128 GB
Samsung A7 2017	Android 8.0 Oreo	ARM Cortex-A53 1.9 GHz 8 çekirdek Exynos	Mali-T830 MP3	3 GB	32 GB

3.1.3 Önerilen Yazılımlar ve Donanımlar

Uygulamanın sorunsuz bir şekilde çalışması ve yüksek performanslı olabilmesi amacıyla bazı minimum donanım özellikleri belirlenmiştir. İlk olarak Android cihazlarda çalışabilmesi için minimum Android 8.0 (Oreo) veya üzeri önerilmektedir. iOS için ise minimum gereksinim iOS 11 veya üzeridir.

İşlemci tarafında minimum olarak çift çekirdeklı 1.5GHz işlemci önerilmekte, ancak daha yüksek performans için dört çekirdeklı 2.0 GHz işlemcinin tercih edilmesi tavsiye edilmektedir.

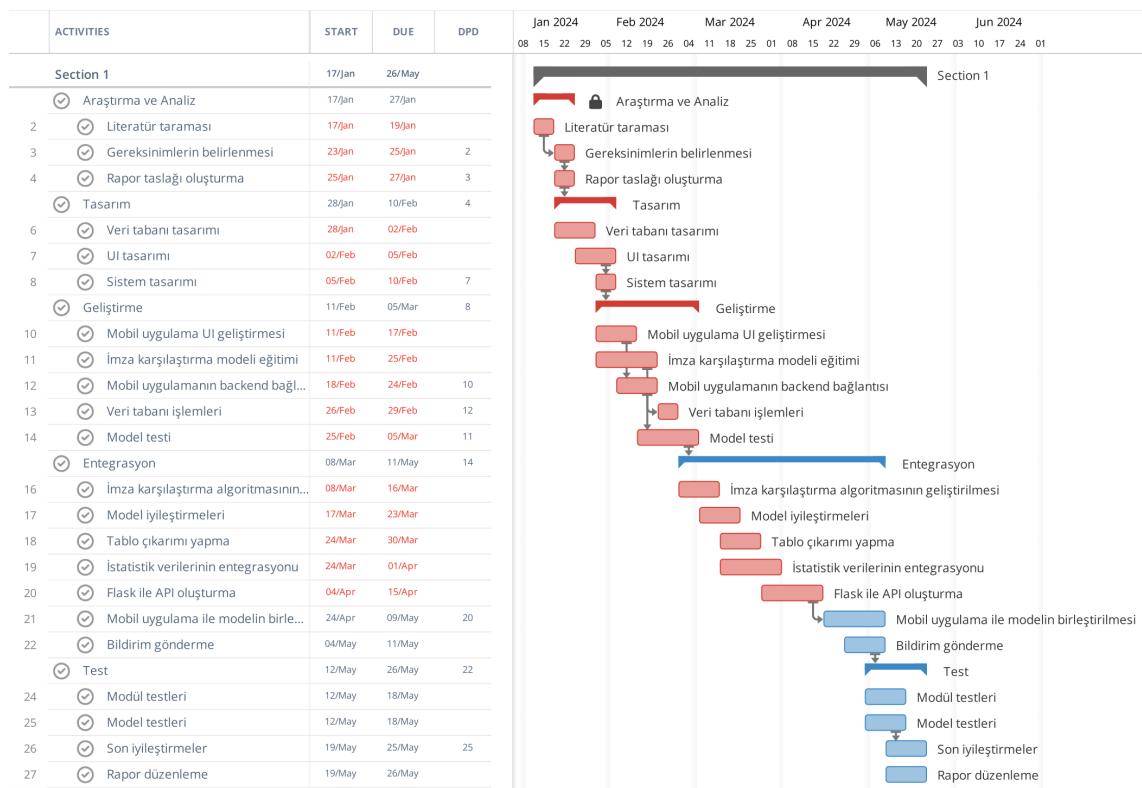
Bellek (RAM) açısından, uygulamanın sorunsuz çalışabilmesi için minimum 2 GB RAM önerilmekte, ancak daha iyi performans için 4 GB RAM'lı bir cihaz tavsiye edilmektedir. Depolama alanı olarak ise minimum 4 GB, tercih edilen ise 8 GB olarak belirlenmiştir.

Projenin görüntü işleme ve karşılaştırma odaklı olduğu göz önüne alındığında, kamera özellikleri büyük bir önem taşımaktadır. Bu nedenle minimum 8 MP arka kamera önerilmekte, daha iyi sonuçlar için ise 12 MP veya üzeri kameralar tavsiye edilmektedir.

Ekran boyutu ve çözünürlük için minimum gereksinim 4 inç, 720p (HD) çözünürlüktür. Ortalama 5.5 inç, 1080p (Full HD) veya üzeri çözünürlükler daha iyi bir deneyim sunacaktır.

3.2 İş Gücü ve Zaman Planlaması

Projenin zaman planlaması Şekil 3.1 'deki gibidir.



Şekil 3.1 Gantt Şeması

3.3 Yasal Fizibilite

6698 sayılı Kişisel Verilerin Korunması Kanunu (KVKK)'na göre, öğrenci ve akademisyenlerin bilgileri "kişisel veri" olarak kabul edilmekte ve bu verilere ilişkin işlemler KVKK'nın genel ilkelerine ve özel hükümlerine uygun şekilde gerçekleştirilmektedir. Bu kapsamda, veri tabanında saklanan öğrenci ve akademisyen bilgilerinin KVKK'deki ilkelere göre paylaşılması, hukuka aykırı olarak işlenmesi ve bilgilere erişilmesinin önlenmesi amacıyla tedbirler alınmaktadır. 8. madde

kapsamında öğrenci ve akademisyenlerin bilgileri, önceden belirlenmiş, açık ve meşru amaçlar doğrultusunda işlenmektedir. Bu amaçlar, projenin amacı ve kapsamı ile sınırlı olmaktadır. Ayrıca, projede öğrenci imzaları saklanmamaktadır; imzalar yalnızca yoklama doğrulama sürecinde anlık olarak işlenip analiz edilmekte ve bu işlemler sonrasında herhangi bir biçimde veritabanında tutulmamaktadır. Bu yaklaşım, öğrencilerin kişisel verilerinin güvenliğini maksimize ederken, KVKK gerekliliklerine tam uyum sağlanması olanak tanımaktadır.

Fikri Haklar Kanunu (FHK)'na göre, bilgisayar programları ve diğer fikri eserler yasal koruma altındadır. Uygulamanın geliştirilmesi için gerekli olan Windows ile MacOS işletim sistemleri ve Visual Studio Code gibi çeşitli uygulamaların lisansları, onaylanan lisans sözleşmesine uygun olarak kullanılmıştır. Tasarım, geliştirme ve raporlama aşamalarında yararlanılan tüm kaynaklar kaynakçada belirtilmiştir.

3.4 Ekonomik Fizibilite

Uygulamanın geliştirilmesi için ihtiyaç duyulan öğeler ve aylık planlamalara dair ücretlendirmeler Tablo 3.2'de verilmiştir. Proje ekibi iki kişiden oluştuğundan gerekli öğelerin adetleri bu doğrultuda belirlenmiştir. Tüm bileşenlerin toplamı 138.703,00 Türk Lirası tutmaktadır.

Tablo 3.2 Ekonomik Fizibilite Tablosu

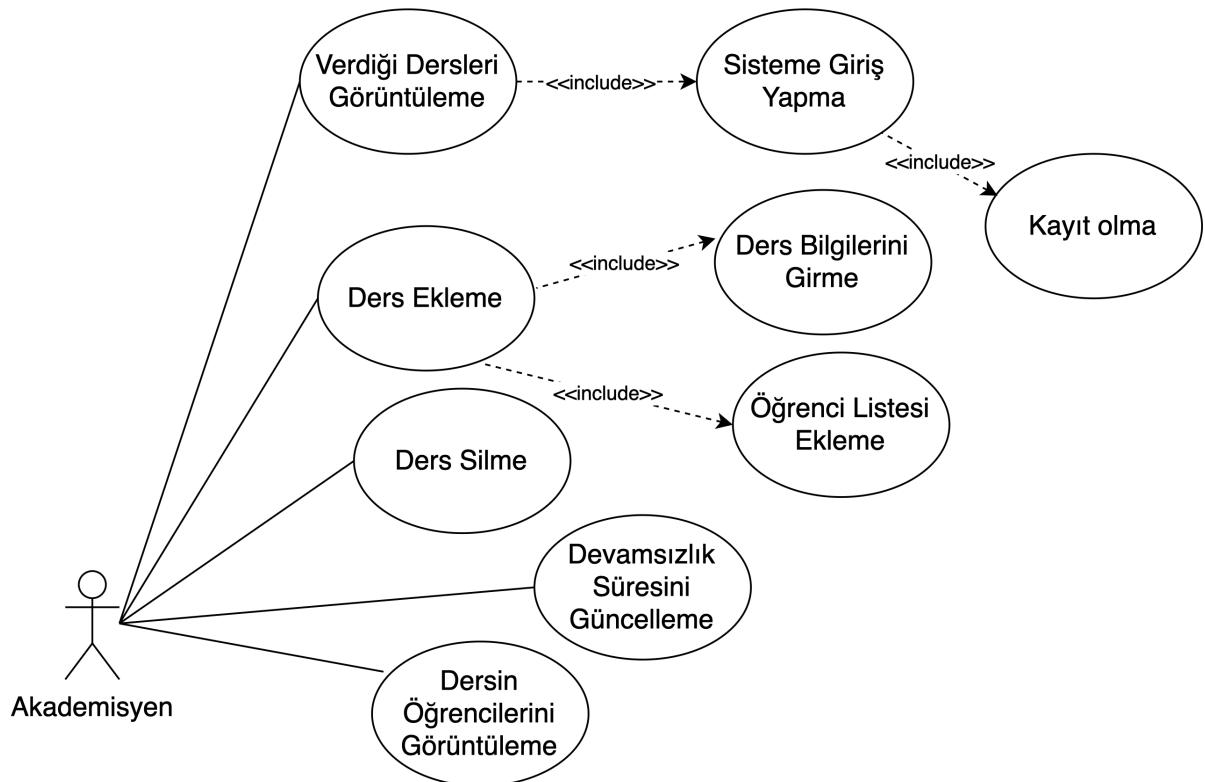
Öge	Birim Fiyat (TL)	Adet	Toplam Fiyat
Laptop	25000	2	50000
İşletim Sistemi(Windows 11)	6599	2	13198
Visual Studio Code	0	2	0
React Native	0	2	0
Expo Go (6 ay)	19200	1	19200
Python	0	2	0
MongoDB	11040	1	11040
Google Colab	165	2	165
Android Telefon (Samsung Galaxy S22)	20000	1	20000
iOS Telefon (iPhone 11)	22000	1	22000
Microsoft 365	3100	1	3100

4 Sistem Analizi

Bu bölümde, sistem analizi sürecinde geliştirilen diyagramlar ve ilgili dokümantasyonlar bulunmaktadır.

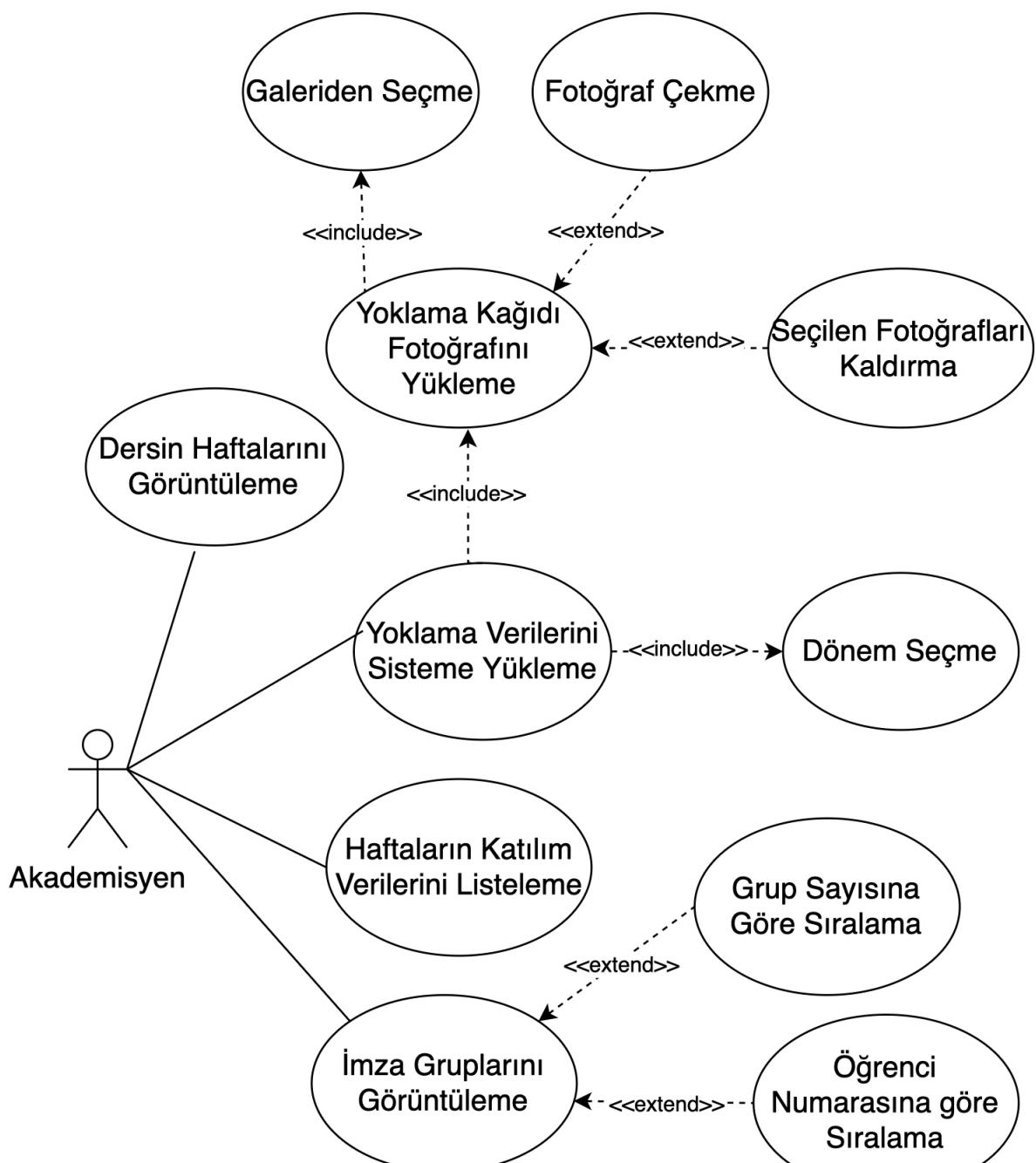
4.1 Kullanım Senaryosu Diyagramı

Akademisyenler uygulamanın temel kullanıcııdır. Ders ekleme, yoklama kağıdını uygulamaya yükleme, devamsızlık istatistiklerine erişme ve öğrencilere mail gönderme gibi çeşitli yeteneklere sahiptir. İşlevlere dair tasarlanan kullanım senaryosu, rahat anlaşılabilmesi amacıyla, aşağıdaki gibi farklı yapılarda gösterilmiştir.



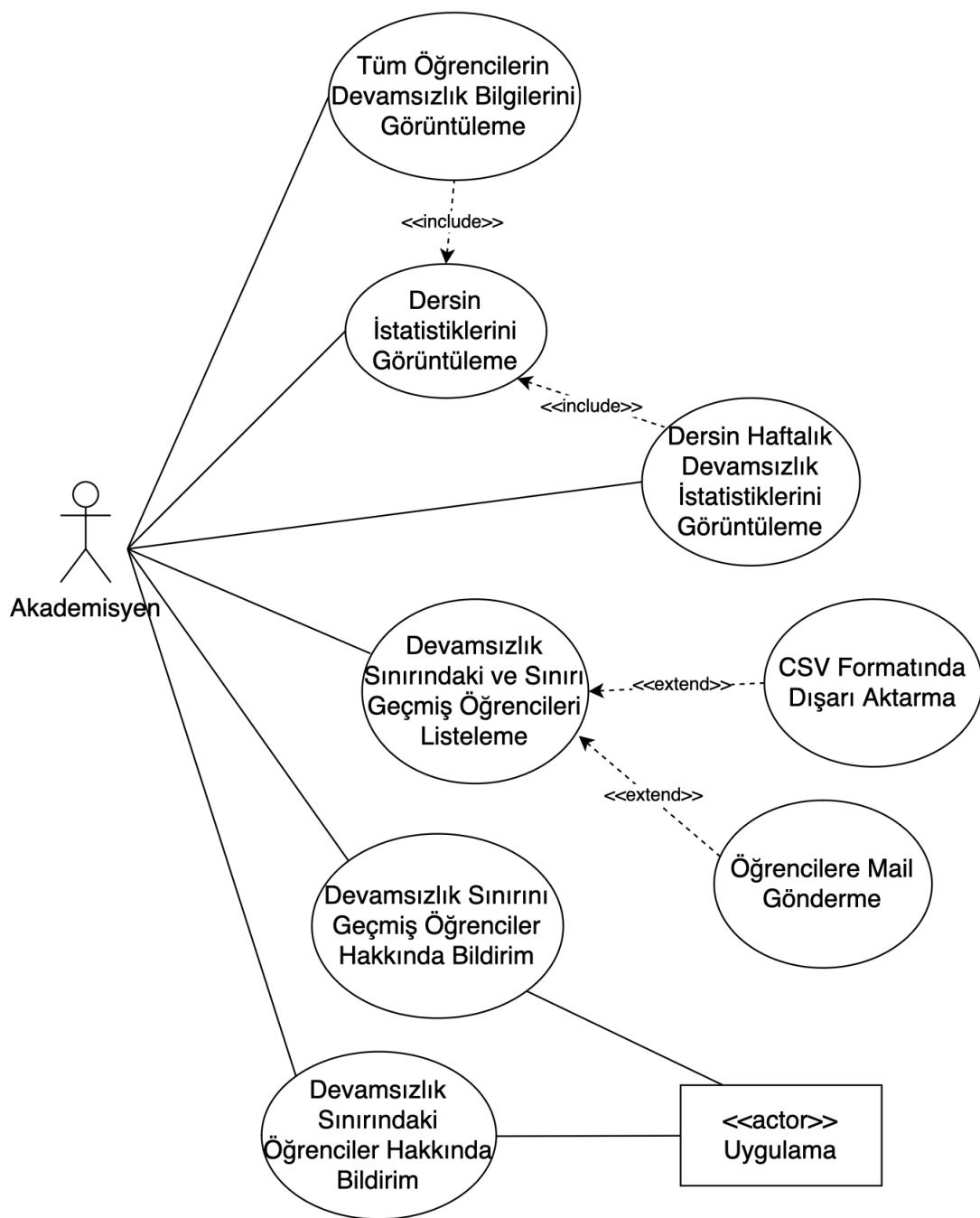
Şekil 4.1 Temel Ders İşlemleri

Uygulamanın yoklama işlemleri ile ilgili modülleri Şekil 4.2'de verilmiştir.



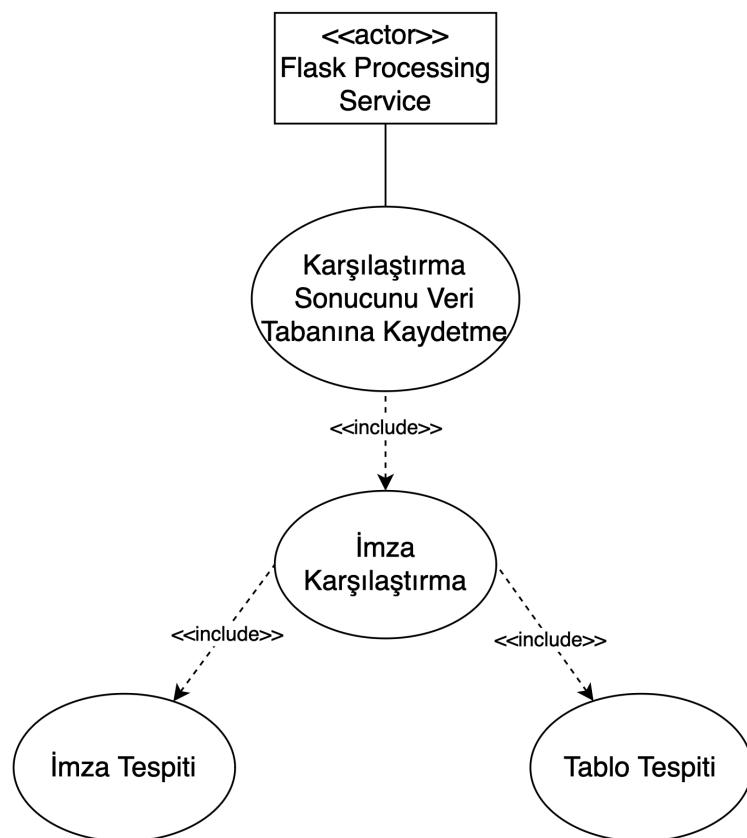
Şekil 4.2 Yoklama İşlemleri

Uygulamanın istatistik görüntüleme ve bildirim gönderme işlemleri ile ilgili modüller 4.3'te verilmiştir.



Şekil 4.3 İmza Karşılaştırma ve Bildirim

Flask servisi ile gerçekleşen imza karşılaştırma ve kaydetme işlemi 4.4'te verilmiştir.



Şekil 4.4 İstatistik İşlemleri

4.2 Kullanım Senaryosu Dokümantasyonu

Kullanım senaryosu kapsamında gerekli dokümanlar aşağıdaki gibi hazırlanmıştır.

Tablo 4.1 Kayıt Olma

Senaryo	Kayıt Olma
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, uygulamaya giriş yapabilmek için kaydolmak ister.
Ön Koşullar	Akademisyen bir okul mailine sahip olmalıdır.
Son Koşullar	Kayıt başarıyla gerçekleşir.
Ana Başarılı Senaryo	1. Akademisyen, uygulama açıldığında gelen ekrandaki 'Kayıt ol.' yazısına tıklar. 2. Ekrandaki ilgili alanları doldurarak 'Kayıt Ol' butonuna tıklar.
Alternatif Akuş	1-2a. Hatalı veya eksik veri 1. Giriş verileri doğru ve tam girilmediğinde ekranda hata mesajı yazar.

Tablo 4.2 Sisteme Giriş

Senaryo	Sisteme Giriş
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, gerekli işlemleri gerçekleştirebilmek için sisteme giriş yapmak ister.
Ön Koşullar	Akademisyen bir mail ve parolaya sahip olmalıdır.
Son Koşullar	Giriş başarıyla yapılır.
Ana Başarılı Senaryo	<ol style="list-style-type: none"> 1. Akademisyen, giriş ekranındaki alanlara sırasıyla mail ve parolasını doğru bir şekilde girer. 2. 'Giriş Yap' butonuna tıklar.
Alternatif Akış	<ol style="list-style-type: none"> 1-2a. Hatalı veya eksik veri 1. Giriş verileri doğru girilmediğinde ekranda hata mesajı yazar.

Tablo 4.3 İlgili Olduğu Dersleri Görüntüleme ve Ekleme

Senaryo	İlgili Olduğu Dersleri Görüntüleme ve Ekleme
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, verdiği dersleri görüntülemek ister.
Ön Koşullar	Akademisyen okul maili ve parolası ile sisteme giriş yapar.
Son Koşullar	Dersler başarıyla görüntülenir.
Ana Başarılı Senaryo	<ol style="list-style-type: none"> 1. Akademisyen, giriş yaptıktan sonra çıkan ekranda derslerini görüntüler. 2. 'Artı' butonuna tıklayarak ekleyeceği dersin detaylarını girer. 3. Öğrenci bilgilerini içeren CSV dosyasını seçtikten sonra 'Ekle' butonuna basar.
Alternatif Akış	<ol style="list-style-type: none"> 1-3a. Hatalı veya eksik veri 1. Ders verilerini eksik girdiğinde veya CSV dosyası seçmediğinde hata alır.

Tablo 4.4 İlgili Olduğu Dersleri Silme

Senaryo	İlgili Olduğu Dersleri Silme
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, eklediği bir dersi silmek ister.
Ön Koşullar	Akademisyen okul maili ve parolası ile sisteme giriş yapar.
Son Koşullar	Ders başarıyla silinir.
Ana Başarılı Senaryo	<ol style="list-style-type: none"> 1. Akademisyen, giriş yaptıktan sonra çıkan ekranda derslerini görüntüler. 2. Ders kartındaki 'Çöp Kutusu' simgesine tıklayarak seçtiği dersi siler.
Alternatif Akış	-

Tablo 4.5 Bir Dersin Devamsızlık Sınırını Değiştirme

Senaryo	Bir Dersin Devamsızlık Sınırını Değiştirme
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, eklediği bir dersin devamsızlık sınırını güncellemek ister.
Ön Koşullar	Akademisyen okul maili ve parolası ile sisteme giriş yapar.
Son Koşullar	Devamsızlık sınırı başarıyla güncellenir.
Ana Başarılı Senaryo	<p>1. Akademisyen, giriş yaptıktan sonra çıkan ekranda derslerini görüntüler.</p> <p>2. Ders kartındaki 'Kalem' simgesine tıklayarak devamsızlık sınırına yeni bir değer girer.</p> <p>3. 'Güncelle' butonuna tıklayarak değişikliği kaydeder.</p>
Alternatif Akış	<p>1-3a. Hatalı veya eksik veri</p> <p>1. Devamsızlık sınırını eksik girdiğinde hata alır.</p>

Tablo 4.6 Öğrenci Listesini Görüntüleme

Senaryo	Öğrenci Listesini Görüntüleme
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, eklediği bir derse kayıtlı öğrencileri görüntülemek ister.
Ön Koşullar	Akademisyen okul maili ve parolası ile sisteme giriş yapar.
Son Koşullar	Öğrenciler başarıyla görüntülenir.
Ana Başarılı Senaryo	<p>1. Akademisyen, istediği bir dersin altındaki 'Öğrenci Listesi' butonuna tıklar.</p> <p>3. Öğrenci listesi ekranda listelenir.</p>
Alternatif Akış	-

Tablo 4.7 Yıklama Kağıdının Fotoğrafını Ekleme

Senaryo	Yıklama Kağıdının Fotoğrafını Ekleme
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, öğrencilerin derslerdeki devamsızlıklarını sisteme otomatik olarak ekleyebilmek ister.
Ön Koşullar	Akademisyen okul maili ve parolası ile sisteme giriş yapar.
Son Koşullar	Yıklama kağıdı başarıyla yüklenir.
Ana Başarılı Senaryo	<p>1. Akademisyen, istediği bir dersin altındaki 'Haftalar' butonuna tıklar.</p> <p>2. Sağ üstte bulunan kamera butonuna tıklar.</p> <p>3. Görselleri ve dönemi seçtikten sonra 'Yükle' butonuna tıklar.</p>
Alternatif Akış	<p>2-3a. Fotoğraf seçimi</p> <p>1. Fotoğrafı kamera ile çekebilir.</p> <p>2. Fotoğrafı galeriden seçebilir.</p>

Tablo 4.8 İmza Karşılaştırılması ve Kaydedilmesi

Senaryo	İmza Karşılaştırılması ve Kaydedilmesi
Birincil Aktör	Akademisyen, Flask Servisi
İlgililer ve Beklentileri	Akademisyen, yoklama kağıdındaki imzaları karşılaştırarak sisteme işlemek ister.
Ön Koşullar	Akademisyen yoklama kağıdının fotoğrafını sisteme girer.
Son Koşullar	Karşılaştırma başarıyla yapılarak sonuçlar kaydedilir.
Ana Başarılı Senaryo	<ol style="list-style-type: none"> 1. Görüntü, Flask servisi ile işlenir. 2. Tablolar algılanır ve imzalar belirlenir. 3. İmza karşılaştırma algoritması ile karşılaştırmalar yapılır. 4. Sonuçlar veri tabanına kaydedilir.
Alternatif Akış	-

Tablo 4.9 Bir Dersin Haftalık Katılımını Görüntüleme

Senaryo	Bir Dersin Haftalık Katılımını Görüntüleme
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, öğrencilerin devamsızlık bilgilerini görüntülemek ister.
Ön Koşullar	Akademisyen maili ve parolası ile sisteme giriş yapar.
Son Koşullar	Devamsızlık bilgileri başarıyla görüntülenir.
Ana Başarılı Senaryo	<ol style="list-style-type: none"> 1. Akademisyen ilgili dersi seçikten sonra 'Haftalar' butonuna tıklar. 2. Listelenen haftalardan birindeki bulunan 'Katılım Listesi' butonuna tıklar ve liste görüntülenir.
Alternatif Akış	-

Tablo 4.10 Derslerin Haftalık Katılım İstatistiklerini Görüntüleme

Senaryo	Derslerin Haftalık Katılım İstatistiklerini Görüntüleme
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, derslerin haftalık katılım istatistiklerini görüntülemek ister.
Ön Koşullar	Akademisyen maili ve parolası ile sisteme giriş yapar.
Son Koşullar	Katılım istatistikleri başarıyla görüntülenir.
Ana Başarılı Senaryo	<ol style="list-style-type: none"> 1. Akademisyen ilgili dersi seçer ve 'İstatistikler' butonuna tıklar. 2. Haftalık katılım grafiği ve katılımcı sayıları ekrana gelir.
Alternatif Akış	-

Tablo 4.11 Devamsızlık ile İlgili Bildirim Gönderme

Senaryo	Devamsızlık ile İlgili Bildirim Gönderme
Birincil Aktör	Uygulama
İlgililer ve Beklentileri	Uygulama, devamsızlık sınırlarındaki öğrenciler için akademisyene bildirim gönderemek ister.
Ön Koşullar	Yoklama verileri uygulama üzerinden yüklenmelidir.
Son Koşullar	Bildirim başarıyla gönderilir.
Ana Başarılı Senaryo	<ol style="list-style-type: none"> 1. Seçilen fotoğraflar işlenmesi için gönderilir. 2. Devamsızlık sınırlarındaki ve sınırı geçmiş öğrenciler tespit edilir. 3. ‘Yoklama bilgileri güncellendi!’ şeklindeki bildirim telefona gönderilir.
Alternatif Akış	-

Tablo 4.12 Devamsızlığı Olan Öğrencileri Bilgilendirme

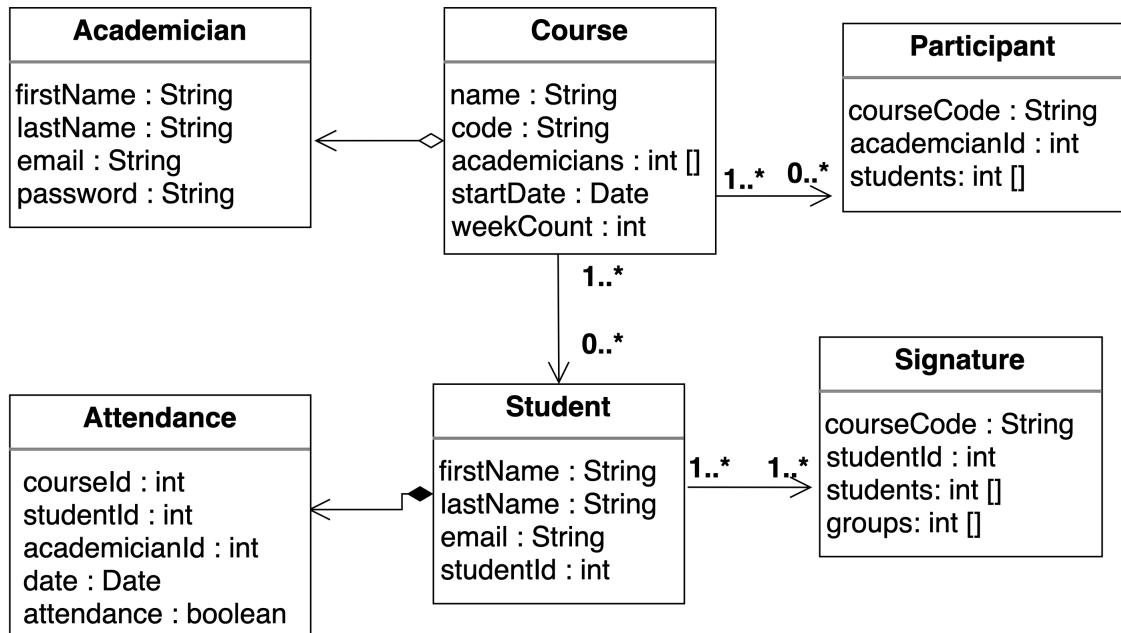
Senaryo	Devamsızlığı Olan Öğrencileri Bilgilendirme
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, devamsızlık sınırında veya sınırı geçmiş öğrencileri bilgilendirmek ister.
Ön Koşullar	Yoklama verileri sistemde bulunmalıdır.
Son Koşullar	Öğrenciler başarıyla bilgilendirilir.
Ana Başarılı Senaryo	<ol style="list-style-type: none"> 1. Akademisyen, istediği bir dersin altındaki ‘Devamsızlıklar’ butonuna tıklar. 2. Sınırı geçmiş veya sınırın altındaki öğrenciler ekranda listenir. 3. Sağ üstteki simgeye tıklanarak seçenekler açılır. 4. ‘Dışarı Aktar’ seçeneği ile listeyi öğrencilerle paylaşır.
Alternatif Akış	4a. Bu seçenek yerine ‘Mail Gönder’ seçeneği ile öğrencilere bir mail gönderir.

Tablo 4.13 İmza Detaylarını Görüntüleme

Senaryo	İmza Detaylarını Görüntüleme
Birincil Aktör	Akademisyen
İlgililer ve Beklentileri	Akademisyen, hangi haftalarda imzaların farklılaşlığı bilgisini görüntülemek ister.
Ön Koşullar	Yoklama verileri sistemde bulunmalıdır.
Son Koşullar	İmza grupları başarıyla görüntülenir.
Ana Başarılı Senaryo	<ol style="list-style-type: none"> 1. Akademisyen, istediği bir dersin altındaki ‘İmza Benzerlikleri’ butonuna tıklar. 2. Öğrencilerin numaraları ve imza grupları ekranda listenir.
Alternatif Akış	-

4.3 Kavramsal Sınıf Diyagramı

Sistem analizi kapsamında tasarlanan kavramsal sınıf diyagramı Şekil 4.5'te gösterilmiştir.



Şekil 4.5 Kavramsal Sınıf Diyagramı

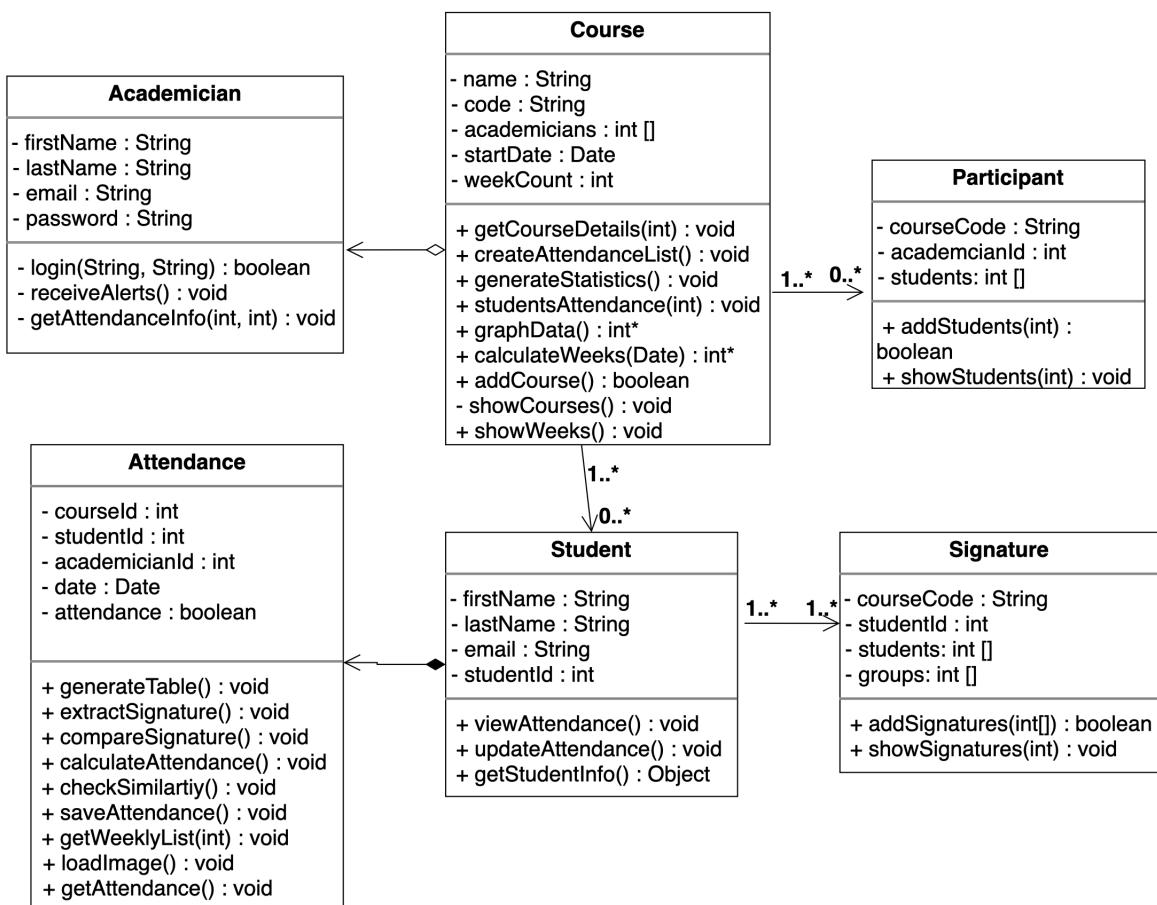
5 Sistem Tasarımı

Bu bölümde, sistem tasarımı için geliştirilen diyagramlar ve görüntü işleme adımları detaylı bir şekilde yer almaktadır.

5.1 Mobil Uygulama İçin Yazılım Tasarımı

5.1.1 Sınıf Diyagramı

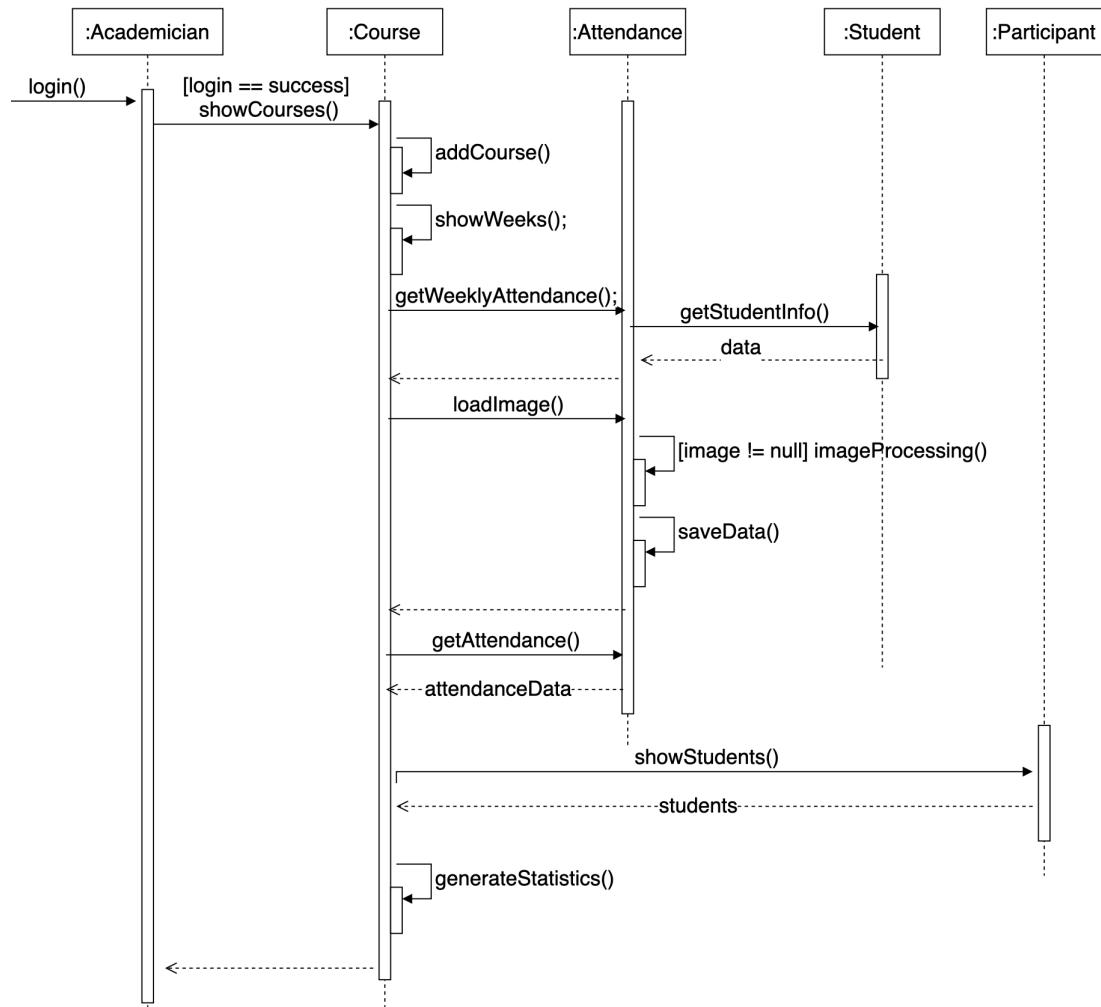
Yazılım tasarımı kapsamında tasarlanan sınıf diyagramı şekil 5.1'de gösterilmiştir.



Şekil 5.1 Sınıf Diyagramı

5.1.2 Ardışıl Diyagram

Yazılım tasarımı kapsamında tasarlanan ardışıl diyagram şekil 5.2'de gösterilmiştir.



Şekil 5.2 Ardışıl Diyagram

5.2 Görüntü İşleme İçin Tasarım

İki imza arasındaki benzerliğin tespit edilmesi sürecinde, öncelikle sınıflandırma temeline dayanan bir CNN modeli hazırlanmıştır. Bu CNN modeli, işlenmiş iki imza görüntüsünün yatay eksende birleştirilmesi ile oluşan veriler ile eğitilmiştir. Kayıp fonksiyonu olarak ise kategorik çapraz entropi (categorical cross entropy) kullanılmıştır. İki imza görüntüsü, bu model ile 0 (orijinal) veya 1 (sahte) olarak sınıflandırılmıştır. Ancak bu model, iki görüntü arasındaki vektörel açıya bağlı olarak bir benzerlik oranı elde etmek yerine sınıflandırma işlemi yaptığı için projenin amacına yönelik bir model olamamıştır. Bu sebepten ötürü tasarım olarak bir SNN modeli kullanılmıştır.

Raporun bu kısmındaki sistem tasarımlı bölümü, el yazısı imzalarını karşılaştırın ve bu imzalar arasındaki benzerlik oranını belirleyen bir SNN modelinin ve bu model için kullanılan verilerin kapsamlı bir açıklamasını içermektedir. Tasarımın temel amacı, bir bireye ait imzalar arasındaki benzerlikleri ve farkları belirleyerek sahteciliği tespit etmektir.

5.2.1 Veri Seti Tasarımı

Model eğitimi için kullanılan veri seti, gerçek kişilere ait el yazısı imzalarını içermektedir. Bu veri setinde, her farklı birey için belirli sayıda orijinal ve sahte imzalar bulunmaktadır. Modelin doğruluğunu ve genelleme yeteneğini artırmak için veri toplama, etiketleme ve bölümlendirme aşamaları özenle tamamlanmıştır.

5.2.1.1 Veri Toplama Süreci

Veri seti oluşturulurken farklı kaynaklar kullanılmıştır. Bu işlemin yapılmasıındaki amaç, birbirinden farklı örüntülere sahip veri setleri sayesinde modelin genelleme yeteneğini artırmaktır. Veri toplama sürecinde belge analizi ve tanıma konusunda araştırma merkezi olan CEDAR tarafından hazırlanmış imza görüntüler [5] ile Kaggle platformunda Signature Verification Dataset [6] başlığı altında bulunan imza görüntülerini kullanılmıştır. Bu görüntülere ek olarak çevreden toplanan imzalar da veri setine eklenmiştir. Veri toplama süreci sonucunda toplamda 115 farklı kişiye ait birçok imza elde edilmiştir. 63 birey için 24 adet orijinal ve sahte imza, 52 birey için ise 12 adet orijinal ve sahte imza vardır.



Şekil 5.3 CEDAR İmza Veri Seti -
Örnek İmza



Şekil 5.4 Kaggle Veri Seti -
Örnek İmza

Tablo 5.1 Veri Seti Kaynakları ve İmza Sayıları

Veri Seti Kaynağı	Orijinal İmza Sayısı	Sahte İmza Sayısı	Toplam İmza Sayısı
CEDAR (55 klasör)	1320	1320	2640
Kaggle (45 klasör)	540	540	1080
Çevreden Toplanan (İlk 8 klasör)	192	192	384
Çevreden Toplanan (Son 7 klasör)	84	84	168
Toplam	2136	2136	4272

5.2.1.2 Veri Etiketleme Süreci

İmza görüntülerinin karşılaştırma sonuçları CSV uzantılı bir dosyaya yazılmıştır. Bu CSV dosyası üç sütundan oluşmaktadır. İlk sütunda karşılaştırılan ilk imza görüntüsünün dizini bulunurken, ikinci sütunda ikinci imza görüntüsünün dizini bulunmaktadır. Üçüncü sütunda ise bu imzaların karşılaştırılma sonu bulunmaktadır. Bu sütun; eğer iki imza da aynı kişiye ait orijinal imzalar ise sonuç 0, bu imzalardan biri orijinal, diğeri sahte ise sonuç 1 olacak şekilde oluşturulmuştur.

```
1/org_1_22.jpg,1/org_1_23.jpg,0  
1/org_1_22.jpg,1/org_1_24.jpg,0  
1/org_1_23.jpg,1/org_1_23.jpg,0  
1/org_1_23.jpg,1/org_1_24.jpg,0  
1/org_1_24.jpg,1/org_1_24.jpg,0  
1/org_1_1.jpg,1/frg_1_1.jpg,1  
1/org_1_1.jpg,1/frg_1_2.jpg,1  
1/org_1_1.jpg,1/frg_1_3.jpg,1  
1/org_1_1.jpg,1/frg_1_4.jpg,1  
1/org_1_1.jpg,1/frg_1_5.jpg,1
```

Şekil 5.5 CSV Dosyasının İçeriği

5.2.1.3 Veri Ayırma İşlemi

CSV uzantılı dosyada bulunan karşılaştırma içerikleri; %70 eğitim, %15 doğrulama ve %15 test olmak üzere üç farklı kümeye bölünmüştür ve farklı üç dosya olarak kaydedilmiştir. Bu üç CSV dosyasının her birinde eşit oranda 0 ve 1 sonucu bulunmaktadır. Bu dosyalar, daha sonra görüntü işleme adımlarında kullanılacaktır.

Tablo 5.2 CSV Veri Ayrımları

Etiket	Anlamı	Toplam	Eğitim	Doğrulama	Test
0	(Orijinal İmza x) vs. (Orijinal İmza y)	22956	16069	3444	3443
1	(Orijinal İmza) vs. (Sahte İmza)	22876	16013	3432	3431

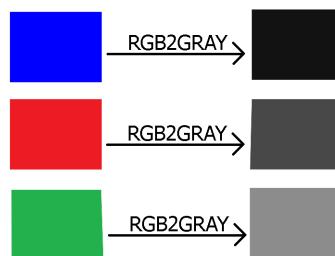
5.2.2 Yazılım Tasarımı

5.2.2.1 Model İçin Görüntü İşleme Adımları

Görüntü işleme, yapay zeka modellerinin eğitiminde kullanılacak verilerdeki kusurları azaltmak ve bu verilerin kalitesini artırıp özellik çıkarma işlemlerini kolaylaştırmak için kritik bir öneme sahiptir. Oluşturulan veri setindeki imzaların modelin eğitiminde kullanılması sürecinde birkaç farklı teknik uygulanmıştır. Bu teknikler için Python dilinde bulunan OpenCV kütüphanesi kullanılmıştır. Her bir teknik, CSV dosyasındaki ilk iki sütunda dizinleri bulunan her iki imza görüntüsü için sırasıyla uygulanmıştır. Bu ilk iki sütundaki imza görüntülerinin verisi, ayrı iki tane girdi dizisi olarak; üçüncü sütunda bulunan sonuç değerleri (etiketler) ise çıktı dizisi olarak model eğitiminde kullanılmıştır.

- Adım 1: Gri Tonlama (Grayscale) ile Görüntü Okuma

Görüntüler, OpenCV kütüphanesi kullanılarak gri tonlamada okunmaktadır. Elimizdeki imza görüntülerinde renk bilgisi öbensizdir, piksellerdeki yoğunluk ve parlaklık bilgisi imza şeklinin algılanması için yeterlidir. Bu sebeple gri tonlama kullanılmıştır.



Şekil 5.6 Gri Tonlama [7]

- Adım 2: Boyutlandırma

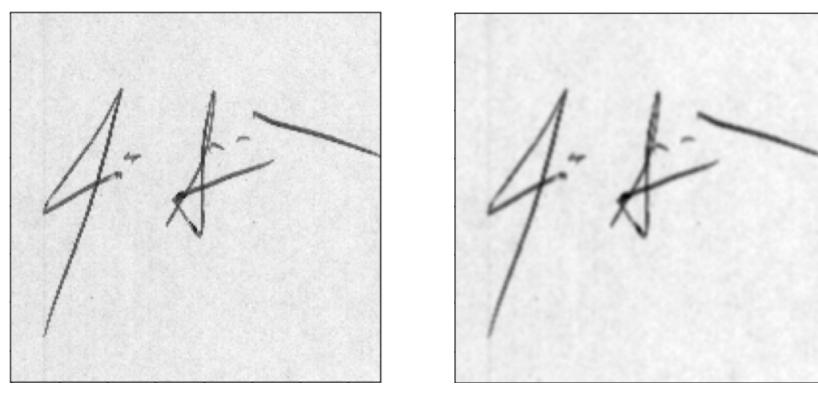
Okunan her görüntü için yeniden boyutlandırma işlemi uygulanmıştır. Boyutlandırma işlemi için 128x128 piksel boyutu kullanılmıştır. Model eğitimi için 64x64 ve 224x224 piksel boyutları da denenmiştir. Ancak 64x64 boyutundaki görüntülerde veri kaybı çoktur ve bu görüntüler, model eğitimi için verimli olamamıştır. Öte yandan, 224x224 görüntüler, veri boyutunu ve işlem yükünü artırılmıştır ve model test aşamasında 128x128 görüntüler ile arasında belirgin bir performans farkı olmamıştır. Bu sebeplerden dolayı görüntü boyutu olarak 128x128 piksel tercih edilmiştir. Bu boyut, görüntü kalitesi ve işlem verimliliğinde iyi bir denge sağlamıştır.



Şekil 5.7 Boyutlandırma

- Adım 3: Gauss Bulanıklığı (Gaussian Blur) Uygulama

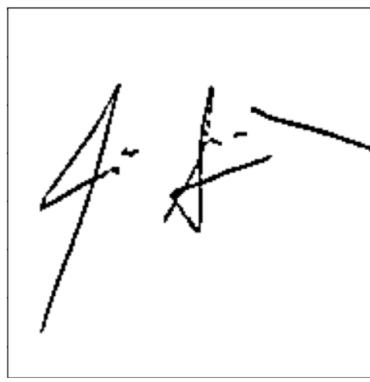
İlk iki işlemden sonra, her bir görüntüye Gauss Bulanıklaştırma uygulanarak görüntülerdeki gürültüler azaltılmıştır. Bulanıklaştırma işlemi için 3x3 boyutunda Gaussfiltresi kullanılmıştır. Bu işlem sayesinde görüntü üzerindeki istenmeyen keskin geçişler yumuşatılmış, görüntü daha pürüzsüz hale getirilmiş ve modelin daha temel özelliklere odaklanması sağlanmıştır.



Şekil 5.8 Gauss Bulanıklaştırma İşlemi

- Adım 4: Adaptif Eşikleme (Adaptive Threshold)

Adaptif eşikleme metodu için OpenCVde bulunan işlevde parametre olarak ADAPTIVE-THRESH-GAUSSIAN-C sabit değeri kullanılmıştır. Bu değer, bloklarda bulunan piksellerin merkez noktaya olan uzaklıklarının bir Gauss fonksiyonuna göre ağırlıklarının hesaplanmasıyla oluşturulan eşik noktasıdır [8]. Bu yöntem, görüntüdeki aydınlatma gibi değişimlere iyi uyum sağlayabilmekte ve tutarlı sonuçlar üretebilmektedir.



Adaptif Eşikleme Uygulanmış Görüntü

Şekil 5.9 Adaptif Eşikleme İşlemi

- Adım 5: Normalizasyon

Son olarak her bir pikselin değeri, belirli bir aralığa sıkıştırılmaktadır. Bu adımda $[0, 1]$ aralığı kullanılmıştır. Bu işlem sayesinde model daha hızlı eğitilmekte ve daha iyi genelleme yeteneğine sahip olmaktadır. Öte yandan, aşırı uyum (overfitting) riski azaltılmakta ve optimizasyon süreçleri iyileştirilmektedir.

Yapılan işlemlere dair özet bir tablo aşağıda verilmiştir.

Tablo 5.3 Model İçin Görüntü İşleme Adımları - Özeti Tablo

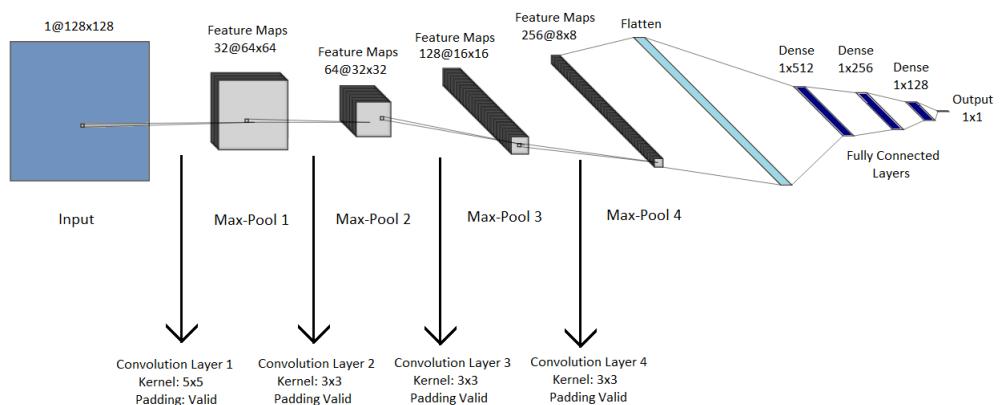
Adım	İşlem	Açıklama
1	Gri Tonlama (Grayscale)	Renk bilgisi gri tonlamaya dönüştürülmüştür.
2	Boyutlandırma	Görüntüler 128x128 piksel boyutuna ayarlanmıştır.
3	Gauss Bulanıklığı (Gaussian Blur)	Gürültü azaltılmış, keskin geçişler yumusatılmıştır.
4	Adaptif Eşikleme (Adaptive Threshold)	Piksellerin eşik noktaları belirlenmiştir.
5	Normalizasyon	Piksel değerleri $[0, 1]$ aralığına getirilmiştir.

Bu görüntü işleme adımları, imzaların tanınması ve karşılaştırılmasındaki model performansını iyi derecede artırır. Bu adımlar sayesinde ham görüntüler, daha düzgün ve tutarlı hale gelir. Sonuç olarak kullanılan görüntü işleme teknikleri, modelin eğitim ve doğrulama süreçlerinde daha yüksek tutarlılık seviyelerine ulaşmasını sağlar.

5.2.2.2 Model Mimarisi

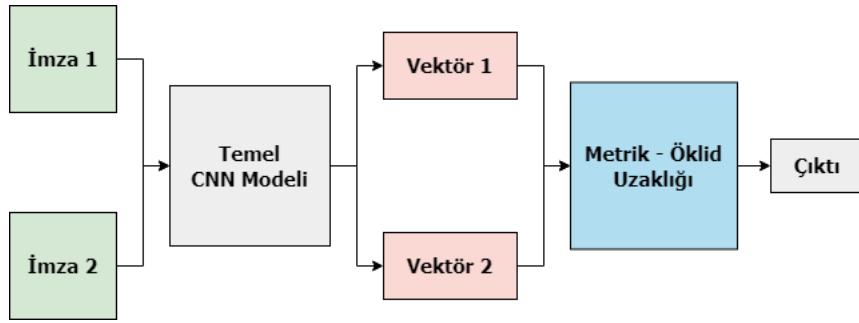
- CNN ve SNN

CNN (Convolutional Neural Network), derin öğrenme topluluğunda oldukça yaygın şekilde kullanılan, başarılı bir yapay sinir ağı mimarisidir. CNN mimarisi, genellikle üç katmandan oluşur. Bunlardan ilki, ağırlıklardan oluşan bir filtrenin, görüntüden özellik çıkarımında kullanıldığı evrişim katmanıdır. İkincisi, doğrusal olmayan katmandır. Bu katmanda, özellik haritalarına bir etkinleştirme fonksiyonu (activation function) uygulanır ve ağıın doğrusal olmayan fonksiyonları modellemesi sağlanır. Üçüncüsü ise havuzlama katmanıdır. Bu katman, özellik haritasındaki komşulukları belirli istatistiksel hesaplamalarla (ortalama, maksimum, vb.) değiştirerek mekansal çözünürlüğünü azaltır. Katmanlardaki birimler yerel olarak birbirlerine bağlıdır. CNN'nin her katmanı, girişi nöron aktivasyonunun çıkışına taşır, böylece tamamen bağlantılı katmanlar oluşturulur. Tüm işlemlerden sonra giriş verileri, bir boyutlu özellik vektörü haline getirilir [9].



Şekil 5.10 Temel Alınan CNN Mimarisi

SNN (Siamese Neural Network), iki özdeş alt CNN içeren bir ağ mimarisidir. Buradaki iki alt CNN mimarisi, aynı parametrelerin ve ağırlıkların bir uzaklık ölçüsü (distance metric) ile birleştirildiği eşit bir yapılandırmaya sahiptir. Bu mimari tipinde, iki alt ağdan biri giriş olarak birinci imza görüntüsünü alırken, diğeri giriş olarak ikinci imza görüntüsünü alır. Daha sonra her iki ağ, verilen imza görüntüsünü girdilerinden özellik çıkarımı yapar. Son olarak, bu iki alt ağın her birinin kullandığı özniteliklerin benzerliği-farkı, son çıkış katmanında uygulanan uzaklık ölçüsü hesaplanarak bulunur. İmza görüntülerinin benzerliklerini karşılaştırılan senaryolarda, SNN mimarisi başarılı sonuçlar vermiştir. Bu sebeple bu mimarinin kullanılmasına karar verilmiştir [9].



Şekil 5.11 Temel CNN Mimarisinin Alt Ağ Olarak Kullanıldığı SNN Tasarımı

- **Mimaride Kullanılan Metrik - Öklid Uzaklığı**

Öklid uzaklığı iki gerçek değerli vektör arasındaki mesafedir. İki vektördeki elemanların kare farklarının toplamının kareköküyle hesaplanır [10].

$$\text{Öklid Uzaklığı (Euclidean Distance)} = |X - Y| = \sum_{i=1}^n (x_i - y_i)^2$$

X = X vektörü veya dizisi

Y = Y vektörü veya dizisi

x_i = Koordinat düzlemindeki yatay eksen değerleri

y_i = Koordinat düzlemindeki dikey eksen değerleri

n = Gözlem sayısı

Öklid uzaklığı, iki imza arasındaki farkı belirlemek için kullanılır. İki vektörel girdi arasındaki Öklid uzaklığı ne kadar küçükse bu görüntüler arasındaki fark o kadar küçüktür. Farkın küçük olması iki görüntünün benzer olduğu ve aynı kişiye ait orijinal imzalar olduğu anlamına gelir. Bu sayede hesaplanan uzaklık değerine göre bir benzerlik hesaplayıcı kullanılabilir. Genel olarak Öklid uzaklığının kullanımı, proje kapsamında kullanılan modelin eğitim ve tahmin sürecinde etkili ve tutarlı olmuştur. Bu sebeple tercih edilmiştir.

5.2.2.3 Model Derlemesinde ve Eğitiminde Kullanılan Metrikler

- **Kayıp Fonksiyonu**

Model eğitimi sırasında kullanılan kayıp fonksiyonu, modelin tahminlerinin gerçek sonuç değerlerinden (etiketler) ne kadar uzak olduğunu ölçer ve bu farkı minimize etmeye çalışır. İmza benzerlik modeli için, siyam ağlarında yaygın kullanılan karşılaştırmalı kayıp fonksiyonu (contrastive loss function) seçilmiştir.

Karşılaştırmalı kayıp, birbirine yakın olan örnekler arasındaki mesafeyi küçültmeyi ve birbirinden uzak olan örnekler arasındaki mesafeyi artırmayı hedefler. Bu şekilde model, benzer örnekleri bir araya getirirken farklı örnekleri birbirinden ayırma becerisi kazanır.

$$L(s_1, s_2, y) = \alpha(1 - y)D_w^2 + \beta y \{\max(0, m - D_w)\}^2$$

Yukarıdaki formülde bulunan 's1' ve 's2' değişkenleri, iki girdi görüntüsü örneğini temsil eder. Öte yandan 'y' değişkeni, iki örneğin karşılaştırmalarının sonuç değerini (benzer ise 0, farklı ise 1) gösteren ikili bir göstergedir. 'α' ve 'β' değişkenleri, benzerlik ve farklılık durumları için iki tane sabiti ifade ederken, 'm' değişkeni, marj değerini ifade eder. Kullanılan verilerde farklı görüntülerin sonuç değeri '1' olarak işaretlendiği için bu tasarımda marj değeri '1' olarak atanır. Son olarak 'Dw' değişkeni, iki girdi görüntüsünün vektörel biçimleri arasındaki Öklid uzaklığını temsil etmektedir. Bu kayıp fonksiyonu sayesinde siyam ağları, ikili benzerlik etiketlerini çiftlere atayan geleneksel yaklaşımlardan farklı davranışır. Sonuç olarak bu işlemler ile birlikte oluşturulan SNN mimarisi, benzer olarak etiketlenen girdi çiftleri için çıkış özellik vektörlerini yakınlaştırır; farklı olarak etiketlenen girdi çiftleri için çıkış özellik vektörlerini uzaklaştırır [11].

- **F1-Skoru Ölçüsü** F1-Skoru hesaplanma işlemini anlamak için öncelikle karışıklık matrisi (confusion matrix), kesinlik (precision) ve duyarlılık (recall) kavramlarını bilmek gereklidir. Karışıklık matrisi, modelin bir veri kümesindeki tahmin performansını temsil eder. İkili (binary) sınıfı (0 veya 1, negatif veya pozitif) bir veri kümesi için bir karışıklık matrisinde dört temel bileşen bulunur:
 - Doğru Pozitifler (TP) "Pozitif" olarak doğru bir şekilde tahmin edilen örneklerin sayısı.
 - Yanlış Pozitifler (FP) "Pozitif" olarak yanlış bir şekilde tahmin edilen örneklerin sayısı.
 - Doğru Negatifler (TN) "Negatif" olarak doğru bir şekilde tahmin edilen örneklerin sayısı.
 - Yanlış Negatifler (FN) "Negatif" olarak yanlış bir şekilde tahmin edilen örneklerin sayısı [12]

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Şekil 5.12 Karışıklık Matrisi (Confusion Matrix) [13]

Karışıklık matrisinin bileşenlerini kullanarak, sırasıyla doğruluk, kesinlik, duyarlılık ve f1-skoru metrikleri tanımlanabilir.

- Doğruluk (Accuracy) ölçüsü, modelin tahminleri arasından toplam doğru sınıflandırmanın toplam veri sayısına bölünmesini ölçer [13].

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- Kesinlik (Precision) ölçüsü, toplam doğru pozitifin toplam tahmin edilen pozitiflere bölünmesini ölçer. Yani bu ölçü, modelin tahmin ettiği doğru pozitif durumların tüm pozitif tahminlerine oranıdır [13].

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{Predictive Results}}$$

- Hatırlama/Duyarlılık (Recall) ölçüsü, toplam doğru pozitifin toplam gerçek pozitiflere bölünmesini ölçer. Yani bu ölçü, modelin tahmin ettiği doğru pozitif durumların toplam gerçek pozitif durumlara oranıdır [13].

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{Actual Results}}$$

- F1-Skoru ölçüsü, kesinlik ve hatırlamanın harmonik ortalaması olan bir ölçütür. Genel olarak modelin tahminlerinin tutarlılığı hakkında fikir verir [13].

$$F\text{-score} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}$$

Bu metrikler sayesinde model performansı ölçülebilir ve değerlendirmelerle birlikte model optimize edilebilir [13].

5.3 Yoklama Şablonu Tasarımı

Bu bölümde, yoklama şablonlarının işlenmesi ve imzaların otomatik olarak tespit edilmesi amacıyla geliştirilen yöntemler ele alınmıştır. Görüntü işleme teknikleri kullanılarak yoklama kağıtlarındaki imza ve katılım bilgileri işlenmiştir. Ayrıca, şablonların karşılaştırılması ve yazılım tasarımı sürecine dair teknik detaylar incelenmiştir.

5.3.1 Yoklama Şablonu Oluşturma Süreci

Yoklama şablonlarının karşılaştırılması sürecinde, iki farklı şablon kullanılmış ve bu şablonların performansları değerlendirilmiştir.

Aşağıdaki şekilde verilmiş olan ilk şablonda, kırptan bölgelerden elde edilen imza görüntülerinin çözünürlük değerleri, görüntü işleme adımlarında birtakım problemlere yol açmıştır. Bu problemlerin başında, adaptif eşikleme işlemi ile elde edilen siyah-beyaz görüntüde istenilen durumun elde edilememesi gelmektedir.

Öğrenci No	Adı Soyadı	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta
10010001	AD-SOYAD-1							
10010002	AD-SOYAD-2							
10010003	AD-SOYAD-3							
10010004	AD-SOYAD-4							
10010005	AD-SOYAD-5							
10010006	AD-SOYAD-6							
10010007	AD-SOYAD-7							
10010008	AD-SOYAD-8							
10010009	AD-SOYAD-9							
10010010	AD-SOYAD-10							
10010011	AD-SOYAD-11							
10010012	AD-SOYAD-12							
10010013	AD-SOYAD-13							
10010014	AD-SOYAD-14							
10010015	AD-SOYAD-15							
10010016	AD-SOYAD-16							
10010017	AD-SOYAD-17							
10010018	AD-SOYAD-18							
10010019	AD-SOYAD-19							
10010020	AD-SOYAD-20							

Şekil 5.13 Yoklama Şablonu 1

İkinci şablon ile yapılan denemelerde kırılan bölgelerden elde edilen imza görüntülerinin çözünürlük değerleri daha iyi sonuçlar vermiştir. Adaptif eşikleme işlemi ile elde edilen siyah-beyaz görüntüde istenen durum başarılı bir şekilde elde edilmiştir.

Öğrenci No.	Ad Soyad	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta
10010001	AD-SOYAD-1							
10010002	AD-SOYAD-2							
10010003	AD-SOYAD-3							
10010004	AD-SOYAD-4							
10010005	AD-SOYAD-5							
10010006	AD-SOYAD-6							
10010007	AD-SOYAD-7							
10010008	AD-SOYAD-8							
10010009	AD-SOYAD-9							
10010010	AD-SOYAD-10							
10010011	AD-SOYAD-11							
10010012	AD-SOYAD-12							
10010013	AD-SOYAD-13							
10010014	AD-SOYAD-14							
10010015	AD-SOYAD-15							

Şekil 5.14 Yıklama Şablonu 2

Arial ve Times New Roman fontları, kullanılan OCR (Optik Karakter Tanıma) yazılımı ile test edilmiştir ve deneysel sonuçlar, OCR yazılımının öğrenci numaralarını Arial fontunda daha tutarlı ve doğru bir şekilde tespit ettiğini göstermiştir. Arial fontunun OCR performansını artırdığı ve daha iyi sonuçlar verdiği belirlenmiştir. Bu nedenle, şablonun okunabilirliğini ve veri doğruluğunu maksimize etmek amacıyla Arial fontu tercih edilmiştir.

Ayrıca, yıkama şablonunda ilgili haftada gelmeyen öğrenci için akademisyen tarafından "/" işaretinin yazılması kararlaştırılmıştır. Bu yöntem, ilgili bölgede imza olmadığınnın anlaşılmasını kolaylaştırarak veri girişinde doğruluğun ve hızın artırılmasını sağlamaktadır. Böylece, yıkama kayıtlarının daha tutarlı ve anlaşılır olması hedeflenmiştir.

Yazılım Tasarımı başlığı altında, imzalanmış ve devamsızlıkların işaretlenmiş olduğu dolu bir şablon örneğine yer verilmiştir.

5.3.2 Yazılım Tasarımı

5.3.2.1 Şablon İçin Görüntü İşleme Adımları

Şablon için görüntü işleme adımlarında, yoklama kağıdının fotoğrafı çekilerek örnek bir görüntü oluşturulmuştur. Bu görüntü üzerinde çeşitli görüntü işleme teknikleri uygulanmıştır.

Sample Image

Öğrenci No.	Ad Soyad	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta
10010001	AD-SOYAD-1	/	Ez	/	Ez	/	Ez	Ez
10010002	AD-SOYAD-2	Ez	/	Ez	/	Ez	/	Ez
10010003	AD-SOYAD-3	/	Mes	Mes	Mes	Mes	Mes	/
10010004	AD-SOYAD-4	Dz	Dz	/	/	Dz	Dz	Dz
10010005	AD-SOYAD-5	Hpd	/	Hpd	Hpd	Hpd	/	Hpd
10010006	AD-SOYAD-6	/	Ls	Ls	/	Ls	Ls	/
10010007	AD-SOYAD-7	L	L	L	L	L	L	L
10010008	AD-SOYAD-8	Mes	Mes	Mes	Mes	/	/	Mes
10010009	AD-SOYAD-9	/	Zet	Zet	/	Zet	Zet	Zet
10010010	AD-SOYAD-10	Qz	Qz	/	Qz	Qz	Qz	/
10010011	AD-SOYAD-11	Zent	Zent	Zent	Zent	/	Zent	Zent
10010012	AD-SOYAD-12	Z	Z	Z	Z	Z	Z	Z
10010013	AD-SOYAD-13	X	X	X	X	X	X	X
10010014	AD-SOYAD-14	Rm	Rm	/	Rm	Rm	Rm	Rm
10010015	AD-SOYAD-15	A	/	A	A	A	A	A

Şekil 5.15 Örnek Görüntü

- Adım 1: Boyutlandırma

A4 kağıdının oranı yaklaşık olarak 1.41'dir ve çekilen fotoğraf görüntülerindeki oran da bu değere yakın bir değerdir. Bu sebeple, çekilen şablon görüntüsü 1750x1250 çözünürlüğünne yeniden boyutlandırılmıştır. Sabit bir çözünürlük kullanılarak imza bölgelerinin tespitinde kullanılacak parametrelerin sabit kalması sağlanmıştır. Boyutlandırma işlemlerinde 1400x1000 de denenmiştir ancak bu çözünürlükteki netlik, öğrenci numarasının OCR ile tutarlı şekilde okunması işleminde yetersiz kalmıştır.

- Adım 2: Gri Tonlama

Renkli görüntülerde renk bilgisi, imza tespiti açısından önemli olmadığından, gri tonlama işlemi uygulanmıştır. Gri tonlama işlemi ile görüntülerdeki renk bilgisi kaldırılarak sadece yoğunluk bilgisi korunmuş, böylece işlem yükü azaltılmış ve imza tespiti işlemleri daha verimli hale getirilmiştir.

Gray Image

Öğrenci No.	Ad Soyad	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta
10010001	AD-SOYAD-1	/	Ez	/	Ez	/	Ez	Ez
10010002	AD-SOYAD-2	Ez	/	Ez	/	Ez	/	Ez
10010003	AD-SOYAD-3	/	Muz	Muz	Muz	Muz	Muz	/
10010004	AD-SOYAD-4	Diz	Diz	/	/	Diz	Diz	Diz
10010005	AD-SOYAD-5	Hpd	/	Hpd	Hpd	Hpd	/	Hpd
10010006	AD-SOYAD-6	/	Kes	Kes	/	Kes	Kes	/
10010007	AD-SOYAD-7	f	f	f	f	f	f	f
10010008	AD-SOYAD-8	Muz	Muz	Muz	Muz	/	/	Muz
10010009	AD-SOYAD-9	/	Red	Red	/	Red	Red	Red
10010010	AD-SOYAD-10	Qz	Qz	/	Qz	Qz	Qz	/
10010011	AD-SOYAD-11	Zant	Zant	Zant	Zant	/	Zant	Zant
10010012	AD-SOYAD-12	Z	Z	Z	Z	Z	Z	Z
10010013	AD-SOYAD-13	X	X	X	X	X	X	X
10010014	AD-SOYAD-14	WMAA	WMAA	/	WMAA	WMAA	WMAA	WMAA
10010015	AD-SOYAD-15	A	/	A	A	A	A	A

Şekil 5.16 Gri Tonlama

- Adım 3: Bulanıklaştırma

Gri tonlamadan sonra, görüntüdeki gürültüyü azaltmak ve keskin geçişleri yumusatmak amacıyla Medyan Filtre kullanılarak bulanıklaştırma işlemi uygulanmıştır. Medyan filtre, özellikle tuz ve biber gürültüsü (salt and pepper noise) gibi ani parlaklık değişikliklerini ortadan kaldırmada etkilidir ve bu işlem, her bir pikselin değerini, o pikselin çevresindeki belirli bir komşuluk alanındaki piksel değerlerinin medyanı ile değiştirmek amacıyla gerçekleştirilir [14]. Medyan, istatistiksel olarak ortanca değeri ifade eder ve bu da komşuluk içindeki değerlerin sıralanması ve ortanca değerin seçilmesi anlamına gelir. Medyan filtreleme işlemi, 3x3 boyutunda bir çekirdek kullanılarak yapılmıştır. Bu çekirdek, her pikselin etrafındaki 8 komşu pikseli kapsar.

Sonuç olarak, bu adımda uygulanan medyan filtreleme işlemi, görüntüdeki istenmeyen gürültüleri azaltarak daha pürüzsüz ve analiz edilebilir bir görüntü elde edilmesini sağlamıştır. Bu işlem, takip eden adımlar olan adaptif eşikleme ve kenar tespiti gibi süreçlerin doğruluğunu artırmak için kritik bir öneme sahiptir.

Blurred Image

Öğrenci No	Ad Soyad	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta
10010001	AD-SOYAD-1	/	Ez	/	Ez	/	Ez	Ez
10010002	AD-SOYAD-2	Ez	/	Ez	/	Ez	/	Ez
10010003	AD-SOYAD-3	/	Mes	Mes	Mes	Mes	Mes	/
10010004	AD-SOYAD-4	De	De	/	/	De	De	De
10010005	AD-SOYAD-5	Hep	/	Hep	Hep	Hep	/	Hep
10010006	AD-SOYAD-6	/	Le	Le	/	Le	Le	/
10010007	AD-SOYAD-7	f	f	f	f	f	f	f
10010008	AD-SOYAD-8	M	M	M	M	/	/	M
10010009	AD-SOYAD-9	/	Red	Red	/	Red	Red	Red
10010010	AD-SOYAD-10	Qd	Qd	/	Qd	Qd	Qd	/
10010011	AD-SOYAD-11	Qd	Qd	Qd	Qd	/	Qd	Qd
10010012	AD-SOYAD-12	3	3	3	3	3	3	3
10010013	AD-SOYAD-13	X	X	X	X	X	X	X
10010014	AD-SOYAD-14	W	W	/	W	W	W	W
10010015	AD-SOYAD-15	4D	/	4D	4D	4D	4D	4D

Şekil 5.17 Bulanıklaştırma

- Adım 4: Adaptif Eşikleme

Adaptif eşikleme, farklı aydınlatma koşulları ve yüzey çeşitlerinden etkilenmeden tutarlı siyah-beyaz görüntüler elde etmek için kullanılmıştır. Bu adımda, adaptif Gauss eşikleme yöntemi kullanılmıştır. Belirli bir blok boyutundaki piksellerin parlaklık ortalaması hesaplanarak uygun eşik değeri belirlenmiştir.

Bu işlem, görüntüdeki her bir pikselin, kendi komşuluk bölgesindeki piksellerin ağırlıklı ortalamasına göre sınıflandırılmasını sağlar. Eğer bir piksel değeri bu ağırlıklı ortalamanın altındaysa siyah (0), üstündeyse beyaz (255) olarak belirlenir. Bu süreç, özellikle imza tespiti gibi ince detayların bulunduğu görüntülerde büyük önem taşır, çünkü yerel aydınlatma değişimlerinden bağımsız olarak tutarlı sonuçlar üretir.

Öğrenci No.	Ad Soyad	Threshold						
		1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta
10010001	AD-SOYAD-1	/	Eski	/	Eski	/	Eski	Eski
10010002	AD-SOYAD-2	Eski	/	Eski	/	Eski	/	Eski
10010003	AD-SOYAD-3	/	Mes	Mes	Mes	Mes	Mes	/
10010004	AD-SOYAD-4	Top	Top	/	/	Top	Top	Top
10010005	AD-SOYAD-5	Top	/	Top	Top	Top	/	Top
10010006	AD-SOYAD-6	/	X	X	/	X	X	/
10010007	AD-SOYAD-7	X	X	X	X	X	X	X
10010008	AD-SOYAD-8	N	N	N	N	/	/	N
10010009	AD-SOYAD-9	/	Det	Det	/	Det	Det	Det
10010010	AD-SOYAD-10	Det	Det	/	Det	Det	Det	/
10010011	AD-SOYAD-11	Det	Det	Det	Det	/	Det	Det
10010012	AD-SOYAD-12	Det	Det	Det	Det	Det	Det	Det
10010013	AD-SOYAD-13	X	X	X	X	X	X	X
10010014	AD-SOYAD-14	Det	Det	/	Det	Det	Det	Det
10010015	AD-SOYAD-15	Det	/	Det	Det	Det	Det	Det

Şekil 5.18 Adaptif Eşikleme

- Adım 5: Kenar Tespiti

Kenar tespiti işlemi, görüntüdeki önemli kenarların ve hatların belirlenmesi amacıyla gerçekleştirilmiştir. Bu adımda, Canny kenar tespiti algoritması kullanılmıştır. Canny algoritması, kenarların belirginleştirilmesi için çift eşikleme yöntemi ile çalışır.

İlk olarak, kenar tespiti için düşük (100) ve yüksek (225) eşik değerleri belirlenmiştir. Kenarların belirginleştirilmesi için morfolojik işlemler uygulanmış; dilatasyon ile kenarlar kalınlaştırılmış, ardından erozyon ile inceltilmiştir. Bu adımlar, önemli kenarların ve hatların net bir şekilde tespit edilmesini sağlamıştır.



Şekil 5.19 Kenar Tespiti

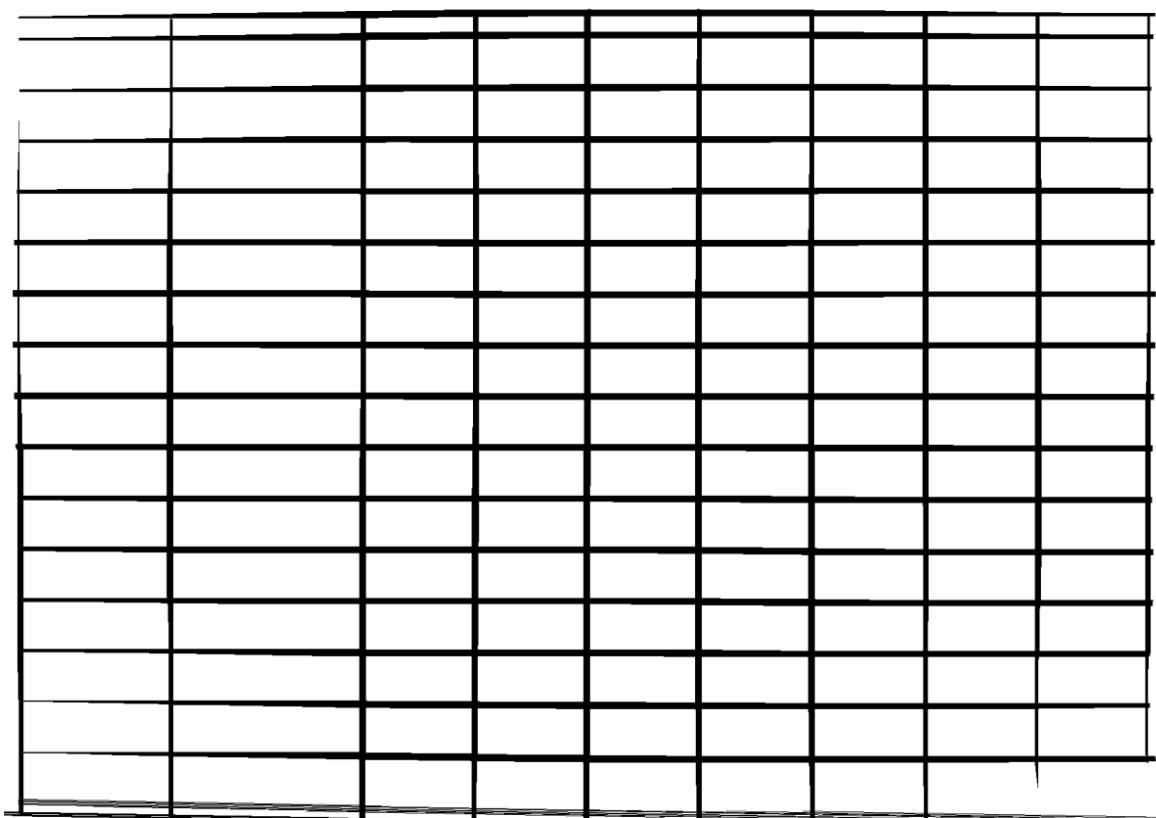
- Adım 6: Çizgi Tespiti

Çizgi tespiti adımda, görüntüdeki yatay ve dikey çizgilerin belirlenmesi için Probabilistic Hough Line Transform yöntemi kullanılmıştır. Bu yöntem, belirli eşik değerleri ile çalışarak görüntüdeki çizgileri tespit eder.

İlk olarak, kenar tespitinden elde edilen görüntü üzerinde morfolojik işlemler uygulanmıştır. Yatay ve dikey çizgilerin daha belirgin hale gelmesi için yatay ve dikey çekirdekler kullanılarak açılma işlemi yapılmıştır. Yatay çizgiler için geniş (50, 1) boyutunda, dikey çizgiler için ise uzun (1, 50) boyutunda çekirdekler kullanılmıştır.

Ardından, HoughLinesP fonksiyonu ile çizgiler tespit edilmiştir. Bu fonksiyon, belirli bir eşik değeri (250), minimum çizgi uzunluğu (500), ve maksimum çizgi aralığı (30) kullanarak çizgileri belirler. Tespit edilen çizgiler, temiz bir görüntü üzerine çizilmiştir.

Clean Image



Şekil 5.20 Çizgi Tespiti

- Adım 7: Çizgi Tamamlama

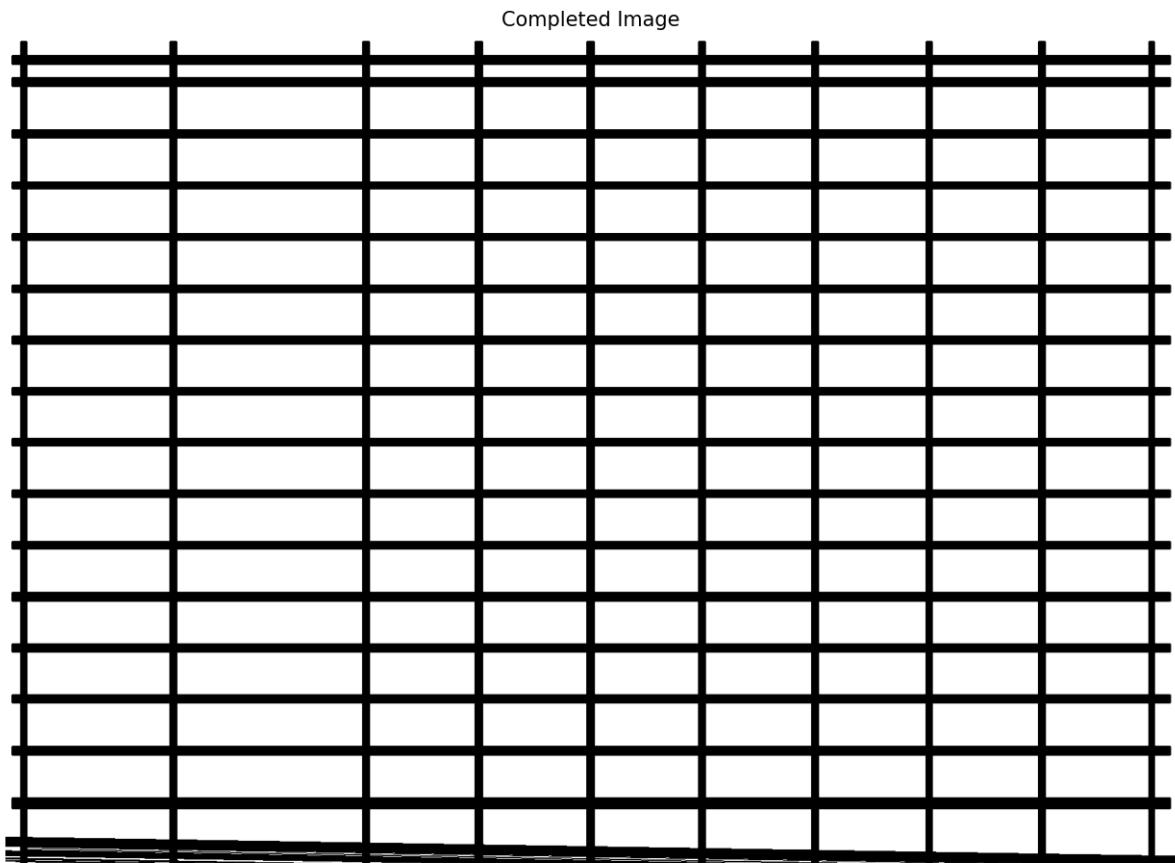
Çizgi tamamlama işlemi sayesinde, Hough Lines Transform ile tespit edilen eksik çizgiler tamamlanmıştır. Bu işlem, yoklama kağıdının tam bir tablo yapısı oluşturması için kritiktir. Hough Lines Transform güçlü olsa da çizgiler; gürültü, düşük kontrast, aydınlatma değişiklikleri veya karmaşık arka plan nedeniyle kesintiye uğrayabilir.

Bu adımda eksik çizgiler, tespit edilen eğim ve kesişim noktalarına göre uzatılmıştır. Dikey çizgiler yukarıdan aşağıya, yatay çizgiler soldan sağa uzatılarak tam bir tablo oluşturulmuştur. İki nokta arasındaki eğim (m) ve y-eksenini kestiği nokta (b) şu şekilde hesaplanır:

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$b = y_1 - m \cdot x_1$$

Dikey çizgilerde eğim sonsuz olup y ekseninde, yatay çizgilerde ise eğim sıfır olup x ekseninde uzatılır. Diğer çizgiler belirlenen eğim ve kesişim noktalarıyla uzatılır.



Şekil 5.21 Çizgi Tamamlama

Yukarıdaki adımlar sonucunda, yoklama kağıdındaki imzalar ve diğer bilgiler daha doğru ve tutarlı bir şekilde tespit edilebilmiştir. Görüntü işleme adımları, şablonun net ve okunabilir hale getirilmesi için kritik bir öneme sahiptir. Bu adımlar sayesinde imzaların doğru bir şekilde tespit edilmesi ve karşılaştırılması sağlanmıştır.

Çizgi tespiti ve tablonun temiz bir görüntüsünün oluşturulması işlemleri, yoklama kağıdındaki öğrenci numarası ve imza bölgelerinin daha tutarlı bir şekilde tespit edilmesi amacıyla gerçekleştirılmıştır. Eğer bu çizgi tespiti ve temiz tablo oluşturma işlemleri yapılmadan doğrudan orijinal görüntünden kontur tespiti yapılrsa, satır ve sütun çizgileriyle birleşen imzalar nedeniyle bölgeler parçalanabilir ve istenilen alanlar doğru şekilde tespit edilemeyecektir. Bu nedenle, önce çizgilerin tespiti ve tamamlanması, ardından temiz bir tablo görüntüsü oluşturulması gereklidir. Bu yöntem, bölgelerin doğru ve tutarlı bir şekilde tespit edilmesini ve analiz edilmesini sağlamıştır.

Yapılan işlemlere dair özet bir tablo aşağıda verilmiştir.

Tablo 5.4 Yoklama Şablonu İçin Görüntü İşleme Adımları - Özeti Tablo

Adım	İşlem	Açıklama
1	Boyutlandırma	Şablon görüntüsü 1750x1250 çözünürlüğüne yeniden boyutlandırılmıştır. Sabit çözünürlük, imza tespitinde kullanılan parametrelerin tutarlığını sağlar.
2	Gri Tonlama	Renk bilgisi kaldırılarak gri tonlama uygulanmış, böylece işlem yükü azaltılarak verimlilik artırılmıştır.
3	Bulanıklaştırma	3x3 Medyanfiltresi kullanılarak görüntülerdeki gürültü azaltılmış ve keskin geçişler yumusatılmıştır.
4	Adaptif Eşikleme	Adaptif Gauss eşikleme ile aydınlatma farklılıklarını giderilmiş ve tutarlı siyah-beyaz görüntüler elde edilmiştir.
5	Kenar Tespiti	Canny kenar tespit algoritması kullanılarak önemli kenarlar ve hatlar belirginleştirilmiştir.
6	Çizgi Tespiti	Hough dönüşümü kullanılarak yatay ve dikey çizgiler tespit edilmiştir.
7	Çizgi Tamamlama	Eksik kalan çizgiler uzatılarak tam bir tablo yapısı elde edilmiştir.

5.3.2.2 Şablonda Kontur Tespiti

- Adım 1: Genel Kontur Tespiti

Şablonun son adımı olan "Adım 7: Çizgi Tamamlama" ile oluşturulan temiz tablo görüntüsündeki konturlar, OpenCV kütüphanesindeki findContour fonksiyonu kullanılarak tespit edilmiştir. Bu konturlar arasından öğrenci numaraları ve imza alanları, belirli genişlik ve yükseklik parametreleri ile ayrılmıştır.

Öncelikle, temiz tablo görüntüsü üzerinde kontur tespiti gerçekleştirilmiştir. Tespit edilen kontur alanları, dikdörtgenler halinde belirlenmiş ve bu alanlar x ve y koordinatlarına göre sıralanmıştır. Bu sıralama, konturların doğru ilişkilendirilmesi ve işlenmesi için önemlidir.

Kontur tespiti sırasında, öğrenci numaraları ve imza alanlarının bulunduğu bölgeler belirli parametreler ile filtrelenmiştir. Öğrenci numaralarının bulunduğu alanlar ve imza alanlarının genişlik parametreleri farklı olduğu için, öncelikle numara alanları, ardından imza alanları tespit edilmiştir. Bu işlem, yoklama şablonundaki bilgilerin doğru ayırtılmasını ve analiz edilmesini sağlamıştır.

All Contours									
Öğrenci No.	Ad Soyad	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta	8. Hafta
10010001	AD-SOYAD-1	/	Ez	/	Ez	/	Ez	Ez	Ez
10010002	AD-SOYAD-2	Ez	/	Ez	/	Ez	/	Ez	Ez
10010003	AD-SOYAD-3	/	Mes	Mes	Mes	Mes	Mes	/	
10010004	AD-SOYAD-4	Hes	Hes	/	/	Hes	Hes	Hes	Hes
10010005	AD-SOYAD-5	Hes	/	Hes	Hes	Hes	/	Hes	Hes
10010006	AD-SOYAD-6	/	Bes	Bes	/	Bes	Bes	/	
10010007	AD-SOYAD-7	f	f	f	f	f	f	f	f
10010008	AD-SOYAD-8	Mes	Mes	Mes	Mes	/	/	Mes	
10010009	AD-SOYAD-9	/	Reet	Reet	/	Reet	Reet	Reet	
10010010	AD-SOYAD-10	Qaz	Qaz	/	Qaz	Qaz	Qaz	/	
10010011	AD-SOYAD-11	Zaz	Zaz	Zaz	Zaz	/	Zaz	Zaz	
10010012	AD-SOYAD-12	3z	3z	3z	3z	3z	3z	3z	
10010013	AD-SOYAD-13	X	X	X	X	X	X	X	
10010014	AD-SOYAD-14	(W)zz	(W)zz	/	(W)zz	(W)zz	(W)zz	(W)zz	
10010015	AD-SOYAD-15	40	/	10	10	40	40	40	40

Şekil 5.22 Tespit Edilen Tüm Konturlar

- Adım 2: Öğrenci Numarası Tespiti

Öğrenci numaralarının tespiti, yoklama şablonundaki öğrenci numarası alanlarının belirlenmesi ve OCR teknolojisi kullanılarak bu numaraların okunmasını içerir. İlk olarak, belirli yükseklik ve genişlik parametrelerine göre öğrenci numarası alanları tespit edilmiştir.

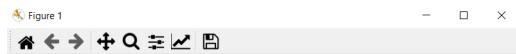
Tespit edilen alanların boş olup olmadığı kontrolü, piksel değerlerinin ortalaması hesaplanarak yapılmıştır. Belirli bir eşik değerin altında olan alanlar dolu kabul edilmiştir ve bu alanlardaki öğrenci numaraları OCR ile okunmuştur. Öğrenci numaralarının OCR ile okunması için Tesseract OCR ve Easy OCR teknolojileri karşılaştırılmıştır. 15 farklı öğrenci numarası içeren yoklama şablonunun farklı koşullarda çekilen 5 fotoğrafında, Tesseract OCR %77.34 (58/75) başarı, Easy OCR ise %94.67 (71/75) başarı göstermiştir. Daha yüksek doğruluk oranı nedeniyle Easy OCR tercih edilmiştir.

Tablo 5.5 OCR Teknolojilerinin Başarı Oranları

OCR Teknolojisi	Doğru Tespit Sayısı	Başarı Oranı (%)
Tesseract OCR	58/75	77.34
Easy OCR	71/75	94.67

Detection Image - Student ID								
Öğrenci No.	Ad Soyad	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta
10010001	AD-SOYAD-1	/	Ez	/	Ez	/	Ez	Ez
10010002	AD-SOYAD-2	Ez	/	Ez	/	Ez	/	Ez
10010003	AD-SOYAD-3	/	Mz	Mz	Mz	Mz	Mz	/
10010004	AD-SOYAD-4	tz	tz	/	/	tz	tz	tz
10010005	AD-SOYAD-5	Mz	/	Mz	Mz	Mz	/	Mz
10010006	AD-SOYAD-6	/	zz	zz	/	zz	zz	/
10010007	AD-SOYAD-7	f	f	f	f	f	f	f
10010008	AD-SOYAD-8	Mz	Mz	Mz	Mz	/	/	Mz
10010009	AD-SOYAD-9	/	zz	zz	/	zz	zz	zz
10010010	AD-SOYAD-10	Qz	Qz	/	Qz	Qz	Qz	/
10010011	AD-SOYAD-11	zz	zz	zz	zz	/	zz	zz
10010012	AD-SOYAD-12	zz	zz	zz	zz	zz	zz	zz
10010013	AD-SOYAD-13	X	X	X	X	X	X	X
10010014	AD-SOYAD-14	zz	zz	/	zz	zz	zz	zz
10010015	AD-SOYAD-15	AD	/	AD	AD	AD	AD	AD

Şekil 5.23 Tespit Edilen Öğrenci Numaraları



Image

10010001



Image

10010015

Şekil 5.24 Tespit Edilen Bölge - Örnek
1: 10010001

Şekil 5.25 Tespit Edilen Bölge - Örnek
2: 10010015

```
--- Easy OCR Öğrenci Numarası Tespiti Sonuçları ---
- Öğrenci 1      : 10010001
- Öğrenci 2      : 10010002
- Öğrenci 3      : 10010003
- Öğrenci 4      : 10010004
- Öğrenci 5      : 10010005
- Öğrenci 6      : 10010006
- Öğrenci 7      : 10010007
- Öğrenci 8      : 10010008
- Öğrenci 9      : 10010009
- Öğrenci 10     : 10010010
- Öğrenci 11     : 10010011
- Öğrenci 12     : 10010012
- Öğrenci 13     : 10010013
- Öğrenci 14     : 10010014
- Öğrenci 15     : 10010015
```

Şekil 5.26 Easy OCR Öğrenci Numarası Tespiti Sonuçları

Sonuç olarak, öğrenci numarası alanları OCR ile okunmuş ve elde edilen metin verisi ile öğrenci numaralarının doğru bir şekilde tespiti sağlanmıştır.

- Adım 3: İmza Bölgesi Tespiti

İmza bölgesi tespiti, yoklama şablonundaki imza alanlarının belirlenmesi ve işlenmesini kapsar. Bu işlemde, öğrenci numarası tespitinden farklı parametreler kullanılarak imza alanları belirlenmiştir.

Detection Image - Signature Box

Öğrenci No.	Ad Soyad	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta
10010001	AD-SOYAD-1	/	Ez	/	Ez	/	Ez	Ez
10010002	AD-SOYAD-2	Ez	/	Ez	/	Ez	/	Ez
10010003	AD-SOYAD-3	/	Mes	Mes	Mes	Mes	Mes	/
10010004	AD-SOYAD-4	Dz	Dz	/	/	Dz	Dz	Dz
10010005	AD-SOYAD-5	Mpo	/	Mpo	Mpo	Mpo	/	Mpo
10010006	AD-SOYAD-6	/	Ls	Ls	/	Ls	Ls	/
10010007	AD-SOYAD-7	f	f	f	f	f	f	f
10010008	AD-SOYAD-8	Mes	Mes	Mes	Mes	/	/	Mes
10010009	AD-SOYAD-9	/	Edt	Edt	/	Edt	Edt	Edt
10010010	AD-SOYAD-10	Qz	Qz	/	Qz	Qz	Qz	/
10010011	AD-SOYAD-11	Zzz	Zzz	Zzz	Zzz	/	Zzz	Zzz
10010012	AD-SOYAD-12	Z	Z	Z	Z	Z	Z	Z
10010013	AD-SOYAD-13	X	X	X	X	X	X	X
10010014	AD-SOYAD-14	(Mmm)	(Mmm)	/	(Mmm)	(Mmm)	(Mmm)	(Mmm)
10010015	AD-SOYAD-15	10	/	10	10	10	10	10

Şekil 5.27 Tespit Edilen İmza Bölgeleri

Elde edilen imza bölgeleri, adaptif eşikleme uygulanmış yoklama şablonu görüntüsünden kırılmıştır. Ancak, adaptif eşikleme uygulanmış yoklama tablosu görüntüsü 0-255 piksel aralığındadır. Modele eğitim için verilen imza görüntüleri ise 0-1 değer aralıklarına normalize edilmiştir. Bu nedenle, yoklama şablonundan kırılan imza görüntülerine normalizasyon işlemi uygulanmıştır. Ayrıca, modelin girdi şekli 128x128 piksel olduğundan, kırılan imza görüntüleri bu çözünürlüğe yeniden boyutlandırılmıştır.

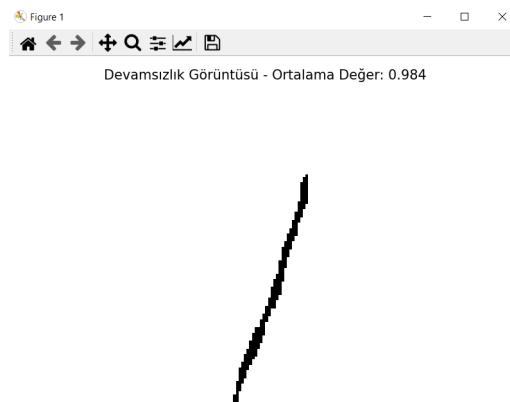
Bu işlemler sonucunda, öğrencilerin imza bölgeleri tespit edilerek analiz edilmiştir. Öğrencilerin imza ve devamsızlık durumları, tespit edilen veriler ile güncellenmiştir.

5.3.2.3 Devamsızlık Tespiti İşlemi

Devamsızlık tespiti işlemi, yoklama şablonunda belirlenen imza bölgelerindeki verilerin analiz edilmesiyle gerçekleştirilmiştir. İlk olarak, her bir imza bölgesinin boş olup olmadığını tespiti yapılmıştır. Bu tespit, ilgili bölgedeki piksel değerlerinin ortalamasının hesaplanmasıyla yapılmıştır. Piksel değerlerinin ortalamasının belirli bir eşik değerin üzerinde olması durumunda (beyaz renge yakın olma durumu), bölge boş olarak kabul edilmiştir. Bölge doluyken ise, bu bölgede imza veya devamsızlık sembolü olan "/" işaretinin bulunup bulunmadığı belirlenmiştir.

Devamsızlık sembollerinin ortalama piksel değerleri 0.98295, imza görüntülerinin ortalama piksel değerleri ise 0.88601 olarak hesaplanmıştır. Bu değerlere dayanarak, belirlenen eşik değeriyle devamsızlık sembollerinin ve imzaların ayırmayı yapılmıştır. Bu ayırım, öğrencinin haftalık devamsızlık durumunun tutarlı bir şekilde güncellenmesini sağlamıştır.

İşlem sonucunda, öğrencilerin 7 haftalık süreçteki devamsızlık durumları güncellenmiş ve devamsızlık sembolü olan "/" işaretini tespit edilen haftalarda ilgili öğrencinin devamsızlık bilgisi 1'den 0'a çevrilmiştir.



Şekil 5.28 Devamsızlık Görüntüsü
Örnek 1 - Ortalama Değer: 0.984



Şekil 5.29 Devamsızlık Görüntüsü
Örnek 2 - Ortalama Değer: 0.978

Şekil 5.28 ve 5.29'da, devamsızlık sembollerinin örnek görüntüleri gösterilmiştir. Bu semboller, ortalama piksel değerlerine göre analiz edilmiştir. Ortalama piksel değerlerinin belirlenen eşik değerine göre sınıflandırılması, devamsızlık ve imza ayırmayı kolaylaştırmıştır. Bu işlem, yoklama şablonundaki bilgilerin doğru ve güvenilir bir şekilde ayırtılmasını sağlamıştır.

--- DEVAMSIZLIK BİLGİSİ ---

- Öğrenci No: 10010001, Yoklama Durumu: [0, 1, 0, 1, 0, 1, 1]
- Öğrenci No: 10010002, Yoklama Durumu: [1, 0, 1, 0, 1, 0, 1]
- Öğrenci No: 10010003, Yoklama Durumu: [0, 1, 1, 1, 1, 1, 0]
- Öğrenci No: 10010004, Yoklama Durumu: [1, 1, 0, 0, 1, 1, 1]
- Öğrenci No: 10010005, Yoklama Durumu: [1, 0, 1, 1, 1, 0, 1]
- Öğrenci No: 10010006, Yoklama Durumu: [0, 1, 1, 0, 1, 1, 0]
- Öğrenci No: 10010007, Yoklama Durumu: [1, 1, 1, 1, 1, 1, 1]
- Öğrenci No: 10010008, Yoklama Durumu: [1, 1, 1, 1, 0, 0, 1]
- Öğrenci No: 10010009, Yoklama Durumu: [0, 1, 1, 0, 1, 1, 1]
- Öğrenci No: 10010010, Yoklama Durumu: [1, 1, 0, 1, 1, 1, 0]
- Öğrenci No: 10010011, Yoklama Durumu: [1, 1, 1, 1, 0, 1, 1]
- Öğrenci No: 10010012, Yoklama Durumu: [1, 1, 1, 1, 1, 1, 1]
- Öğrenci No: 10010013, Yoklama Durumu: [1, 1, 1, 1, 1, 1, 1]
- Öğrenci No: 10010014, Yoklama Durumu: [1, 1, 0, 1, 1, 1, 1]
- Öğrenci No: 10010015, Yoklama Durumu: [1, 0, 1, 1, 1, 1, 1]

Şekil 5.30 Öğrencilerin Yoklama Durumu Bilgisi

Sample Image							
Öğrenci No.	Ad Soyad	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta
10010001	AD-SOYAD-1	/	Ez	/	Ez	/	Ez
10010002	AD-SOYAD-2	Ez	/	Ez	/	Ez	Ez
10010003	AD-SOYAD-3	/	Mes	Mes	Mes	Mes	/
10010004	AD-SOYAD-4	D	D	/	/	D	D
10010005	AD-SOYAD-5	Mes	/	Mes	Mes	/	Mes
10010006	AD-SOYAD-6	/	X	X	/	X	/
10010007	AD-SOYAD-7	f	f	f	f	f	f
10010008	AD-SOYAD-8	Mes	Mes	Mes	Mes	/	Mes
10010009	AD-SOYAD-9	/	Zat	Zat	/	Zat	Zat
10010010	AD-SOYAD-10	Zat	Zat	/	Zat	Zat	/
10010011	AD-SOYAD-11	Zat	Zat	Zat	Zat	/	Zat
10010012	AD-SOYAD-12	Z	Z	Z	Z	Z	Z
10010013	AD-SOYAD-13	X	X	X	X	X	X
10010014	AD-SOYAD-14	Mes	Mes	/	Mes	Mes	Mes
10010015	AD-SOYAD-15	A	/	A	A	A	A

Şekil 5.31 Yoklama Durum Bilgisi İle Karşılaştırmak İçin Örnek Görüntü

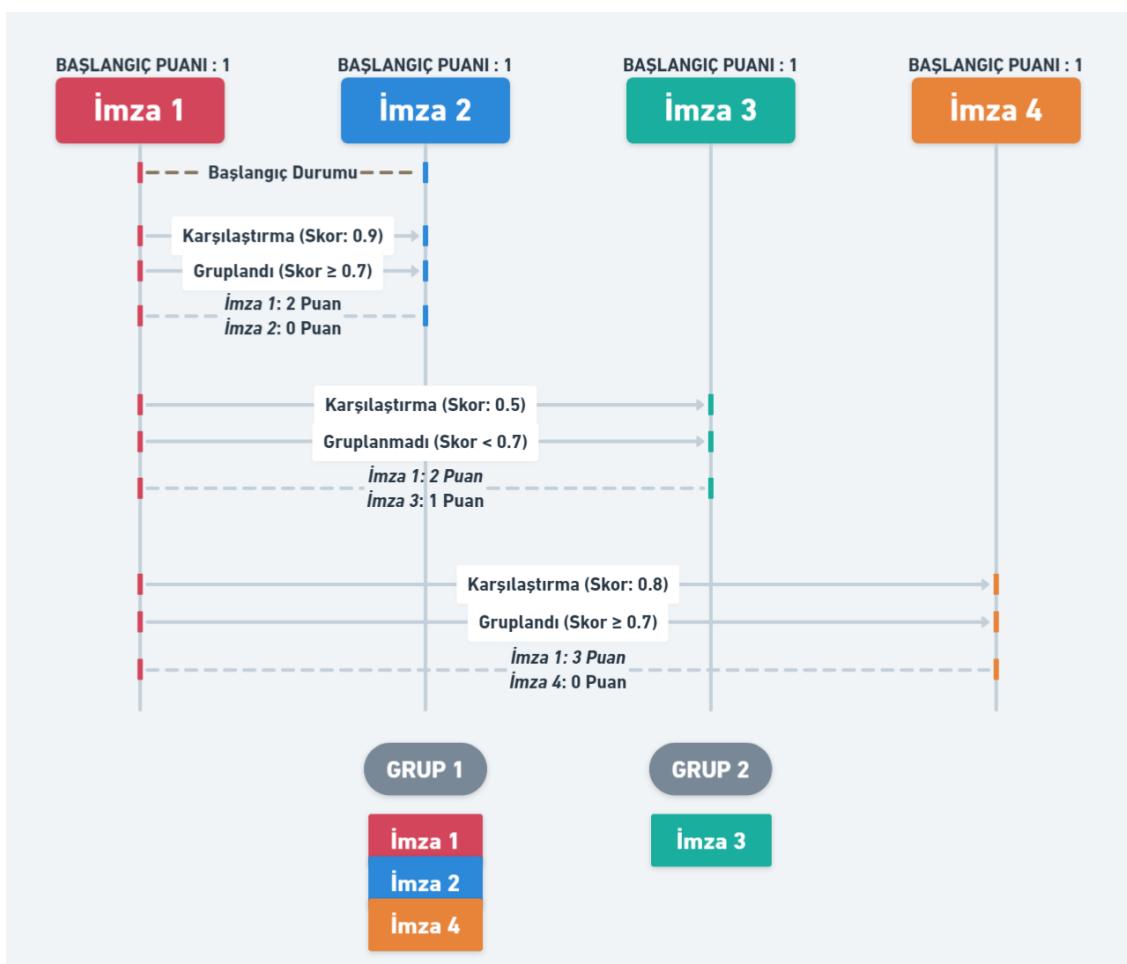
Öğrencilerin yoklama durumu bilgisi Şekil 5.30'da gösterilmiştir. Her bir öğrencinin 7 haftalık süreçteki durumu 1 ve 0 değerleri ile belirtilmiştir. Eğer haftada imza varsa, bu hafta 1 değeri ile gösterilmiştir. Eğer imza yoksa ve devamsızlık sembolü ("/") varsa, bu hafta 0 değeri ile gösterilmiştir. Bu bilgiler, öğrencilerin yoklama durumunu kolayca takip etmek ve analiz etmek için kullanılmıştır.

5.3.2.4 İmza Karşılaştırma Algoritması

İmza karşılaştırma algoritması, imzalar arasındaki benzerlikleri değerlendirip, belirli bir eşik değere dayanarak gruplar oluşturur.

Başlangıçta her imza ayrı bir gruptur ve puanları 1'dir. İlk adımda, birinci imza diğer tüm imzalarla karşılaştırılır. Eğer karşılaştırma sonucu oluşan benzerlik skoru, eşik değerin üzerinde ise diğer imzaların puanları, birinci imzanın puanına aktarılır ve son değerleri 0 olarak güncellenir. Bu adım sonrasında birinci imza ve ona benzeyen imzalar aynı grupta tutulur. Bu grup içerisindeki imzalar bir daha karşılaştırılmaz.

Birinci imza ile gruplaştırılan imzalar dışında, skoru 0'dan büyük olan açıkta kalan ilk imza, diğer açıkta kalan imzalarla karşılaştırılır. Benzerlik skoru eşik değerin üzerinde olan imzalar gruplaştırılır ve karşılaştırma işlemine tekrar alınmaz. Bu süreç, tüm imzalar değerlendirilene kadar devam eder.



Şekil 5.32 İmza Karşılaştırma ve Gruplandırma Süreci

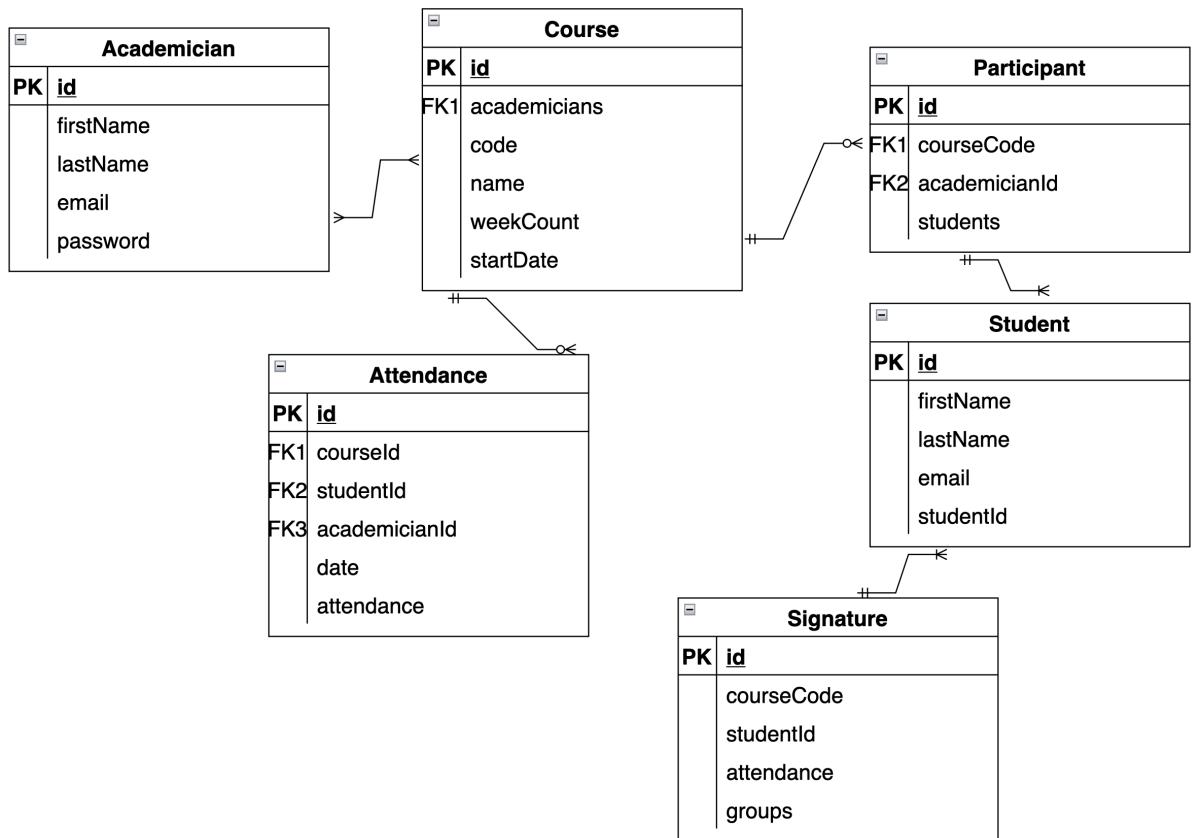
Durum	Karmaşıklık	Açıklama
En İyi Durum	$O(n)$	İlk imza tüm imzalarla benzerlik gösterir ve tek grup oluşur. Her imza bir kez karşılaştırılır.
En Kötü Durum	$O(n^2)$	Hiçbir imza diğer imzalarla benzerlik göstermez ve her imza ayrı grup oluşturur. Her imza tüm imzalarla karşılaştırılır.
Ortalama Durum	$O(n \log n)$	İmzalar benzerlik oranlarına göre gruptara ayrılır. Her grup oluşturma işlemi logaritmik olarak azalan bir imza sayısına dayanır.

Tablo 5.6 Algoritmanın Karmaşıklık Analizi

Bu algoritma, tüm imzaların birbirleriyle karşılaştırılması yerine, yalnızca gerekli karşılaştırmaları yaparak etkin çalışır. Bu durum, karmaşıklığı ve işlem süresini önemli ölçüde azaltmıştır. Ayrıca, referans bir imzanın veri tabanında bulundurulmasının kişisel haklar açısından uygun olmaması nedeniyle, en büyük grubu orijinal imza olarak kabul etme yaklaşımı benimsenmiştir. Bu sayede, her yeni imza seti bağımsız olarak değerlendirilmiş ve kişisel verilerin saklanması gerekliliği ortadan kalkmıştır.

5.4 Veri Tabanı Tasarımı

Veri tabanı tasarımlı kapsamında tasarlanan varlık-ilişki diyagramı şekil 5.33'te gösterilmiştir.



Şekil 5.33 Varlık-İlişki Diyagramı

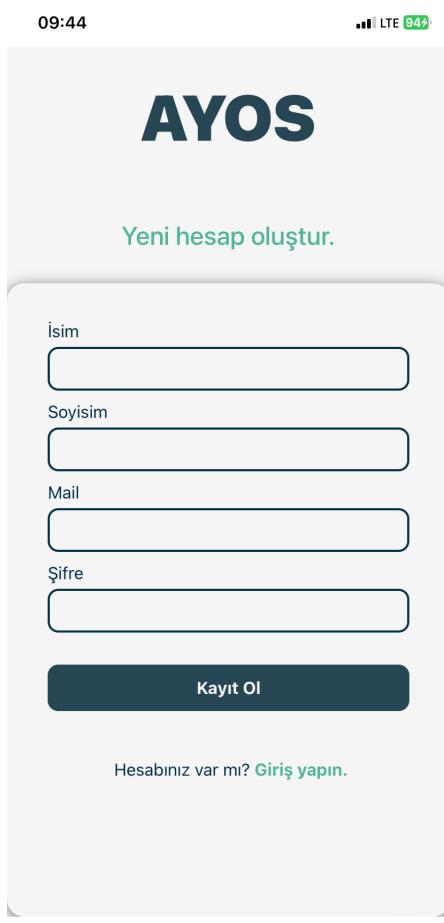
5.5 Girdi Çıktı Tasarımı

Projede kullanılan kullanıcı arayüzleri, tamamen mobil cihazlara uygun şekilde tasarlanmıştır. Bu arayüzlerin oluşturulması sürecinde herhangi bir UI/UX geliştirme aracı kullanılmamış ve arayüzler, kodlama aşamasında şekillendirilmiştir. Var olan arayüzler ise 'Uygulama' başlığı altında gösterildiğinden, raporda tekrara düşmemek adına bu başlık altında verilmemiştir.

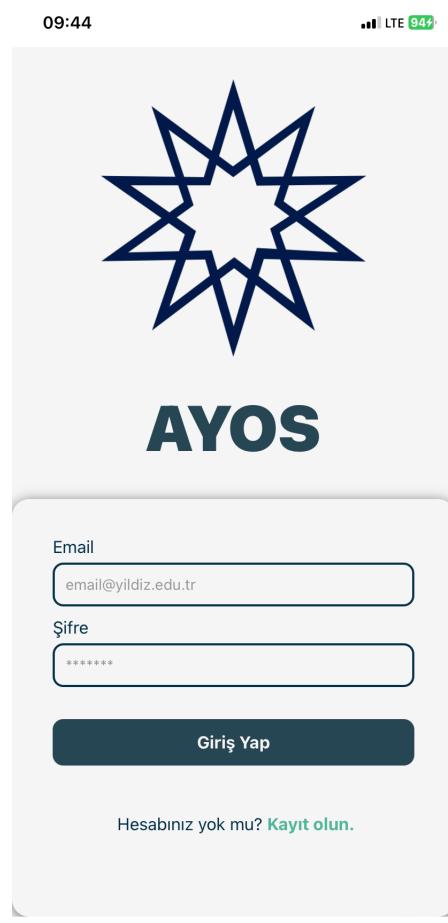
6 Uygulama

Bu bölümde, proje kapsamında geliştirilen mobil uygulamanın görselleri bulunmaktadır.

Uygulamaya giriş yapmak için öncelikle 6.1'deki ekranдан kayıt olunması gerekmektedir. Ardından email ve şifre ile giriş yapılabilir.

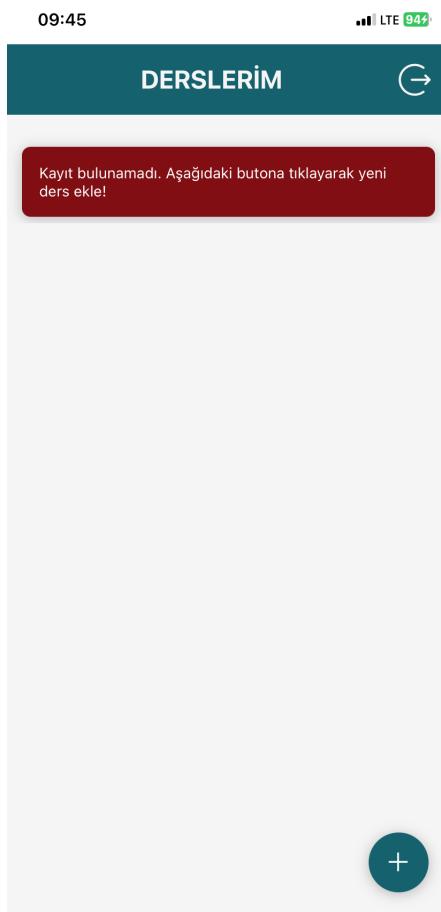


Şekil 6.1 Kayıt Olma

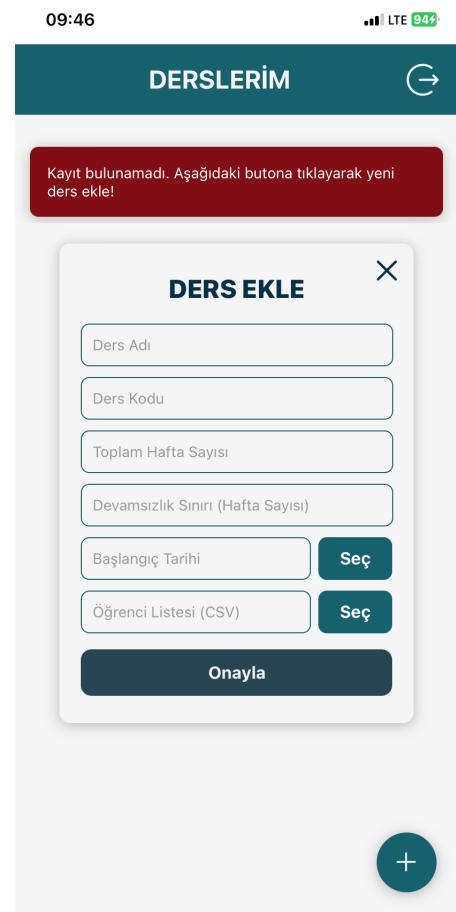


Şekil 6.2 Giriş Ekranı

Uygulamaya ilk defa giriş yaptıktan sonra herhangi bir ders kaydı bulunmadığı için sayfa 6.3'deki gibi görülmektedir. Sağ alttaki butona tıklayarak yeni bir ders eklenebilmektedir.

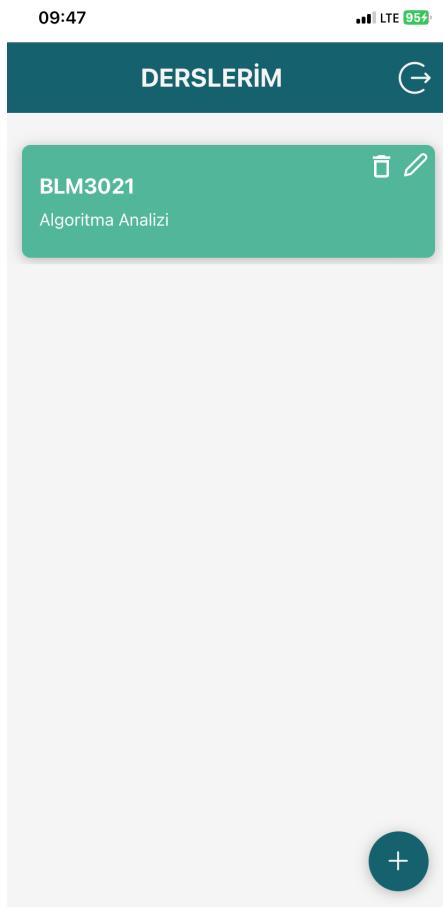


Şekil 6.3 Uygulamaya İlk Giriş

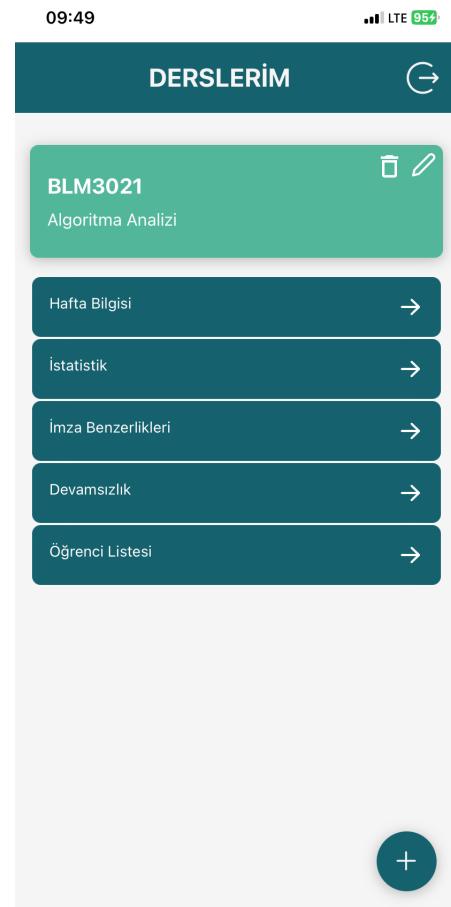


Şekil 6.4 Ders Ekleme

Yeni bir ders eklendikten sonra Şekil 6.6'daki gibi dersin üzerine tıklayarak seçenekler açılabilir mektedir.

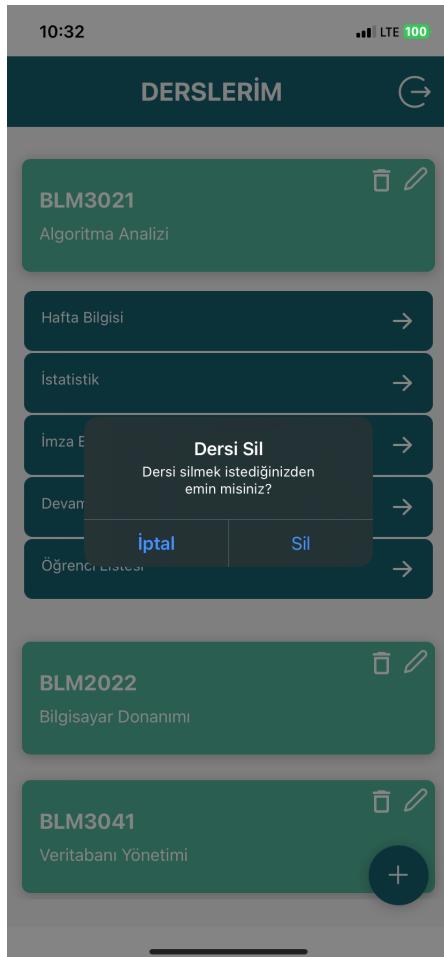


Şekil 6.5 Eklenen Ders



Şekil 6.6 Ders Seçenekleri

Ders kartının üzerindeki çöp kutusu simgesine tıklanarak ders silinebilir. Kalem simgesine tıklanarak ise devamsızlık sınırında değişiklik yapılabilir.



Şekil 6.7 Ders Silme



Şekil 6.8 Sınır Güncelleme

Bir ders kartının seçeneklerinde 'Hafta Bilgisi' başlıklı seçeneğe tıklandığında ilgili dersin haftaları Şekil 6.9'daki gibi listelenmektedir. Herhangi bir yoklama bilgisi sisteme girilmediği zaman haftaların katılım durumları '0/0' şeklinde görülmektedir. Bu sebeple 'Katılım Listesi' butonuna tıklandığında Şekil 6.10'da görüldüğü öğrenci kaydı bulunmamaktadır.

HAFTALAR

Hafta	Katılım	Aksiyon
7.02.2024	0/0	Katılım Listesi
14.02.2024	0/0	Katılım Listesi
21.02.2024	0/0	Katılım Listesi
28.02.2024	0/0	Katılım Listesi
6.03.2024	0/0	Katılım Listesi
13.03.2024	0/0	Katılım Listesi

KATILIM LİSTESİ

Kayıt bulunamadı.

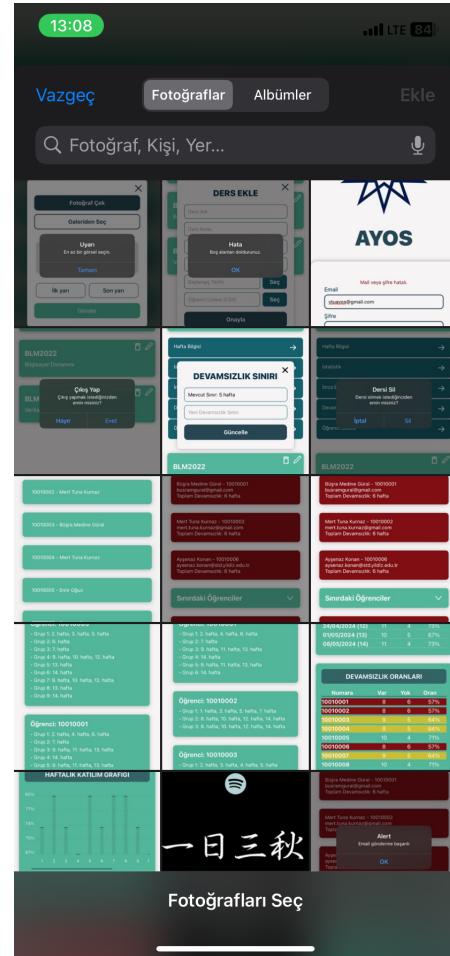
Şekil 6.9 Haftalar

Şekil 6.10 Katılım Lisesi

Sağ üstteki kamera simgesine tıklandığında yoklama kağıdının yüklenebilmesi amacıyla ekrana Şekil 6.11'deki gibi bir pop-up gelmektedir. Yoklama kağıdının fotoğrafının çekilmesi veya galeriden seçilerek yüklenmesi mümkündür.



Şekil 6.11 Fotoğraf Ekleme



Şekil 6.12 Galeriden Seçme

Herhangi bir fotoğraf veya dönem seçilmeden gönderim yapıldığında kullanıcıya aşağıdaki şekillerdeki gibi hata mesajı gösterilmektedir.



Şekil 6.13 GörSEL Uyarısı



Şekil 6.14 Dönem Uyarısı

Fotoğraflar eklendikten sonra Şekil 6.15'teki gibi dönem seçimi yapılmalıdır. 'İlk yarı' dönemin ilk 7 haftası için yükleme yapıldığında, 'son yarı' ise dönemin kalan haftaları için yükleme yapıldığında seçilmelidir. Yükleme yapıldıktan sonra katılım sayısında değişiklikler görülmektedir.



Şekil 6.15 Yoklama Gönderme

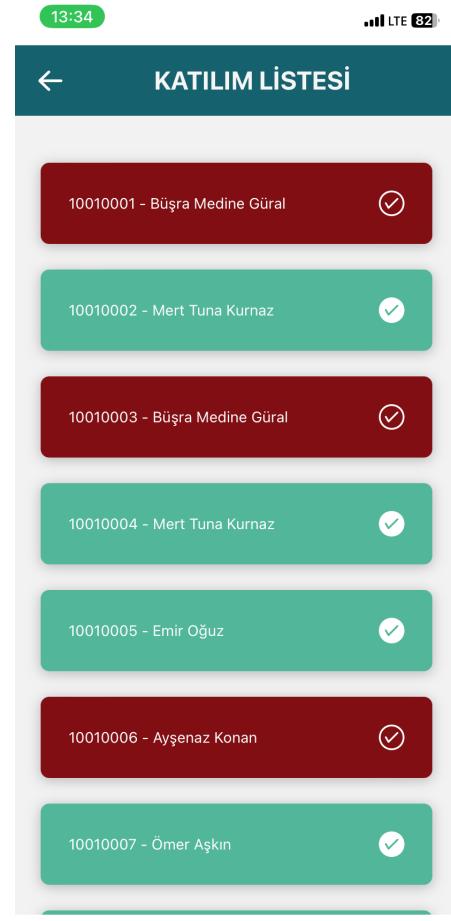


Şekil 6.16 Güncellenmiş Haftalar

Yoklama kağıdındaki bilgiler sisteme işlendikten sonra bir bildirim gelmektedir. Bu bildirim, kullanıcıyı ders seçeneklerinde de görülebilen 'Devamsızlıklar' adlı sayfaya yönlendirmektedir. Bu sayfada devamsızlık sınırındaki ve sınırı geçmiş öğrenciler listelenmektedir. Hafta kartlarındaki butonlara tıklandığında ise güncel katılım verileri Şekil 6.18'deki gibi görülebilmektedir.

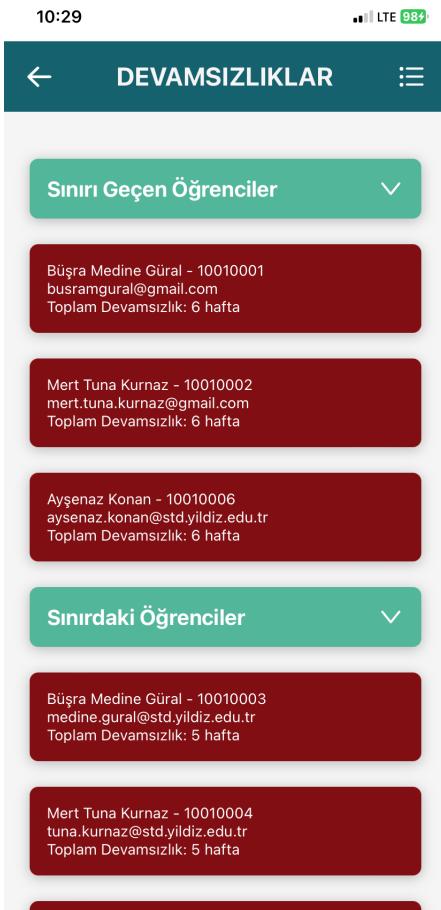


Şekil 6.17 Bildirim



Şekil 6.18 Haftalık Katılım

'Devamsızlık' sayfasındaki öğrencilerin listesi bir CSV dosyası ile dışarıya aktarılabilmektedir. Aynı zamanda bu öğrencilere Şekil 6.20'de göründüğü gibi devamsızlık durumları ile ilgili bir mail gönderilebilmektedir.

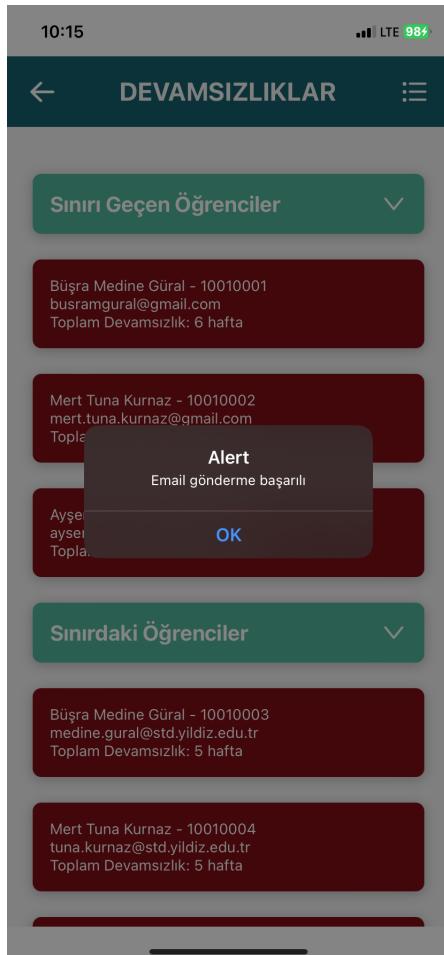


Şekil 6.19 Devamsızlıklar

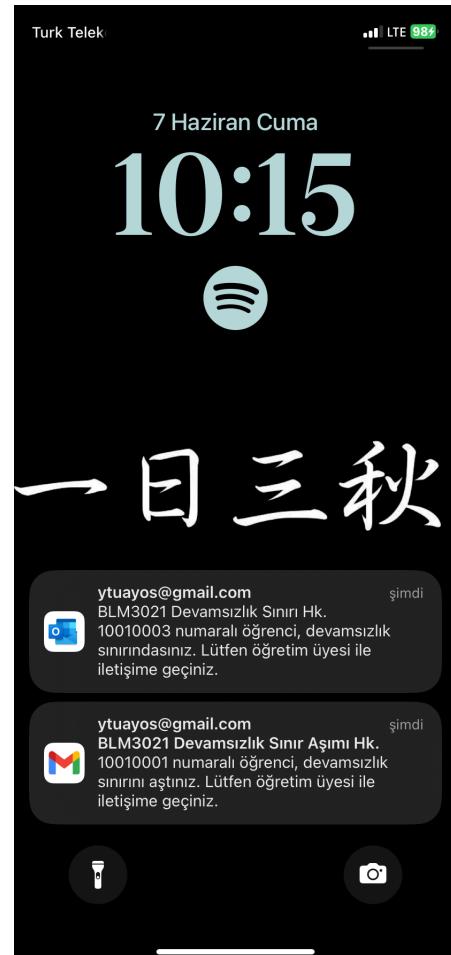


Şekil 6.20 Seçenekler

Mail, Şekil 6.22'de görüldüğü gibi iki farklı kategorideki öğrencilere göre özelleşmiş bir şekilde gönderilmektedir.



Şekil 6.21 Mail Gönderme



Şekil 6.22 Gelen Mail

'Dersler' sayfasındaki seçeneklerden gidilebilecek bir 'İstatistik' sayfası aşağıdaki gibidir. Bu sayfada, dersin her bir haftasındaki katılım oranları, grafikleri ve öğrencilerin katılım sayıları detaylı bir şekilde bulunmaktadır. 'Devamsızlık Oranları' başlığının altındaki tabloda kırmızı ile sınırı geçenler ve sarı ile sınırda bulunan öğrenciler gösterilmektedir.



Şekil 6.23 Haftalık İstatistikler

Numara	Var	Yok	Oran
10010001	8	6	57%
10010002	8	6	57%
10010003	9	5	64%
10010004	9	5	64%
10010005	10	4	71%
10010006	8	6	57%
10010007	9	5	64%
10010008	10	4	71%
10010009	10	4	71%
10010010	10	4	71%
10010011	12	2	86%
10010012	14	0	100%
10010013	14	0	100%
10010014	12	2	86%
10010015	12	2	86%

Şekil 6.24 Öğrenci Devamsızlıklarını

'Dersler' sayfasındaki seçeneklerden gidilebilecek 'İmza Benzerlikleri' sayfası aşağıdaki gibidir. Bu sayfada öğrenci numarasına ve grup sayısına göre sıralama yapılmaktadır. Bir kişinin atmış olduğu imzalar birbiriyle karşılaştırıldığında, benzerliklerine göre çeşitli gruplar oluşabilmektedir. Örneğin Şekil 6.25'deki görselde, sisteme ilk yüklenilen 7 haftalık yoklama kağıdında 10010001 numaralı kişinin 2, 3 ve 4. haftalardaki imzalarının birbirine benzendiği ancak 7. haftadaki imzasının diğerlerinden farklı olduğu tespit edilmiştir. Tüm dönem için yoklama bilgisi sisteme işlendiğinde grup sayısında artış meydana gelmektedir. Çünkü iki ayrı yüklenilen yoklama kağıdındaki imzalar birbiriyle karşılaştırılamamaktadır. Haliyle 14 haftalık veri için imza benzerliğinin yüksek olduğu durumlarda (örneğin 1001002 numaralı öğrenci) iki grup görülmesi en ideal durumdur.

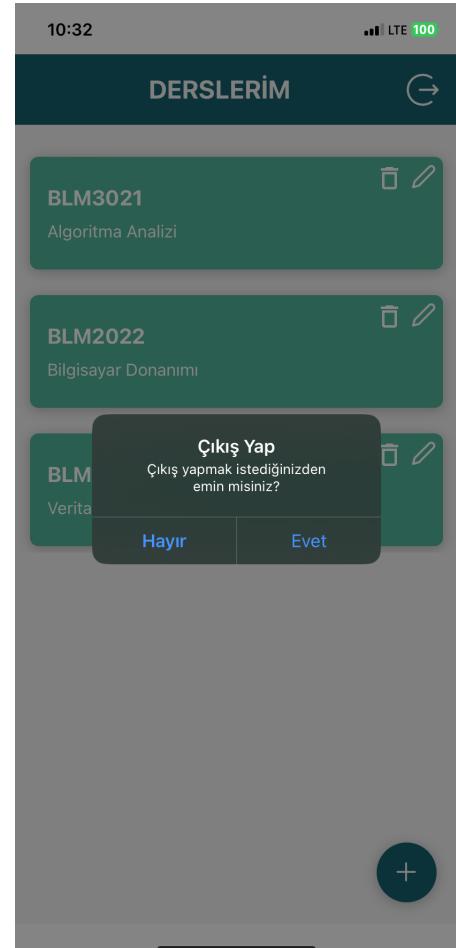
Şekil 6.25 İmza Benzerliği-1

Şekil 6.26 İmza Benzerliği-2

'Dersler' sayfasındaki seçeneklerden gidilebilecek 'Öğrenci Listesi' sayfası 6.27'deki gibidir. Bu sayfa, ilgili derse kayıtlı olan öğrencilerin görüntülenebilmesi içindir. Şekil 6.28'deki görselde ise, derslerin listelendiği sayfada sağ üstte bulunan simgeye tıklanarak uygulamadan çıkış yapılmaktadır.



Şekil 6.27 Öğrenci Listesi



Şekil 6.28 Çıkış

7

Deneysel Sonuçlar

Bu bölümde, önceki aşamalarda tasarlanan veri seti ve metrikler ile birlikte modelin eğitimi ve performansının değerlendirilmesi işlemleri gerçekleştirılmıştır. Model eğitimi adımda, belirlenen veri seti üzerinde modelin öğrenmesi sağlanmış ve eğitim sonuçları incelenmiştir.

7.1 Model Eğitimi

Model eğitimi, farklı optimizasyon algoritmaları ve kayıp fonksiyonları kullanılarak gerçekleştirılmıştır. RMSProp ve Adam optimizasyon algoritmaları ile Binary Crossentropy ve Contrastive Loss fonksiyonları kullanılarak çeşitli denemeler yapılmıştır. Eğitim, doğrulama ve test kümeleri üzerinde performans değerlendirilmiştir.

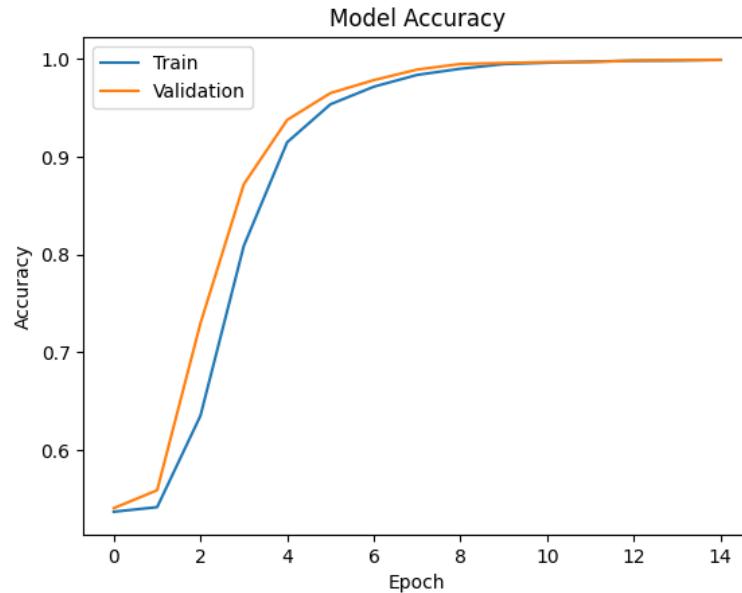
7.2 Model Performansı

Bu bölümde, geliştirilen modelin performansı üzerine yapılan deneysel sonuçlar ve analizler sunulmuştur. Farklı deneme ve parametreler kullanılarak elde edilen sonuçlar Tablo 7.1'de özetlenmiştir.

Tablo 7.1 Model Karşılaştırma Tablosu

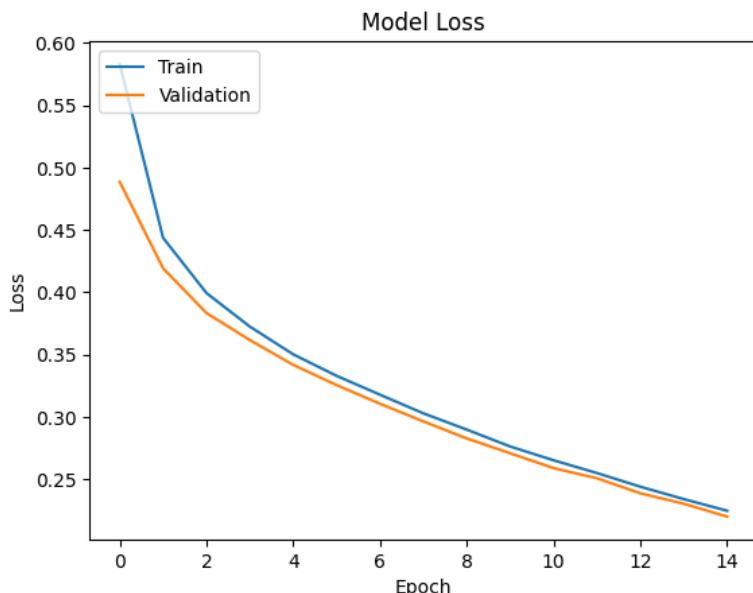
Deneme	Optimizer	Kayıp Fonksiyonu	Batch Boyutu	Epoch	Eğitim Başarımı	Doğrulama Başarımı	Test Başarımı
1	RMSProp	Binary Crossentropy	128	10	0.702	0.677	0.673
2	RMSProp	Contrastive Loss	128	10	0.892	0.885	0.887
3	Adam	Binary Crossentropy	128	10	0.843	0.838	0.841
4	Adam	Contrastive Loss	128	10	0.930	0.924	0.918
5	Adam	Contrastive Loss	256	15	0.996	0.995	0.998

Tablo 7.1'de gösterilen sonuçlar, farklı optimizasyon algoritmaları ve kayıp fonksiyonları ile elde edilmiştir. En iyi sonuçlar, beşinci denemedede Adam optimizasyon algoritması ve Contrastive Loss fonksiyonu kullanılarak elde edilmiştir. Bu model; eğitim, doğrulama ve test setlerinde yüksek doğruluk oranlarına ulaşmıştır.



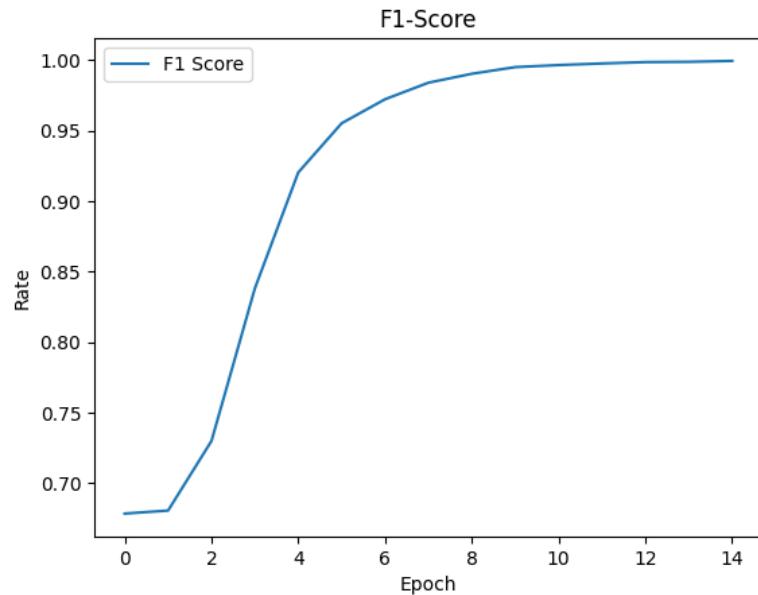
Şekil 7.1 Model Doğruluk Grafiği

Şekil 7.1'de modelin eğitim ve doğrulama setlerindeki doğruluk oranları gösterilmiştir. Modelin doğruluk oranı, epoch sayısı arttıkça düzenli bir şekilde artmakta ve yüksek bir doğruluk seviyesine ulaşmıştır.



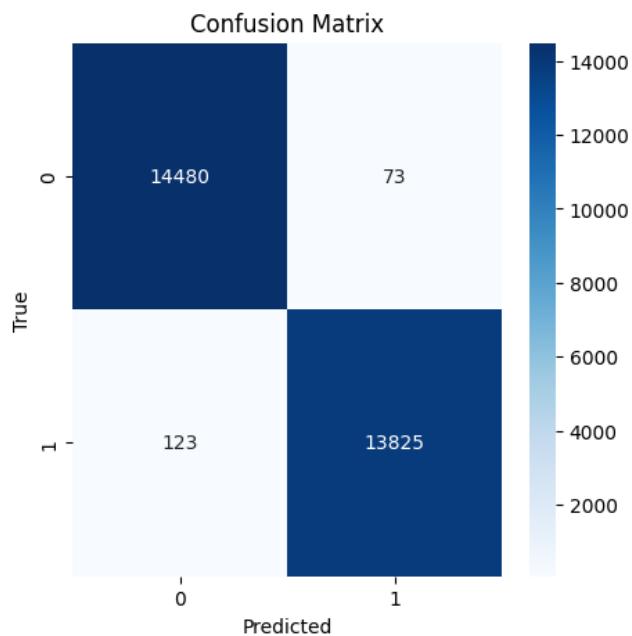
Şekil 7.2 Model Kayıp Grafiği

Şekil 7.2'de modelin eğitim ve doğrulama setlerindeki kayıp oranları gösterilmiştir. Kayıp oranı, epoch sayısı arttıkça azalmaktadır ve modelin öğrenme sürecinde başarılı bir performans sergilediğini göstermiştir.

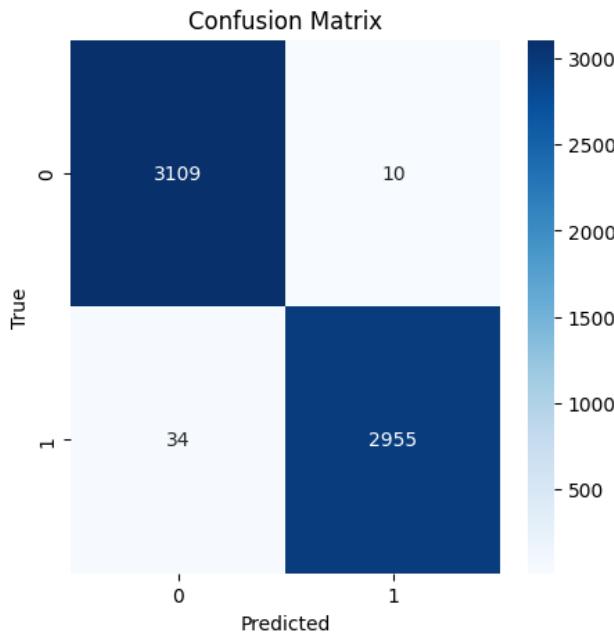


Şekil 7.3 Model F1-Skoru Grafiği

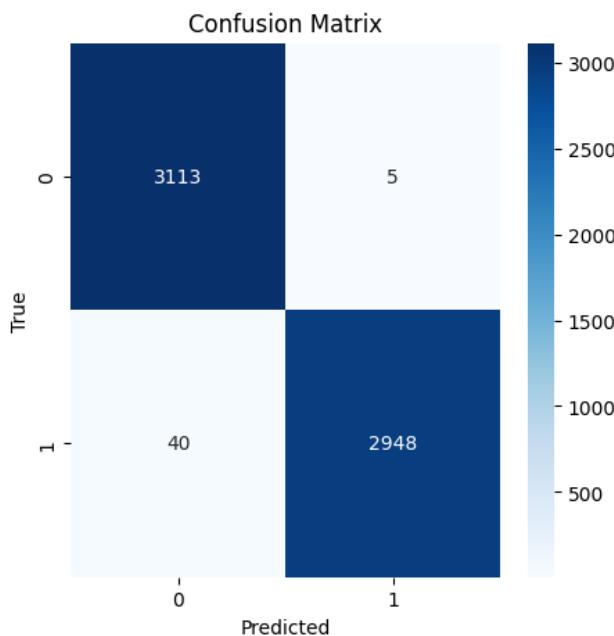
Şekil 7.3'de modelin F1-skoru gösterilmiştir. F1-skoru, epoch sayısı arttıkça artmakta ve yüksek bir seviyeye ulaşmaktadır, bu da modelin sınıflandırma performansının iyİ olduğunu göstermiştir.



Şekil 7.4 Eğitim (Training) Seti Karışıklık Matrisi



Şekil 7.5 Doğrulama (Validation) Seti Karışıklık Matrisi



Şekil 7.6 Test Seti Karışıklık Matrisi

Şekil 7.4, 7.5 ve 7.6'da sırasıyla eğitim, doğrulama ve test setleri için oluşturulan karışıklık matrisleri gösterilmektedir. Bu matrisler, modelin doğru ve yanlış sınıflandırma performansını göstermekte olup, modelin genel başarımını değerlendirmede önemli bir rol oynamıştır.

Deneysel sonuçlar, modelin eğitim, doğrulama ve test setlerinde yüksek performans gösterdiğini ve sınıflandırma görevlerinde başarılı olduğunu göstermiştir.

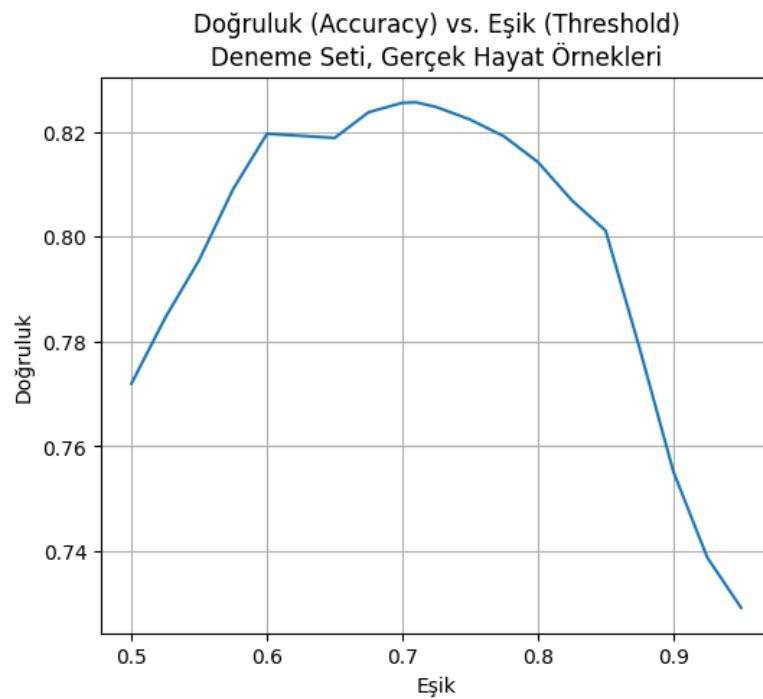
7.3 Model Tahmin Sonucu Oluşan Çıktıların Değerlendirilmesi

Bu kısımda; eğitim, doğrulama ve test kümelerinin herhangi birinden rastgele seçilen iki imzanın karşılaşılması ile oluşan sonuçlar çıktı olarak elde edilmiştir ve bu iki imza görüntülenmiştir. Bu tahminler sonucu oluşan öklid uzaklıguna göre bir benzerlik skoru hesaplanmıştır. Öklid uzaklığı için eşik değeri 0.65 olarak belirlenmiştir, yani bu benzerlik skoru için eşik değerinin 0.7 olduğu anlamına gelir. Bu durum aşağıdaki formülden anlaşılmaktadır.

$$\text{Benzerlik Skoru} = \max\left(0, \min\left(1, \frac{1 - \text{Öklid Uzaklılığı}}{0.5}\right)\right)$$

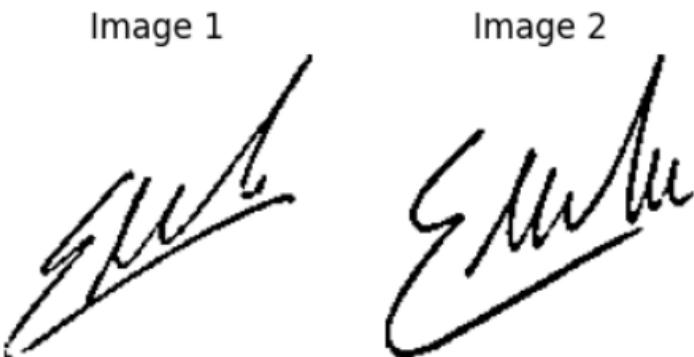
Model, veri seti dışında gerçek hayattan örnek veriler ile test edilmiştir. Modelin, veri setinden tamamen farklı örnekler üzerindeki başarı oranı, %82.63 olmuştur.

Modelin bu örneklerdeki doğruluk (accuracy) sonuçları, farklı eşik değerleri kullanılarak analiz edilmiştir. Şekil 7.7'de gösterilen grafikte, doğruluk değerlerinin eşik değerine göre değişimi gözlemlenmiştir. Eşik değerinin 0.7 civarında seçilmesinin sebebi, bu değerde modelin en yüksek doğruluk değerine ulaşmasıdır. Bu analiz, modelin optimal eşik değerinin belirlenmesinde önemli rol oynamıştır.



Şekil 7.7 Doğruluk (Accuracy) vs. Eşik (Threshold)

- Tahmin 1:

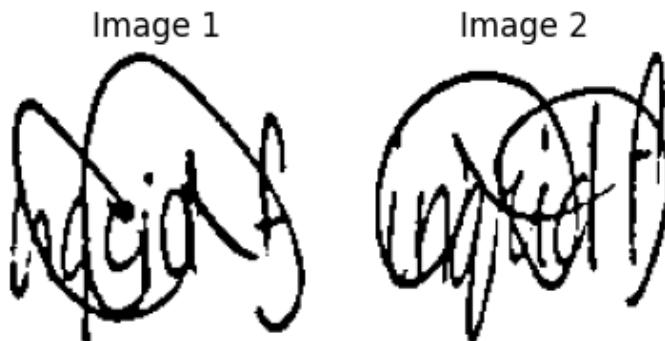


Şekil 7.8 Karşılaştırma 1 - İmza Görüntüleri

- Gerçek Cevap (Etiket) : 1 (forged / different)
- Öklid Uzaklığı : 0.931
- Benzerlik Skoru : 0.138
-> DOĞRU TAHMİN!

Şekil 7.9 Karşılaştırma 1 - Çıktı

- Tahmin 2:

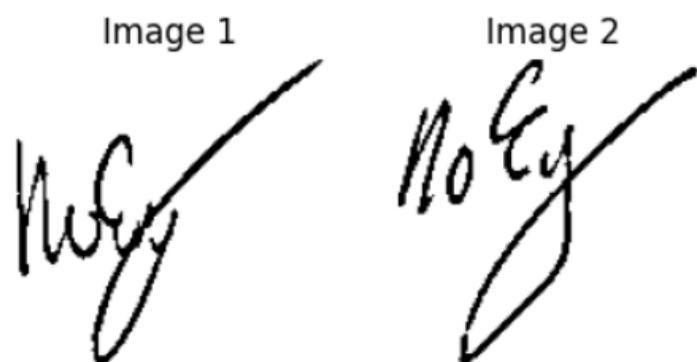


Şekil 7.10 Karşılaştırma 2 - İmza Görüntüleri

- Gerçek Cevap (Etiket) : 1 (forged / different)
- Öklid Uzaklığı : 0.569
- Benzerlik Skoru : 0.862
-> YANLIŞ TAHMİN!

Şekil 7.11 Karşılaştırma 2 - Çıktı

- Tahmin 3:



Şekil 7.12 Karşılaştırma 3 - İmza Görüntüleri

```
- Gerçek Cevap (Etiket) : 1 (forged / different)
- Öklid Uzaklığı       : 0.887
- Benzerlik Skoru      : 0.227
-> DOĞRU TAHMİN!
```

Şekil 7.13 Karşılaştırma 3 - Çıktı

7.4 Örnek Şablon Çıktısı

Şekil 7.14'te örnek bir yoklama şablonu üzerindeki imza karşılaştırma algoritmasının sonuçları verilmiştir. Her bir öğrenci için model tahminleri sonucunda tespit edilen orijinal imzalar yeşil renkli sınırlayıcı kutular ile gösterilmiştir, sahte imzalar kırmızı renkli kutular ile gösterilmiştir. Buradaki bir öğrenciye ait imzaların hepsi o öğrenciye ait orijinal imzalarıdır, yani kırmızı renk ile gösterilmiş bölgeler aslında yanlış tahminlerdir. Bireyler, kendi imzalarını atarken bazı zamanlarda kendi imza örüntüsünden farklı şekilde imza atabilmektedir. Örnek olarak, 10010007 numaralı öğrencinin beşinci haftadaki imzası diğer imzalarına göre farklıdır ve model bunu da sahte imza olarak tespit etmiştir. Bu tarz durumlar göz önünde bulundurulmalıdır. Bunun dışında, model genel olarak orijinal imzaların tespitinde başarılı olmuştur. Bu örnek, bu projenin gerçek zamanlı uygulamalarda kullanılabilirliğini göstermiştir.

Öğrenci No.	Ad Soyad	1. Hafta	2. Hafta	3. Hafta	4. Hafta	5. Hafta	6. Hafta	7. Hafta
10010001	AD-SOYAD-1	/	Ez	/	Ez	/	Ez	Ez
10010002	AD-SOYAD-2	Ez	/	Ez	/	Ez	/	Ez
10010003	AD-SOYAD-3	/	Mes	Mes	Mes	Mes	Mes	/
10010004	AD-SOYAD-4	Atan	Atan	/	/	Atan	Atan	Atan
10010005	AD-SOYAD-5	Hpo	/	Hpo	Hpo	Hpo	/	Hpo
10010006	AD-SOYAD-6	/	Kes	Kes	/	Kes	Kes	/
10010007	AD-SOYAD-7	F	F	F	F	F	F	F
10010008	AD-SOYAD-8	Mes	Mes	Mes	Mes	/	/	Mes
10010009	AD-SOYAD-9	/	Ded	Ded	/	Ded	Ded	Ded
10010010	AD-SOYAD-10	Qaz	Qaz	/	Qaz	Qaz	Qaz	/
10010011	AD-SOYAD-11	Zant	Zant	Zant	Zant	/	Zant	Zant
10010012	AD-SOYAD-12	Zes	Zes	Zes	Zes	Zes	Zes	Zes
10010013	AD-SOYAD-13	X	X	X	X	X	X	X
10010014	AD-SOYAD-14	Waaa	Waaa	/	Waaa	Waaa	Waaa	Waaa
10010015	AD-SOYAD-15	AJ	/	AJ	AJ	AJ	AJ	AJ

Şekil 7.14 Yoklama Şablonu - Örnek Çıktı

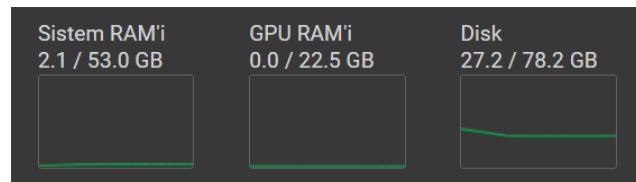
8

Performans Analizi

Bu bölümde, veri işleme ve model eğitimi süreçlerinin Google Colab üzerinde sistem kaynakları üzerindeki etkileri incelenmiştir. Aşağıdaki görsellerde, veri işleme ve model eğitimi sırasında kullanılan sistem RAM'i, GPU RAM'i ve disk kullanım gösterilmiştir. Görseller, sistem kaynaklarının kullanımını ve işleme sürelerini teknik ve detaylı bir şekilde sunmaktadır.

8.1 Veri İşleme Süresi ve Sistem Kullanımı

Veri işleme süresi ve sistem kaynaklarının kullanımı, eğitim, doğrulama ve test veri kümeleri için ayrı ayrı değerlendirilmiştir.



Şekil 8.1 Veri İşleme Sürecinde Sistem Kullanımı

Şekil 8.1'de gösterilen sistem kaynaklarının kullanım grafiği, veri işleme sırasında sistem RAM'inin, GPU RAM'inin ve disk kullanımının başlangıç ve işleme sonrası durumlarını göstermektedir. Veri işleme süresi, her bir veri kümesi için ayrı ayrı hesaplanmış ve hem Şekil 8.2'de hem de Tablo 8.1'da sunulmuştur.

```

# process the data
X_train, y_train = process_data(train_data_csv, data_dir, img_size)
X_val, y_val = process_data(val_data_csv, data_dir, img_size)
X_test, y_test = process_data(test_data_csv, data_dir, img_size)

Data processing time: 33 minutes and 16.28 seconds
Data processing time: 1 minutes and 19.32 seconds
Data processing time: 1 minutes and 19.33 seconds

```

Şekil 8.2 Veri İşleme Süresi Kod Çıktısı

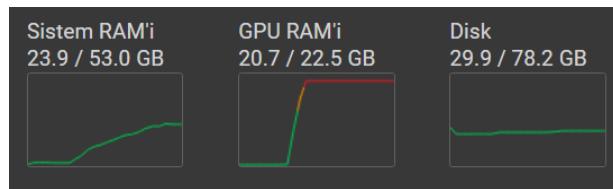
Tablo 8.1 Veri İşleme Süresi

Veri Kümesi	İşleme Süresi
Eğitim	33 dakika 16.28 saniye
Doğrulama	1 dakika 19.32 saniye
Test	1 dakika 19.33 saniye

Tablo 8.1, veri işleme süresinin eğitim veri kümesi için daha uzun sürdüğünü, doğrulama ve test veri kümeleri için ise benzer sürelerde tamamlandığını göstermektedir.

8.2 Model Eğitimi Sırasında Sistem Kullanımı

Model eğitimi sırasında kullanılan sistem kaynakları, eğitim süresince önemli ölçüde artış göstermiştir. Aşağıdaki görselde, eğitim süreci boyunca sistem RAM'i, GPU RAM'i ve disk kullanımındaki değişimler gözlemlenmiştir.



Şekil 8.3 Model Eğitimi Sırasında Sistem Kullanımı

Şekil 8.3'te gösterilen grafikler, model eğitimi sırasında sistem RAM'ının önemli ölçüde kullanıldığını ve GPU RAM'ının tam kapasiteye yakın çalıştığını göstermektedir. Disk kullanımı ise eğitim süresince sabit kalmıştır.

Bu analizler, modelin eğitimi ve veri işleme süreçlerinin sistem kaynakları üzerindeki etkilerini değerlendirmede önemli bir rol oynamaktadır.

8.3 Yoklama Şablonu İşleme Süresi

Yoklama şablonu görüntüsünün yüklenmesinden, sonucun elde edilmesine kadar geçen tüm süre ve bu süreç içerisindeki modelin tüm imzaları karşılaştırdığı süre aşağıda detaylandırılmıştır.

Tablo 8.2 Yoklama Şablonu İşleme Süresi

İşlem Adımı	Süre
Modelin Görüntüdeki Tüm İmzaları Karşılaştırma Süresi	4.61 saniye
Toplam İşlem Süresi (Görüntü İşleme ve Karşılaştırma)	11.37 saniye

Tablo 8.2'de gösterilen veriler, yoklama şablonu görüntüsünün yüklenmesinden sonra modelin imzaları karşılaştırma süresini ve toplam işlem süresini özetlemektedir. Model karşılaştırma süresi 4.61 saniye iken, tüm işlem süresi 11.37 saniyedir. Bu, modelin imzaları hızlı şekilde karşılaştırdığını ve verimli çalıştığını göstermiştir..

Hesaplanan süreler, modelin gerçek zamanlı uygulamalarda kullanılabilir olduğunu göstermiştir.

9 Sonuç

Bu çalışma, yoklama süreçlerinde akademisyenlere yardımcı olmayı hedefleyerek yoklama süreçlerinin dijitalleşmesini amaçlayan bir uygulamanın geliştirilmesini amaçlamaktadır. Kullanılan görüntü işleme teknikleri ile ooklama kağıdı fotoğrafları üzerinden öğrenci numaraları algılanarak, her bir öğrencinin imza benzerlikleri kontrol edilmektedir. İşlemler sonucunda oluşan çıktılar ise akademisyen ile bir mobil arayüz aracılığıyla şeffaf bir şekilde paylaşılmaktadır.

Gerçekleştirilen çalışmalar sırasında modelin performansı; farklı optimizasyon algoritmaları, kayıp fonksiyonları ve batch boyutlarının kullanılmasıyla incelenmiştir. Modelin performasının maksimize edilmesi, Adam optimizatörü ve karşılık kaybı fonksiyonunun kullanıldığı senaryoda gerçekleştirilmiştir.

Bu projede imza görüntülerini dijital ortamda saklanmamaktadır. Bu sebeple hassas verileri koruma kapsamında güvenlik açısından problemlerle karşılaşılmamıştır. Ancak imzaların saklanmamasının bir dezavantajı bulunmaktadır: Bir yoklama kağıdında bulunan hafta sayısı kadar imza kendi içinde karşılaştırılmaktadır. Bu nedenle iki ayrı yoklama kağıdı fotoğraflarındaki var olan tüm haftaların birbiriyle benzerliği tespit edilememektedir. Bunun çözümü için bir kağıttaki hafta sayısı maksimize edilmiştir. Böylelikle bir dönem ikiye ayrılmış ve bir ders için iki ayrı yedi haftalık yoklama kağıdı şablonu hazırlanmıştır. Fazla sayıda imza kendi arasında karşılaştırıldığı için benzerlik sonucu daha başarılı bir şekilde optimize edilebilmiştir.

Projede belirlenen başlangıç hedeflerine çok büyük oranda ulaşılmıştır. Kullanıcı dostu bir arayüz ile akademisyenlerin yoklama süreçlerini yönetebilmesi mümkün kılınmıştır. Ancak dikkat edilmesi gereken bazı noktalar bulunmaktadır. Yoklama kağıdı fotoğraflarının düzgün bir ışık ve açı ile çekilerek sisteme yüklenmesi gerekmektedir. Düşük ışık ya da yanlış açı ile çekilmiş fotoğraflardaki imzaların algılanması zor olacağından sistemin başarısı düşebilmektedir. Bu nedenle, bu alanda çalışmalar yapmak isteyen araştırmacılar, modelin eğitim sürecinde bu parametreleri de işleme katarak modeli eğitmeli ve performansını artırmalıdır.

Referanslar

- [1] M. S. Akbar, P. Sarker, A. T. Mansoor, A. M. Al Ashray, and J. Uddin, “Face recognition and rfid verified attendance system,” in *2018 International Conference on Computing, Electronics Communications Engineering (iCCECE)*, 2018, pp. 168–172. doi: 10.1109/iCCECOME.2018.8658705.
- [2] K. O. Okopujie, E. Noma-Osaghae, O. J. Okesola, S. N. John, and O. Robert, “Design and implementation of a student attendance system using iris biometric recognition,” in *2017 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2017, pp. 563–567. doi: 10.1109/CSCI.2017.96.
- [3] S. Software. “Education success platform.” (2023), [Online]. Available: <https://www.seatssoftware.com> (visited on 04/20/2024).
- [4] H. Abu Bakar, “Signature recognition system for student attendance system in utp,” 2004.
- [5] M. J. B. Harish Srinivasan Sargur N Srihari. “Cedar signature dataset.” (2001), [Online]. Available: <https://paperswithcode.com/dataset/cedar-signature> (visited on 02/15/2024).
- [6] R. Reni. “Signature verification dataset.” (2023), [Online]. Available: <https://www.kaggle.com/datasets/robinreni/signature-verification-dataset> (visited on 03/25/2024).
- [7] Mobilhanem. “Opencv’de video boyutlandırma ve gri tonlama.” (2020), [Online]. Available: <https://www.mobilhanem.com/opencvde-video-boyutlandirma-ve-gri-tonlama/> (visited on 04/26/2024).
- [8] C. A. Tarık Talan, “Bilgisayar bilimlerinde teorik ve uygulamalı araştırmalar,” in Efe Akademik Yayıncılık, 2021, ch. Görüntü İşleme - Section 8 (?) [Online]. Available: <https://books.google.com.tr/books?id=dIJjEAAAQBAJ&1pg=PA1&hl=tr&pg=PA1#v=onepage&q&f=true> (visited on 04/02/2024).
- [9] M. Varol Arisoy, “Signature verification using siamese neural network one-shot learning,” *International Journal of Engineering and Innovative Research*, vol. 3, no. 3, pp. 248–260, 2021. doi: 10.47933/ijeir.972796.
- [10] GeeksforGeeks. “Öklid mesafesi.” (2024), [Online]. Available: <https://www.geeksforgeeks.org/euclidean-distance/> (visited on 04/23/2024).
- [11] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, “Signet: Convolutional siamese network for writer independent offline signature verification,” *CoRR*, vol. abs/1707.02131, 2017. arXiv: 1707.02131. [Online]. Available: <http://arxiv.org/abs/1707.02131>.

- [12] V. Labs. “F1 score guide: Everything you need to know.” (2023), [Online]. Available: <https://www.v7labs.com/blog/f1-score-guide> (visited on 04/20/2024).
- [13] D. Team. “What is a confusion matrix in machine learning?” (2023), [Online]. Available: <https://www.datacamp.com/tutorial/what-is-a-confusion-matrix-in-machine-learning> (visited on 04/22/2024).
- [14] H. Z. H. Abadi, S. Samavi, and N. Karimi, “Image noise reduction by low complexity hardware median filter,” in *2013 21st Iranian Conference on Electrical Engineering (ICEE)*, 2013, pp. 1–5. DOI: 10.1109/IranianCEE.2013.6599571.