**T.C.**

SAKARYA UYGULAMALI BİLİMLER ÜNİVERSİTESİ   
TEKNOLOJİ FAKÜLTESİ

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

DERİN ÖĞRENME YÖNTEMİ KULLANILARAK EL YAZISININ DİJİTALE ÇEVRİLMESİ

Bilgisayar Mühendisliği Tasarımı 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 YAZISININ DİJİTALE ÇEVRİLMESİ

Bilgisayar Mühendisliği Tasarımı 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

13.06.2024

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](#_Toc155810554)

[İÇİNDEKİLER………………………………………………………………………ii](#_Toc155810555)

[SİMGELER VE KISALTMALAR LİSTESİ iv](#_Toc155810556)

[ŞEKİLLER LİSTESİ………………………………………………………….……...v](#_Toc155810557)

[TABLOLAR LİSTESİ……………………………………………..………………...vi](#_Toc155810558)

[ÖZET……………………………………………………………………………......vii](#_Toc155810559)

[ABSTRACT………………………………………………………………………..viii](#_Toc155810560)

[BÖLÜM 1.   
GİRİŞ………………………………………………………………………………...9](#_Toc155810561)

[BÖLÜM 2.  
ÖNCEKİ ÇALIŞMALARIN ARAŞTIRILMASI………………………………... ....11](#_Toc155810562)

[2.1. El Yazısı Karakter Tanıma ve Resim Sınıflandırmada Derin Ögrenme Yaklaşımları…………………………………………………………….11](#_Toc155810563)

[2.2. Handwritten Text Recognition and Conversion Using Convolutional Neural Network (CNN) Based Deep Learning Model 11](#_Toc155810564)

[2.3. Neural Network Based Handwritten Character Recognition System Without Feature Extraction 12](#_Toc155810565)

[2.4. Makine Öğrenmesi Yöntemleriyle El Yazısı Tanıma 12](#_Toc155810566)

[2.5. El Yazısı Rakam Tanıma İçin Destek Vektör Makinelerinin Ve Yapay Sinir Ağlarının Karşılaştırması 13](#_Toc155810567)

[BÖLÜM 3.   
YÖNTEM VE TASARIM 14](#_Toc155810568)

[3.1. Materyal……………...............................................................................14](#_Toc155810569)

[3.1.1. MNIST (Mixed National Institute of Standards and Technology) 14](#_Toc155810570)

[3.1.2. EMNIST (Extended MNIST) 15](#_Toc155810571)

[3.1.3. Kütüphaneler 15](#_Toc155810572)

[3.2. Yöntem………………………………………………………….............16](#_Toc155810573)

[3.2.1. Verinin Eldesi ve Segmentasyonu 16](#_Toc155810574)

[3.2.2. Sınıflandırma İçin Ön İşleme 17](#_Toc155810575)

[3.2.3. Sınıflandırma 18](#_Toc155810576)

[3.2.3.1. CNN (Convolutional Neural Network) 18](#_Toc155810577)

[BÖLÜM 4.   
BULGULAR………………………………………………………………………...21](#_Toc155810578)

[4.1. MNIST Veri Seti İçin Elde Edilen Bulgular 21](#_Toc155810579)

[4.2. EMNIST Veri Seti İçin Elde Edilen Bulgular 24](#_Toc155810580)

[BÖLÜM 5.   
TARTIŞMA VE SONUÇ 25](#_Toc155810581)

[KAYNAKLAR…………………….. 26](#_Toc155810582)

[EK-1…………………………………………………………………………………28](#_Toc155810583)

[EK-2…………………………………………………………………………………28](#_Toc155810584)

[ÖZGEÇMİŞ……………………………………………………………………........29](#_Toc155810585)

SİMGELER VE KISALTMALAR LİSTESİ

|  |  |
| --- | --- |
| CNN  CTC  DNN  EMNIST  IAM  LSTM | : Convolutional Neural Network  : Connectionist Temporal Classification  : Deep Neural Network  : Extended MNIST  : Identity and Access Management  : Long Short-Term Memory |
| MNIST  Numpy  RGB  RNN  SVM  YSA | : Modified National Institute of Standards and Technology  : Numerical Python  : Red, Green, Blue  : Recurrent Neural Network  : Support Vector Machine  : Yapay Sinir Ağı |

ŞEKİLLER LİSTESİ

[Şekil 3. 1 MNIST Veri Seti Örnekleri [1]. 14](#_Toc155810514)

[Şekil 3. 2 EMNIST Veri Seti Örnekleri [1]. 15](#_Toc155810515)

[Şekil 3. 3 Sistemin Genel Çalışma Şeması 16](#_Toc155810516)

[Şekil 3. 4 Segmentasyon Örneği 17](#_Toc155810517)

[Şekil 4. 1 MNIST Test Verisi Örnekleri ve Tahmin/Gerçek Etiketleri 22](#_Toc155810519)

[Şekil 4. 2 El Yazısı Örneklerinin Orijinal ve Normalizasyon Sonrası Görüntüleri 22](#_Toc155810520)

[Şekil 4. 3 El Yazısı Örneklerinin Orijinal ve Ön İşleme Sonrası Görüntüleri 23](#_Toc155810521)

[Şekil 4. 4 Ön İşleme Sonrasında Elde Edilen Görüntülerin Sınıflandırma Sonucundaki Etiket Tahminleri 23](#_Toc155810522)

[Şekil 4. 5 EMNIST Veri Setinin Görselleştirilmiş Örnekleri 24](#_Toc155810523)

[Şekil 4. 6 EMNIST Test Verilerinden Örnek Görüntüler ve Etiket Tahminleri 24](#_Toc155810524)

TABLOLAR LİSTESİ

[Tablo 1 CNN Mimarisi Özeti 19](#_Toc155809984)

[Tablo 2 MNIST ve EMNIST Veri Setleri Performans Metrikleri Sonuçları 21](#_Toc155809985)

ÖZET

Anahtar kelimeler: El yazısı tanıma, harf, rakam, dijital, makine öğrenmesi, derin öğrenme

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 dijitalleştirmek üzerine yapılan çalışmalar incelenerek bir sistem sunulmuştur. Bu sistem segmentasyon, ön işleme, sınıflandırma ve birleştirme gibi farklı aşamaları içermektedir. Bu aşamalarda girdi olarak verilen el yazısı analiz edilerek dijitale 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.

Bu proje kapsamında segmentasyon ve birleştirme aşamaları gerçekleştirilmediğinden sadece karakter sınıflandırma işlemi yapılmıştır. Bu işlemde hem MNIST ve EMNIST test verileri hem de özel olarak oluşturulmuş test verileri kullanılarak yapılan etiket tahmin sonuçları görsellerle sunulmuştur. MNIST veri seti için eğitim başarısı %98, test başarısı ise %99; EMNIST veri seti için eğitim başarısı %93, test başarısı ise %96 olarak elde edilmiştir. Son olarak sistemdeki eksikler ve gelecekteki planlar sunulmuştur.

ABSTRACT

Keywords: Handwriting recognition, letter, digit, digital, machine learning, deep learning

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

Since segmentation and fusion stages were not performed in this project, only character classification was performed. In this process, label prediction results using both MNIST and EMNIST test data and specially created test data are presented with visuals. For the MNIST dataset, the training success was 98% and the test success was 99%; for the EMNIST dataset, the training success was 93% and the test success was 96%. Finally, the deficiencies in the system and future plans are presented.

# GİRİŞ

Günümüzde çeşitli belge ve dokümanlar üzerinde bulunan el yazısını dijitale 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 tabanıyla 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. Sonrasında yatay ve dikey kestirim yöntemleri ile metin içindeki satırlar sonrasında da karakterler tanınarak elde edilir. Ön işleme aşamasında, segmente edilmiş karakterler MNIST ve EMNIST veri setlerine uygun hale getirilmek için çeşitli normalizasyon işlemlerine ve farklı filtreleme algoritmalarına tabii tutulur. Sınıflandırma aşamasında, ön işleme sonrasında elde edilen karakterler etiketlendirilerek tanınmış 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 etiketlendirilmiş karakterler birleştirilerek anlamlı kelimeler elde edilmiştir. Böylelikle el yazısı metni dijitale çevirme işlemini gerçekleştiren sistem tamamlanmıştır. Ayrıca kullanıcıların sisteme kolaylıkla ulaşabilmesi için Flask Framework kullanılarak bir web uygulaması oluşturulmuştur.

Projenin tasarım kısmında model oluşturulduktan sonra MNIST ve EMNIST test verileri ile özel olarak hazırlanmış test verileri kullanılarak testler gerçekleştirilip elde edilen sonuçlar her bir test verisi için yapılan etiket tahminleri ile görsel olarak sunulmuştu. Projenin son halinde ise özel olarak hazırlanan çeşitli form şablonları kullanılarak testler gerçekleştirilmiştir. Ayrıca modelin performansını değerlendiren metrikler ve sonuçları bir tablo halinde sunulmuştur. Çalışmanın eksikleri ve gelecek planları ile sistemin başarısının arttırılması için hedeflenen planlar sunulmuştur.

# ÖNCEKİ ÇALIŞMALARIN ARAŞTIRILMASI

* 1. El Yazısı Karakter Tanıma ve Resim Sınıflandırmada Derin Ögrenme 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].

* 1. 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 kelimeden 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].

* 1. 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].

* 1. 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].

* 1. 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].

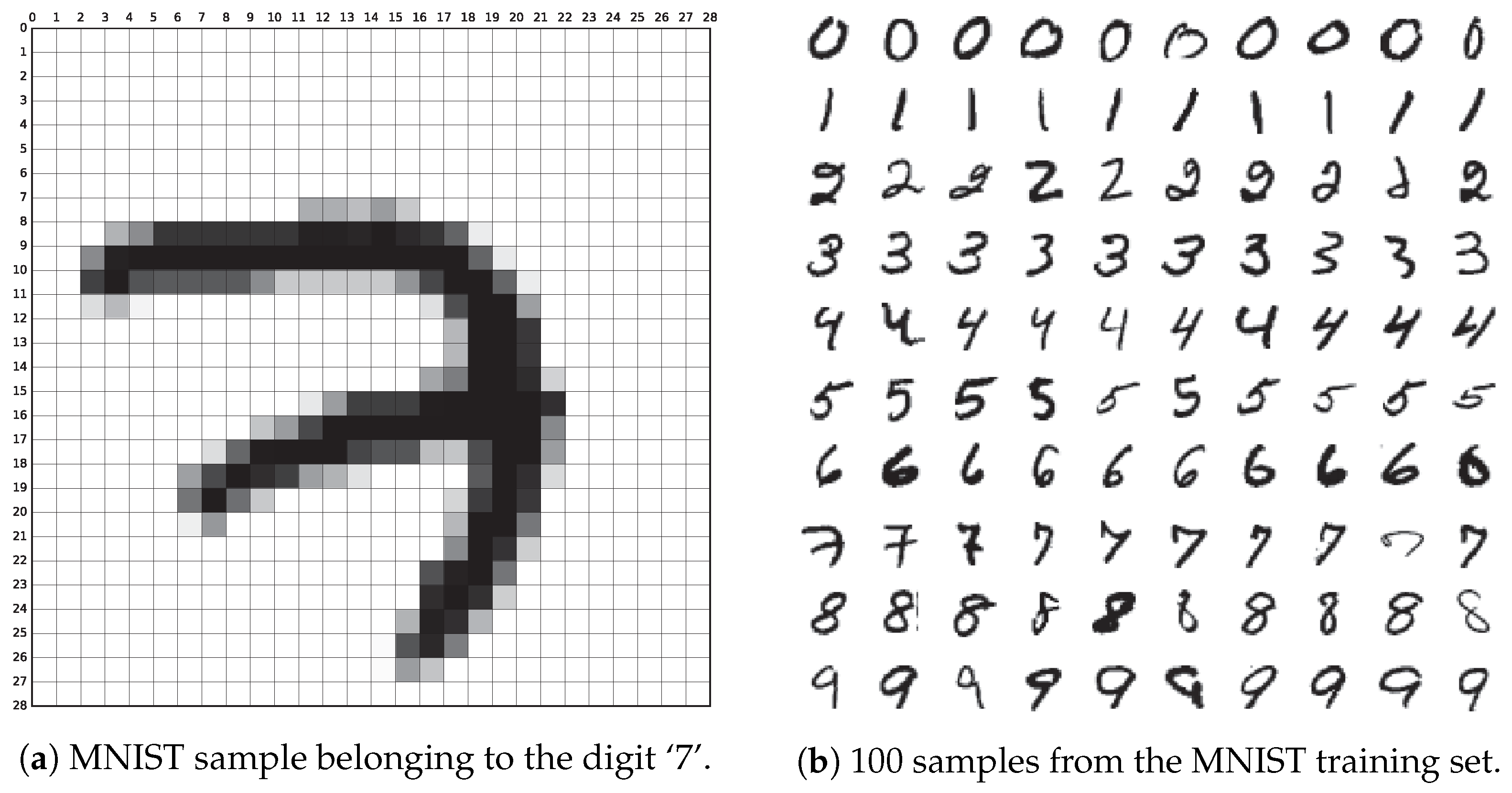
# YÖNTEM VE TASARIM

* 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.

* + 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].

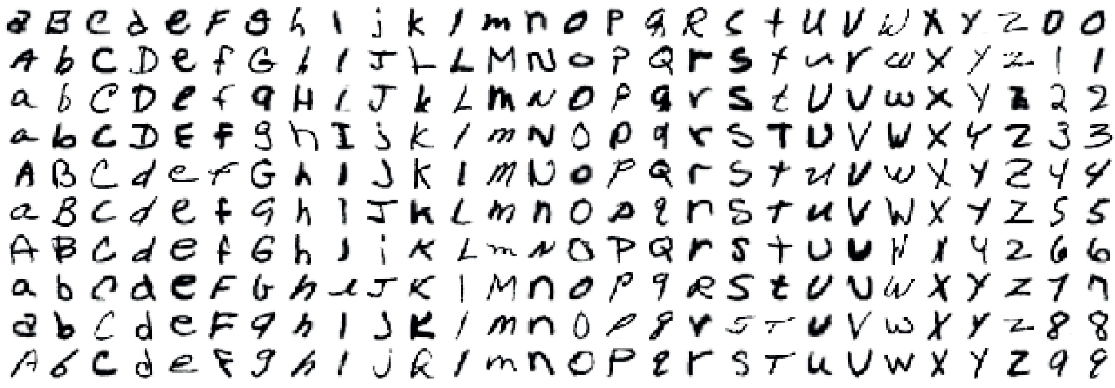


Şekil 3. 1 MNIST Veri Seti Örnekleri [1].

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].

* + 1. 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].

* + 1. Kütüphaneler

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,

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].

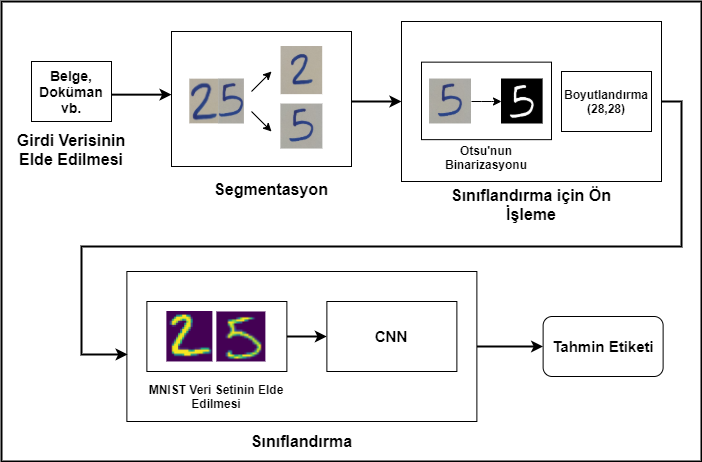
Keras,

OpenCV,

Görselleştirmeler için seaborn ve matplotlib kütüphaneleri kullanılmıştır.

* 1. Yöntem

Yöntem kısmında planlanan sistemin genel çalışma prensibi sunulmaktadır. Sistem verinin eldesi, segmentasyon, ön işleme, sınıflandırma ve birleştirme gibi farklı aşamalardan oluşmaktadır. Segmentasyon aşaması satır ve karakter segmentasyonu olmak üzere iki adımdan oluşmaktadır ve her iki adım öncesinde de ön işleme aşaması gerçekleştirilmiştir. Şekil 3.3’te sistemin genel çalışma şeması verilmiştir.



Şekil 3. 3 Sistemin Genel Çalışma Şeması

* + 1. Verinin Eldesi

Bu çalışma için çeşitli bilgilerin alındığı 3 farklı form şablonu oluşturulmuştur.

Bu form şablonlarının fotoğrafları sadece el yazısının bulunduğu kısmı içerecek şekilde kırpılarak sisteme girdi olarak verilir. Girdi olarak verilen görüntü segmentasyon işlemine tabii tutulacaktır. Görüntü segmentasyonu, bir görüntüyü her biri içerisinde farklı özelliklerin tutulduğu anlamlı bölgelere ayırma işlemidir. Örneğin, görüntü içerisinde benzer parlaklıklar olabilir ve bu parlaklıklar ilgili görüntünün farklı bölgelerindeki nesneleri temsil edebilir.

FORM ÖRNEĞİ EKLE

* + 1. Satır Segmentasyonu

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

metin, yazı tipi, el yazısı, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

SATIR ÖRNEK RESMİ

* + 1. 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.

Girdi olarak bir görüntü alır ve bu görüntüdeki yazı karakterlerini tanımlayarak her birini ayrı ayrı ele alır.

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.

* + 1. Harf ve Rakam Modellerinin Oluşturulması

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

* + - 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.

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 ağın 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.

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.
   * + 1. 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 ve tip dönüşümü gibi ön işleme adımları uygulanmıştır. Son olarak bu veriler 3.2.4.1’de anlatılan CNN modeli üzerinde eğitilmiştir. Eğitim sonucunda rakam tahmininde kullanılacak olan H5 uzantılı “numbers\_model” model elde edilmiştir.

* + - 1. EMNIST Veri Seti İçin Model Eğitimi

EMNIST veri seti, CSV uzantılı “emnist-letters-train” ve “emnist-letters-test” dosyalarından çekilerek kullanılmıştır. 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.

Orijinal-güncel hali ss

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 3.2.4.1’de anlatılan CNN modeli üzerinde eğitilmiştir. Eğitim sonucunda harf tahmininde kullanılacak olan H5 uzantılı “letters\_model” modeli ve CSV uzantılı ayrı bir mapping dosyası elde edilir. CSV uzantılı dosya üzerinde değişiklikler yapılmıştır. Orijinal-güncel hali ss

* + 1. Karakter Tahminleri

Karakterleri içeren bileşenler harf ve rakam etiketinin tahminleri için aynı ön işleme adımlarına tabi tutulmuştur. Rakam tanıma işlemi için H5 uzantılı “numbers\_model” modeli kullanılır. Harf tahmin işlemi için ise H5 uzantılı “letters\_model” modeli kullanılmıştır. Ayrıca harf tanıma işleminde model eğitimi sonucunda elde edilen mapping içeriği bölüm 3.2.4.2’deki gibi değiştirilerek kullanılmıştır.

* + 1. Karakterlerin Birleştirilmesi

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.

* + 1. Flask Web Uygulaması

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

Bu işlemler yapıldıktan sonra “continue” butonu ile şekil..deki sayfaya geçilir. Bu kısımda kullanıcıdan formdaki el yazısı kısımları içeren bir fotoğraf yüklemesi beklenir.

Fotoğraf seçildikten “upload” butonu ile yüklenip şekil..deki sonuç sayfasına geçilir. Bu sayfada da kullanıcıdan ilk sayfada alınan bilgiler ile Bölüm 3.2.6’da elde edilen python sözlük yapısı bir tablo şeklinde ekrana yazdırılır.

# BULGULAR

Çalışmanın bu kısmında oluşturulan sistem test edilmiştir ve test sonucunda elde edilen tahmin çıktıları ile başarı değerleri sunulmuştur. Ayrıca web uygulaması üzerinden örnek dönüşüm işlemleri de sunulmuştur. “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.

* 1. MNIST Modeli İçin Değerlendirmeler

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

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

Şekil..’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.

metin, çizgi, diyagram, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Açıklama otomatik olarak oluşturuldu

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

metin, ekran görüntüsü, diyagram, kare içeren bir resim

Açıklama otomatik olarak oluşturuldu

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

metin, ekran görüntüsü, yazı tipi, grafik içeren bir resim

Açıklama otomatik olarak oluşturuldu

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

yazı tipi, ekran görüntüsü, grafik, simge, sembol içeren bir resim

Açıklama otomatik olarak oluşturuldu

* 1. EMNIST Modeli İçin Değerlendirmeler

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

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

Şekil..de 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.

metin, çizgi, diyagram, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Açıklama otomatik olarak oluşturuldu

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

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

EMNIST veri seti, 26 farklı karakter içeren İngiliz alfabesinden oluşturulduğu için etiketleri sıralı olarak 1-25 aralığında verilmektedir. Örnek olarak “A” harfi “1”, “Z” harfi ise “25” olarak etiketlenir, bu etiketler başlık 3.2.4.3’te anlatılan “mapping.csv” dosyasına göre belirlenmiştir

* 1. Sistemin Test Edilmesi

# 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. 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. Çalışmanın yöntem kısmında anlatılan segmentasyon ve birleştirme aşamaları gerçekleştirilmemiştir. Gerçekleştirilen kısımların devamı olarak karakterlerin tanınması, belge üzerindeki yazılardan gerçekleştirileceği için segmentasyon aşaması önem taşımaktadır. Ayrıca tanınan karakterlerin birleştirilerek segmentasyondan önceki hallerinin elde edilmesi için birleştirme aşaması da önemlidir.
2. Yapılan çalışmada sadece CNN modeli kullanılmıştır. Devamında farklı derin öğrenme ve makine öğrenmesi yöntemleri de uygulanarak başarı sonuçları karşılaştırılabilecek ve sonucunda en uygun yöntem seçilebilecektir. Bu sayede tahmin sonuçlarının doğruluğunun arttırılması hedeflenmektedir.
3. Yapılan çalışmanın test aşamasında MNIST ve EMNIST test verilerinin yanı sıra özel test örnekleri de oluşturulmuştur. Bu örneklerin çeşitliliğinin ve sayılarının arttırılması hedeflenmektedir.
4. Oluşturulan test örneklerine kıyasla hazır test verilerinin tahmin doğruluk oranının daha yüksek olduğu görülmüştür. Oluşturulan test örneklerinde de bu doğruluk oranının artması için sistemde değişikliklerin yapılması hedeflenmektir.

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 MNİST 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 (ICCUBEA)*. 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. Kannojia, 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. "EVRİŞİMLİ 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 Universitesi (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.

EK-1

TABLO 3, İŞ ZAMAN ÇİZELGESİ

|  |  |  |
| --- | --- | --- |
| İş Paketi | Kimler Tarafından Yapıldı | Hafta Aralığı |
| Konu Seçimi | A, H | 1-3 |
| Literatür Taraması | A, H | 3-6 |
| Veri Seti Seçilmesi | A, H | 6-7 |
| Sistem Tasarımı | A, H | 7-9 |
| Sınıflandırma Modelinin Oluşturulması | A, H | 9-11 |
| Modelin Test Edilmesi | A, H | 11-12 |
| Rapor Düzenleme | A, H | 10-14 |

EK-2

TABLO 4, RİSK ANALİZİ

|  |  |
| --- | --- |
| Risk | Risk Yönetimi |
| Verilerin yetersiz olması | Özel veri setinin örnek sayısı arttırılacaktır. |
| 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 sunulmuştur.

1. **Çalışmanın tasarım boyutunu açıklayınız.**

Sunulan tasarım yeni bir projedir. Tasarım, projenin %40’ını oluşturmaktadır.

1. **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.

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

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

Görüntü işleme dersinden ise ön işlemede kullanılan yöntemler konusunda yararlanıldı.

1. **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.

1. **Kullandığınız veya dikkate aldığınız gerçekçi kısıtlar nelerdir? Lütfen çalışmanıza uygun yanıtlarla doldurunuz.**
2. **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 karşılaşılacak 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 dijitale 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 ÇEVRİLMESİ** |
| **Çalışmayı Hazırlayanlar** | **Ayşegül TOPTAŞ**  **Havvanur BOZKURT** |
| **Danışman Onayı** | **Muhammed Ali Nur ÖZ** |