



Recognition and Performance Evaluation of Handwritten Numbers Using a New Segmentation Method

Sultan Murat Yılmaz^{1,a,*}, Serap Çakar^{2,b}

¹Giresun University, Giresun, 28200, Türkiye

²Sakarya University, Sakarya, 54187, Türkiye

*Corresponding author

Research Article

History

Received: 17/10/2022

Accepted: 13/06/2023

ABSTRACT

Machine learning and computer vision applications have been very trendy recently and have been used in many applications and fields, such as biometric recognition, disease diagnosis and character analysis. Within the scope of this study, an application was developed to make it easier to read and integrate the written or test exam grades in schools and the performances of the classifiers used in the application were evaluated. Grades written by hand on exam papers were defined by using image processing methods. At the stage of recognition of grades, pre-processing, classification and slicing processes were performed on the image and classification processes were carried out on the data set used. Especially in slicing process, a new method was developed by using neighborhood relations of pixels and its performance was evaluated. MNIST dataset, which is frequently preferred in academic studies, was used to train and test the system. In this dataset, there are 60,000 handwritten samples of numbers 0-9 taken from 250 different people. In the image processing stage, a new technique for slicing the characters was used, and appropriate character separation was applied for all situations. After proper slicing, Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN) and K-Nearest Neighbor (K-NN) algorithms were used to classify the data and the performance values were obtained as 98%, 99.4% and 86%, respectively.

Keywords: handwriting recognition, machine learning, classification

Yeni Bir Dilimleme Yöntemi Kullanarak El Yazısı Rakamlarının Tanınması ve Performans Değerlendirmesi

Süreç

Geliş: 17/10/2022

Kabul: 13/06/2023

ÖZ

Makine öğrenmesi ve bilgisayar görmesi uygulamaları son zamanlar oldukça trend olmaya ve biyometrik tanıma, hastalık teşhisi ve karakter analizi gibi uygulamalar başta olmak üzere birçok uygulama ve alanda kullanılmaya başlanmıştır. Bu çalışma kapsamında okullarda yapılan yazılı veya test sınav notlarının daha kolay okunup sistem üzerine entegre edilmesi için bir uygulama geliştirilmiş ve uygulamada kullanılan sınıflandırıcıların performansları değerlendirilmiştir. Sınav kağıtları üzerine elle yazılan başarı notlarının görüntü işleme yöntemleri kullanılarak tanımlanması yapılmıştır. Başarı notlarının tanınması aşamasında görüntü üzerinde ön işleme, dilimleme ve sınıflandırma işlemleri yapılarak kullanılan veri seti üzerinde sınıflandırma işlemleri gerçekleştirilmiştir. Özellikle dilimleme işleminde piksellerin komşuluk ilişkilerinden yararlanarak yeni bir yöntem geliştirildi ve performansı değerlendirildi. Sistemi eğitmek ve test etmek için akademik çalışmalarda sıklıkla tercih edilen MNIST veri seti kullanılmıştır. Bu veri setinde 0-9 arasındaki rakamların 250 farklı kişiden alınan 60.000 el yazısı örneği bulunmaktadır. Görüntü işleme aşamasında karakterlerin dilimleme işlemleri için yeni bir teknik kullanılarak tüm durumlar için uygun karakter ayrımı gerçekleştirilmiştir. Uygun dilimleme işlemi gerçekleştirildikten sonra verileri sınıflandırmak için Yapay Sinir Ağları (YSA- Artificial Neural Networks (ANN)), Evrimsel Sinir Ağları (Convolutional Neural Networks- CNN) ve K-En Yakın Komşu (K-Nearest Neighbor- K-NN) algoritmaları kullanılmış ve performans değerleri sırasıyla %98, %99,4, %86 olarak elde edilmiştir.

Anahtar Kelimeler: el yazısı tanımlama, makine öğrenmesi, sınıflandırma

Copyright



This work is licensed under
Creative Commons Attribution 4.0
International License

^a s.muratyilmaz@giresun.edu.tr

^{id} <https://orcid.org/0000-0001-5386-2219>

^b scakar@sakarya.edu.tr

^{id} <https://orcid.org/0000-0002-3682-0831>

How to Cite: Yılmaz SM, Çakar S (2023) Recognition and Performance Evaluation of Handwritten Numbers Using a New Segmentation Method, Journal of Engineering Faculty, 1(1): 38-47

Giriş

Makine öğrenmesi ve bilgisayar görmesi uygulamalarında özellikle el yazısı tanıma işlemleri bilgi işlemede önemli bir rol oynamaktadır. Son yıllarda geliştirilen teknoloji ve yöntemler sayesinde el yazısı tanıma uygulamalarının kullanımı ve doğruluk oranları artmıştır.

Yazı yazma stilleri kişiden kişiye farklılık gösterir. Bu durum insan el yazısı rakam tanımayı araştırma ve uygulama için ana alan haline getirmiştir. Rakamların farklı stiller ile yazılması ve çok sayıda yazıtların kısa sürede okunabilmesi zor olabileceğinden, bir bireyin rakamları gözden geçirmesi oldukça zor bir görev haline gelir. Bu noktada insanların yerini bilgisayarların alması, süreci kısaltacak ve kolaylaştıracaktır.

Günümüzde el yazısı tanıma aktif bir araştırma alanıdır. Son yıllarda, yüksek tanıma doğruluğu ve güvenilirliği talep eden pratik uygulamalar için birçok el yazısı rakam tanıma sistemi önerilmiştir. El yazısı tanıma günümüzde yaygın olarak kullanılan Makine Öğrenmesi ve Derin Öğrenme yöntemleri mevcuttur. Bu yöntemler el yazılarını tanıma (sınıflandırmada) iyi yöntemler olsalar da el yazısı tanıma hâlâ bir açık araştırma alanıdır. Tanıma doğruluğu, çalışma süreleri, hesaplama karmaşıklığı yönünden çalışma performansını arttıracak yeni yöntemler geliştirilmesi talep edilmektedir. Bu açıdan günümüzde trend bir araştırma alanı olup bu konu ile ilgili literatürde çeşitli çalışmalar mevcuttur.

Tanya Makkar ve arkadaşları [1] yaptıkları çalışmada MNIST data setini kullanarak el yazısı rakamlarının K-NN ve CNN ile tanınması ile ilgili çalışma gerçekleştirdiler. Yapılan bu çalışmada görüntü işlemede en iyi sonuçların alınmasında iki ana etken olduğuna değinilmiştir. Birincisi görüntü özniteliklerinin uygun ve doğru bir şekilde çıkarılması, ikincisi ise iyi bir sınıflandırma yönteminin belirlenmesidir. Görüntü özellikleri çıkarmada Gauss dağılım metodunu kullanarak K-NN ve CNN performanslarını ve zaman karmaşıklığını karşılaştırdılar. Yapılan çalışmada CNN'in K-NN'e göre daha iyi performans verdiğine ancak CNN'in eğitim süresinin göz ardı edilecek seviyede olduğuna değinilmiştir.

Preman Ghadekar ve arkadaşları [2] çalışmalarında MNIST ve EMNIST veri setleri üzerinde K-NN ve Destek Vektör Makinesi (SVM) yöntemlerini kullanarak el yazısı rakamlarının tespitini gerçekleştirdiler. Yapılan bu çalışmada görüntü işleme safhasında görüntülerin özellik çıkarımları için iki ayrı yöntem kullanılmıştır. Hibrit Ayrık Dalgacık Dönüşümü ve Ayrık Kosinüs Dönüşümü kullanılarak görüntü özelliklerini çıkardılar ve bu özellikler K-NN ve SVM yöntemleri kullanılarak rakamların tespiti gerçekleştirildi. SVM yönteminin rakam tespitinde daha avantajlı olduğu sonucuna varılmıştır.

E. Lejeune [3] yaptığı çalışmada farklı bir yöntem kullanarak görüntü işlemede daha az maliyet ve karmaşıklığa sahip meta model yöntemi kullanmıştır. Bu yöntemde Neo-Hookean materyal modeli kullanmıştır. Materyal modeli ile farklı bir özellik çıkarma yöntemi kullanılarak ileri beslemeli YSA modeline göre CNN modelinde daha yüksek performans elde edilmiştir.

O. Kaziha ve T. Bonny [4] geliştirdikleri çalışmada MNIST veri setinden gelen verilerin ve görüntü işlemede kullanılacak görüntülerin boyutlarını kullandıkları yöntem ile küçültürken CNN ve Uzun Kısa Süreli Bellek (LSTM) sinir ağlarında performansını gözlemlidiler. Her iki sınıflandırma yöntemi için 784 olan giriş verilerini 128'e düşürmüşlerdir. Bu sayede sınıflandırma eğitiminde zamandan kazanç sağlanmış, performanslarında CNN için %99,45, LSTM için %99,22 doğruluk oranı elde edilmiştir.

Palvanov A. ve Cho Yi. [5] el yazısı rakamlarının tespitinde en uyumlu derin öğrenme yönteminin seçilmesi için dört farklı derin öğrenme algoritmasını denemişlerdir. MNIST veri setini kullanarak dört farklı derin öğrenme yöntemini gerçek zaman uygulamalarında eğitim ve test sürelerini, bellek kullanımının miktarlarını ve temel indeks değerlerini karşılaştırdılar.

Ricardo F. Alvear-Sandoval ve arkadaşları [6] MNIST veri seti üzerinde CNN performansının iyileştirilmesi için bir çalışma gerçekleştirdiler. CNN performansının iyileşmesi için iki yönlü takip işlemi uyguladılar. Birincisi yığılmış gürültü giderme otomatik kodlayıcı sınıflandırıcılarının CNN birimlerine uygulanması, ikincisi ise CNN mimarilerinin sınırlamalarından yararlanmaya çalışarak önceki tasarımların en iyilerinin çıktısına bir Yığılmış Gürültü Giderici Otomatik Kodlayıcı sınıflandırıcısı uygulamaktan ibarettir. Bu sayede normal CNN modelinden daha üstün performans elde edilmiştir.

E. S. Gati ve arkadaşları [7] Kannada alfabesi için Kannada-MNIST ve Dig-MNIST olarak adlandırılan yeni el yazısı rakamları veri kümesi üzerinde yeni bir boru hattı oluşturularak veriler üzerinde eğitimin başarı oranını artırdılar. Geliştirilen modelde doğruluk oranlarını Kannada-MNIST'de %96,8'den %97,53'e ve Dig-MNIST veri setinde %76,1'den %85,02'ye çıkarmışlardır.

D. Beohar ve A. Rasool [8] çalışmalarında MNIST veri setini ve YSA ve CNN sınıflama yöntemlerini kullanarak el yazısı rakamlarının tanınmasını gerçekleştirdiler. Bu çalışmada el yazısı rakamlarının tanınmasında beş aşamalı bir yöntem önerildi. Aynı zamanda çoklu sinir ağlarında lokal Gradyan yöntemi kullanılarak Gradyan patlaması durumunun engellenmesi sağlanmıştır.

K. Zhang ve arkadaşları [9] MNIST veri setini kullanarak YSA uygulamalarında oluşan hiper parametreleri ve L-2 norm düzenlemelerinin etkilerini gözlemlidiler. YSA uygulamalarında optimizasyon yapmak için manuel, ızgara, rastgele arama ve Gauss Süreçlerini kullanan Bayes optimizasyonu gibi bir dizi yöntem önerilmiştir. Çok küçük öğrenme oranlarının aşırı uyum problemi oluşturacağından L2-norm düzenlemesi ile aşırı uyumun ortadan kaldırılmasının kritik öneme sahip olduğu belirtilmiştir.

A. Garin ve G. Tauzin [10] makina öğrenmesi yöntemi için topolojik veri analizini kullanarak yeni bir öğrenme metodunu sundular. Gri seviyeli resimler üzerinden belirli filtreleme yöntemlerini kullanarak MNIST veri seti üzerinden eğim yapılmasıyla el yazısı rakamlarının tanınması ile ilgili çalışma gerçekleştirdiler.

Dong-yuan Ge ve arkadaşları [11] MNIST veri setini kullanarak CNN tabanlı el yazısı rakam tanıması için yüksek

doğrulukla detektör tasarımı gerçekleştirdiler. TensorFlow üzerinde gerçekleştirilen bu uygulamada konvolüsyon katmanlarındaki özellik haritaları yüksek boyutta ve derinlikte çıkartılmıştır. Bu sayede CNN'in son katmanında fazla sayıda nöron kullanılarak sınıflandırmanın hassasiyeti ve performansı artırılmıştır.

Birjit G. ve arkadaşları [12] MNIST veri setini kullanarak el yazısı rakamların tanınmasında kullanılan beş farklı makine öğrenmesi yönteminin performanslarını karşılaştırdılar. Bu yöntemler için hangi durumlarda hangi yöntemin performansının daha iyi olabileceği belirlenmeye çalışıldı.

AG Hochuli ve arkadaşlarının [16] yaptıkları çalışmada el yazısı rakamlarının dilimleme işlemlerini gerçekleştirerek CNN modelinde eğitimini sağladılar. Yapılan çalışmada el yazısı rakamlarının boylarının (rakamların üst ve alt tepe noktaları farkı) yarısı alınarak C medyan değeri hesaplanıyor. Bu değer üzerinden bit taraması yapılarak rakamların dilimlenmesi sağlanıyor. Bu yöntemde birleşik rakamların ve iç içe geçmiş rakamların dilimlenme işlemleri doğru bir şekilde gerçekleşmiyor. Yapılan ölçümlerde dilimleme işlemi %85 oranında doğru bir şekilde yapılmıştır. Sınıflandırma işleminde CNN modeli kullanılmıştır. NIST veri seti kullanılarak CNN modelinin eğitimi yapıldı. Dilimleme işleminden gelen veriler CNN modeline uygulanmasında %97,12 oranında başarı elde edildi.

Abdülcelil Gattal ,Youcef Chibani ve Bilal Hacı [17] yaptıkları çalışmada el yazısı rakamlarının uygun şekilde dilimlenmesini ve sınıflandırmasını gerçekleştirdiler. Dilimlenme işlemi için görüntü ön işleme kısmında dikey iz düşüm histogram analizi, kontur analizi ve kayan pencere Radon dönüşümüne dayalı olarak üç farklı segmentasyon yöntemi kullanarak dilimleme işlemi gerçekleştirildi. Bu yöntem kesintili rakamların tanınması ve birleşik rakamların tanınmasını sağlamaktadır. Ancak iç içe geçmiş rakamlar için dilimleme aşamasında uygun çözümler sunamamaktadır. Sınıflandırma aşamasında karar destek vektör makinası kullanılmıştır. Veri seti olarak NIST veri seti kullanıldı. Gerçekleştirilen modelin performansı %96,91 olarak ölçüldü.

Mohsine Elkhayati ve arkadaşlarının [18] yapmış olduğu çalışmada Arapça yazıların dilimlenmesi gerçekleştirdiler. Arapça yazılar birleşik yazıldığından bu çalışmada matematik morfolojik işlemler kullanıldı. Ancak bu yöntemin kullanılmasında el yazılarında veri kaybına neden oldu. Bu durumu engellemek için Küresel Genişleme ve Küresel Aşınma yöntemleri kullanıldı. Sınıflandırma için CNN yöntemi kullanıldı. Tüm işlemler sonucunda performans olarak %97,35 oranında doğruluk oranına sahiptir.

Guillaume Renton ve arkadaşlarının [19] yapmış oldukları çalışmada el yazısı metin satırı segmentasyonu için öğrenmeye dayalı bir yöntem sunuyorlar. Tam konvolüsyon ağlarını kullanarak giriş verilerinin giriş verilerinin çözünürlüğünü düşürmeden piksel bazda tarama yapılarak her satırın x ekseninde tespiti yapıldı. Kullanılan bu yöntem %91,3 lük bir performans sağlıyor ancak bu yöntemin kullanılması bellek yönetimi açısından maliyetli olabileceği unutulmamalıdır.

Manoj Kumar Sharma ve Vijay Pal Singh Dhaka [20] yaptıkları çalışmada destek vektör makinası sınıflandırma

yöntemini kullanarak el yazısı kelimelerinin dilimleme işlemini gerçekleştirdiler. Bu çalışmada kayıt, eğitim ve segmentasyon olmak üzere üç aşamaya ayrılmıştır. Eğitim ve test işlemlerinde ICDAR veri seti kullanılmıştır. Segmentasyon aşaması sekiz alt bölümden oluşur ve dilimleme işlemlerinde SURF (Hızlandırılmış Dayanıklı Özellikler) algoritması kullanıldı. Filtrelenmiş destek vektör makinaları kullanılarak dilimleme işlemleri sonucunda %96,48 oranında bir başarı elde edilmiştir.

Payal Jindal ve Balkrishan Jindal [21] yapmış oldukları çalışmada orta nokta algılama tekniği kullanılarak Gurmukhi yazısıyla yazılmış el yazısı metin belgelerinin satır ve kelime bölünmesi işlemi gerçekleştirildi. Bu çalışmada yazarların geliştirdiği satır bölümlenmesi algoritması ve kelime bölümlenme algoritması kullanılarak hem satır hem de kelime bölünmesi işlemi gerçekleştirildi. Burada önerilen orta nokta tespitine dayalı yaklaşımla, basitçe iki satırı veya iki kelimeyi ayıran boşlukların tanınmasına dayanmaktadır. Yapılan testler sonucunda önerilen yöntemin hem satır hem de kelime segmentasyonu için %95 oranında bir performansla çalıştığı bulunmuştur.

Son yıllarda yapılan çalışmalar ve gelişmeler sonucunda yapay zekâ ve derin öğrenme teknikleri karakter tanımda oldukça yüksek performans çıkarmıştır. Bu çalışma kapsamında özellikle kalabalık sınıflarda yapılan yazılı veya test sınav kağıtlarındaki el yazısı notlarının dijital ortamda anlaşılır ve hatasız olarak tespit edilmesi için bir uygulama geliştirilmiştir. Kağıtların üzerindeki notların görüntüsü tarama işleminden geçirilerek dijital ortama taşınmıştır. Dijital ortama aktarılan resimler gerekli ön işleme, dilimleme ve sınıflandırma işlemlerine tabi tutularak el yazısı rakamlarının tanınması işlemleri gerçekleştirilmiştir. Sınıflandırma işlemleri yapılırken Yapay Sinir Ağları (YSA - Artificial Neural Networks), K En yakın Komşu Algoritması (K-NN - K-Nearest Neighbor) ve Evrimsel Sinir Ağları (CNN - Convolutional Neural Networks) yöntemleri kullanılmış ve bu yöntemler arasında performans farklılıkları gözlemlenmiştir.

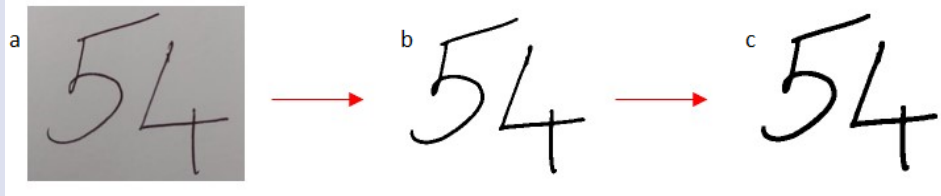
El Yazısı ile Rakam Tanıma Uygulaması

Görüntü işleme uygulamalarında el yazısı tanıma işlemleri çevrim içi (online) ve çevrim dışı (offline) olmak üzere iki kısımda incelenir. Çevrim içi yöntemler anlık el yazısı tanıma uygulamalarıdır. Çevrim içi yöntemlerde özel ekranlar (tablet, dokunmatik ekran vs.) üzerinden el yazısı hareketlerinden yararlanarak el yazıları tanımlanır. Çevrim dışı yöntemler ise el yazıları görüntülerinin analiz için tasarlanan sisteme görüntü olarak aktarılıp, sistem üzerinden tanınmasını sağlayan yöntemdir [13].

Bu çalışma kapsamında çevrim dışı yöntem kullanılmıştır. Sınav kağıtlarında notların yazıldığı alanların görüntüleri alınarak tasarlanan sistem üzerinden görüntünün giriş, ön işleme, dilimleme, sınıflandırma ve sonuç olmak üzere beş aşamalı bir sistemden geçerek el yazısı rakamlarının tanıma işlemi gerçekleştirilmiştir (Şekil 1).



Şekil 1. Uygulama işlem basamakları.
Figure 1. Application process steps.



Şekil 2. (a) Orijinal resim.; (b) Blur, Laplace filtresi ve Siyah-Beyaz dönüşüm uygulaması.; (c) Morfolojik işlem.
Figure 2. (a) Original image.; (b) Blurring, Laplace filtering and black-white conversion applications.; (c) Morphological process.

El yazısı görüntüleri tarayıcı yardımıyla sisteme girişi sağlanır. Sisteme alınan ham görüntüler ön işleme aşamasında belirli manipüle işlemlerinden geçerek özellik çıkarımları yapılır. Dilimleme aşamasında geliştirilen yeni yöntem kullanılarak piksellerin komşuluk ilişkisinden yararlanılarak rakamların dilimlenmesi yapılır. Sınıflandırma aşamasında K-NN yöntemi, evrişimli sinir ağları (CNN) ve yapay sinir ağları (ANN) kullanılarak sınıflandırma işlemleri gerçekleştirilir. Evrişimli sinir ağları ve yapay sinir ağlarının eğitimi için MNIST veri seti kullanılmıştır.

MNIST Veri Seti

Bu çalışmada sistemin eğitimi ve testler için kullanılan veri seti olarak "Modified National Institute of Standards and Technology" (MNIST) veri seti [14] kullanılmıştır. El yazısı rakamlardan oluşan MNIST veri seti uluslararası alanda sıklıkla kullanılmaktadır. MNIST veri seti sıfır ile dokuz arasındaki rakamların görüntülerinden oluşan bir veri setidir. MNIST veri setindeki verilerin 60,000 adedi eğitim verisi, 10.000 adedi test verisi olarak bulunmaktadır. Rakamların her biri 28x28 piksel ölçüsüne sahip görüntülerden oluşmaktadır.

Ön İşleme

Ön işleme bölümünde sisteme alınan ham görüntülerin istenmeyen gürültü, parazit ve gereksiz alanlarını ayırarak görüntülerin istenilen (benzersiz) alanlarını ön plana çıkarmak için kullanılan bir takım manipüle işlemleri gerçekleştirilir. Bu bölümde görüntülerin benzersiz özelliklerini ön plana çıkarmak için belirli manipülasyon işlemleri ve filtreleme yöntemleri kullanılmıştır. Çalışmada sınav notlarının tanınması işlemi gerçekleştirileceğinden her basamak için onluk sayma sistemi rakamlarını (0-9 arası rakamları) tanıma yapılmıştır. Dijital rakamlara baktığımızda tüm rakamlar bütünlük içindedir. Yani dijital rakamlarda yazım stillerinden kaynaklanan kesikler, rakamın tam ve doğru yazılmaması gibi maddi hatalar yoktur. Aslında el yazısı

rakamlarında bu şekilde yazılması beklenilmektedir. İdeal durumda el yazısı rakamların ortak özelliği yazım esnasında bir bütünlüğü olmasıdır. El yazısı ile yazılan her rakam kalemi kaldırmadan yazılabilir durumda olmalıdır. Bu durum da her rakam için bir bütünlük durumu oluşması beklenmektedir. Ön işleme aşamasında temel motivasyonumuz rakamların bir bütünlük içinde olmasıdır. Ancak kişilerin yazım stillerinden kaynaklanan farklılıklar sebebiyle her zaman bu ideal yazım şekli gerçekleşmeyebilir. Bu durumlar için görüntü üzerinde gerekli manipüle işlemi gerçekleştirerek her basamak değeri için hem rakamların bütünlüğünü hem de rakamların birbirlerinden ayrıklığı sağlanmalıdır.

Ön işleme aşamasında sisteme giriş sağlanan el yazısı not görüntülerinde bireysel yazım stillerinden dolayı el yazısı rakamları bütünlük içinde olmayabilir. Bu durum bizim için istenmeyen bir durumdur. Çünkü dilimleme aşamasında bütünlük özelliğinden yararlanarak el yazısı rakamlarının ayrıştırılması sağlanacaktır. Bu durumu engellemek için ön işleme aşamasında gerekli manipüle işlemleri gerçekleştirilir. Aynı zamanda yazım stilleri nedeniyle görüntülerin birbirine bitişik olması da söz konusu olabilir. Bu durum için de bitişik görüntülerin uygun şekilde ayrılması ve saptanması gerekmektedir.

Ön işleme aşamasında öncelikle görüntüdeki bütünlüğü arttırmak ve istenmeyen özellikleri yumuşatmak için 5x5 matris özellikli blur filtrelemesi uygulanır. Ardından istenilen kenar ve özelliklerin çıkarılması için düşük seviyede Laplace filtrelemesi kullanılır. Daha sonra görüntü siyah beyaz görüntüye çevrilir. Görüntüyü siyah beyaz forma dönüştürürken uç özelliklerinden yararlanıldı. Yani görüntü matrisinde siyaha yakın olan piksel değerleri "0", beyaza yakın olan piksellerin matris değerleri ise "255" olarak belirlendi. Bu işlemlerden sonra rakamlarda bütünlük oluşmayabilmektedir. Bu durumun önüne geçmek için ve görüntüdeki rakamların bütünlüğünü sağlamak amacıyla morfolojik işlemlerden yararlanıldı. Görüntümüzdeki son işlem olarak 2x2 matrisli Morfolojik Genişletme işlemi

uygulandı. Tüm işlemlerden sonra görüntüler şekil 2 (a), (b) ve (c)'deki gibi olacaktır.

Dilimleme

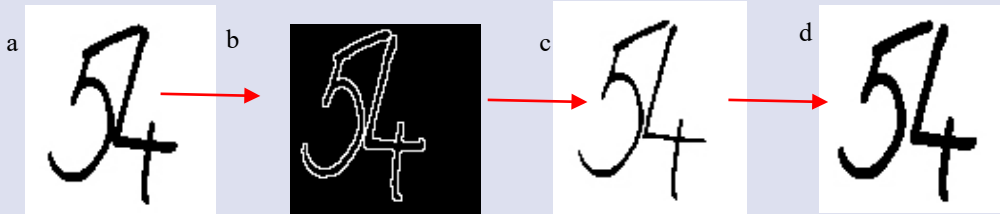
Ön işleme aşaması tamamlandıktan sonra oluşan görüntü birden fazla basamak değerine sahipse görüntüyü basamak değerlerine ayırmak gerekmektedir. Ön işlemeden sonra gelen görüntü (Şekil 2 (c)) iki boyutlu matris halindedir. Görüntüde not değerlerinin olduğu matris değerleri (siyah alanlar) "0", diğer tüm alanların değerleri (beyaz alanlar) "255" değerindedir. Bu değerlerin uç noktalarda tutulması dilimleme ve sınıflandırma aşamasında bize kolaylık sağlayacaktır.

Ön işlemeden gelen görüntüdeki her bir rakam bir bütünlük içerisinde. Rakamlarda kişilerin yazı stillerinden oluşan kopmalar morfolojik işlemler kullanılarak tamamlanmış ve bütünlük sağlanmıştır. Rakamların tamamının bütünlük içinde olması dilimleme işlemleri için kolaylık sağlayacaktır. Ancak burada rakamların bitişik yazılma durumlarının da kontrol edilmesi gerekmektedir. Bunun için ön işleme aşamasından gelen görüntü (Şekil 3 (a)) üzerinde bazı manipüle işlemleri gerçekleştirilmesi gerekmektedir. Bunun için öncelikle görüntüye 2x2 matrisli Morfolojik Gradyan işlemi uygulanır (Şekil 3 (b)). Bu işlemle birleşik görüntünün sınırları belirlenir. Sınır belirleme işleminden sonra belirlenen sınır dışı alanlar 2x2 matrisli Morfolojik Erozyon (Şekil 3 (c)) işlemi ile temizlenir. Temizlenme işlemi sonunda görüntüde bütünlük kaybı olamaması için son işlem olarak 2x2 matrisli Morfolojik Genişletme (Şekil

3 (d)) işlemi yapılır. Bu sayede uç uca birleşik olan rakamlar ayrılmış olur. Bu işlem sonrasında görüntüdeki rakamı ifade eden pikseller "0" değerini taşır. Diğer bütün pikseller "255" değerini taşır. Yani rakamları ifade eden pikseller siyah diğer bütün pikseller beyaz rengini taşır.

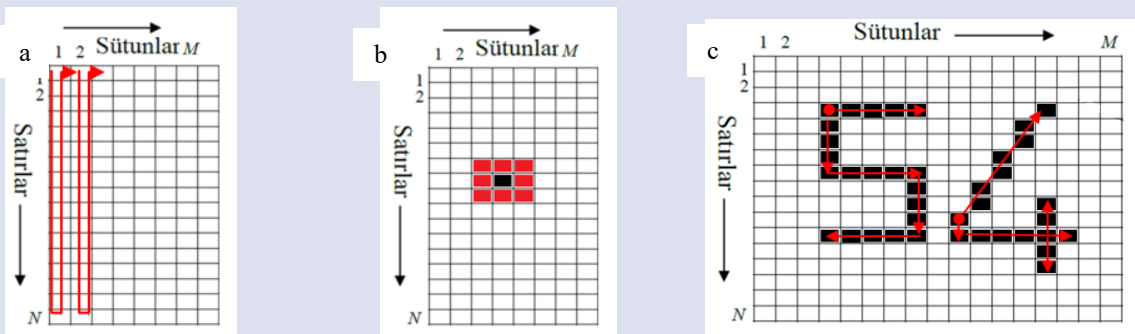
Dilimleme işleminde geliştirilen yeni yöntem CNN ağlarının konvolüsyon işlemlerinden ilham alınarak geliştirilmiştir. Ön işlemeden gelen görüntüdeki rakamlar bir bütünlük içinde olacağından her rakama ait görüntüdeki piksellerin değerleri "0" (Sıfır) olacaktır ve rakamların görüntüleri birbirlerini kesmeyecektir. Rakamların arası ve diğer tüm alanlardaki pikseller "255" değerine sahip beyaz görüntü pikselleri olacaktır. İlk olarak ön işlemeden gelen işlenmiş görüntü başlangıç sütunundan başlanarak sütun bazlı taramaya başlanılır (Şekil 4 (a)).

Taramada bulunan ilk "0" değerine ait matris adresi bizim referans noktamız olarak kabul edilir. Referans noktası bulunduktan sonra ön işlemeden gelen görüntü boyutunda tüm elemanları "255" olan yeni bir matris oluşturulur. Ön işlemeden gelen görüntünün referans noktasından başlanarak komşu pikseli "0" değerine sahip tüm matris adreslerinin değeri tespit edilir ve oluşturulan yeni matriste bu adreslere "0" değeri atanır. Bir piksele ait tüm komşu piksel değerleri şekil 4(b)'de belirtilmiştir. Şekil 4 (b)'de belirtilen siyah alana sahip pikselin tüm komşuları şekil 4 (b)'de belirtilen kırmızı alanlardır. Referans noktasından başlanılarak tespit edilen tüm sıfır değerine sahip pikseller bize görüntüdeki rakamın ilk basamak değerine sahip rakamı verecektir (Şekil 4 (c)).



Şekil 3. (a) Ön işlemeden gelen resim.; (b) Morfolojik Gradyan (2x2).; (c) Morfolojik Erozyon (2x2).; (d) Morfolojik Genişletme (2x2).

Figure 3. (a) Image from pre-processing.; (b) Morphological Gradient (2x2).; (c) Morphological Erosion (2x2).; (d) Morphological Expansion (2x2).



Şekil 4. (a) Sütun bazlı tarama işlemi.; (b)Piksel komşuluk gösterimi.; (c) Komşuluk ilişkisinden yararlanarak rakam belirleme işlemi.

Figure 4. (a) Column-based scanning process.; (b) Pixel neighborhood display.; (c) The process of determining numbers using the neighborhood relationship.

Tespit edilen rakam oluşturulan yeni matriste saklanır. Tespit edilen "0" değerine sahip pikseller içinde en büyük sütun değerine sahip piksel yeni referans değerimiz olur. Başlangıç satır değeri (ilk satır) ve yeni referans değerine sahip olan pikselin sütun numarası (0 x yeni referans değeri sütun numarası) değerine sahip pikselden başlanılarak sütun bazlı taramaya devam edilir. Tarama işleminde bulunan ilk "0" değeri yeni referans noktası değeri olur. Bu noktaya sahip pikselden başlanılarak komşusu "0" değerine sahip olan pikseller için tüm değerleri "255" olan işlenmiş görüntü boyutlarında yeni bir matris oluşturulur. Oluşturulan yeni matrise komşuluk ilişkisinden yararlanarak "0" değerine sahip piksellerin atama işlemi gerçekleştirilir (Şekil 4 (c)). Komşuluk ilişkisinden yararlanarak değeri "0" olan pikseller bittiğinde işlem tamamlanır ve en büyük ikinci basamak değerine sahip rakamın görüntüsü oluşturulmuş olunur. Komşusu "0" olan değerler içinde sütun değeri en büyük olan değer hafızada tutulur ve yeni referans değeri olur. Bu işlemler (Şekil 5) görüntüdeki tüm rakamlar tespit edilene kadar (Şekil 6) devam eder. Böylece görüntülerin piksel değerlerinin komşuluk ilişkilerinden yararlanarak tüm basamak değerlerine sahip rakamların değerleri tespit edilir. Dilimleme işlemine ait kaba kod şekil 5'de belirtilmiştir. Dilimleme işleminde tespit edilen tüm rakamlara ait görüntüler sınıflandırma işlemlerinde MNIST veri seti kullanıldığı için son aşamada siyah piksellerin olduğu alanlardan kırılarak 28x28 boyutuna dönüştürülür.

Dilimleme İşleminin Değerlendirilmesi

El yazısı rakamlarının dilimlenme yönteminde çevrim içi el yazılarının tespitine benzer bir yöntem geliştirilmiştir. Bu

yöntemde CNN ağlarının konvolüsyon aşamasına benzer piksel taramalı bir yöntem uygulanmıştır. Literatürde genel olarak el yazısı rakamlarının dilimlenmesinde histogram analizi, kontur analizi, yazı konum bilgilerinden yararlanma, yazı şekilleri vb. yöntemler kullanılmaktadır. Bu çalışma kapsamında sınav kağıtlarında el yazısıyla yazılan başarı notları rakamlarının dilimlenmesi için piksellerin komşuluk ilişkilerinden yararlanılarak yeni bir dilimleme yöntemi geliştirildi. Geliştirilen bu yöntemle rakam görüntüleri ön işleme aşamasında belirtilen manipülasyon işlemlerinden geçirilir. Ön işleme aşamasından başarılı olarak gelen görüntünün siyah piksellerinin komşuluk ilişkisinden yararlanarak dilimleme işlemleri gerçekleştirilmiştir. El yazısı görüntülerinde yazım stillerinden dolayı kaynaklanan küçük kesikli rakamlar morfolojik işlemler kullanılarak bütünleme işlemleri gerçekleştirilir. Ancak yazım stillerinden kaynaklanan büyük kesikli rakamlarda birleştirme işlemleri uygun bir şekilde tamamlanamıyor.

El yazısı görüntülerinde bütün ve ayrı rakamlar ile bitişik bir şekilde yazılan rakamlar ön işleme aşamasından tespit edilip dilimleme işlemi başarılı bir şekilde gerçekleştiriliyor. Ancak iç içe geçmiş el yazısı rakamlarında bu yöntem başarılı kabul edilemez.

Dilimleme işlemlerinin testi için 20 kişiden 50 farklı en az 2 basamaklı rakamlardan oluşan yazım örnekleri alınmıştır. Her kişiden el yazısı rakamlarının 30 tanesi bütün ve ayrı rakamlar (Şekil 2 gibi), 10 tanesi birleşik şekilde yazılan rakamlar (Şekil 3 gibi), 10 tanesi ise iç içe geçmiş rakamlardan oluşmaktadır. Dilimleme işleminin performansı tabloda 1'de belirtilmiştir.

```

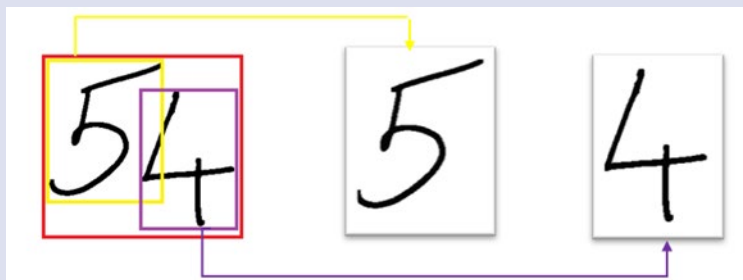
1  Goruntu_on_isleme[m][n]
2  a=0;b=0;
3  i:
4  while(a<m+1){
5  while(b<n+1){
6  if(Goruntu_on_isleme[a][b]==0){
7  tum_komsu_bul(Goruntu_on_isleme[a][b]);
8  a=en_byk_px1_stun+1;//komşuluk ilişkisinden gelen en büyük sütun değerinin 1 fazlası
9  b=0;
10 goto i;
11 }b++;
12 }a++;

```

tum_komsu_bul() /* ilk siyah renk piksel değeri bulunduğundan sonra komşuluk ilişkilerinde yararlanarak rakama ait tüm "0" değerine sahip piksel değerleri bulunur ve yeni matrise atanır. Atanan değerlerin tamamının ön işlemdeki yeni değerleri "255" olur. */

Şekil 5. Dilimleme İşlemi Kaba Kodu.

Figure 5. Pseudo code of slicing process.

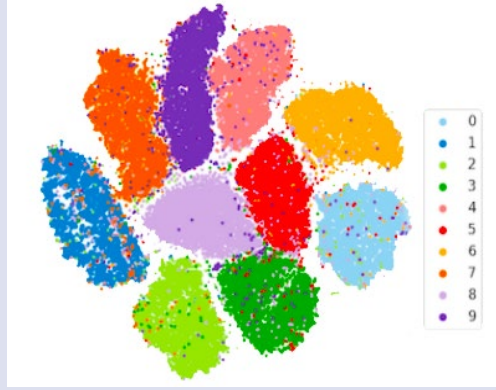


Şekil 6. Görüntü Dilimleme İşlemi.

Figure 6. Image slicing process.

Tablo 1. Geliştirilen dilimleme yöntemi performans değerleri.
Table 1. Performance values of developed slicing method

	Bütün ve Ayrık Rakamlar	Birleşik Rakamlar	İç içe Geçmiş Rakamlar
Örnek Sayısı	600	200	200
Doğru Dilimleme Sayısı	592	160	5
Yanlış Dilimleme Sayısı	2	40	195
Performans	%99,67	%80	%2,5



Şekil 7. MNIST verilerinin K-NN algoritmasına göre sınıflandırılması [15].
Figure 7. Classification of MNIST data according to the K-NN algorithm [15].

Tablo 1'e bakıldığında bütün ve ayrık rakamlardan oluşan görüntülerde %99,67 oranında başarımla elde edilmiştir. Birleşik rakamlarda %80 başarımla elde edilmiştir. İç içe geçmiş görüntülerde %2,5 performans elde edilmiştir. Geliştirilen bu yöntem bütün ve ayrık rakamlar ve birleşik rakamların dilimlenmesi için başarılı sonuçlar elde edilirken iç içe geçmiş rakamlar için başarılı sonuçlar elde edilememiştir.

Sınıflandırma

K-En Yakın Komşu Algoritması (K-NN - K-Nearest Neighbor) ile Uygulama

K-en yakın komşu (K-NN) algoritması belirli bir veri seti içinde yeni bir noktanın hangi en yakın gruba ait olacağını belirleyen en temel makine öğrenmesi algoritmalarından birisidir. Algoritmadaki "K" değeri en yakın kaç adet grup sayısının kontrol edileceği değerdir. K-NN algoritmalarında bir noktanın hangi sınıfa ait olacağını belirlemek için kullanılan veri setine ait verilerin koordinat sistemi üzerindeki konumlarının sınıflama için kullanılan noktaya ait uzaklıkları tespit edilir. Bu uzaklıklardan en yakın "K" adet komşu üyeleri seçilir. Genelde komşu üyesi sayıları eşit çıkmaması için "K" değeri tek sayı seçilir. Seçilen en yakın "K" adet komşudan en fazla gruba ait olan sınıfa yeni nokta dahil edilir.

K-NN algoritmasını kullanarak Python programlama dili ile sınıflandırma işlemi gerçekleştirilmiştir. K-NN uygulamasında K değeri "5" olarak alınmıştır. Mesafe algoritması olarak "Öklid mesafe" algoritması kullanılmıştır. Öklid mesafe algoritması kullanmak için Minkowski yönteminden yararlanılmıştır ve "p" değeri 2 olarak kabul edilmiştir. MNIST verilerinin K-NN algoritması ile kümeleneceği Şekil 7'de gösterilmiştir.

"K" değeri için farklı değerler denenmiştir fakat hem zaman hem de performans açısından en iyi sonuç "5" değerinde elde edilerek, %86 oranında bir performans sağlanmıştır. K-NN algoritması hesaplama ve sınıflama işlemini aynı anda yaptığından büyük verilerle çalışması biraz zaman alabilmektedir. MNIST veri setinde, 60000 eğitim 10000 test verisi olmak üzere toplam 70000 veri olduğundan hesaplama süresi 15,28 dakika sürmüştür.

Yapay Sinir Ağları (YSA-Artificial Neural Networks (ANN)) ile Uygulama

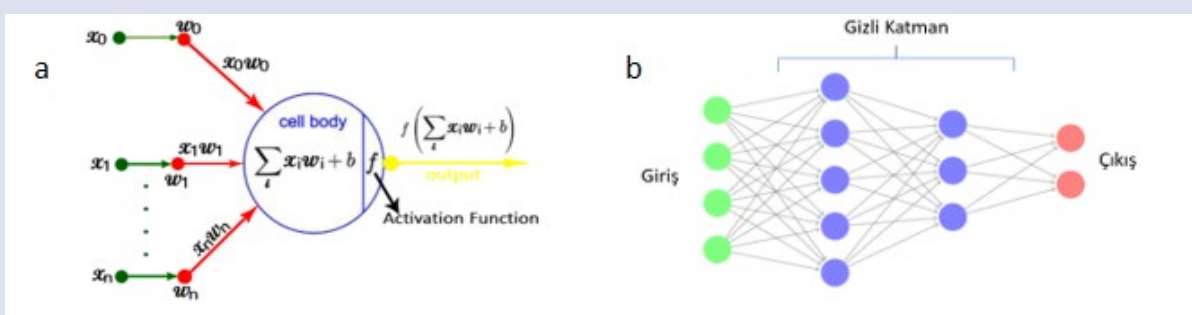
Yapay Sinir Ağları (YSA) beyin temel yapı taşı olan sinir hücrelerinin çalışma prensibinden örnek alınarak geliştirilen bir yöntemdir. Biyolojik sinir hücresi yapısına benzetim yöntemi ile matematiksel sinir hücresi oluşturulmuştur.

Yapay sinir hücresi (Şekil 8. (a)) üzerinden giriş yapan veri sinir hücresinin ağırlık değeri ile çarpılır. Yapay sinir hücresinin çekirdek yapısında bir toplama fonksiyonu bulunur. Bu toplama fonksiyonu yapay sinir hücresine giriş yapan bütün değerlerin ağırlıkları ile çarpımlarını toplar ve bu toplam değerine yapay sinir hücresinin bias değerini ekler. Elde edilen değer lineer bir değerdir. Bu değer direkt çıkışa verilmesi durumunda reel olmayan sonuçlar elde edilir. Bu durumun engellenmesi için aktivasyon fonksiyonları kullanılır. YSA'larda aktivasyon fonksiyonu kullanılmazsa YSA sınırlı öğrenme gücüne sahip bir doğrusal bağlantı (linear regression) gibi davranacaktır. Bu durum gerçek uygulamalar için istenmeyen durumdur ve yanlış sonuçlar elde edilir. Böylelikle yapay sinir hücresinin matematiksel modeli oluşturulmuş olur. Elde edilen sonuç bir sınıflandırma değeri olabileceği gibi başka bir sinir hücresinin giriş değeri de olabilir.

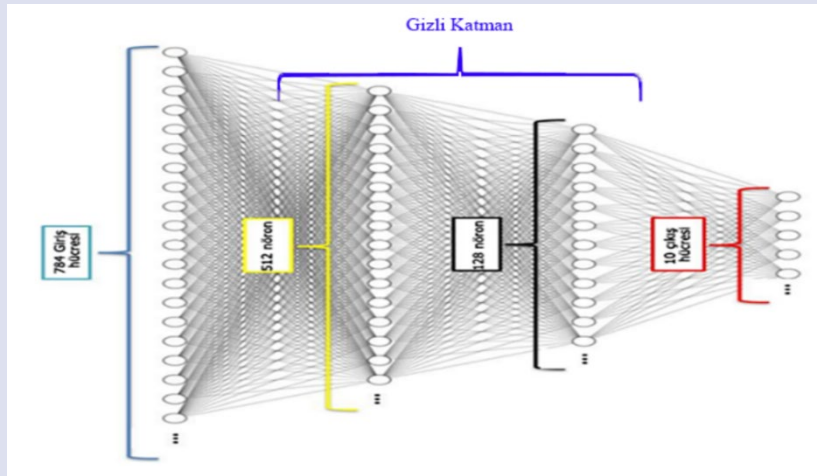
Yapay sinir hücreleri kullanılarak YSA'lar oluşturulur. YSA'lar tek katmanlı ve çok katmanlı olarak kullanılabilir. Çok katmanlı YSA'lar (Şekil 8. (b)) birden fazla gizli katmana sahip olabilir.

YSA'lar kullanarak dilimleme işleminden gelen görüntüler python üzerinden sınıflandırma işlemine tabi tutulmuştur. MNIST veri seti 28x28 boyutunda görüntülere sahiptir. Dilimleme işleminden gelen her bir görüntü de 28x28 boyutundadır. Görüntülerden gelen verilerin tamamı sinir ağına modeline uygulanır. Giriş verileri 28x28 boyutunda olduğu için giriş nöronları 784 adet olmalıdır. Sinir ağı modeli giriş katmanı, gizli (hidden) katman ve çıkış katmanı olmak üzere üç katmandan oluşturulmuştur. Gizli

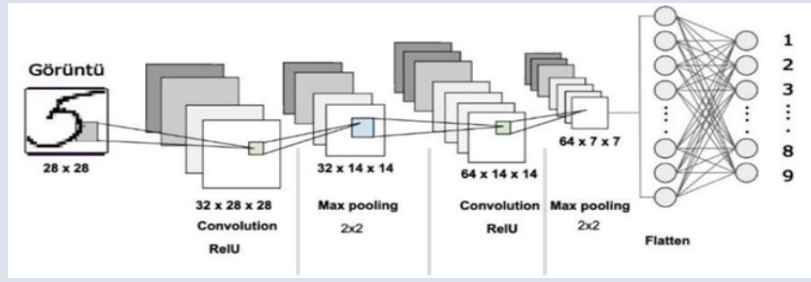
katman iki aşamalıdır. Giriş katmanında 784 adet nöronun, gizli katman ise sırasıyla 512 ve 128 nöronun oluşmaktadır. Hem giriş hem de gizli katmanda aktivasyon fonksiyonu olarak "ReLU" fonksiyonu kullanılmıştır. Çıkış katmanında sınıflandırma işlemi sıfır ile dokuz arasındaki rakamları sınıflanacağı için 10 adet nöron bulunur ve aktivasyon fonksiyonu olarak "softmax" fonksiyonu uygulanmıştır. YSA modeline MNIST veri setinden gelen 60000 adet veri eğitim için 10000 adet veri ise test için kullanılmıştır. Modelimize dahil edilen verilerin sonucunda sistemin doğruluğu %98 olarak hesaplanmıştır. YSA modelinin eğitimi için kullanılan süre 11,85 dakikadır. Kullanılan YSA modeli Şekil 9'de gösterilmiştir.



Şekil 8. (a) Yapay sinir hücresi; (b) Çok katmanlı YSA.
Figure 8. (a) Artificial neuron; (b) Multilayer ANN.



Şekil 9. Kullanılan Yapay Sinir Ağları Modeli.
Figure 9. Proposed artificial neural network model.



Şekil 10. Uygulamada kullanılan CNN modeli.
Figure 10. CNN model used in the application.

Tablo 2. Kullanılan sınıflandırıcıların performans değerleri
Table.2. Performance values of the used classifiers.

Performans	KNN	YSA	CNN
Doğruluk	%86	%98	%99,4
Eğitim Süresi(dk)	9,4	11,85	15,28

Evrişimsel Sinir Ağları (Convolutional Neural Networks – CNN) ile Uygulama

Evrişimsel sinir ağları (CNN) (Şekil 10) son yıllarda görüntü işlemede sıklıkla kullanılan derin öğrenme yöntemlerinden birisidir. CNN görüntü işlemede sınıflandırılacak görüntünün çeşitli filtreleme yöntemleri kullanarak özellik çıkarımı yapar ve bu çıkarımlar üzerinden YSA'ları kullanarak bir sınıflandırma gerçekleştirir.

CNN beş katmandan oluşur. Birinci ve en önemli katman konvolüsyon katmanıdır. Bu katmanda görüntü üzerinde belirli filtreleme yöntemi kullanılarak özellik çıkarımı gerçekleştirilir. Filtreleme yapılan görüntülerin tamamı özellik (feature) matrisi denilen matriste saklanır. İkinci katman doğrusal olmayan (Non-Linearity) katmandır. Bu katman aktivasyon katmanı (Activation Layer) olarak adlandırılır çünkü aktivasyon fonksiyonlarından birisi bu katmanda kullanılır. Üçüncü katman havuzlama (pooling) katmanıdır burada pooling işlemleri gerçekleştirilir. Bu sayede özelliklerin ağırlıkları azaltılır ve uygunluğu denetlenir. Dördüncü katman düzleme (Flattening) katmanıdır. Bu katman son ve en önemli katman olan Tam Bağlantılı (Fully Connected) katmanının girişindeki verileri hazırlar. Sinir ağları üzerinde giriş verileri tek boyutlu bir dizi üzerinden alır. Sinir ağlarında kullanılan giriş verileri ise konvolüsyon ve Pooling katmanından gelen matrislerin tek boyutlu diziyeye çevrilmiş halidir. Beşinci ve son katman Tam Bağlantılı (Fully Connected) katmandır. Verileri düzleme katmanından alır ve sinir ağı yoluyla öğrenme işlemini gerçekleştirir.

CNN kullanılarak dilimleme işlemlerinden gelen veriler sınıflandırma işlemine tabi tutulur. Dilimleme işleminden gelen görüntü 28x28 boyutunda bir matris şeklindedir. Bu görüntüleri eğitim için sırasıyla CNN'e tabi tutulur. CNN modeli (Şekil 10) oluştururken iki tane konvolüsyon katmanı kullanılmıştır. Her konvolüsyon katmanından sonra havuzlama (pooling) katmanı da olacağından iki tane de havuzlama (pooling) katmanı kullanılmıştır. Evrişimsel sinir ağlarının ilk katmanı olan konvolüsyon katmanında ilk olarak görüntülerin 32'li özellik haritaları çıkarılmıştır İlk konvolüsyon katmanında özellikleri çıkarılan görüntüleri havuzlama katmanında 2x2 max pooling (havuzlama) işlemine tabi tutulmuştur. İkinci

konvolüsyon katmanında 64'lü özellik haritası çıkarılmış ve havuzlama katmanında 2x2'li max pooling işlemi uygulanmıştır. Konvolüsyon katmanlarında aktivasyon fonksiyonu olarak "ReLU" aktivasyon fonksiyonu kullanılmıştır. Düzleme katmanında 586 tane giriş verisi tek bir dizi şeklinde tam bağlantılı katmana sınıflandırma için gönderilmiştir. Tam bağlantı katmanında 586 tane giriş 10 tane çıkış nöronları vardır. Giriş nöronlarına "ReLU" aktivasyon fonksiyonu, çıkış nöronlarına "softmax" aktivasyon fonksiyonu uygulanmıştır. CNN modeli şekil 10'da belirtilmiştir. Modelin eğitimi için MNIST veri setinden 60000, testi için 10000 adet veri kullanılmıştır. Oluşturulan modelin doğruluk oranı %99,4 olarak gözlemlenmiştir. Verilerin eğitimi için geçen süre 15,8 dakika olarak hesaplanmıştır.

Performansların Değerlendirilmesi

Sınıflandırma işlemlerinin sonucunda hesaplanan tüm değerler Tablo 2'de gösterilmiştir. CNN yönteminde eğitim süresi 15,28 dakika, doğruluk oranı %99,4 olarak ölçülmüştür. YSA'da modelin eğitim süresi 11,85 dakika, doğruluk oranı %98 olarak ölçülmüştür. K-NN algoritmasında ise eğitim ve hesaplama süresi 9,4 dakika doğruluk oranı %86 olarak ölçülmüştür. Eğitim için harcanan süreler karşılaştırıldığında CNN modeli 15,28, YSA modeli 11,85 ve K-NN uygulaması 9,4 dakika olarak hesaplanmıştır. Bu süreler 3.2 GHZ 32 işlemcili ve 512 GB RAM'e sahip server bilgisayar üzerinde hesaplanmıştır kullanılan donanımlara göre değişkenlik gösterebilir. Burada modellerin eğitim hızları karşılaştırılmasında en hızlı K-NN, onu takip eden YSA ve CNN sınıflandırma yöntemleri gelir. Ancak performans yönünden değerlendirildiğinde CNN sınıflandırıcının en yüksek performansı gösterdiği, onu takip eden YSA ve K-NN modellerinin olduğu gözlemlenmektedir.

YSA ve CNN yöntemlerinde genel bir eğitim modeli oluşturulur ve bu model her sınıflandırma işlemde kullanılır. Yani sistem bir kere eğitilir ve eğitim sonuçlarında çıkan model her sınıflandırma işlemleri için kullanılacak genel model olur. Bu sayede tekrar model oluşturmaya gerek kalmadan gerçekleştirilecek her sınıflandırma için genel eğitim modeli kullanılacaktır. Bu sayede model eğitildikten

sonra gerçekleştirilecek sınıflandırmalar çok daha kısa sürelerde olacaktır. Ancak K-NN sınıflandırma yönteminde her sınıflandırma için K-NN algoritması tekrar yeni hesaplamalar yapmak zorundadır.

Sonuçlar

Literatürde genel olarak el yazısı rakamlarının dilimlenmesinde histogram analizi, kontur analizi, yazı şekilleri vb. yöntemler kullanılmaktadır. Bu çalışma kapsamında sınav kağıtlarında el yazısıyla yazılan başarı notları rakamlarının yeni bir dilimleme yöntemi geliştirilerek bütün ve ayırık rakamlar ve birleşik rakamlar başarılı bir şekilde dilimlenmiş iç içe geçen rakam görüntüleri için dilimleme yöntemi yetersiz kalmıştır. Geliştirilen bu yöntem bütün ve ayırık rakamlar ve birleşik rakamlar için başarılı olarak çalışmaktadır.

Başarılı bir şekilde dilimleme de ayrıştırılan tüm rakamlar CNN modeli, YSA modeli ve K-NN algoritmasına tâbi tutularak performans değerlerinin tablo halinde karşılaştırılması yapılmıştır. MNIST veri tabanı kullanılarak yapılan sınıflandırma işlemlerinin performansları kıyaslandığında CNN modeli %99.4, YSA modeli %98 ve K-NN uygulaması %86 olarak tespit edilmiştir. YSA ve CNN modelinde eğitim işlemi tamamlandıktan sonra model kaydedilip eğitilmiş model yeni durumlar için kullanılabilir. Bu durum da bize yeni veri geldiğinde, sistemi tekrar eğitmeden kullanım olanağı sağlar. Ancak K-NN uygulamalarında yeni durumlar için tekrar eğitim işlemi gerçekleştirilmesi gerekmektedir.

Kaynaklar

- [1] Makkar T., Kumar Y., Dubey A.K., Rocha Á., Goyal A., Analogizing time complexity of KNN and CNN in recognizing handwritten digits, 2017 Fourth International Conference on Image Information Processing (ICIIP), (2017) 1-6.
- [2] Ghadekar P., Ingole S., Sonone D., Handwritten Digit and Letter Recognition Using Hybrid DWT-DCT with KNN and SVM Classifier, 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), (2018) 1-6.
- [3] E. Lejeune, Mechanical MNIST: A benchmark dataset for mechanical metamodels, *Extreme Mechanics Letters*, 36 (2020) (100659) 1-7.
- [4] Kaziha O., Bonny T., A Comparison of Quantized Convolutional and LSTM Recurrent Neural Network Models Using MNIST, 2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA), (2019) 1-5.
- [5] Palvanov A, Cho Y.I. Comparisons of Deep Learning Algorithms for MNIST in Real-Time Environment, *IJFIS*, 18 (2018) 126-134.
- [6] Alvear-Sandoval R.F., Sancho-Gómez J.L., Figueiras-Vidal A.R., On improving CNNs performance: The case of MNIST, *Information Fusion*, 52 (2019)106-109.
- [7] Gati E.S., Nimo B.D., Asiamah E.K., Kannada-Mnist Classification Using Skip CNN, 16th International Computer Conference on Wavelet Active Media Technology and Information Processing, (2019) 245-248.
- [8] Beohar D., Rasool A., Handwritten Digit Recognition of MNIST dataset using Deep Learning state-of-the-art Artificial Neural Network (ANN) and Convolutional Neural Network (CNN), 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), (2021) 542-548.
- [9] Zhang K., Su H., Dou Y., Shen S., Evaluation of the Influences of Hyper-Parameters and L2-Norm Regularization on ANN Model for MNIST Recognition, 2019 International Conference on Intelligent Computing Automation and Systems (ICICAS), (2019) 379-386.
- [10] Garin A., Tauzin G., A Topological "Reading" Lesson: Classification of MNIST using TDA, 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), (2019) 1551-1556.
- [11] Ge D.-y., Yao X.-f., Xiang W.-j., Wen X.-j., Liu E.-c., Design of High Accuracy Detector for MNIST Handwritten Digit Recognition Based on Convolutional Neural Network, 2019 12th International Conference on Intelligent Computation Technology and Automation (ICICTA), (2019) 658-662.
- [12] Gope B., Pande S., Karale N., Dharmale S., Umekar P., Handwritten Digits Identification Using Mnist Database Via Machine Learning Models, *IOP Conference Series: Materials Science and Engineering*, 1022 (2021) 1-11.
- [13] KARAKAYA K., Handwriting Recognition Using Machine Learning, Master Thesis, Sakarya Üniversitesi, Institute of Natural Sciences, 2020.
- [14] GitHub, Mahanteshambi/Clustering-MNIST page, 2020, NOS. Available at: <https://github.com/Mahanteshambi/Clustering-MNIST>, Retrieved August 8, 2023.
- [15] Pythonawesome, A 1D analogue of the MNIST dataset for measuring spatial biases page, NOS. Available at: <https://pythonawesome.com/a-1d-analogue-of-the-mnist-dataset-for-measuring-spatial-biases>. Retrieved August 8, 2023.
- [16] Hochuli A.G., Oliveira L.S., Britto Jr A.S., Sabourin R., Handwritten Digit Segmentation: Is it Still Necessary?, *Pattern Recognition*, 78 (2018) 1-11.
- [17] Gattal A., Chibani Y., Hadjadji B., Segmentation and Recognition System for Unknown-length Handwritten Digit Strings, *Pattern Analysis and Applications*, 20 (2017) 307-323.
- [18] Elkhayati M., Elkettani Y., Mourchid M., Segmentation of Handwritten Arabic Graphemes Using a Directed Convolutional Neural Network and Mathematical Morphology Operations, *Pattern Recognition*, 122 (2022) (108288) 1-15.
- [19] Renton G., Soullard Y., Chatelain C., Adam S., Kermorvant C., Paquet T., S Fully Convolutional Network With Dilated Convolutions for Handwritten Text Line Segmentation, *International Journal on Document Analysis and Recognition (IJ DAR)*, 21 (2018) 177-186.
- [20] Sharma M. K., Dhaka V. S., Segmentation of Handwritten Words Using Structured Support Vector Machine, *Pattern Analysis and Applications*, 23 (2020) 1355-1367.
- [21] P. Jindal, B. Jindal, Line and Word Segmentation of Handwritten Text Documents Written in Gurmukhi Script Using Mid Point Detection Technique, 2015 2nd International Conference on Recent Advances in Engineering & Computational Sciences (RAECS), (2015) 1-6.