

Konvolüsyonel Sinir Ağları (CNN) ile Çin Sayı Örüntülerinin Sınıflandırması

Classification of Chinese Number Patterns with Convolutional Neural Networks (CNN)

Nihal Zuhul KAYALI¹ , Sevinç İLHAN OMURCA² 

¹Bilgisayar Mühendisliği Bölümü, Türk-Alman Üniversitesi, İstanbul, Türkiye

²Bilgisayar Mühendisliği Bölümü, Kocaeli Üniversitesi, Kocaeli, Türkiye

(nihal.kayali@tau.edu.tr, silhan@kocaeli.edu.tr)

Received: Sep.2,2021

Accepted: Sep.26,2021

Published: Oct.20, 2021

Özetçe— Makine Öğrenmesi ve Derin Öğrenme uygulamaları için bir standart haline gelen MNIST veri seti oldukça popülerdir. Bu çalışmada MNIST veri setinin bir varyasyonu olarak el yazısı ile yazılmış Çince sayıların görüntülerini içeren veri kümesi tanıtılmıştır. Makine Öğrenmesi yöntemlerinin güncel ve zorlu araştırma konularından çok katmanlı yapay sinir ağlarının bir türü olan Konvolüsyonel Sinir Ağları (ConvNN ya da CNN) modeli uygulanıp bir sınıflandırma çalışması yapılmıştır. Sınıflandırma aşamasında %97 doğru sınıflandırma oranına ulaşılmıştır. Sonuç olarak, önerilen modelin Çin sayı örüntüleri üzerinde yüksek başarımla kullanılabileceği düşünülmektedir.

Anahtar Kelimeler : Görüntü İşleme, Sınıflandırma, Konvolüsyonel Sinir Ağları, Derin Öğrenme

Abstract— The MNIST dataset, which has become a standard for Machine Learning and Deep Learning applications, is quite popular. In this study, a dataset containing images of handwritten Chinese numbers is introduced as a variation of the MNIST dataset. Convolutional Neural Networks (ConvNN or CNN) model, which is a type of multi-layer artificial neural networks, which is one of the current and challenging research topics of Machine Learning methods, has been applied and a classification study has been made. At the classification stage, a correct classification rate of 96.5% was achieved. As a result, it is thought that the proposed model can be used on Chinese number patterns with high performance.

Keywords : Image Processing, Classification, Convolutional Neural Networks, Deep Learning

1.Giriş

Gelişen teknoloji, her saniye artan veri miktarı ve alanın popülerleşmesiyle birlikte kapsamı da büyüyüp gelişen Yapay Zekâ kavramının tanımı pek çok şekilde yapılabilir. En basit şekilde ifade etmek gerekirse ilk olarak “zeki makineler özellikle de zeki bilgisayar programları yapma bilimi ve mühendisliği” şeklinde ortaya atılmıştır (McCarthy, 2007). Bir başka ifade ile Yapay Zekâ, insanda bulunan doğal zekâyı modelleyebilmek adına insana özgü yetilerin makineye kazandırılmasıdır. Yapay Zekânın en önemli alt dalı olarak geliştirilen Makine Öğrenmesi ise verideki anlamlı örüntüleri ortaya

çıkarmak için çeşitli öğrenme kuralları kullanan algoritmalar olarak tanımlanabilir. Öğreticili, öğreticisiz, takviyeli ve hibrit öğrenme kuralları ile çalışılan veri üzerinden kendi kendine öğrenip tahminde bulunur (Fyfe, 2000). Makine Öğrenmesi mühendislik, sağlık ve finans başta olmak üzere pek çok alanda yaygın olarak kullanılmaktadır.

Makine Öğrenmesinin sıklıkla kullanılan popüler modellerinden biri Yapay Sinir Ağlarıdır. İnsan beyninin öğrenme yapısı modellenmeye çalışılarak yapay sinir ağlarının biyolojik sinir ağlarına benzetilme çabası ile ortaya çıkarılmıştır. Yapay Sinir Ağları ile makinelerin eğitilip, öğrenmesi ve bunun sonucunda karar verebilmesi amaçlanmaktadır. Kendi kendine öğrenen, çeşitli koşullara göre değişebilen ve değerlendirme yapabilen Yapay Sinir Ağları katman sayılarına göre kategorize edilebilir. Sadece girdi ve çıktı katmanından oluşan yapay sinir ağları Tek Katmanlı Yapay Sinir Ağları olarak adlandırılır. Girdi ve çıktı katmanları arasında katmanlar var ise bu tipe Çok Katmanlı Yapay Sinir Ağları olarak adlandırılır. Çok Katmanlı Yapay Sinir Ağlarında girdi ve çıktı katmanları arasında eğer birden fazla gizli katman varsa bu durumda Derin Sinir Ağlarından söz etmek gerekir (Uğuz, 2019).

Yapay Sinir Ağı kavramının ortaya çıkması ve konu üzerinde yapılan çalışmaların ilerleyişinden sonra Derin Öğrenme kavramı ortaya atılarak bu konu üzerinde yoğunlaşmıştır. Derin Öğrenme kavramı yapay sinir ağlarının başarımını ve etkililiğini arttırmak hedefiyle geliştirilmiştir. Derin Öğrenme özellikle sınıflandırma, tanıma ve tespit için kullanılmaktadır (Deng&Yu, 2014). Üstün performansların elde edilmesi ile son zamanlarda Derin Öğrenme çalışmalarına eğilimi artmıştır. Derin Öğrenme modeli çok sayıda gizli katmandan oluşan yapay sinir ağları modellerini kullanır. Her katman kendinden sonra gelen katmanın giriş katmanı işlevini görür. Derin Öğrenme mimarilerinden biri de Konvolüsyonel Sinir Ağlarıdır (CNN).

Alanda son yıllarda CNN ile yapılmış görüntü tabanlı sınıflandırma çalışmalarına bakıldığında, 2020 yılında yapılmış bir çalışmada yaklaşık 1200 görüntü içeren bir veri seti ile iki tip Mango sınıflandırma yaklaşımı sunulmuştur. Görüntü tanımadaki yaygın olarak uygulanan bir derin öğrenme tekniği olan CNN algoritmaları kullanılmıştır. Sonuçlar, CNN güdümlü Mango sınıflandırma uygulamalarının sınıflandırma otomasyonunda kullanıldığında insanların mango türünü doğru bir şekilde bilmesini sağladığını bulmuştur. Eğitilen model, test setinde %100 doğruluk elde edip yaklaşımın uygulanabilirliğini göstermiştir (Mettleq vd., 2020).

2020 yılında böceklerin görüntülerini içeren veri setinden onların tür düzeyinde sınıflandırılıp sınıflandırılmayacağını test eden bir CNN çalışması yapılmıştır. Çalışmada 361 yer böceği türünden oluşan 65841 adet müze örneğini içeren bir görüntü veri tabanı oluşturulmuştur. Bu çalışmada vücut büyüklüğü, cins içindeki tür sayısı ve tür kimliği ile sınıflandırmanın doğruluk değerlendirilmesi yapılmıştır. CNN ile 19.164 test görüntüsünün %51,9'unu tür düzeyinde ve %74,9'unu cins düzeyinde doğru olarak sınıflandırmıştır (Hansen vd., 2020).

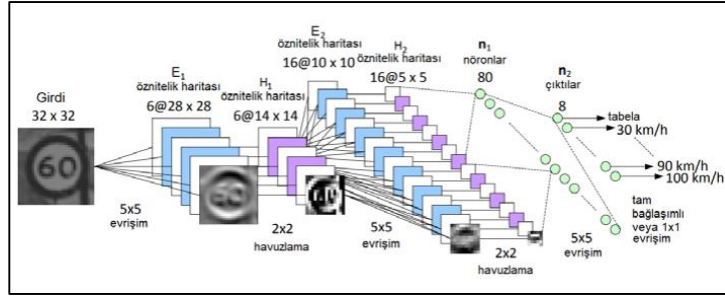
2020 yılında biyolojik görüntüler içeren veri setindeki görüntüleri doğru sınıflandırmak için özgün bir CNN modeli önerilmiştir. Çalışmadaki model, sınıflandırma performansını arttırmak için giriş görüntü boyutlarının artmasıyla doğru orantılı şekilde artan hesaplama maliyeti problemini çözmek üzerinedir. Bunun için CNN içindeki bazı modüller değiştirilmiştir. Önerilen modelin mevcut görüntü sınıflandırma yöntemlerine kıyasla ağ parametrelerini ve hesaplama maliyetini azalttığı tespit edilmiştir (Qin vd., 2020).

2020 yılında yapılmış bir başka çalışmada ise daha iyi bir sınıflandırma sonucu elde edebilmek amacıyla CNN modeli kullanılarak retinal fundus görüntülerinden öznelik çıkarılmıştır. Elde edilen CNN çıkış sonuçları önerilen sistemde farklı makine öğrenmesi sınıflandırıcıları için girdi olarak kullanılmış ve sonuçlar değerlendirilmiştir (Gayathri vd., 2020).

Bu çalışmada, el ile yazılmış 15 farklı Çin rakam biriminden oluşan özgün 15000 görüntü üzerinde CNN yöntemi kullanılarak görüntülerin hangi rakam olduğu şeklinde sınıflandırılması amaçlanmıştır.

2. Materyal ve Metot

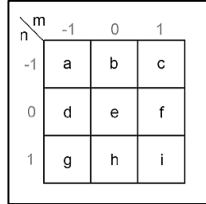
Geleneksel CNN mimarisi beş esas katmandan oluşur. Bunlar Girdi Katmanı, Konvolüsyon Katmanı, Havuzlama Katmanı, Tam Bağlantılı Katman ve Çıktı Katmanıdır. CNN yapısını içeren model örneği Şekil 1'deki gibidir.



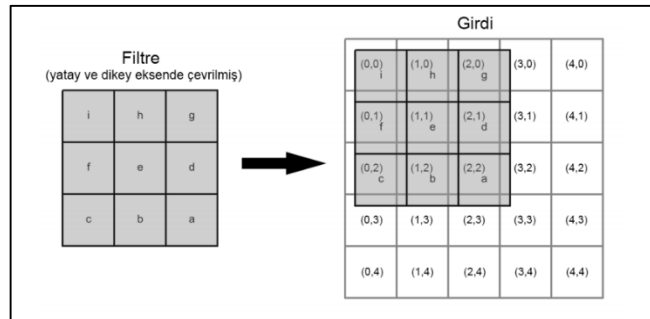
Şekil 1. Geleneksel CNN mimarisi (Tüfekçi & Karpaz, 2019)

Girdi katmanı resim, sinyal, ses dosyası gibi verilerden oluşabilir. Bu girdi verilerinin türüne göre sinir ağının gerçekleştireceği işlemler de farklılaşır. Girdi verisi bir resim olduğunda yapılan işlem renk analizi veya obje kenarlarının bulunması buna örnek olarak verilebilir. Bir sinir ağında bulunan katmanların arasında nöronların bağlantıları ve eğitim süreci neticesinde her bir nöronun çıktı değeri için ne ölçüde önem ifade ettiğini gösteren sayısal değerleri yani ağırlıklar bulunur. Her bir nöronda hesaplanan çıkış değeri ağırlığı ile çarpılarak diğer katmandaki nöronun girdi değerini oluşturur. Girdi katmanından çıktı katmanına doğru ilerleyen bu hesaplama akışına ileri yayılım algoritması denir. İleri yayılım algoritması neticesinde sinir ağına ait tahmin değerleri ve bu değerlere ait hata oranları elde edilir. Hatayı azaltmak için kullanılacak algoritmalarından biri ise geri yayılım algoritmasıdır. Bu algoritmaya göre, ileri yayılım algoritması ile alınan sonuçların hata değeri, çıktı katmanından girdi katmanına doğru çeşitli türev işlemlerinden geçip ardından öğrenme katsayısı olarak belirlenen sayı ile çarpılarak geriye yayılma gösterir (Tüfekçi & Karpaz, 2019).

Konvolüsyon konsepti LeCun tarafından ortaya atılmıştır (LeCun vd., 1989). Gelişmiş bir sinyal işleme tekniği olan Konvolüsyon özelleştirilmiş doğrusal bir işlemdir. En az bir katmanında matris çarpımı yerine Konvolüsyon işlemi olan ağlar Konvolüsyonel Sinir Ağları olarak adlandırılırlar (Kurt, 2018) Şekil 2’de verilen örnek filtre yatay ve dikey olarak döndürülüp girdi ile işleme girmektedir. Filtrenin döndürülmeden giriş verisi ile işleme girdiği durumlar da vardır. Bu işlemde Çapraz Korelasyon Fonksiyonu uygulanır. Şekil 3’te ise döndürülmüş filtrenin giriş verisine uygulanma örneği gösterilmiştir. Konvolüsyon işlemi sonrasında aktivasyon fonksiyonu devreye girmektedir.



Şekil 2. Örnek filtre (Kurt, 2018)



Şekil 3. Örnek Konvolüsyon hesaplama işlemi (Kurt, 2018)

Bir sinir ağındaki nöronun başlıca görevi bilgi iletimidir. Yapay Sinir Ağlarının en küçük parçası olan nöronlar, n sayıda girdi alır ve tek bir çıktı üretirler. Bir yapay nöron x girdileri ile w ağırlıklarını çarpıp bias değeri eklenerek elde edilir. Derin Öğrenme uygulamalarında amacımız oluşturduğumuz

model için en iyi doğruluk ve hata skorlarını verecek “w” ve “b” parametre değerlerini hesaplamaktır. Aktivasyon fonksiyonu (1)’de görüldüğü üzere y değerini kontrol etmek için bir diğer ifadeyle söz konusu nöronun aktiflik durumunun kararını belirlemek için kullanılmaktadır.

$$y = \text{Aktivasyon} (\sum (w * x + b)) \quad (1)$$

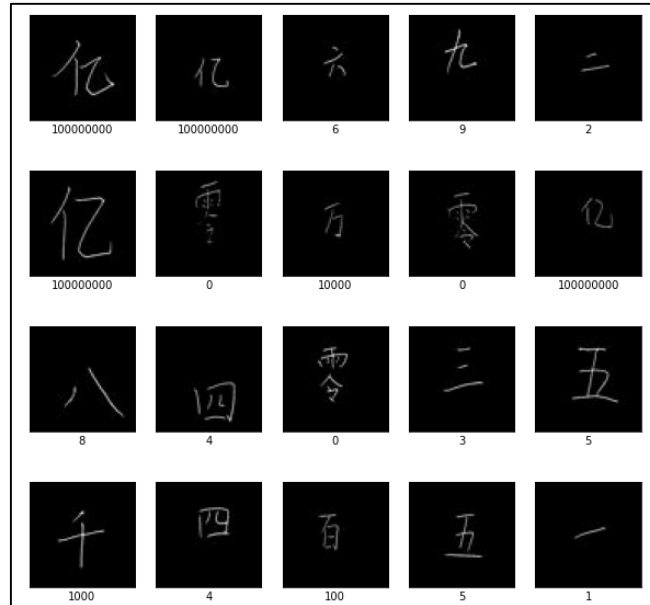
Çeşitli aktivasyon fonksiyonları modele uygun düşecek şekilde seçilebilir. Ardından çıktı verilerinin boyutunu düşürmek için ortalama veya maksimum değerini alma gibi bazı yöntemler ile alt bölgeleri özetleyerek Havuzlama işlemi yapılır. Esas amaç ardından gelecek Konvolüsyon katmanı için giriş boyutunu azaltmaktır. Girdide havuzlama işlemi kayan pencere yöntemi ile her seferinde belirlenen havuzlama fonksiyonuna göre kapsadığı girdi alanından bir değer oluşturur. Oluşan değeri çıktı katmanına ekler (Dumoulin & Visin., 2016; Kurt, 2018).

2.1. Veri Seti

MNIST veri seti Makine Öğrenmesi ve Derin Öğrenme uygulamalarında adeta bir klasik haline gelmiştir. Bu veri setinin bir varyasyonu da Newcastle Üniversitesi’nde bir proje çerçevesinde toplanan veriler ile oluşturulmuştur. Bu çalışma neticesinde toplanan, el yazısı ile yazılmış Çince sayıların bazılarını içeren bu veri seti üzerinde CNN ile bir model oluşturulup sınıflandırma uygulaması yapılmıştır. Uygulama Python dilinde Google Colab ortamında gerçekleştirilmiştir.

Çince sayma sistemi kullandığımız sayı sistemine göre farklılıklar içermektedir. Örnek vermek gerekirse on bin sayısını ifade eden 万 adında farklı bir karakter, yüz milyon sayısını ifade eden ayrı bir karakter 亿 vardır. Çalışmada kullanılan veri setinde yer alan değerler 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 100, 1000, 10000 ve 100000000’dur.

Çalışmaya 100 gönüllü Çin vatandaşı katılmıştır. İçinde 15 farklı belirlenmiş bölüm olan bir A4 kağıdına 15 farklı sayının tamamını standart bir siyah tükenmez kalemle yazılması istenmiştir. Bu işlem her bir gönüllü için 10 kez tekrarlanmış ve A4 kağıtları 300x300 piksel çözünürlüğünde taranmıştır. 15000 görüntüden oluşan bir veri kümesi elde edilmiştir. Her bir görüntü 64x64 boyutundadır. Örnek veri görüntüleri Resim 1’deki gibidir.



Resim 1. Örnek veri görüntüleri

Taranan görüntüler şu şekilde adlandırılmıştır: “suite_id: 1”, “sample_id: 3”, “kod: 4” ⇒ “input_1_3_4.jpg”. Burada “suite_id” gönüllü numarasını, “sample_id” bu gönüllünün kaçınıcı tekrarı olduğunu “kod” ise yazdığı sayının id’sini ifade etmektedir. Her bir görüntü dosyasının bilgilerini içeren

“suite_id”, “sample_id”, “code”, “value”, “character” sütunları csv formatında bir excel tablosunda yer almıştır. “Value” yazılan sayının değerini, “character” ise Çince karakter karşılığı yer almaktadır. Excel tablosu colab ortamına çekilerek tabloya adlandırma sütunu da eklenmiştir. Excel tablosunun içeriği Tablo 1’de görüldüğü gibidir.

	suite_id	sample_id	code	value	character	file_path
0	1	1	10	9	九	input_1_1_10.jpg
1	1	10	10	9	九	input_1_10_10.jpg
2	1	2	10	9	九	input_1_2_10.jpg
3	1	3	10	9	九	input_1_3_10.jpg
4	1	4	10	9	九	input_1_4_10.jpg
...
14995	99	5	9	8	八	input_99_5_9.jpg
14996	99	6	9	8	八	input_99_6_9.jpg
14997	99	7	9	8	八	input_99_7_9.jpg
14998	99	8	9	8	八	input_99_8_9.jpg
14999	99	9	9	8	八	input_99_9_9.jpg

15000 rows x 6 columns

Tablo 1. Excel tablosu içeriği

Çalışmada kullanılmak üzere veri setinin tamamı üç bölüme ayrılmıştır. Bunlar eğitim seti, doğrulama seti ve test setidir. Eğitim seti, her bir girdi verisinin çıktı verisiyle birlikte yer aldığı veri setidir. Doğrulama seti, eğitim setinin içinden seçilir. Eğitim aşamasında modelin performansını arttırmak ve uygun parametrelerin belirlenmesinin tespiti için ayrılan küçük bir bölüttür. Test seti ise modelin daha önce hiç işlemediği örneklerden sadece giriş verisi olarak modele verir. Bunun sonucunda sınıflandırmanın ne ölçüde doğru bulunduğunun bilgisi elde edilir. Veri setinin %80’i eğitim seti, %20’si test seti için ayrılmıştır. Doğrulama seti için eğitim setinin %20’si kullanılmıştır.

2. 2. Kullanılan Önemli Kütüphaneler

- Tensorflow: Google tarafından geliştirilip pek çok platformda verimli bir şekilde çalışan Tensorflow açık kaynak kodlu ve ücretsiz bir yazılım kütüphanesidir.
- Keras: Kapsamlı fonksiyonları sayesinde derin öğrenme modellerini tasarlamayı ve algoritmalarını ifade etmeyi kolaylaştırmak amacıyla hızlı sonuç almaya odaklanılarak geliştirilmiş bir Python kütüphanesidir (Team vd., 2016).
- Numpy: Python’un bilimsel işlemlerin yapılmasına olanak sağlayan bir matematik kütüphanesidir.
- Pandas: Verilerin analize hazır hale getirilmesi için veri okuma, veri işleme ve veri temizleme aşamalarını hızlı ve kolay bir şekilde yapılmasına imkân veren, kullanışlı veri yapılarına sahip açık kaynak kodlu bir Python kütüphanesidir.

3. Oluşturulan Model ve Bulgular

Çalışmada her bir görüntü 64x64x1 boyutundaki bir matristen oluşmaktadır. Her bir matris bir giriş verisidir. Bu giriş verileri rastgele seçilen ağırlıklar ile çarpılır. Ağın sahip olduğu ağırlıklar veriler için en uygun değerler olmalıdır. Bu ağırlık değerleri modelin tekrar tekrar çalışması sonucunda en uygun değerlerin bulunmasına odaklanılmasıyla sonuçlanır. Bu süreç ağın eğitilmesidir. Öğrenme işlemi ise eğitim aşamasının sonucunda öğrenme işlemi gerçekleşmiş olur. Veri seti üzerine uygulanan model Tablo 2’de görüldüğü gibidir.

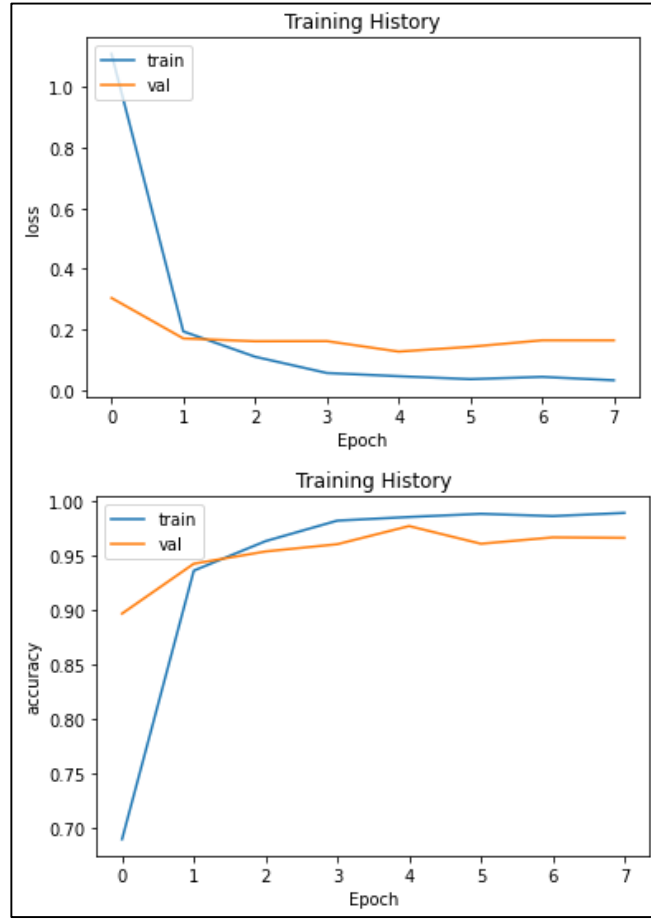
Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 64, 64, 256)	2560
activation (Activation)	(None, 64, 64, 256)	0
max_pooling2d (MaxPooling2D)	(None, 32, 32, 256)	0
conv2d_1 (Conv2D)	(None, 32, 32, 128)	295040
activation_1 (Activation)	(None, 32, 32, 128)	0
max_pooling2d_1 (MaxPooling2D)	(None, 16, 16, 128)	0
conv2d_2 (Conv2D)	(None, 16, 16, 64)	73792
activation_2 (Activation)	(None, 16, 16, 64)	0
max_pooling2d_2 (MaxPooling2D)	(None, 8, 8, 64)	0
flatten (Flatten)	(None, 4096)	0
dense (Dense)	(None, 2048)	8390656
activation_3 (Activation)	(None, 2048)	0
dropout (Dropout)	(None, 2048)	0
dense_1 (Dense)	(None, 15)	30735
activation_4 (Activation)	(None, 15)	0
Total params: 8,792,783		
Trainable params: 8,792,783		
Non-trainable params: 0		

Tablo 2. Uygulanan model

Veri setinin model boyunca işlenerek ağırlıkların güncellenmesine Epoch denir. Epoch sayısı çok fazla olduğunda model öğrenmek yerine ezberlemeye girme riski bulunur. Bu sayı az olduğunda ise ağırlıkların en iyi şekilde güncelleneceği duruma ulaşamayacaktır. Çalışmada Epoch sayısı 20 olarak belirlense de eğitim esnasında doğrulama setinde artık gelişme gösterilmiyorsa devreye giren erken durdurma seçeneği modele eklenerek zaman kaybından tasarruf edilmiştir. Epoch içinde işleme giren veri setinin için belirlenen sayıda yeni bir veri dizisi oluşturulur. Bu veri dizisinin sayısını Batch değişkeni belirler. Sınır ağında bu veri dizisi modelin sonuna kadar gidip kalır ve Batch için belirlenen sayı kadar giriş sona ulaştıktan sonra hata hesaplanıp ağırlıklar optimize edilir. Batch sayısı olarak 128 belirlenmiştir.

Model üç Konvolüsyon katmanına sahiptir. Bu katmanlar sırasıyla 256, 128 ve 64 adet 3x3'lük filtreden oluşmaktadır. Konvolüsyon işleminden sonra negatif giriş verileri için 0 değerini, x pozitif giriş verileri için x değerini alan ReLU aktivasyonu kullanılmıştır. Özellikle ağır derin olduğu ve işlem yükünün üst düzeyde olduğu problemlerde tercih edilen ReLU aktivasyon fonksiyonunun uygulanması sonucunda resimlerin boyutunda bir değişiklik olmamakla birlikte içeriği değişmektedir. Havuzlama işleminde 2x2'lik alanlar içinde en yüksek değeri alan değer seçilecek şekilde çalışan maksimum havuzlama yöntemi kullanılmıştır. Flatten sözcük olarak düzleştirmek anlamına gelir. Bu katmanda Konvolüsyon ve Havuzlama katmanlarından gelen matrislerin tek boyutlu bir diziye çevrilmiş hali bulunur. CNN'in son ve en önemli katmanı olan Tam Bağlantılı Katman girişindeki verilerin hazırlanması işlemi gerçekleştirilir. Çıkış katmanında Softmax aktivasyon fonksiyonu kullanılmıştır. Softmax aktivasyon fonksiyonu, çoklu sınıflandırma problemleri için kullanılır. Verilen her bir giriş verisinin bir sınıfa ait olma olasılığını 0 ve 1 arası çıktılar üretmek için göstermektedir. Çalışmamızda Softmax aktivasyon fonksiyonu el yazısı ile yazılmış rakamları 15 sınıf içinden en uygun sınıfa yerleştirmektedir.

Uygulanan modelin eğitim aşamasında Epoch değeri 20 olarak belirlenmişse de erken durdurma seçeneği sayesinde sekizinci döngüde eğitim sonlanmıştır. Eğitim doğruluğu %96,5 ve doğrulama doğruluğu ise %97 olarak bulunmuştur. İlgili sonuçlar Şekil 5'teki grafik şeklinde sunulmuştur.



Şekil 4. Elde edilen kayıp ve doğruluk oranları

4. Sonuç

Makine öğrenmesi çalışmalarında yüksek başarımlı sonuçlar ediliyor olmasına, hazır araç ve kütüphaneler görece bu zorlukları azaltmasına rağmen CNN uygulamaları hala pek çok zorluğa sahiptir. Görüntü gibi işlenecek verilerin boyutlarının günümüz teknolojisi her geçen an daha da yüksek kaliteye sahip olması ile hesaplama, depolama ve işleme başlıca maliyetlere en kolay verilebilecek örneklerdendir. Tutarlı ve başarılı bulunabilecek uygulamaları gerçekleştirebilmek için oluşturulan mimariler günden güne daha derinleşmektedir. Maliyetlerin düşürülmesi ve sonuçların başarıyla elde edilmesinde donanımın yüksek kalitede olmasının yanı sıra kurulan modeldeki parametrelerin önemi büyüktür. Bu çalışma, el yazısıyla yazılmış Çince sayı görüntülerini içeren bir veri tabanına dayalı olarak örnek bir sınıflandırma sağlamıştır. Derin Öğrenme modellerinden Konvolüsyonel Sinir Ağı-CNN kullanılarak geliştirilmiştir. Geliştirilen model ile test sonucu %97 hata oranı ise %13 olarak bulunarak başarılı bir sonuç elde edilmiştir.

Kaynaklar

- McCarthy, J. (2007). What is artificial intelligence? Computer Science Department, Stanford University. Available from: <http://www-formal.stanford.edu/jmc/whatisai.pdf>
- Fyfe, C. (2000). Artificial neural networks and information theory. University of Paisley.
- Uğuz, S. (2019). Makine öğrenmesi teorik yönleri ve Python uygulamaları ile bir yapay zekâ ekolü. Nobel Yayıncılık. Ankara.
- Deng, L., & Yu, D. (2014). Deep learning: methods and applications. *Foundations and trends in signal processing*, 7(3-4), 197-387.
- Tüfekçi, M., & Karpat, F. (2019). Derin Öğrenme Mimarilerinden Konvolüsyonel Sinir Ağları (CNN) Üzerinde Görüntü İşleme-Sınıflandırma Kabiliyetininin Arttırılmasına Yönelik Yapılan Çalışmaların İncelenmesi.
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4), 541-551.
- Kurt, F. (2018). Evrişimli Sinir Ağlarında Hiper Parametrelerin Etkisinin İncelenmesi.
- Dumoulin, V., & Visin, F. (2016). A guide to convolution arithmetic for deep learning. arXiv preprint arXiv:1603.07285.
- Metteq, A. S. A., Dheir, I. M., Elsharif, A. A., & Abu-Naser, S. S. (2020). Mango Classification Using Deep Learning. *International Journal of Academic Engineering Research (IJAER)*, 3(12).
- Hansen, O. L., Svenning, J. C., Olsen, K., Dupont, S., Garner, B. H., Iosifidis, A., ... & Høye, T. T. (2020). Species-level image classification with convolutional neural network enables insect identification from habitus images. *Ecology and evolution*, 10(2), 737-747.
- Qin, J., Pan, W., Xiang, X., Tan, Y., & Hou, G. (2020). A biological image classification method based on improved CNN. *Ecological Informatics*, 58, 101093.
- Gayathri, S., Gopi, V. P., & Palanisamy, P. (2020). A lightweight CNN for Diabetic Retinopathy classification from fundus images. *Biomedical Signal Processing and Control*, 62, 102115.
- Data: https://data.ncl.ac.uk/articles/dataset/Handwritten_Chinese_Numbers/10280831/1
- Team, T. T. D., Al-Rfou, R., Alain, G., Almahairi, A., Angermueller, C., Bahdanau, D., ... & van Tulder, G. (2016). Theano: A Python framework for fast computation of mathematical expressions. arXiv preprint arXiv:1605.02688.