KNN ile Optik Karakter Tanıma

Yusuf Çiçek

[1*] Beykent Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü, İstanbul, Türkiye,

(İlk Geliş Tarihi 06.01.2022)

Özet

Bilgisayarların işlem gücü kabiliyetleri geliştikçe insanoğlu yeni çözümler üretmeye başladı. Bu çözümlerin en başında optik karakter tanıma gelmekteydi. Bu projede KNN ile Optik karakter tanıma yapılmıştır. Sınıflandırma için %97, CER için %100 doğruluk ile çalışmaktadır.

Anahtar Kelimeler:OCR, OpenCV, KNN

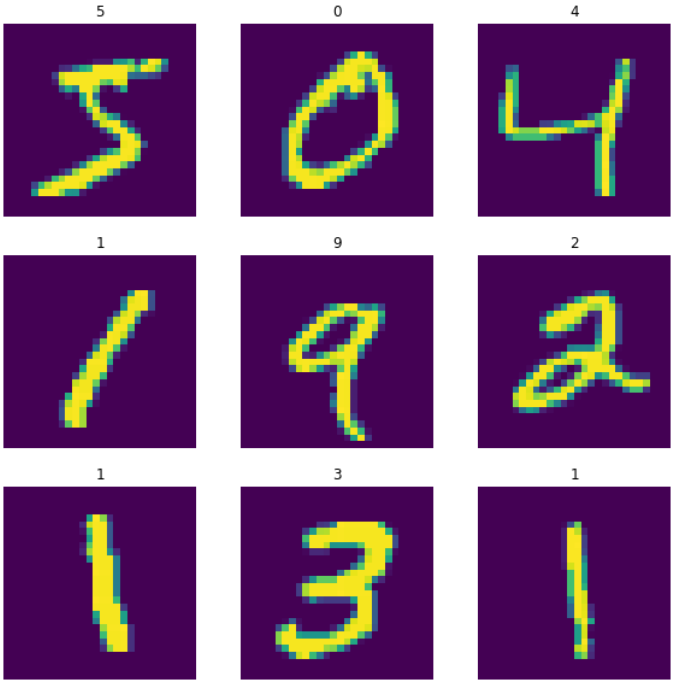
1. Giriş

Bilgisayarların işlem gücü kabiliyetleri geliştikçe insanoğlu yeni çözümler üretmeye başladı. Bu çözümlerin en başında optik karakter tanıma gelmekteydi. Bu problemin hazır bir dökümandan, el yazısından ve doğal resimlere kadar geniş bir yelpazesi vardır. Bu projede K - En Yakın Komşu algoritması ile el yazısı tanıma yapılmıştır. Optik karakter tanıma için en çok bilinen tesseract OCR KNN ve OpenCV temelli çalışmaktadır.

2. Materyaller ve Method

2.1 Veri Setinin Açıklanması

Bu projede tensorflow kütüphanesinin içinde bulunan MNİST veri seti kullanılmıştır. Bu veri setinde toplam 70000 adet resim vardır. Bu resimler rakamlardan oluşmaktadır. Her bir fotoğraf siyah-beyaz renkli ve 28'e 28 çözünürlüğündedir. Veri rakamlardan oluştuğundan ötürü her rakam kendi etiket değerini taşır. Yani "0" resminin etiket değeri "0" dır. Bundan dolayı bu veri setinde sınıflandırma kullanmak doğru olacaktır.



Şekil 1.1

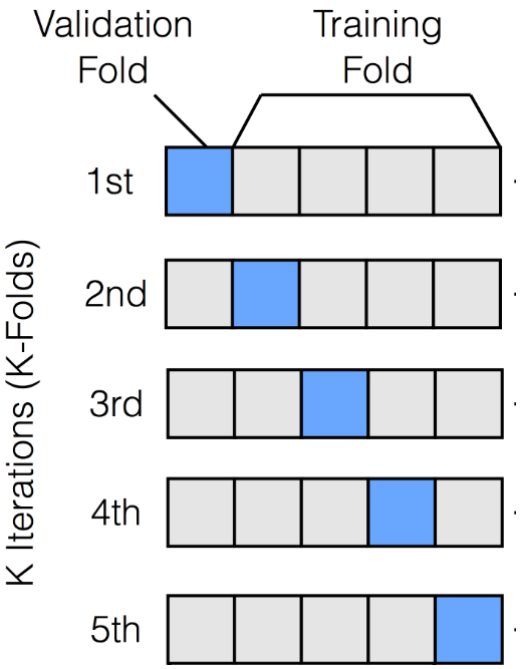
2.2 Kullanılacak Yöntemin Açıklanması

Makine öğrenmesi yöntemleri arasından KNN görüntü işlemede sıklıkla kullanılan bir yöntemdir. Bunun sebebi KNN veri setini öğrenmez ve uzaklığa göre çalışır. Her gözlem için uzaklık hesabı yapar ve en yakın N tane komşusuna bakarak sonuca karar verir. Uzaklık için Manhattan Öklid gibi uzaklık formülleri kullanılır.

$$d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

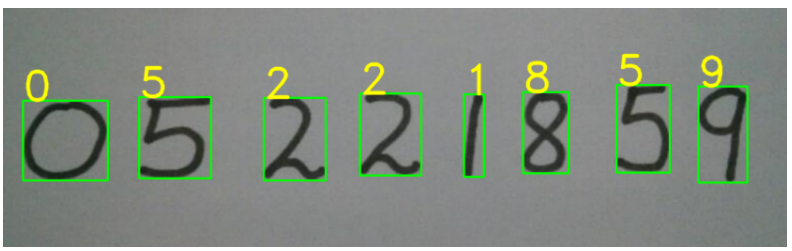
Şekil 1.2

Bu projede 1'den 20'ye kadar olan komşu sayıları için 5 katlı K - Katlı çapraz doğrulama yapılmıştır. En yüksek doğruluklu komşu sayısı için sonuç %97 ile 3'tür.



Şekil 1.3

Sıralı ifade ya da bir metin üzerinden optik karakter tanıma için her karakterin tespitinin yapılması ve KNN ile tanınması gerekmektedir. Karakterlerin tespiti için OpenCV ile kenar tespiti yapılır ve karakterlerin bulunduğu dikdörtgenler bulunur. Her dikdörtgenin ayrı karakter için sınıf değeri bulunur. Bu şekilde tüm işlemler tamamlanmış olur.



Şekil 1.4

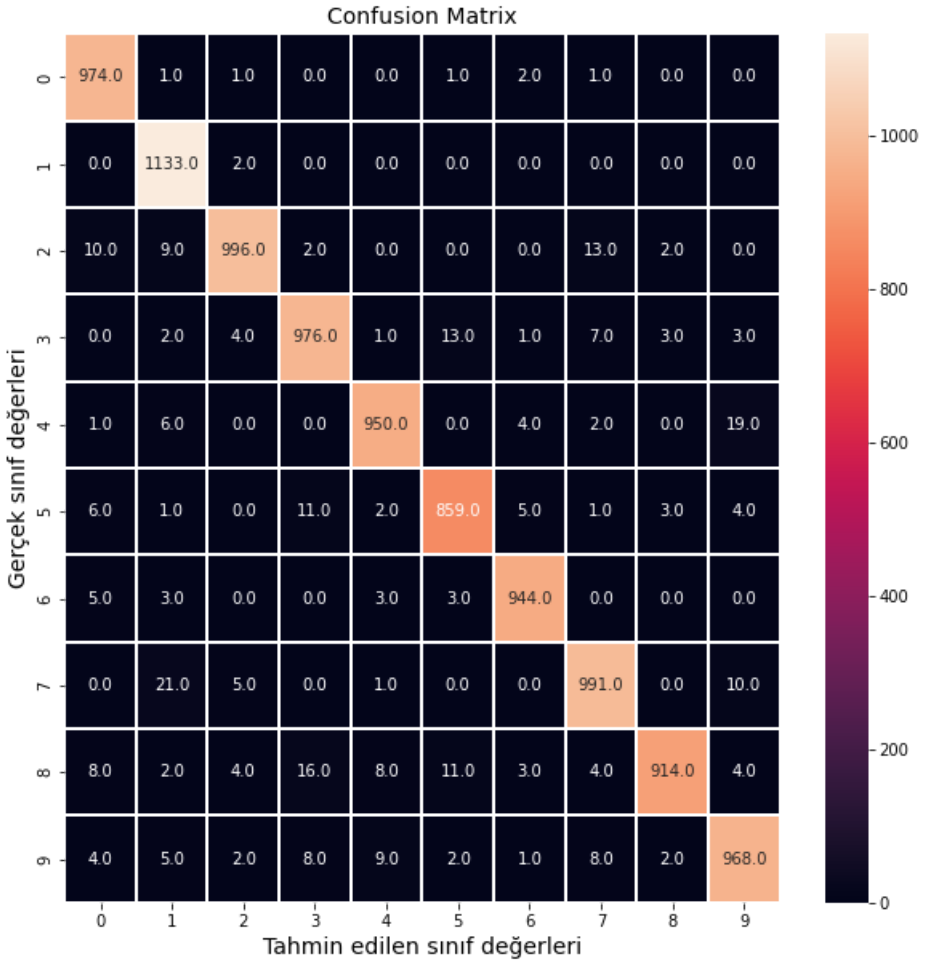
3. Araştırma Sonuçları ve Tartışma

3.1 Analiz için kullanılacak performans metrikleri

3.1.1 Karakter sınıflandırması için

Bu projede model başarımının ölçülmesi için test seti için 10000 resim ve eğitim seti için 60000 resim kullanılmıştır. Değerlendirmesi için accuracy score(Hata oranı), precision(hassasiyet), recall(kesinlik), f1-score, roc curve ve CER (Karakter Hata Oranı) kullanılmıştır.

	precision	recall	f1-score	support
0	0.97	0.99	0.98	980
1	0.96	1.00	0.98	1135
2	0.98	0.97	0.97	1032
3	0.96	0.97	0.96	1010
4	0.98	0.97	0.97	982
5	0.97	0.96	0.96	892
6	0.98	0.99	0.98	958
7	0.96	0.96	0.96	1028
8	0.99	0.94	0.96	974
9	0.96	0.96	0.96	1009
accuracy			0.97	10000
macro avg	0.97	0.97	0.97	10000
weighted avg	0.97	0.97	0.97	10000



Şekil 2.2

3.1.2 Optik Karakter tanıma için

Optik karakter tanıma için WER(Kelime Hata Oranı) ve CER(Karakter Hata Oranı) üzerinden performans metrikleri ölçülmektedir. Bu projede ve şekil 1.4 'deki örnekte CER metriğine bakılır.

$$CER = \frac{S + D + I}{N}$$

Şekil 2.3

CER formülü için S=Karakter değiştirerek, D=Karakter silerek ve I=Karakter ekleyerek referans değeri bulunmasıdır. Şekil 1.4 'deki örneğe bakılırsa CER %100'dür. Fakat tek bir resim için pek de gerçekçi bir sonuç elde edilemez.

4. Sonuç

KNN modeli sınıflandırmada %97 ve CER'de %100 doğrulukla çalışmıştır. Daha karmaşık verilerde doğru çalışmayacaktır. Bunun sebebi veri setinin büyüklüğünden ziyade çeşitliliği ve karakter tespitinin istenilen türde olmamasıdır. Popülerliğini devam ettiren yapay sinir ağları ile OCR modellerinin başarım oranları çok yüksek seviyelerdedir. Daha karmaşık ve daha zor OCR problemleri için başarımı yüksek yapay sinir ağları kullanılabilir.

5. Kaynakça

[1] Handwritten Recognition Using SVM, KNN and Neural Network
https://arxiv.org/ftp/arxiv/papers/1702/1702.00723.pdf
[2] https://github.com/tesseract-ocr/tesseract