

**T.C.
SAKARYA UYGULAMALI BİLİMLER ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

DERİN ÖĞRENME YÖNTEMİ KULLANILARAK EL YAZISINI DİJİTALE ÇEVİREN WEB UYGULAMASI

Bilgisayar Mühendisliği Bitirme Raporu

**B200109011 Ayşegül TOPTAŞ
B200109028 Havvanur BOZKURT**

Danışmanı : Dr. Öğr. Üyesi Muhammed Ali Nur ÖZ

Haziran 2024

T.C.
SAKARYA UYGULAMALI BİLİMLER ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

**DERİN ÖĞRENME YÖNTEMİ KULLANILARAK EL
YAZISINI DİJİTALE ÇEVİREN WEB UYGULAMASI**

Bilgisayar Mühendisliği Bitirme Raporu

B200109011 Ayşegül TOPTAŞ
B200109028 Havvanur BOZKURT

Danışmanı : Dr. Öğr. Üyesi Muhammed Ali Nur ÖZ

**Bu rapor 00.00.2024 tarihinde aşağıdaki jüri tarafından oybirliği / oyçokluğu ile
kabul edilmiştir.**

Danışman

Üye

Üye

BEYAN

Rapor içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, raporda yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir çalışmada kullanılmadığını beyan ederim.

Ayşegül TOPTAŞ
Havvanur BOZKURT

07.06

TEŞEKKÜR

Bitirme projesi süresince değerli bilgi ve deneyimlerinden yararlandığımız, her konuda bilgi ve desteğini almaktan çekinmediğimiz, araştırmanın planlanmasından yazılmasına kadar tüm aşamalarında yardımlarını esirgemeyen, teşvik eden, aynı titizlikte bizi yönlendiren değerli danışman hocamız Dr. Öğr. Üyesi Muhammed Ali Nur ÖZ'e teşekkürlerimizi sunarız.

Ayrıca her türlü koşulda öncelikleri biz öğrencilerin eğitim kalitesi yüksek, özverili mühendisler olarak mezun olması olan, üniversitemizin değerli hocalarına bizlere katkılarından dolayı teşekkürlerimizi sunarız.

İÇİNDEKİLER

| | |
|---|------|
| TEŞEKKÜR..... | i |
| İÇİNDEKİLER..... | ii |
| SİMGELER VE KISALTMALAR LİSTESİ..... | iv |
| ŞEKİLLER LİSTESİ..... | v |
| TABLolar LİSTESİ..... | vii |
| ÖZET..... | viii |
| ABSTRACT..... | ix |
| BÖLÜM 1. | |
| GİRİŞ..... | 10 |
| BÖLÜM 2. | |
| ÖNCEKİ ÇALIŞMALARIN ARAŞTIRILMASI..... | 13 |
| 2.1. El Yazısı Karakter Tanıma ve Resim Sınıflandırmada Derin Öğrenme Yaklaşımları..... | 13 |
| 2.2. Handwritten Text Recognition and Conversion Using Convolutional Neural Network (CNN) Based Deep Learning Model | 13 |
| 2.3. Neural Network Based Handwritten Character Recognition System Without Feature Extraction | 14 |
| 2.4. Makine Öğrenmesi Yöntemleriyle El Yazısı Tanıma..... | 14 |
| 2.5. El Yazısı Rakam Tanıma İçin Destek Vektör Makinelerinin Ve Yapay Sinir Ağlarının Karşılaştırması | 15 |
| BÖLÜM 3. | |
| YÖNTEM VE TASARIM..... | 16 |
| 3.1. Materyal..... | 16 |
| 3.1.1. MNIST (Mixed National Institute of Standards and Technology) | 16 |
| 3.1.2. EMNIST (Extended MNIST) | 17 |
| 3.1.3. Kütüphaneler ve Araçlar | 17 |

| | |
|---|-----------|
| 3.2. Yöntem..... | 18 |
| 3.2.1. Verinin Eldesi | 20 |
| 3.2.2. Satır Segmentasyonu | 21 |
| 3.2.3. Karakter Segmentasyonu | 21 |
| 3.2.4. Harf ve Rakam Modellerinin Oluşturulması | 23 |
| 3.2.4.1. CNN (Convolutional Neural Network) | 23 |
| 3.2.4.2. MNIST Veri Seti İçin Model Eğitimi | 26 |
| 3.2.4.3. EMNIST Veri Seti İçin Model Eğitimi | 26 |
| 3.2.5. Sınıflandırma..... | 28 |
| 3.2.6. Birleştirme..... | 29 |
| 3.2.7. Flask Web Uygulaması | 30 |
| BÖLÜM 4. | |
| BULGULAR..... | 32 |
| 4.1. MNIST Modeli İçin Değerlendirmeler | 32 |
| 4.2. EMNIST Modeli İçin Değerlendirmeler | 34 |
| 4.3. Sistemin Test Edilmesi..... | 37 |
| BÖLÜM 5. | |
| TARTIŞMA VE SONUÇ..... | 41 |
| KAYNAKLAR..... | 42 |
| EK-1..... | 44 |
| EK-2..... | 44 |
| ÖZGEÇMİŞ..... | 45 |

SİMGELER VE KISALTMALAR LİSTESİ

| | |
|--------|---|
| ASCII | : American Standard Code for Information Interchange |
| CNN | : Convolutional Neural Network |
| CSV | : Comma-Separated Values |
| CTC | : Connectionist Temporal Classification |
| DNN | : Deep Neural Network |
| EMNIST | : Extended MNIST |
| HTML | : HyperText Markup Language |
| IAM | : Identity and Access Management |
| LSTM | : Long Short-Term Memory |
| MNIST | : Modified National Institute of Standards and Technology |
| Numpy | : Numerical Python |
| PDF | : Portable Document Format |
| RGB | : Red, Green, Blue |
| RNN | : Recurrent Neural Network |
| SVM | : Support Vector Machine |
| TXT | : Text |
| URL | : Uniform Resource Locator |
| YSA | : Yapay Sinir Ağı |

ŞEKİLLER LİSTESİ

| | |
|--|----|
| Şekil 3. 1 MNIST Veri Seti Örnekleri [1]..... | 16 |
| Şekil 3. 2 EMNIST Veri Seti Örnekleri [1]..... | 17 |
| Şekil 3. 3 Sistemin Genel Çalışma Şeması..... | 19 |
| Şekil 3. 4 Form Şablonları..... | 20 |
| Şekil 3. 5 Satır Segmentasyonu..... | 21 |
| Şekil 3. 6 Karakter Segmentasyonu..... | 22 |
| Şekil 3. 7 Hatalı Karakter Segmentasyonu..... | 23 |
| Şekil 3. 8 TXT Dosyasının Güncellenmesi..... | 27 |
| Şekil 3. 9 CSV Dosyası İçeriği..... | 28 |
| Şekil 3. 10 Sınıflandırma ile Etiket Tahmini..... | 29 |
| Şekil 3. 11 Örnek Python Sözlüğü..... | 29 |
| Şekil 3. 12 Web Arayüzü 1..... | 30 |
| Şekil 3. 13 Web Arayüzü 2..... | 30 |
| Şekil 3. 14 Web Arayüzü 3..... | 31 |
| Şekil 3. 15 Web Arayüzü 4..... | 31 |
| Şekil 4. 1 MNIST Modeli Doğruluk ve Kayıp Grafiği..... | 33 |
| Şekil 4. 2 MNIST Modeli Karışıklık Matrisi..... | 33 |
| Şekil 4. 3 Rakam Tahminleri..... | 34 |
| Şekil 4. 4 Yanlış Rakam Tahminleri..... | 34 |
| Şekil 4. 5 EMNIST Modeli Doğruluk ve Kayıp Grafikleri..... | 35 |
| Şekil 4. 6 Harf Tahminleri..... | 36 |
| Şekil 4. 7 Giriş Sayfası..... | 37 |
| Şekil 4. 8 Fotoğraf Yükleme ve Kırpma Sayfası..... | 38 |
| Şekil 4. 9 Sonuç Sayfası..... | 39 |

| | |
|---|----|
| Şekil 4. 10 PDF İçeriği..... | 39 |
| Şekil 4. 11 Adres Bilgisi Formu Sonucu..... | 40 |
| Şekil 4. 12 Kişisel Bilgi Formu Sonucu..... | 40 |

TABLÖLAR LİSTESİ

| | |
|---|----|
| Tablo 1 CNN Mimarisi Özeti..... | 24 |
| Tablo 2 MNIST Modeli Performans Değerlendirme Sonuçları..... | 32 |
| Tablo 3 EMNIST Modeli Performans Değerlendirme Sonuçları..... | 34 |

ÖZET

Anahtar kelimeler: El yazısı, harf, rakam, dijital, derin öğrenme, etiket tahmini, web uygulaması, segmentasyon

El yazısını dijital hale çevirmek, çeşitli alanlarda kolaylık sağlayan ve günümüzde de farklı çalışmalara konu olan bir işlemdir. Kağıt tabanlı verilerin dijital ortama aktarılması, verilerin saklanması ve analizi açısından büyük bir öneme sahiptir.

Bu projede, kağıt üzerindeki el yazısını dijitalleştirme üzerine yapılan çalışmalar incelenerek bir sistem sunulmuştur. Bu sistem segmentasyon, sınıflandırma ve birleştirme gibi farklı aşamaları içermektedir. Bu aşamalarda girdi olarak verilen el yazısı analiz edilerek dijitalle çevrilir. Sınıflandırma aşamasında bir derin öğrenme yöntemi olan CNN modeli kullanılmıştır. Modelin eğitimi için bu alanda en çok kullanılan MNIST ve EMNIST veri setleri seçilmiştir.

Kullanıcıların tasarlanan sistemi kolaylıkla kullanabilmesi açısından sistemin entegre edildiği bir Flask web uygulaması geliştirilmiştir. Uygulama üç ayrı form içeriğindeki el yazılarını dijitalle çevirecek şekilde tasarlanmıştır.

Bu proje kapsamında MNIST veri seti ile oluşturulan rakam modelinin eğitim ve test başarıları %99; EMNIST veri seti ile oluşturulan harf modelinin eğitim başarıları %93, test başarıları ise %90 olarak elde edilmiştir. Ayrıca web uygulaması ile bu modellerin testleri gerçekleştirilmiş olup görsellerle sunulmuştur. Son olarak sistemdeki eksikler ve gelecekteki planlar belirtilmiştir.

ABSTRACT

Keywords: Handwriting, letter, digit, digital, deep learning, label prediction, web application, segmentation

Digitizing handwriting is a process that provides convenience in various fields and is the subject of different studies today. Digitizing paper-based data is of great importance for data storage and analysis.

In this project, a system is presented by examining the studies on digitizing handwriting on paper. This system includes different stages such as segmentation, classification and merging. In these stages, the handwriting given as input is analyzed and digitized. CNN model, a deep learning method, is used in the classification stage. MNIST and EMNIST datasets, which are the most widely used in this field, were selected for training the model.

In order for users to easily use the designed system, a Flask web application has been developed in which the system is integrated. The application was designed to digitize the handwritings in three different forms.

Within the scope of this project, the training and test accuracy of the digit model created with the MNIST dataset was 99%; the training and test accuracy of the letter model created with the EMNIST dataset was 93% and 90%, respectively. In addition, the tests of these models were carried out with the web application and presented with visuals. Finally, the shortcomings of the system and future plans are presented.

BÖLÜM 1. GİRİŞ

Günümüzde çeşitli belge ve dokümanlar üzerinde bulunan el yazısını dijitalle çevirmek farklı alanlarda kullanılan bir işlemdir. Verilerin kağıt üzerinde bulunması depolama, paylaşılma ve dosyalama gibi konularda zorluklara yol açtığından bilgisayar ortamına aktarılması için bu işlem önem taşımaktadır. Ayrıca bu verilerin bilgisayar ortamına aktarılmış olması veri analizi işlemleri için kolaylık sağlamaktadır. Analiz edilen bu veriler çeşitli makine öğrenimi veya derin öğrenme uygulamalarında kullanılarak bu verilerden çeşitli bilgiler çıkarılır. Bunlardan yola çıkılarak bu çalışmada el yazısını dijital hale çevirme işleminin gerçekleştirilmesi ele alınıp bir sistem önerilmiştir.

Bu önerilen çalışmanın kararlaştırılma sürecinde konuyla ilgili makalelerden, akademik tezlerden ve internet kaynaklarından literatür taraması yapılmıştır. İncelenen çalışmalarda genel olarak MNIST veri setinin kullanıldığı görülmüştür. Bu veri seti 250'den fazla yazardan alınan, el yazısı rakamlardan oluşan 60.000'i eğitim seti için olmak üzere toplamda 70.000 örnek içermektedir [1]. Önerilen çalışmada sadece rakam değil harf tanımaya da ihtiyaç duyulduğundan yalnızca bu veri setini kullanmak yetersiz kalacaktır. Bu sebeple MNIST veri tabanı ile aynı yapıyı paylaşan, hem el yazısı rakamlardan hem de harflerden oluşan EMNIST veri seti de kullanılmıştır. EMNIST 88.800'ü eğitim seti için olmak üzere toplam 103.600 örnekten oluşmaktadır [2]. Ayrıca bazı çalışmalarda veri seti için IAM El Yazısı Veritabanı'nın da kullanıldığı görülmüştür [3].

Literatür taramasındaki çalışmalarda karakter tanıma işlemi için farklı derin öğrenme ve makine öğrenimi algoritmaları ve bu algoritmaların karşılaştırılmaları yer almaktadır. Makine öğrenimi, bilgisayarlara belirli görevleri nasıl gerçekleştireceklerini öğrenme yeteneği sağlayan bir yapay zeka dalıdır. Bu teknik, verilerden öğrenen ve yeni veriler verildiğinde tahminler yapan modellerin tasarımına

dayanır [4]. Derin öğrenme, makine öğreniminin bir alt kümesidir. Özellik çıkarımı ve dönüştürme için birden çok işlem birimi katmanından oluşur. Veri girişine yakın alt katmanlar basit özellikleri öğrenirken, üst katmanlar önceki katmanlardan türetilen daha karmaşık özellikleri öğrenir. Böylelikle hiyerarşik ve güçlü bir özellik temsili oluşturulur [5]. Literatürdeki çalışmaların geneline bakıldığında, derin öğrenme algoritmalarının büyük sayıda örnek içeren veri kümelerinin işlenmesinde daha etkili olduğu bilindiğinden [6], makine öğrenimine göre daha çok tercih edildiği görülmüştür.

Sistemin geneli segmentasyon, ön işleme, sınıflandırma ve birleştirme gibi farklı aşamalardan oluşmaktadır. Segmentasyon aşaması, satır segmentasyonu ve karakter segmentasyonu olmak üzere iki farklı bölümden oluşmaktadır. İlk olarak RGB formatındaki fotoğraf ve taranmış dokümanlar girdi verisi olarak alınıp segmentasyon aşaması için uygun hale getirilir. Segmentasyon aşamasında görüntü her biri içerisinde farklı özelliklerin tutulduğu anlamlı bölgelere ayrılır. Sınıflandırma aşamasında, segmente edilmiş karakterler MNIST ve EMNIST veri setlerine uygun hale getirilmek için çeşitli ön işleme adımlarına tabi tutulur. Sonrasında elde edilen bu karakterler etiketlenilerek sınıflandırılmış olur. Sınıflandırma işlemi için bir derin öğrenme yöntemi olan CNN modeli, MNIST ve EMNIST verileri için ayrı ayrı eğitilerek iki ayrı model elde edilmiştir. Bu modellerin performansı doğruluk, hassasiyet, duyarlılık ve F1 puanı gibi metriklerle değerlendirilmiştir. Birleştirme aşamasında ise etiketlenilmiş karakterler birleştirilerek anlamlı kelimeler elde edilmiştir. Böylelikle el yazısı metni dijitalle çevirme işlemini gerçekleştiren sistem tamamlanmıştır. Ayrıca kullanıcıların sisteme kolaylıkla ulaşabilmesi için Flask Framework kullanılarak bir web uygulaması geliştirilmiştir.

Projede model oluşturulduktan sonra, MNIST ve EMNIST test verileri ile özel olarak hazırlanmış test verileri kullanılarak testler gerçekleştirilmiş ve elde edilen sonuçlar, her bir test verisi için yapılan etiket tahminleriyle görsel olarak sunulmuştur. Ayrıca, özel olarak hazırlanan çeşitli form şablonları kullanılarak ek testler yapılmıştır. Modelin performansını değerlendiren metrikler ve sonuçlar, tablolar ve görseller

halinde sunulmuştur. Son olarak çalışmanın eksikleri, gelecek planları ve sistemin başarısını artırmaya yönelik hedefler açıklanmıştır.

BÖLÜM 2. ÖNCEKİ ÇALIŞMALARIN ARAŞTIRILMASI

2.1. El Yazısı Karakter Tanıma ve Resim Sınıflandırmada Derin Öğrenme Yaklaşımları

Bu tezde el yazısı karakter tanıma çalışması yapılmıştır. Kullanılan veri setlerine göre algoritmaların karşılaştırılması yapılmış, benzer çalışmalarla sonuçlar kıyaslanmış ve bulgular sunulmuştur. 60000 eğitim görüntüsü ve 10000 test görüntüsünden oluşan Mnist veri seti ve eğitim için 13440, test için 3360 karakterden oluşan Arapça veri seti kullanılmıştır. Bu veri setleri ile yapılan çalışmalarda DNN, CNN ve RNN derin öğrenme modelleri kullanılmıştır. Çalışmada Tensorflow ve Keras kütüphaneleri kullanılmıştır. Bu çalışmada oluşturulan modellerin değerlendirilmesi için doğruluk ve kayıp değerleri hesaplanmıştır. Kayıp fonksiyonu olarak ikili çapraz entropi ve seyrek kategorik çapraz entropi kullanılmıştır. Doğruluk fonksiyonu, doğru öngörülen sınıfın toplam test sınıfına bölünerek ve daha sonra 100 katı alınarak hesaplanmıştır. Mnist veri setinde doğruluk oranları DNN için %99.53, CNN için %99.88 ve RNN için %99.05 olarak; kayıp oranları DNN için 0.039, CNN için 0.042, RNN için 0.048 olarak elde edilmiştir. Arapça veri setinde doğruluk oranları DNN için %96.48, CNN için %99.00 ve RNN için %96.94 olarak; kayıp oranları DNN için 0.150, CNN için 0.028, RNN için 0.077 olarak elde edilmiştir [7].

2.2. Handwritten Text Recognition and Conversion Using Convolutional Neural Network (CNN) Based Deep Learning Model

Önerilen çalışmada, el yazısı ile yazılmış bir metnin resmi girdi olarak alınıp dijital metne dönüştürülmektedir. Çalışmada eğitim için 600'den fazla yazarın el yazısı örneklerini ve 100.000'den fazla kelimededen oluşan görselleri içeren IAM El Yazısı Veritabanı kullanılmıştır. Çalışmada birden fazla görüntü örneğinden benzer nesnelerin özelliklerini incelemek ve bunları sınıflandırmak için CNN ve metin sıralı

veri olduğundan uzun belleğe sahip olan LSTM kullanılmıştır. Ayrıca görüntüdeki metnin farklı yerleşimleri için CTC kaybı kullanılmıştır. Tensorflow, OpenCv ve Numpy kütüphaneleri kullanılmıştır. Model, yaklaşık 86.000 insan el yazısı örneğiyle eğitilmiş ve 10.000 örnekle doğrulanmıştır. Birden fazla eğitimden sonra modelde, eğitim verilerinde %94 doğruluk ve 0,147 kayıp; doğrulama verilerinde ise %85 doğruluk ve 1,105 kayıp kaydedilmiştir [3].

2.3. Neural Network Based Handwritten Character Recognition System Without Feature Extraction

Bu makalede, çok katmanlı İleri Beslemeli sinir ağı kullanılarak özellik çıkarımı yapılmadan İngiliz alfabesi kullanılarak el yazısıyla yazılmış karakterlerin tanınmasına yönelik bir yöntem sunulmuştur. Sinir ağının eğitimi için elli farklı karakter veri seti kullanılmıştır ancak kullanılan veri setlerinin isimleri makalede verilmemiştir. Önerilen sistem, özellik çıkarma tekniklerini kullanan yöntemlere kıyasla daha az karmaşıktır. Karakterleri sınıflandırmak için kullanılan birkaç sinir ağı mimarisi arasında, her biri 100 nöronlu iki gizli katmana sahip olanın %90.19 ile en yüksek tanıma doğruluğunu sağladığı bulunmuştur [8].

2.4. Makine Öğrenmesi Yöntemleriyle El Yazısı Tanıma

Bu çalışmada 6 farklı makine öğrenimi algoritması ile el yazısı tanıma işlemi gerçekleştirilmiştir ve elde edilen başarı sonuçları karşılaştırılmıştır. Çalışmada MNIST veri seti kullanılmıştır. Bu veri seti 250 farklı kişiden alınan 60.000 örnek içerir. Çalışma el yazısı tanıma işlemi için ön işleme, karakter dilimleme, öznitelik çıkarımı, sınıflandırma ve son işleme adımlarını içerir. Kullanılan algoritmaların başarı oranları; Destek Vektör Makinesi için %90, Karar Ağaçları için %87, Rastgele Orman ve Yapay Sinir Ağı için %97, K-En Yakın Komşu Algoritması için %96 ve K-Ortalama Algoritması için %98 olarak elde edilmiştir [9].

2.5. El Yazısı Rakam Tanıma İçin Destek Vektör Makinelerinin Ve Yapay Sinir Ağlarının Karşılaştırması

Bu çalışmada Matlab programlama dili kullanılarak hazırlanan bir uygulama aracılığıyla SVM ve YSA yöntemleri veri seti ile ayrı ayrı eğitilmiştir. Eğitim verileri, MNIST veritabanından rastgele seçilmiş olan 5000, 10000, 20000, 30000, 60000 adetlik kümeler halinde hazırlanmıştır. 5000 verilik veri setinde SVM için %97.06'lık, YSA için %88,30'luk doğruluk; 10000 verilik veri setinde SVM için %99,97'lik, YSA için %89,39'luk doğruluk, 20000'lik veri setinde SVM için %99,98'lik, YSA için %91,78'lik doğruluk; 30000'lik veri setinde SVM için %99,97'lik, YSA için %91,62'lik doğruluk; 60000'lik veri setinde SVM için %99,99'luk, YSA için %91.47'lik doğruluk sağlanmıştır [10].

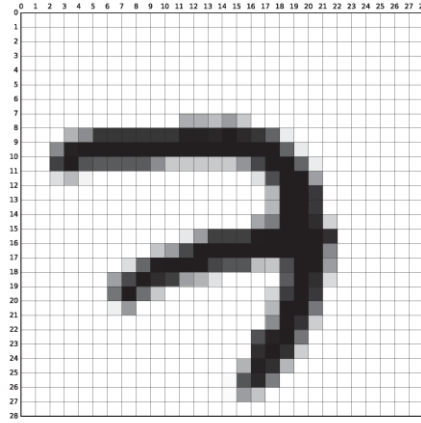
BÖLÜM 3. YÖNTEM VE TASARIM

3.1. Materyal

Bu çalışmada rakamlar için MNIST ve harfler için EMNIST veri setleri kullanılmıştır. Eğitim, Python programlama dili ve kütüphaneleri kullanılarak Visual Studio Code üzerinde gerçekleştirilmiştir. Web uygulaması için Flask Framework kullanılmıştır.

3.1.1. MNIST (Mixed National Institute of Standards and Technology)

MNIST veri kümesi, farklı makine öğrenimi ve örüntü tanıma çalışmaları için test ortamı olarak yaygın bir şekilde kullanılmaktadır. El yazısı rakamlardan oluşan 60.000'i eğitim seti için olmak üzere toplamda 70.000 örnek içermektedir, her bir örnek 28x28 piksel ölçüsündedir [1].



(a) MNIST sample belonging to the digit '7'.



(b) 100 samples from the MNIST training set.

Şekil 3. 1 MNIST Veri Seti Örnekleri [1].

Şekilde 3.1 (a)'da mnist veri setinde “7” rakamına karşılık gelen bir örnek ve şekil 3.1 (b)'de ise eğitim setinden 100 örnek verilmiştir [1].

3.1.2. EMNIST (Extended MNIST)

MNIST veri tabanıyla aynı yapıyı paylaşan EMNIST, hem el yazısı rakamlardan hem de harflerden oluşan veri setidir. 88.800'ü eğitim seti için olmak üzere toplam 103.600 örnekten oluşmaktadır, her bir örnek 28x28 piksel ölçülere ve 8 bit gri seviyeli çözünürlüğe sahiptir [2]. Şekil 3.2'de EMNIST veri setine ait örnekler verilmiştir [1].



Şekil 3. 2 EMNIST Veri Seti Örnekleri [1].

3.1.3. Kütüphaneler ve Araçlar

Scikit-learn, sınıflandırma, regresyon, kümeleme ve karar ağaçları gibi farklı makine öğrenme algoritmalarını birleştiren bir python kütüphanesidir. Ayrıca öğrenme algoritmalarının test edilmesi ve sonuçlarının değerlendirilmesi için kullanılan Precision, Recall, F1 Score ve Accuracy gibi metrikleri içermektedir [11].

Numpy, çok boyutlu dizilerle ve matrislerle çalışmayı ve matematiksel işlemler yapmayı sağlayan bir Python kütüphanesidir [12].

Pandas, Python programlama dilinde veri manipülasyonu ve analizi için kullanılan bir kütüphanedir. Verileri işlemek, temizlemek, dönüştürmek ve analiz etmek için geniş olanaklar sağlar.

TensorFlow, derin öğrenme modelleri oluşturmak için kullanılan bir Python kütüphanesidir. Çeşitli yapay zeka uygulamaları için uygundur. Esnekliğiyle, görüntü

tanıma, doğal dil işleme ve endüstriyel ölçekte üretim tahminleri gibi farklı alanlarda kullanılabilir [13].

Keras, Python'da derin öğrenme modelleri oluşturmak için kullanılan yüksek seviyeli bir yapay zeka kütüphanesidir. Modüler yapısı sayesinde farklı derin öğrenme mimarileri kolayca oluşturulabilir, eğitilebilir ve değerlendirilebilir.

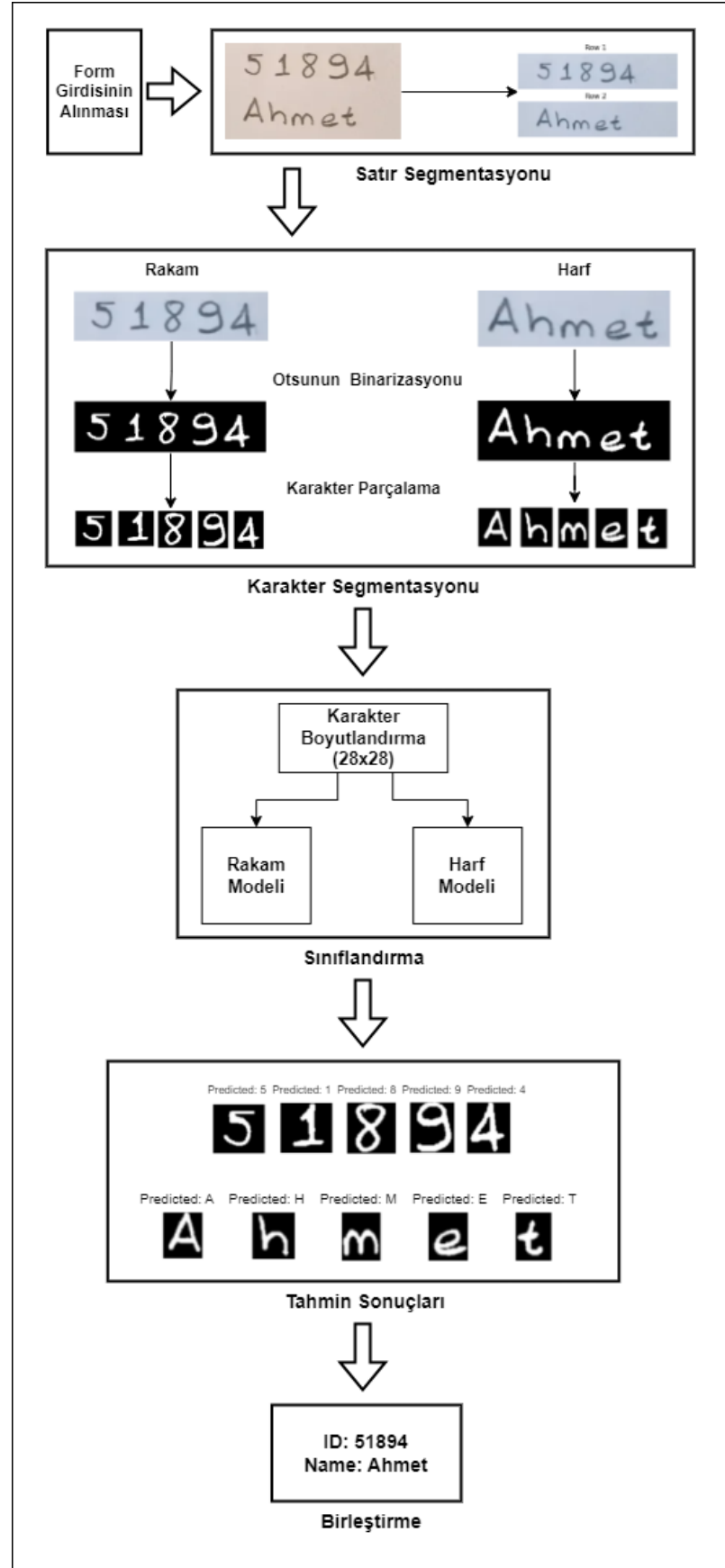
OpenCV, bilgisayarla görme ve görüntü işleme işlemleri için kullanılan açık kaynaklı bir kütüphanedir.

Seaborn ve matplotlib veri görselleştirme işlemleri için kullanılan Python kütüphaneleridir.

Flask, Python tabanlı ve küçük ile orta ölçekli web uygulamaları geliştirmek için kullanılan framework'tür. Açık kaynak kodlu olan Flask, Jinja2 şablon motoruyla dinamik HTML sayfaları oluşturma ve URL'leri belirli Python fonksiyonlarına yönlendirme imkanı sunar.

3.2. Yöntem

Yöntem kısmında planlanan sistemin genel çalışma prensibi sunulmaktadır. Sistem verinin eldesi, satır segmentasyonu, karakter segmentasyonu, sınıflandırma ve birleştirme aşamalarından oluşmaktadır. Karakter segmentasyonu ve sınıflandırma aşamalarında belirli ön işleme adımları gerçekleştirilmiştir. Şekil 3.3'te sistemin genel çalışma şeması verilmiştir.



Şekil 3. 3 Sistemin Genel Çalışma Şeması

3.2.1. Verinin Eldesi

Bu çalışma için üniversite bilgilerinin, adres bilgilerinin ve kişisel bilgilerin alındığı üç farklı form şablonu özel olarak oluşturulmuştur. Şekil 3.4'te bu form şablonları verilmiştir. Bu form şablonlarının fotoğrafları sadece el yazısının bulunduğu kısmı içerecek şekilde kırpılarak sisteme girdi olarak verilir. Girdi olarak verilen görüntü satır segmentasyon işlemine tabi tutulacaktır.

| SAKARYA UNIVERSITY OF APPLIED SCIENCES | |
|---|---|
| University Informations Form | |
| ID | : |
| Name | : |
| Surname | : |
| University | : |
| Faculty | : |
| Department | : |
| Student Number | : |

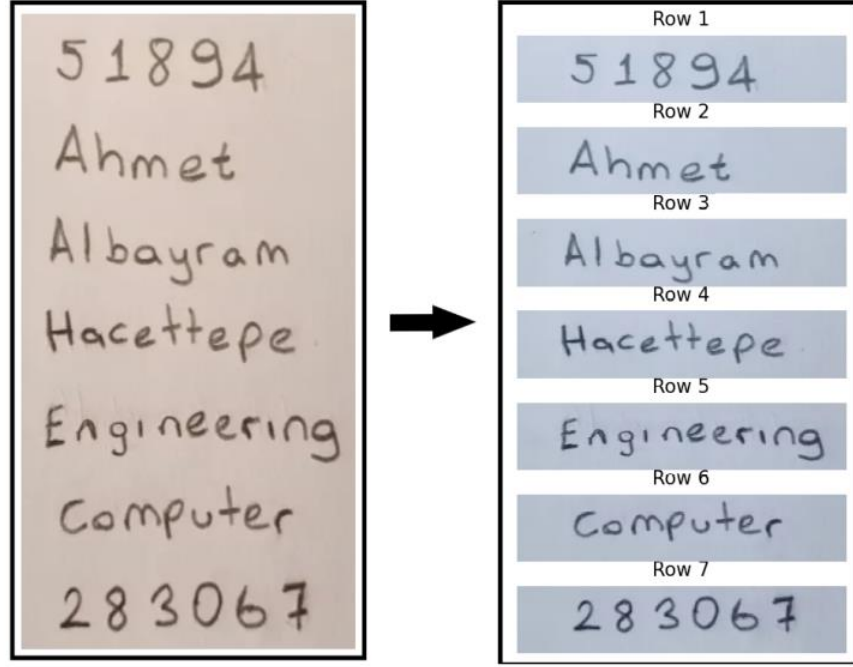
| SAKARYA UNIVERSITY OF APPLIED SCIENCES | |
|---|---|
| Address Informations Form | |
| ID | : |
| Country | : |
| City | : |
| Zipcode | : |
| Street Number | : |
| Apartment Number | : |

| SAKARYA UNIVERSITY OF APPLIED SCIENCES | |
|---|---|
| Personal Informations Form | |
| ID | : |
| Identification Number | : |
| Gender | : |
| Birth Year | : |
| Birthplace | : |
| Disease | : |
| Phone Number | : |

Şekil 3. 4 Form Şablonları

3.2.2. Satır Segmentasyonu

Girdi olarak alınan resmin yükseklik ve genişlik boyutları belirlenir ve form şablonundaki satır sayısına göre eşit şekilde parçalara yani satırlara ayrılır. Şekil 3.5'te satır segmentasyon örneği verilmiştir.



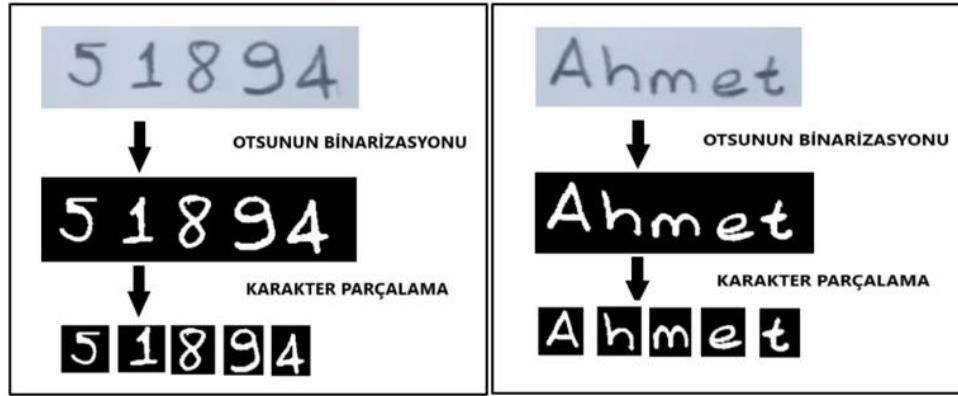
Şekil 3. 5 Satır Segmentasyonu

3.2.3. Karakter Segmentasyonu

Satırlardaki karakterlerin ayrılabilmesi için satır parçaları üzerine karakter segmentasyonu uygulanır. Bu parçaların karakterlere daha kolay ve doğru ayrılabilmesi için eşikleme işlemi uygulanmıştır. Eşikleme işlemi, görüntüye Otsu'nun Binarizasyon yöntemi uygulanarak yapılır. Bu yöntem, görüntünün yalnızca nesne (ön plan) ve arka plandan oluştuğunu ve arka planın heterojenliğinin ve çeşitliliğinin göz ardı edildiğini varsayan doğrusal diskriminant kriteridir [16]. Böylelikle girdi verisi elde edilirken meydana gelen çekim veya ışık gibi farklılıklar en aza indirilmiş olur. Karakter segmentasyonu için arka plan koyu ön plan ise açık renk olacak şekilde ayarlanmıştır.

Dönüştürülmüş görüntüdeki bağlı bileşenlerin analizi yapılır ve her bir bileşenin istatistikleri elde edilir. Bu işlem, görüntüdeki farklı nesnelerin ayrı bileşenler olarak tanımlanmasını sağlar. Daha sonra, bileşenlerin ağırlık merkezlerine göre sıralama yapılır. Bu işlem, bileşenlerin X koordinatlarına göre sıralanmasını sağlar, böylece yazı karakterlerinin soldan sağa doğru sıralanması elde edilir. Sıralı bileşenler, her birinin sınırlayıcı kutularını alarak ayrıştırılır. Bu sınırlayıcı kutular, bileşenlerin boyutlarını 28x28 olarak tanımlar. Boyutlandırma sonucunda tahmin için modele gönderilecek olan bileşenler ile veri setindeki görüntülerin boyutlarının eşitlenmesi sayesinde daha iyi tanınabilirlik sağlar. Böylece tahmin işlemi için kullanılacak karakterler elde edilmiş olur.

Şekil 3.6’da karakter segmentasyonu örneği verilmiştir.



Şekil 3. 6 Karakter Segmentasyonu

Karakterler bitişik yazıldığında parçalama işlemi doğru şekilde gerçekleştirilememektedir. “merhaba” kelimesinde bulunan “e” ve “r” harfleri ile “a” ve “b” harfleri bitişik yazıldığından parçalama işlemi Şekil 3.7’deki gibi gerçekleştirilmiş olur. Bu da karakterlerin yanlış etiketlenmesine sebep olur. Bu yüzden sisteme gönderilen yazılarda karakterlerin birbirinden ayrı yazılması gerekmektedir.



Şekil 3. 7 Hatalı Karakter Segmentasyonu

Eğitim sırasında kullanılan EMNIST veri seti İngiliz Alfabesinden oluştuğu için Türkçe karakterler kullanılmamaktadır. Ayrıca karakter segmentasyon “ı” ve “j” gibi noktalı harflerin noktalarının ayrı bir karakter olarak algılanmasından dolayı tahmin kısmında sorun yaşanabilmektedir. Bundan dolayı test aşamalarında noktalı harfler kullanılmamıştır.

3.2.4. Harf ve Rakam Modellerinin Oluşturulması

Bu aşamada MNIST ve EMNIST veri setleri ile birlikte aynı CNN modeli ayrı ayrı eğitilmiştir.

3.2.4.1. CNN (Convolutional Neural Network)

Yapay sinir ağları, insan beyninin çalışma şekline benzeyen bilgisayar sistemleridir. CNN'ler, her bir nöronun bir girdi alıp doğrusal olmayan bir fonksiyonla işleyerek skaler bir çarpım ürettiği, kendini optimize eden yapay sinir ağlarıdır ve bu yönleriyle geleneksel Yapay Sinir Ağları'na benzerdirler. CNN'de üç tür temel katman vardır: konvolüsyonel katman, havuzlama katmanı ve tam bağlı katman. Bu katmanlar girdi verileri üzerinde farklı görevler gerçekleştirir. Konvolüsyonel katmanda, özellik çıkarımı için çeşitli filtreler uygulanır. Havuzlama katmanı, sırasıyla filtre bölgesindeki maksimum veya ortalama değeri çıkaran maksimum havuzlama veya ortalama havuzlama işlemini gerçekleştirir. Tam bağlı katman, özellik haritalarından gelen bilgileri toplar ve nihai sınıflandırmayı oluşturur [17].

MNIST ve EMNIST veri setleri için aynı CNN modelinin mimarisi kullanılmıştır. Fakat son katmandaki nöron sayısı ve parametre sayısı farklıdır. Tablo 1'de oluşturulan CNN mimarisi özeti verilmiştir.

Tablo 1 CNN Mimarisi Özeti

| Layer (type) | Output Shape | Param # |
|--------------------------------|-------------------------|-------------|
| conv2d (Conv2D) | (None, 26, 26, 32) | 320 |
| conv2d_1 (Conv2D) | (None, 24, 24, 64) | 18496 |
| max_pooling2d (MaxPooling2D) | (None, 12, 12, 64) | 0 |
| conv2d_2 (Conv2D) | (None, 10, 10, 128) | 73856 |
| max_pooling2d_1 (MaxPooling2D) | (None, 5, 5, 128) | 0 |
| conv2d_3 (Conv2D) | (None, 3, 3, 128) | 147584 |
| max_pooling2d_2 (MaxPoolin2D) | (None, 1, 1, 128) | 0 |
| flatten (Flatten) | (None, 128) | 0 |
| dense (Dense) | (None, 512) | 66048 |
| dropout (Dropout) | (None, 512) | 0 |
| dense_1 (Dense) | (None, 256) | 131328 |
| dropout_1 (Dropout) | (None, 256) | 0 |
| dense_2 (Dense) | (None, 128) | 32896 |
| dropout_2 (Dropout) | (None, 128) | 0 |
| dense_3 (Dense) | (None, 10) / (None, 28) | 1290 / 4773 |

Conv2D katmanı görüntüleri analiz edip özellikleri saptamak için kullanılır. Örnek olarak “conv2d” katmanına bakıldığında 32 farklı filtre kullanılarak 26x26 boyutunda bir çıktı üretir. Bu katmanlarda aktivasyon fonksiyonu olarak “ReLU” kullanılmıştır. ReLU, pozitif girdileri aynı tutarken negatif girdileri sıfıra indirger ve bu sayede modelde doğrusal olmayanlık sağlar.

MaxPooling2D katmanı ağırlık sayısını azaltır ve uygunluğu kontrol eder. Örnek olarak “max_pooling2d” katmanı önceki katmanın çıktısını alır ve boyutunu azaltmak için her bir 2x2 piksel bloğundaki en büyük değeri seçer. Bu, veri boyutunu azaltırken önemli özellikleri korur ve hesaplama yükünü azaltır.

Flatten katmanı, bu özellik haritalarını tek bir vektör haline getirip ağıın sonraki Dense katmanlarına uygun bir giriş oluşturur. Örnek olarak “flatten” katmanı 128 uzunluğunda bir çıkış vektörü oluşturur.

Dense katmanı veri üzerinde daha karmaşık ilişkiler öğrenir ve temsil eder. Örnek olarak “dense” katmanı Flatten katmanından gelen vektörel veriyi alır ve 512 nöron içeren bir katman oluşturur. Bu katmanlarda aktivasyon fonksiyonu olarak “ReLU” kullanılmıştır. Son katman olan “dense_3” katmanı MNIST veri seti için 10, EMNIST veri seti için 28 nöron içerir. Bunun nedeni veri setlerindeki sınıf sayısının farklı olmasıdır. Bu katmanda aktivasyon fonksiyonu olarak “softmax” kullanılmıştır. Softmax fonksiyonu, model çıktılarından olasılık dağılımı oluşturarak her bir sınıfın olasılığını 0 ile 1 arasında normalize eder.

Dropout katmanı, rastgele bazı nöronları her eğitim adımında devre dışı bırakarak aşırı uyumu önler. Örnek olarak “dropout” katmanı 512 uzunluğunda bir çıkış vektörü oluşturur.

Mimaride görülen “None” ifadesi, model katmanlarına farklı boyutlarda giriş verileri verebilme esnekliği sağlar. “Param” ifadesi ise modeldeki öğrenilebilir parametrelerin toplam sayısını gösterir.

Tekil sınıf endekslerini içeren ve one-hot kodlanmamış hedef değişkenlerle çalışıldığı için model eğitim sürecinde kayıp fonksiyonu olarak “sparse_categorical_crossentropy” kullanılmıştır. Optimizer olarak ise “adam” algoritması kullanılmıştır.

MNIST veri seti için 5 epochluk bir eğitim süreci gerçekleştirilmiştir, early stopper kullanılmamıştır. EMNIST veri seti için 10 epochluk bir eğitim süreci gerçekleştirilmiştir ancak early stopper ile eğitim 8. epochta sonlanmıştır.

Oluşturulan CNN modelinin başarısını değerlendirmek için farklı performans metrikleri kullanılmıştır [11]:

1. Precision (Hassasiyet): Doğru pozitif tahminlerin tüm pozitif tahminlere oranı alınarak hesaplanır.
2. Recall (Duyarlılık): Gerçek pozitiflerin ne kadarının doğru tahmin edildiğini gösterir.
3. F1 Score: Hassasiyet ve duyarlılık değerlerinin harmonik ortalaması alınarak hesaplanır.
4. Accuracy (Doğruluk): Yapılan tüm tahminlerin ne kadarının doğru tahmin edildiğini gösterir.

3.2.4.2. MNIST Veri Seti İçin Model Eğitimi

MNIST veri seti, “keras.datasets” kütüphanesinden çekilerek kullanılmıştır. Yüklenen veriler, eğitim ve test verileri olmak üzere ayrılmıştır. Sonrasında boyutlandırma, tip dönüşümü ve normalizasyon gibi ön işleme adımları uygulanmıştır. Son olarak bu veriler “Bölüm 3” altında anlatılan CNN modeli üzerinde eğitilmiştir. Eğitim sonucunda rakam tahmininde kullanılacak olan H5 uzantılı “numbers_model” model elde edilmiştir.

3.2.4.3. EMNIST Veri Seti İçin Model Eğitimi

EMNIST veri seti, CSV uzantılı “emnist-letters-train” ve “emnist-letters-test” dosyalarından çekilerek kullanılmıştır. Bu dosyaların ilk sütunlarında harfleri içeren görüntülerin etiketleri, sonraki sütunlarında ise bu görüntülerin piksel değerleri bulunur. Ayrıca etiketlendirme işlemi için TXT uzantılı “emnist-letters-mapping” dosyası kullanılmıştır. Orijinal mapping dosyasının içeriği şekildeki gibi değiştirilmiştir. Etiketleme için büyük ve küçük harflerin ASCII karakter tablosundaki değerleri kullanılmıştır. Örneğin sınıf etiketi 1 olan görüntü, ASCII haritasındaki 65 değerine (“A” harfi) karşılık gelir.

| | | | | |
|----|----|-----|----|----|
| 1 | 65 | 97 | 1 | 65 |
| 2 | 66 | 98 | 2 | 66 |
| 3 | 67 | 99 | 3 | 67 |
| 4 | 68 | 100 | 4 | 68 |
| 5 | 69 | 101 | 5 | 69 |
| 6 | 70 | 102 | 6 | 70 |
| 7 | 71 | 103 | 7 | 71 |
| 8 | 72 | 104 | 8 | 72 |
| 9 | 73 | 105 | 9 | 73 |
| 10 | 74 | 106 | 10 | 74 |
| 11 | 75 | 107 | 11 | 75 |
| 12 | 76 | 108 | 12 | 76 |
| 13 | 77 | 109 | 13 | 77 |
| 14 | 78 | 110 | 14 | 78 |
| 15 | 79 | 111 | 15 | 79 |
| 16 | 80 | 112 | 16 | 80 |
| 17 | 81 | 113 | 17 | 81 |
| 18 | 82 | 114 | 18 | 82 |
| 19 | 83 | 115 | 19 | 83 |
| 20 | 84 | 116 | 20 | 84 |
| 21 | 85 | 117 | 21 | 85 |
| 22 | 86 | 118 | 22 | 86 |
| 23 | 87 | 119 | 23 | 87 |
| 24 | 88 | 120 | 24 | 88 |
| 25 | 89 | 121 | 25 | 89 |
| 26 | 90 | 122 | 26 | 90 |

Şekil 3. 8 TXT Dosyasının Güncellenmesi

Şekil 3.8’de sol tarafta “emnist-letters-mapping.txt” dosyasının orijinal içeriği yer alır, sağ tarafta ise bu dosyanın eğitim için güncellenmiş içeriği yer alır. Orijinal halindeki ilk sütun etiketleri, ikinci sütun ASCII tablosundaki büyük harflerin değerlerini, üçüncü sütun ise ASCII tablosundaki küçük harflerin değerlerini göstermektedir.

Etiketlendirme işlemi için kullanılan kod yapısına göre, orijinal TXT dosyasının üçüncü sütunu etiketleme işlemine dahil edilememiştir. Bu sorunun çözümü için iki sütun aşağı doğru birleştirilerek satır sayısı ve etiket değerleri artırılmıştır. Ancak CSV dosyasındaki eğitim ve test verilerinin etiketleri 26 değerine kadar olduğundan model eğitim sürecinde sorun yaşanmıştır. Bu sebeple orijinal dosya değiştirilerek sağdaki haliyle eğitim sürecine dahil edilmiştir. Bu yüzden etiketleme işlemi sadece büyük harfler olacak şekilde ayarlanmıştır.

Veri setindeki görüntülerin önce yatay olarak simetrisi alınır ve daha sonra saat yönünde 90 derece döndürülür. Bu işlem, eğitim verilerinin çeşitliliğini artırmak için kullanılır. Bu işlemin sonrasında veriler normalize edilir. Normalizasyon, görüntü piksellerinin 0 ila 1 aralığında olmasını sağlar ve modelin daha hızlı öğrenmesine yardımcı olur.

Son olarak, eğitim ve test verileri, mapping dosyası da kullanılarak “Bölüm 3” altında anlatılan CNN modeli üzerinde eğitilmiştir. Eğitim sonucunda harf tahmininde kullanılacak olan H5 uzantılı “letters_model” modeli ve CSV uzantılı “emnist-letters-mapping” isimli bir mapping dosyası elde edilir. Bu dosyasındaki değerlerin, ASCII karakter tablosuna göre kaymış olduğu görülmüştür. Bu nedenle orijinal dosyada “Z” harfi bulunmamaktadır. Etiketlendirme işlemi için “91, Z” değeri eklenerek dosya güncellenmiştir.

```
0,Character
65,@
66,A
67,B
68,C
69,D
70,E
71,F
72,G
73,H
74,I
75,J
76,K
77,L
78,M
79,N
80,O
81,P
82,Q
83,R
84,S
85,T
86,U
87,V
88,W
89,X
90,Y
91,Z
```

Şekil 3. 9 CSV Dosyası İçeriği

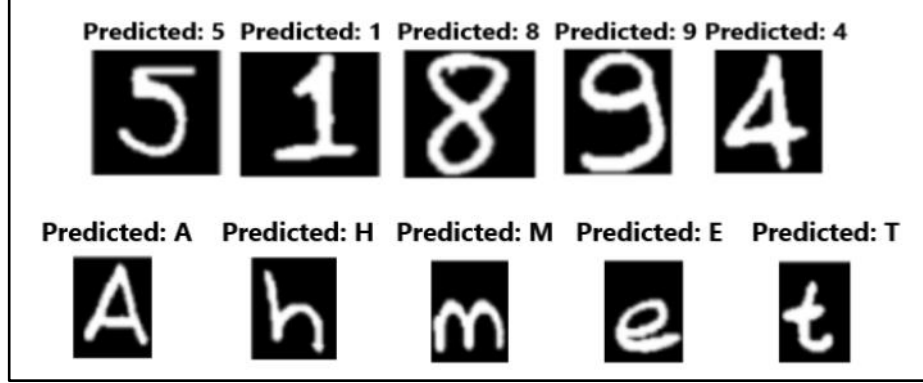
Şekil 3.9’da tahmin işleminde kullanılan “emnist-letters-mapping.csv” dosyasının güncellenmiş içeriği verilmiştir. Bu dosya, tahmin işleminde ASCII değerlerinden karakter etiketlerinin elde edilmesi için decode edilerek kullanılmıştır.

3.2.5. Sınıflandırma

Karakterleri içeren bileşenler harf ve rakam etiketinin tahminleri için aynı ön işleme adımlarına tabi tutulmuştur. Rakam tanıma işlemi için H5 uzantılı “numbers_model” modeli, harf tahmin işlemi için ise H5 uzantılı “letters_model” modeli kullanılmıştır.

Ayrıca harf tanıma işleminde model eğitimi sonucunda elde edilen mapping içeriği “Bölüm 3” altında anlatıldığı gibi değiştirilerek kullanılmıştır.

Şekil 3.10’da harf ve rakam için sınıflandırma ile yapılan etiket tahminleri verilmiştir.



Şekil 3. 10 Sınıflandırma ile Etiket Tahmini

3.2.6. Birleştirme

Parçalanmış karakterlerin etiket tahminleri yapıldıktan sonra elde edilen etiketler bir liste içerisinde toplanır. Bu liste içerisindeki etiketler birleştirilerek kelimenin parçalanmadan önceki hali elde edilmiş olur. Sonrasında bu kelimeler şablonlara göre oluşturulan python sözlüğü yapılarında saklanır.

```
{ 'ID: ': '51894', 'Name: ': 'Ahmet' }
```

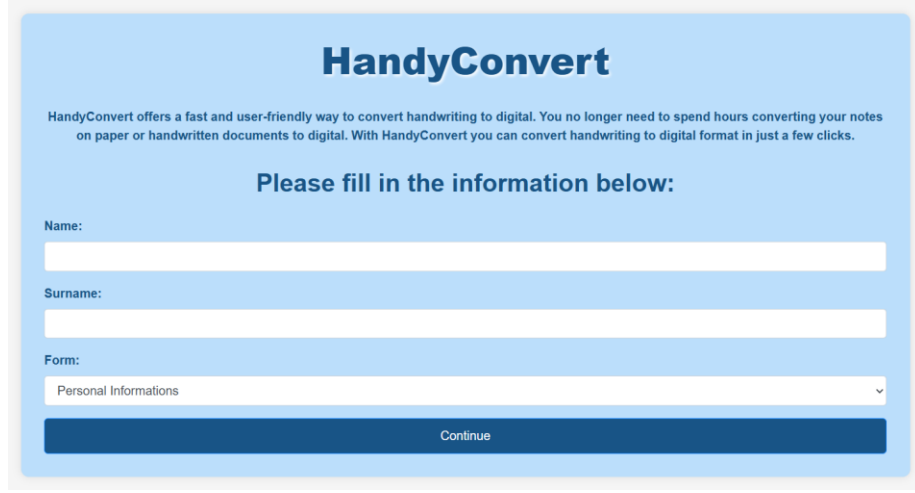
Şekil 3. 11 Örnek Python Sözlüğü

Şekil 3.11’de birleştirme sonucu elde edilen örnek python sözlüğü verilmiştir. Harf sınıflandırma işleminde etiketler sadece büyük harflerden oluştuğundan tahmin edilen etiketler birleştirildikten sonra “title()” metodu ile ilk harfi büyük sonraki harfler küçük olacak şekilde düzenlenmiştir.

Sisteme birden fazla kelime bulunan bir satır verildiğinde birleştirme aşamasında satırdaki bütün karakterler tek bir kelime olarak birleştirilmektedir, yani kelime segmentasyonu sağlanamamaktadır.

3.2.7. Flask Web Uygulaması

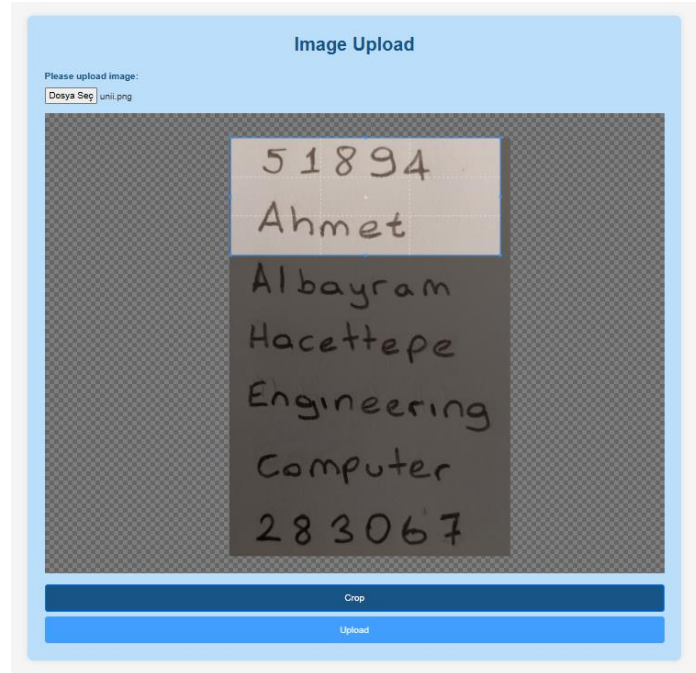
Kullanıcılara kullanım kolaylığı sağlamak açısından Flask Framework kullanılarak bir web uygulaması geliştirilmiştir. Uygulama çalıştırıldığında ilk olarak Şekil 3.12'deki sayfa açılır. Bu sayfada kullanıcıdan ad, soyad bilgilerinin girilmesi ve şablon seçimi yapılması istenir.



The image shows the 'HandyConvert' web interface. At the top, the title 'HandyConvert' is displayed in a large, bold, blue font. Below the title, a paragraph explains the service: 'HandyConvert offers a fast and user-friendly way to convert handwriting to digital. You no longer need to spend hours converting your notes on paper or handwritten documents to digital. With HandyConvert you can convert handwriting to digital format in just a few clicks.' Below this, a prompt says 'Please fill in the information below:'. The form consists of three input fields: 'Name:', 'Surname:', and 'Form:'. The 'Form:' field is a dropdown menu with 'Personal Informations' selected. At the bottom of the form is a blue button labeled 'Continue'.

Şekil 3. 12 Web Arayüzü 1

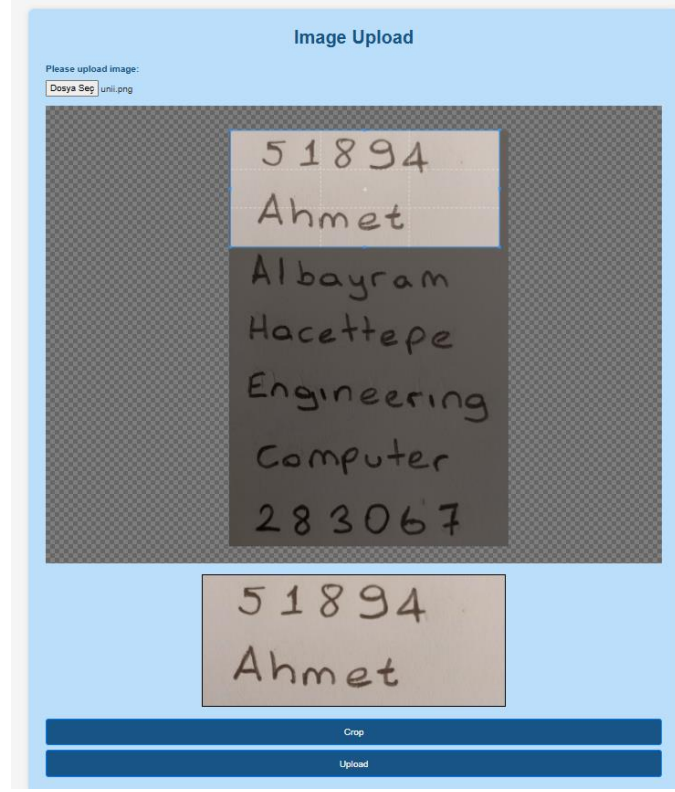
Bu işlemler yapıldıktan sonra “continue” butonu ile Şekil 3.13'teki sayfaya geçilir. Bu kısımda kullanıcıdan el yazısı ile doldurulan form fotoğrafını yüklemesi istenir.



The image shows the 'Image Upload' web interface. At the top, the title 'Image Upload' is displayed in a large, bold, blue font. Below the title, a prompt says 'Please upload image:'. There is a file upload button labeled 'Dosya Seç' and a text input field for the file name, which contains 'untitled.png'. Below the upload area is a large image of a handwritten document. The document contains the following text: '51894', 'Ahmet', 'Albayram', 'Hacettepe', 'Engineering', 'Computer', and '283067'. At the bottom of the image are two blue buttons: 'Crop' and 'Upload'.

Şekil 3. 13 Web Arayüzü 2

Yüklenen fotoğraf üzerinde kırpılmak istenen kısım seçilerek “crop” butonuna basılır ve Şekil 3.14’teki gibi sayfada fotoğrafın kırpılmış hali gösterilir.



Şekil 3. 14 Web Arayüzü 3

Kırpılan fotoğraf “upload” butonu ile yüklenip Şekil 3.15’teki sonuç sayfasına geçilir. Bu sayfada da kullanıcıdan ilk sayfada alınan bilgiler ile “Bölüm 3” altında anlatıldığı gibi elde edilen python sözlük yapısı bir tablo şeklinde ekrana yazdırılır. Ayrıca sözlükteki bu bilgiler bir PDF dosyasına da tablo şeklinde kaydedilir.



Şekil 3. 15 Web Arayüzü 4

BÖLÜM 4. BULGULAR

Çalışmanın bu kısmında “Bölüm 3” altında anlatılan CNN modeli, MNIST ve EMNIST veri setleri kullanılarak eğitilmiştir. Eğitilen bu modelin başarısı yine “Bölüm 3” altında anlatılan performans metriklerine göre değerlendirilmiştir. Oluşturulan sistem test edilmiştir ve test sonucunda elde edilen tahmin çıktıları ile başarı değerleri sunulmuştur. Ayrıca web uygulaması üzerinden örnek uygulamalar da sunulmuştur.

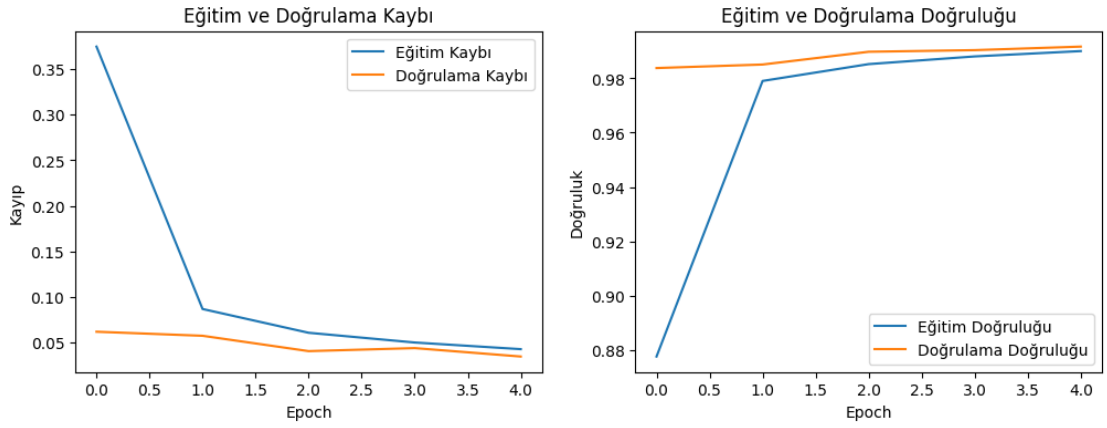
4.1. MNIST Modeli İçin Değerlendirmeler

Tablo 2’de MNIST veri seti için performans değerlendirme sonuçları verilmiştir.

Tablo 2 MNIST Modeli Performans Değerlendirme Sonuçları

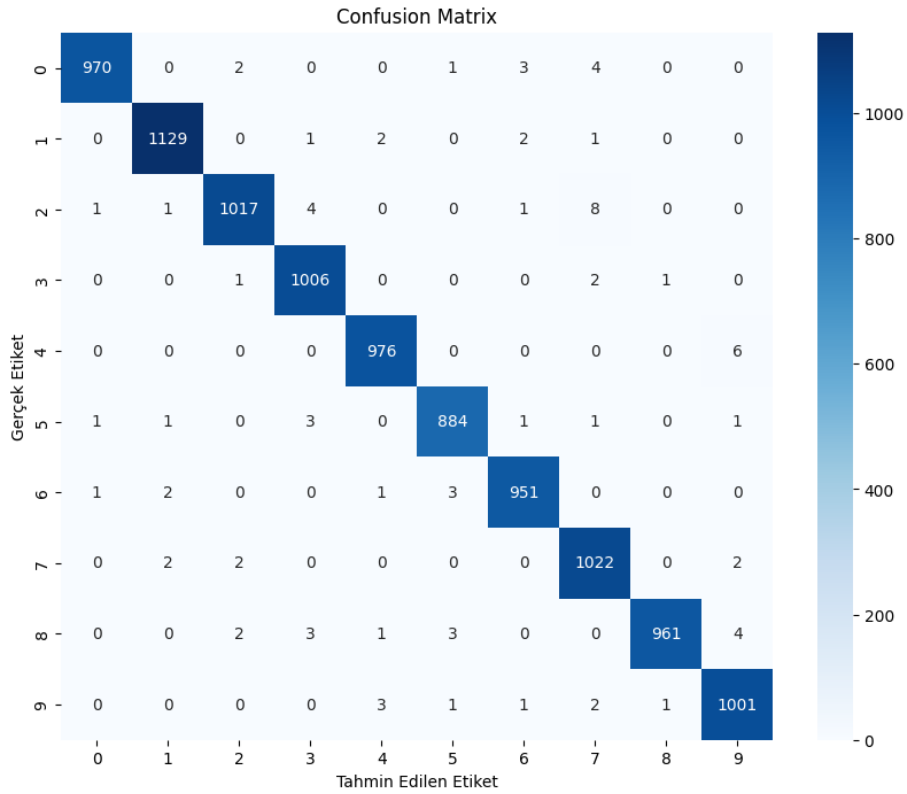
| Performans Metrikleri | MNIST |
|-----------------------|--------|
| Precision | 0.9903 |
| Recall | 0.9903 |
| F1 Score | 0.9902 |
| Test Accuracy | 0.9903 |
| Train Accuracy | 0.9913 |

Şekil 4.1’de MNIST veri seti için eğitim sonucunda elde edilen doğruluk ve kayıp grafikleri verilmiştir. Grafikler incelendiğinde eğitim ve doğrulama verilerinin 5 epochluk sonuçları için kayıp değerlerinin azaldığı, doğruluk değerlerinin ise arttığı gözlemlenmektedir.



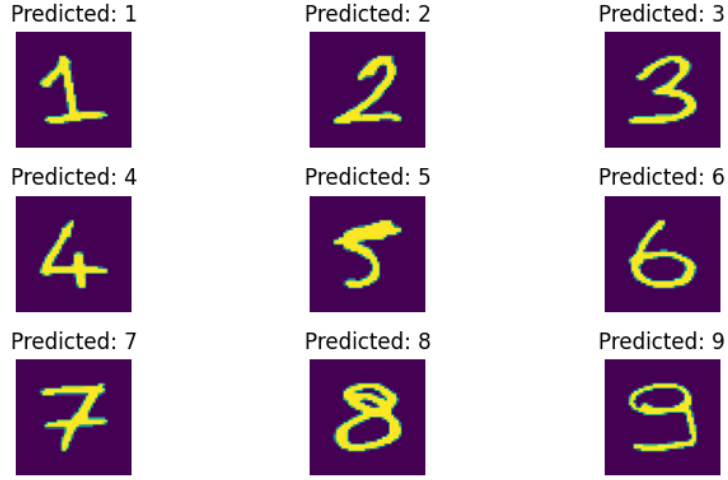
Şekil 4. 1 MNIST Modeli Doğruluk ve Kayıp Grafiği

Şekil 4.2’de MNIST test verileri ile oluşturulan Karışıklık Matrisi (Confusion Matrix) verilmiştir. Tahmin edilen etiketlerin neredeyse tamamının gerçek etiketleri ile aynı olduğu görülmektedir.



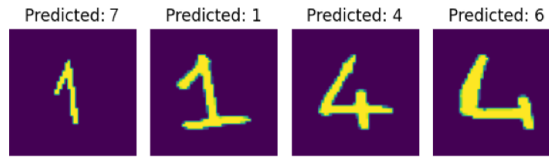
Şekil 4. 2 MNIST Modeli Karışıklık Matrisi

Şekil 4.3'te MNIST modeli kullanılarak el yazısı ile kağıt üzerine yazılmış rakamların tahmin sonuçları verilmiştir. Tüm rakamların doğru tahmin edildiği görülmektedir.



Şekil 4. 3 Rakam Tahminleri

Şekil 4.4'te verilen örnekteki gibi el yazısı ile kağıt üzerine yazılmış “1” ve “4” rakamlarının farklı şekillerde yazılışlarında modelin yanlış tahmin yapabildiği görülmüştür.



Şekil 4. 4 Yanlış Rakam Tahminleri

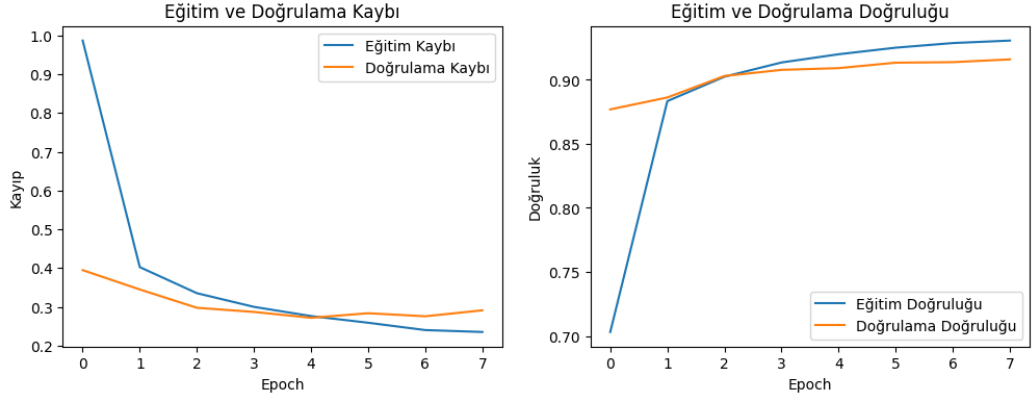
4.2. EMNIST Modeli İçin Değerlendirmeler

Tablo 3'te EMNIST veri seti için performans değerlendirme sonuçları verilmiştir.

Tablo 3 EMNIST Modeli Performans Değerlendirme Sonuçları

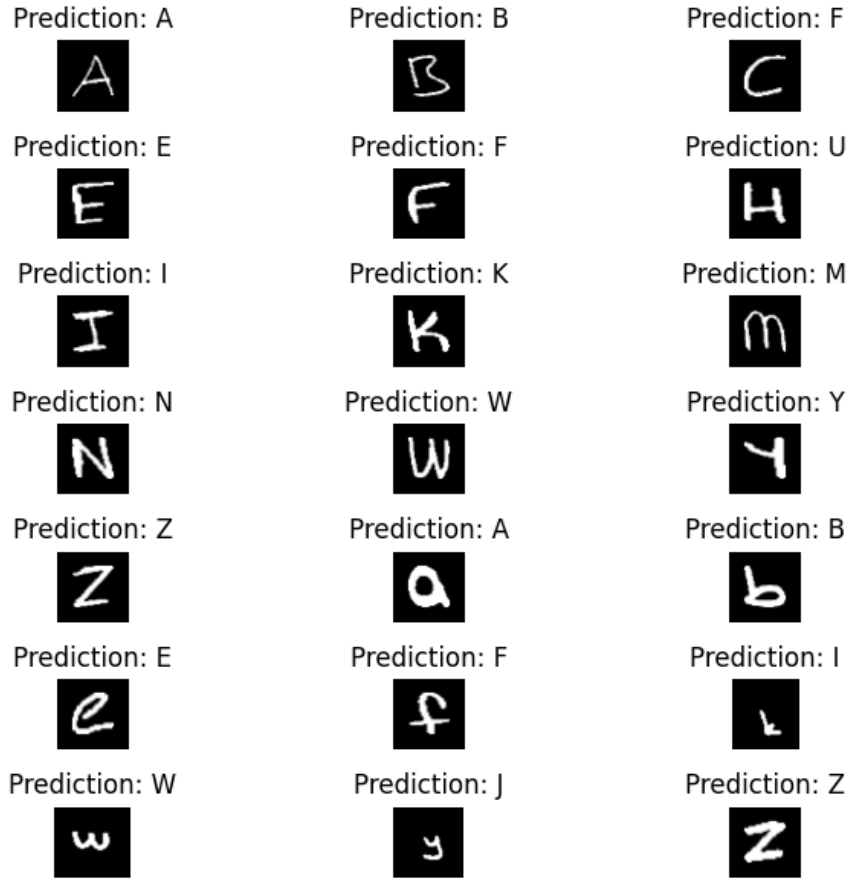
| Performans Metrikleri | EMNIST |
|-----------------------|--------|
| Precision | 0.9221 |
| Recall | 0.9091 |
| F1 Score | 0.9121 |
| Test Accuracy | 0.9091 |
| Train Accuracy | 0.9316 |

Şekil 4.5'te EMNIST veri seti için eğitim sonucunda elde edilen doğruluk ve kayıp grafikleri verilmiştir. Grafikler incelendiğinde eğitim ve doğrulama verilerinin 8 epochluk sonuçları için kayıp değerlerinin azaldığı, doğruluk değerlerinin ise arttığı gözlemlenmektedir.



Şekil 4. 5 EMNIST Modeli Doğruluk ve Kayıp Grafikleri

Şekil 4.6'da EMNIST modeli kullanılarak el yazısı ile kağıt üzerine yazılmış bazı büyük ve küçük harflerin tahmin sonuçları verilmiştir. Tahmin sonuçlarının çoğunun doğru olduğu görülmüştür ancak bazı harflerde, yazılış şekillerinden dolayı model yanlış tahminlerde bulunabilmektedir.



Şekil 4. 6 Harf Tahminleri

EMNIST veri seti, 26 farklı karakter içeren İngiliz alfabesinden oluşturulduğu için etiketler sıralı olarak 1-25 aralığında verilmektedir. Örnek olarak “A” harfi “1”, “Z” harfi ise “25” olarak etiketlenir, bu etiketler “Bölüm 3” altında anlatılan “mapping.csv” dosyasına göre belirlenmiştir.

4.3. Sistemin Test Edilmesi

Bu kısımda geliştirilen web uygulamasının test edilmesi adım adım görsellerle sunulmuştur.

The image shows a web application interface for 'HandyConvert'. At the top, the title 'HandyConvert' is displayed in a large, bold, dark blue font. Below the title, a paragraph of text explains the service: 'HandyConvert offers a fast and user-friendly way to convert handwriting to digital. You no longer need to spend hours converting your notes on paper or handwritten documents to digital. With HandyConvert you can convert handwriting to digital format in just a few clicks.' This text is in a smaller, dark blue font. Below the text, a prompt 'Please fill in the information below:' is centered. Underneath this prompt, there are three input fields. The first is labeled 'Name:' and contains the text 'Henzkta'. The second is labeled 'Surname:' and contains the text 'Parker'. The third is labeled 'Form:' and is a dropdown menu with 'University Informations' selected. At the bottom of the form, there is a dark blue button with the text 'Continue' in white.

Şekil 4. 7 Giriş Sayfası

Şekil 4.7’de giriş sayfasında kullanıcı, istenilen bilgileri doldurmuş ve üniversite bilgilerini içeren “University Informations” formunu seçmiştir.

Image Upload

Please upload image:

**SAKARYA UNIVERSITY OF
APPLIED SCIENCES**

University Informations Form

| | |
|----------------|---------------|
| ID | 44 2576901183 |
| Name | Henzkta |
| Surname | Parker |
| University | Stanford |
| Faculty | Sciences |
| Department | Computer |
| Student Number | 3087746280041 |

44 2576901183
Henzkta
Parker
Stanford
Sciences
Computer
3087746280041

Şekil 4. 8 Fotoğraf Yükleme ve Kırpma Sayfası

Kullanıcı Şekil 4.8’deki sayfada el yazısı ile doldurduğu formun fotoğrafını seçmiş ve sadece el yazısı olan kısmı kırparak sisteme yüklemiştir.

HandyConvert

Result

Name: Henzhta
Surname: Parker
Form: University Informations

Informations:

| |
|--|
| ID: : 442576101783 |
| Name: : Henzhta |
| Surname: : Parker |
| University: : Stanford |
| Faculty: : Sctences |
| Department: : Computer |
| Student Number: : 3087746280061 |

Şekil 4. 9 Sonuç Sayfası

Şekil 4.9'daki sonuç sayfasında el yazısı kısmının tahminlerini içeren ve seçilen forma göre oluşturulmuş sözlük yapısı bir tablo şeklinde ekranda gösterilmiştir. Bazı karakter tahminlerinin yanlış yapılabilirdiği görülmektedir. Örneğin el yazısıyla yazılmış "Sciences" kelimesinde "i" harfi "t" olarak tahmin edilmiştir. Başka bir örnek ise el yazısı ile yazılmış "3087746280041" sayısında sondan ikinci rakam olan "4" rakamı "6" olarak tahmin edilmiştir. Bu yanlış tahminler, yazılış farklılıklarından veya fotoğraf çekim açısından kaynaklanabilmektedir.

HandyConvert Result

Form Type: University Informations

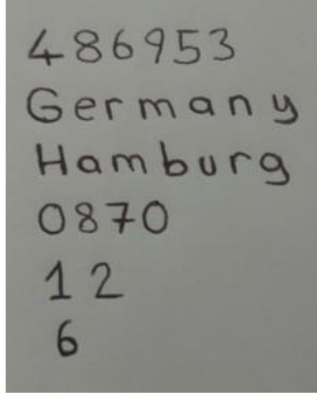
| | |
|-----------------|---------------|
| ID: | 442576101783 |
| Name: | Henzhta |
| Surname: | Parker |
| University: | Stanford |
| Faculty: | Sctences |
| Department: | Computer |
| Student Number: | 3087746280061 |

Şekil 4. 10 PDF İçeriği

Ayrıca sözlük yapısı bir tablo halinde bir PDF dosyasına yazdırılarak kaydedilir. Şekil 4.10’da bu PDF dosyasının içeriği gösterilmiştir.

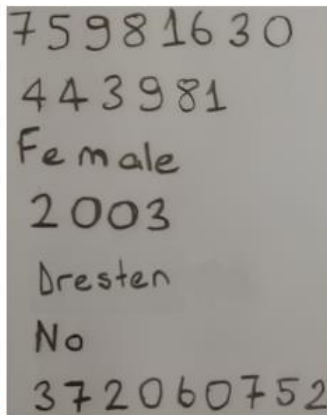
Oluşturulan diğer iki formun da testleri gerçekleştirilmiştir. Şekil 4.11’de adres bilgilerini içeren “Address Informations” formuna göre test sonucu verilmiştir. Şekil 4.12’de ise kişisel bilgileri içeren “Personal Informations” formuna göre test sonucu verilmiştir.

| Form: Address Informations |
|----------------------------|
| Informations: |
| ID: : 486953 |
| Country: : Germany |
| City: : Homburg |
| Zipcode: : 0870 |
| Street Number: : 12 |
| Apartment Number: : 6 |



Şekil 4. 11 Adres Bilgisi Formu Sonucu

| Form: Personal Informations |
|---------------------------------|
| Informations: |
| ID: : 75181630 |
| Identification Number: : 443981 |
| Gender: : Female |
| Birth Year: : 2003 |
| Birthplace: : Dresten |
| Disease: : No |
| Phone Number: : 372060752 |



Şekil 4. 12 Kişisel Bilgi Formu Sonucu

BÖLÜM 5. TARTIŞMA VE SONUÇ

Tartışma ve sonuç kısmında çalışma süresince elde edilen sonuçlar değerlendirilip çalışmanın eksikleri ve gelecek planları sunulmuştur. Bu çalışmada el yazısı rakam ve harfleri tanıyan bir sistem oluşturulmuştur ve bu sistem bir web uygulamasına entegre edilmiştir. Bu sistemde bir derin öğrenme yöntemi olan CNN modeli, MNIST ve EMNIST veri setleri kullanılarak eğitilmiştir. Eğitim sonuçları değerlendirildiğinde model başarısının yüksek olduğu ve karakter tanıma işleminde etiket tahminlerinin çoğunlukla doğru olduğu görülmüştür.

Sonuç olarak çalışmanın eksikleri ve gelecek planları şu şekilde özetlenebilir:

1. Bitişik olarak yazılmış karakterlerin ve noktalı karakterlerin parçalanması doğru bir şekilde gerçekleştirilememektedir. Karakter segmentasyon yöntemi geliştirilerek bu sorunların çözülmesi hedeflenmektedir.
2. Bir satır birden fazla kelime içerdiğinde bu kelimelerin birleştirme aşamasında ayrılamaması sorunu yaşanmaktadır. Kelime segmentasyonu geliştirilerek bu sorunun çözülmesi hedeflenmektedir.
3. Yapılan web uygulamasına yüklenen resimlerin çözünürlüğünün yüksek olması yapılacak işlemler açısından kolaylık sağlamaktadır. Bu sebeple uygulamaya fotoğraf yükleme sırasında çözünürlük artırma işlevi eklenmesi hedeflenmektedir.
4. Sistemde şu an üç farklı form şablonu ile işlemler yapılabilmektedir. Oluşan ihtiyaçlara göre bu formların sayısı artırılabilir ve sistem bu formlara göre düzenlenebilir.

KAYNAKLAR

- [1] Baldominos, A.; Saez, Y.; Isasi, P. A Survey of Handwritten Character Recognition with MNIST and EMNIST. *Appl. Sci.* 2019.
- [2] TOĞAÇAR, Mesut. "Harflerden Oluşan Genişletilmiş MNIST Veri Kümesinin Derin Öğrenme Tabanlı Tasarlanmış Sinir Ağı Modeli ile Sınıflandırılması." *Çukurova Üniversitesi Mühendislik Fakültesi Dergisi* 36.3 (2021): 681-690.
- [3] Jebadurai, Jebaveerasingh, et al. "Handwritten text recognition and conversion using convolutional neural network (CNN) based deep learning model." *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*. IEEE, 2021.
- [4] Méndez, M., Merayo, M.G. & Núñez, M. Machine learning algorithms to forecast air quality: a survey. *Artif Intell Rev* 56, 10031–10066 (2023).
- [5] Shinde, Pramila P., and Seema Shah. "A review of machine learning and deep learning applications." *2018 Fourth international conference on computing communication control and automation (ICCUBE)*. IEEE, 2018.
- [6] Wang, Pin, En Fan, and Peng Wang. "Comparative analysis of image classification algorithms based on traditional machine learning and deep learning." *Pattern Recognition Letters* 141 (2021): 61-67.
- [7] Salouhou, Aoudou. *El yazısı karakter tanıma ve resim sınıflandırmada derin öğrenme yaklaşımları*. MS thesis. Fatih Sultan Mehmet Vakıf Üniversitesi, Lisansüstü Eğitim Enstitüsü, 2019.
- [8] Pradeep, Jayabala, E. Srinivasan, and S. Himavathi. "Neural network based handwritten character recognition system without feature extraction." *2011 international conference on computer, communication and electrical technology (ICCCET)*. IEEE, 2011.
- [9] Karakaya, Rabia. Makine öğrenmesi yöntemleriyle el yazısı tanıma. MS thesis. Sakarya Üniversitesi, 2020.

- [10] DAĞDEVİREN, Bil Müh Engin, and Zeynep ORMAN. "EL YAZISI RAKAM TANIMA İÇİN DESTEK VEKTÖR MAKİNELERİNİN VE YAPAY SİNİR AĞLARININ KARŞILAŞTIRMASI." (2013).
- [11] Kannoja, Suresh Prasad, and Gaurav Jaiswal. "Effects of varying resolution on performance of CNN based image classification: An experimental study." *Int. J. Comput. Sci. Eng* 6.9 (2018): 451-456.
- [12] Harris, C.R., Millman, K.J., van der Walt, S.J. et al. Array programming with NumPy. *Nature* 585, 357–362 (2020).
- [13] Pang, B., Nijkamp, E., & Wu, Y. N. (2020). Deep Learning With TensorFlow: A Review. *Journal of Educational and Behavioral Statistics*, 45(2), 227-248.
- [14] Ahmet, Ü. N. A. L., Ü. N. A. L. Esra, and Dilcan GÜLER. "EVRIŞİMLİ SİNİR AĞLARI KULLANARAK OPTİK KARAKTER TANIMA." *Journal of Business in The Digital Age* 6.1 (2023): 1-12.
- [15] Dölek, İshak. Yapay sinir ağlarıyla optik karakter tanımı kullanılarak günümüz Türkçesinin Osmanlıcaya çevrilmesi. Diss. Sakarya Üniversitesi (Turkey), 2016.
- [16] Yousefi, Jamileh. "Image binarization using Otsu thresholding algorithm." Ontario, Canada: University of Guelph 10 (2011).
- [17] Kadam, Shivam S., Amol C. Adamuthe, and Ashwini B. Patil. "CNN model for image classification on MNIST and fashion-MNIST dataset." *Journal of scientific research* 64.2 (2020): 374-384.
- [18] Şekerci, Murat, and Rembiye Kandemir. "BİRLEŞİK VE EĞİK TÜRKÇE EL YAZISI TANIMADA K-NN SINIFLAMA YÖNTEMİ VE SÖZLÜK KULLANIMI." *Trakya Üniversitesi Fen Bilimleri Dergisi* 10.1 (2009): 97-102.
- [19] Goyani, M., Dani, H., & Dixit, C. (2013). Handwritten Character Recognition-A Comprehensive Review. *International Journal of Research in Computer and Communication Technology*, 2(9), 702-707.
- [20] Djaghbello, S., Abdelouahab, A., & Abderraouf, B. (2023). A survey on text-line segmentation process in historical Arab manuscripts.
- [21] Boraik, O. A., Ravikumar, M., & Saif, M. A. N. (2022). Characters segmentation from arabic handwritten document images: Hybrid approach. *International Journal of Advanced Computer Science and Applications*, 13(4).

EK-1

TABLO 3, İŞ ZAMAN ÇİZELGESİ

| İş Paketi | Kimler Tarafından Yapıldı | Hafta Aralığı |
|---------------------------------------|---------------------------|---------------|
| Konu Seçimi | A, H | 1-2 |
| Literatür Taraması | A, H | 2-5 |
| Veri Seti Seçilmesi | A, H | 5-6 |
| Sistem Tasarımı | A, H | 6-9 |
| Sınıflandırma Modelinin Oluşturulması | A, H | 9-11 |
| Modelin Test Edilmesi | A, H | 11-13 |
| Sistemin İyileştirilmesi | A, H | 12-14 |
| Web Uygulamasının Geliştirilmesi | A, H | 13-14 |
| Rapor Düzenleme | A, H | 13-15 |

A: Ayşegül TOPTAŞ

H: Havvanur BOZKURT

EK-2

TABLO 4, RİSK ANALİZİ

| Risk | Risk Yönetimi |
|-----------------------------------|--------------------------------|
| Model başarısının düşük olması | Farklı modeller denenecektir. |
| Sistem aşamalarında hata oluşması | Sistem yeniden planlanacaktır. |

ÖZGEÇMİŞ

Ayşegül TOPTAŞ, 13.12.2000'de Kocaeli'de doğdu. İlk, orta ve lise eğitimini Kocaeli'de tamamladı. 2019 yılında 24 Kasım Anadolu Lisesi'nden mezun oldu. 2020 yılında Sakarya Uygulamalı Bilimler Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği Bölümü'ne başladı. Mesleki ilgi alanları arasında mühendislik yazılımları, yapay zeka, bilgisayarlı görü yer almaktadır.

Havvanur BOZKURT, 10.05.2002'de Kocaeli'de doğdu. İlk, orta ve lise eğitimini Kocaeli'de tamamladı. 2020 yılında Cahit Elginkan Anadolu Lisesi'nden mezun oldu. 2020 yılında Sakarya Uygulamalı Bilimler Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği Bölümü'ne başladı. Mesleki ilgi alanları arasında mühendislik yazılımları, yapay zeka, görüntü işleme, bilgisayarlı görü yer almaktadır.

STANDARTLAR VE KISITLAR FORMU

1. Çalışmanın amacını özetleyiniz.

Kağıt tabanlı verilerin dijital ortama aktarılması, verilerin saklanması ve analizi gibi çeşitli alanlarda da kolaylık sağlar. Bu amaçlar doğrultusunda kağıt üzerindeki el yazısını dijitalleştirmek için bir sistem geliştirilmiştir. Geliştirilen bu sistem web uygulamasına entegre edilmiştir.

2. Çalışmanın tasarım boyutunu açıklayınız.

Sunulan tasarım yeni bir projedir. Tasarlanan projenin tamamı gerçekleştirilmiştir.

3. Bu çalışmada bir mühendislik problemini kendiniz formüle edip, çözdünüz mü?

Sınıflandırma aşamasında özel olarak oluşturulan veri setindeki karakterler girdi olarak verildiğinde sınıflandırılamıyordu. Yapılan incelemeler sonrasında bu verilerin eğitim verileriyle aynı formatta olmadığı ve bunun da ön işleme adımından kaynaklandığı görülmüştür. Bu sorun, ön işleme için Otsu'nun Binarizasyonu kullanılarak çözülmüştür.

Etiketlendirme aşamasında, tahmin edilen etiketlerde gerçek etiketlere göre kayma olduğu tespit edilmiştir. Eğitim sonrasında etiketlendirme işlemi için oluşturulan CSV dosyasının içeriği güncellenerek bu sorun çözülmüştür.

4. Çalışmada kullandığınız yöntemler nelerdir ve önceki derslerde edindiğiniz hangi bilgi ve becerileri kullandınız? Açıklayınız.

Yapay zeka üzerine alınan derslerden derin öğrenme yöntemleri ile sınıflandırma becerileri kullanıldı.

Görüntü işleme dersinden ön işlemede kullanılan yöntemler konusunda yararlanıldı.

Web programlama dersinden web uygulamasının tasarımı konusunda yararlanıldı.

5. Kullandığınız veya dikkate aldığınız mühendislik standartları nelerdir?

Veri setlerinin seçimi, temizlenmesi ve standartlara uygunluğu projenin kalitesi açısından önemlidir. Bu nedenle “Veri Standardizasyonu” dikkate alınmıştır.

Sunulan rapor, projenin amaçları, kullanılan yöntemler, elde edilen sonuçlar, eksiklikler ve gelecekteki planlar gibi detayları içerdiğinden “Dokümantasyon ve Raporlama Standardizasyonu” dikkate alınarak hazırlanmıştır.

6. Kullandığınız veya dikkate aldığınız gerçekçi kısıtlar nelerdir? Lütfen çalışmanıza uygun yanıtlarla doldurunuz.

a) Ekonomi:

b) Çevre sorunları:

c) Sürdürülebilirlik:

Eldeki veri setlerinin sınırlılığı ve çeşitliliği, bu teknolojinin genelleme yeteneğini ve farklı el yazılarını tanıma performansını kısıtlayabilir. Bunun yanı sıra, projenin gerçekleştirilen kısmının sınıflandırma odaklı olması ve segmentasyon, birleştirme gibi adımları içermemesi, gerçek dünya koşullarında karşılaşılabilecek zorlukları yansıtmayabilir. Bu sebeplerle proje sürdürülebilirlik açısından kısıtlar içermektedir.

d) Üretilebilirlik:

Veri temini, altyapı oluşturulması gibi faktörler üretilebilirlik açısından bu teknolojinin yaygınlaşmasını ve geniş ölçekli kullanımını kısıtlayabilir.

e) Etik:

f) Sağlık:

g) Güvenlik:

Kişisel ve hassas bilgilerin dijitalle aktarılması söz konusu olduğundan, güvenlik riskleri önemlidir. Bu süreçte verilerin yetkisiz erişimden korunması gereklidir. Verilerin şifrelenmesi, transfer edilirken veya depolanırken güvenliğini sağlamak için önemlidir. Ayrıca, veri bütünlüğüne de dikkat edilmelidir; veri kaybı veya bozulma gibi durumlar bütünlüğü tehlikeye atabilir. Bu sebeplerle proje güvenlik açısından kısıtlar içermektedir.

h) Sosyal ve politik sorunlar:

| | |
|--------------------------------|---|
| Çalışmanın Adı | DERİN ÖĞRENME YÖNTEMİ KULLANILARAK EL YAZISININ DİJİTALE ÇEVİRİLMESİ |
| Çalışmayı Hazırlayanlar | Ayşegül TOPTAŞ Havvanur BOZKURT |
| Danışman Onayı | Muhammed Ali Nur ÖZ |