

Derin Öğrenme ile Araç Sınıflandırma

Öz

Bu çalışmada Stanford Üniversitesi Araç veri seti kullanılarak Spark ortamında veri işleme adımları yapılmış ve ardından çeşitli CNN modelleri uygulanmıştır. Veri işleme olarak veri üretimi ile veriler çeşitlendirilmiştir. Model çalışmalarında ise ilk olarak 3 katmanlı CNN, 4 katmanlı CNN, 5 Katmanlı CNN ve son olarak MobileNetV2 uygulanmıştır. Modeller arasından en başarılı model MobileNetV2 olmuştur. Sonuçların geliştirilebilmesi için dengesiz veri seti, dengeli hale getirilebilir, görüntüler üzerine çeşitli filtreler denenebilir ve ResNet, VGG gibi modeller denenebilir.

Anahtar kelimeler: Derin Öğrenme, Görüntü İşleme, MobileNetV2, Görüntü Sınıflandırma

Car Classification with Deep Learning

Abstract

In this study, data processing steps were made in the Spark environment using the Stanford University Tool dataset, and then various CNN models were applied. Data has been diversified with data generation as data processing. In the model studies, firstly, 3-layer CNN, 4-layer CNN, 5-layer CNN and finally MobileNetV2 were applied. Among the models, the most successful model was MobileNetV2. In order to improve the results, the unbalanced data set can be balanced, various filters can be tried on the images, and models such as ResNet and VGG can be tried.

Keywords: Deep Learning, Image Processing, MobileNetV2, Image Classification

1. Giriş (Introduction)

Günümüzde artan araç filosunun yanı sıra izleme güvenlik kameralarının sayısındaki artışla birlikte, trafik verilerinin yakalanması için araç tanınmanın otomatikleştirilmesi gerekmektedir [1]. Otomatik araç sınıflandırma teknikleri yaygın olarak kullanılmaktadır. Aslında, otoyol gözetimi, geçiş ücreti toplama, trafik sıkışıklığı tespiti ve diğerleri gibi birçok gerçek dünya uygulaması bu tekniklerle otomatikleştirilebilir.

Araç sınıflandırması için bazıları kısıtlayıcı senaryoları çözmeye başarılı olan çeşitli yöntemler önerilmiştir. Bununla birlikte, kalan diğer zorluklar arasında aydınlatma, görüntü ölçeği, görüntü kalitesi, boyut ve renkteki değişimler sayılabilir. Bu yazıda, derin öğrenme teknolojisine dayalı bir araç tipi sınıflandırma sistemi sunuyoruz. Bu sistem iki tekniğe dayalı olarak oluşturulmuştur: Veri büyütme ve evrişimli sinir ağları (CNN). Bu çalışma, yol güvenliğini iyileştirmenin bir yolu olarak derin öğrenme yoluyla yol gözetimini otomatikleştirmeyi amaçlayan küresel bir uygulamanın parçasıdır. Bu uygulama, belirli bir araç tipinin algılanmasına göre trafik ışıklarını otomatik olarak kontrol eder; örneğin, acil bir durumda daha uzakta bir ambulans veya polis aracı algılanır algılanmaz, sıkışık bir yolda yeşil ışığa geçmek. Bu, yol tıkanıklığı için otomatik bir sistem kurmamızı sağlayacaktır. Literatür video gözetimi açısından zengindir. Wang ve diğerleri [2], video gözetim

kameralarından alınan görüntülerde aracı sınıflandırmak için hem HOG tanımlayıcısını hem de SVM sınıflandırıcısını kullanır. Llorca ve diğerleri [3] ayrıca trafik görüntülerinde araç logolarını tanımak için HOG tanımlayıcısı ve SVM sınıflandırıcı kullanır; bu örnekler, sabit görüntüler üzerinde sınıflandırmada HOG-SVM birleştirmenin başarısını yansıtmaktadır [4]. Son yıllarda, derin sinir ağları bilgisayarla görme, makine çevirisi, konuşma tanıma vb. gibi birçok alanda kullanılmaktadır. Evrişimsel sinir ağının (CNN) kullanılması nedeniyle bu alanlarda devrim niteliğinde sonuçlar elde edilmiştir. AlexNet, Krizhevsky ve diğerleri [5] tarafından önerildikten sonra, CNN'ler görüntü sınıflandırmada daha iyi performans gösterdi ve büyük ölçüde nesne algılama [6], video sınıflandırma [7] ve segmentasyon [8] için uygulandı. Bu ilerlemeler aynı zamanda yüksek performanslı CNN ağları üzerine araştırmalara yeni bir odaklanmaya da katkıda bulunmuştur. Bu ağların mimarileri de daha derin ve daha geniş yapılar kullanılarak performanslarının arttığını gördü. Simonyon et al. bilgisayar görüşünde derin mimariyi kullanarak araştırmayı kolaylaştıran önerilen VGGNet [9]. Szegedy ve diğerleri [10], ImageNet Challenge için yeni bir sanat eseri tanımlayan başlangıç modüllerini içeren GoogleNet'i sundu.

* Sorumlu yazar. Tel: 123456789

E-posta adresi:

Alındı :
Revizyon :
Kabul :

2. Materyal ve Yöntemler

2.1. Veri Kümesi

Bu veri seti, Stanford Üniversitesi'nin paylaştığı çeşitli araç sınıflarını içeren veri setidir. Arabalar veri

kümesi, 196 araba sınıfının 16.185 görüntüsünü içerir. Veriler 8.144 eğitim görüntüsüne ve 8.041 test görüntüsüne bölünür, burada her bir sınıf kabaca 50-50'ye bölünür. Sınıflar genellikle Marka, Model, Yıl düzeyindedir. Veri kümesi içerisinde örnek görüntüler Şekil 1'de verilmiştir.



Şekil 1. Çeşitli araçların görüntüsü

2.2. Evrimsel Sinir Ağı

Evrimsel sinir ağı (CNN), bir görüntü sınıflandırma modelini oluşturan ileri beslemeli bir ağıdır [13]. Yapısı, biyolojik beynin birincil görsel korteks tedavi sürecinden esinlenmiştir [14]. Bu ilham, bitişik piksellerin sürekli bölgelerini benzersiz değerlerle birleştiren verileri alt örnekleme kırımlı katmanların ekleneceği yatmaktadır [15]. Bu alt örnekleme, Şekil 4'te gösterildiği gibi her bir evrim katmanının çıktısını oluşturur.

Bir CNN'nin yapısı, bir veya daha fazla ardışık kırımlı katmandan ve ardından düzenli tam bağlantılı YSA (Yapay Sinir Ağları) katmanlarından oluşur. Bu yapı Şekil 2'de gösterilmektedir. Kırımlı katmanlar, karakteristik çıkarma katmanları olarak hareket eder ve son tam olarak bağlı YSA katmanları bir sınıflandırma modülü olarak çalışır. Konvolüsyon katmanları üç farklı katmandan oluşur; filtre bankası katmanı, doğrusal olmayan katman ve karakteristik havuz katmanı [16]. Her evrim katmanının girdisi ve çıktısı, ilk iki boyutun görüntünün yüksekliği ve genişliği olduğu ve üçüncü boyutun özellik kartlarının sayısı olduğu bir 3B matristir. Örneğin bir RGB görüntüsü için karakteristik kartlar üç olacaktır; her renk kanalı için bir tane. Böylece, böyle bir görüntü, ilk girişte her biri tek bir renk kanalına sahip üç ayrı görüntü ile temsil edilecektir [17]. Çıktı özelliği haritaları tanımsızdır ve eğitime bağlıdır. Ağ ne kadar derinse, çıktındaki işlevsellik

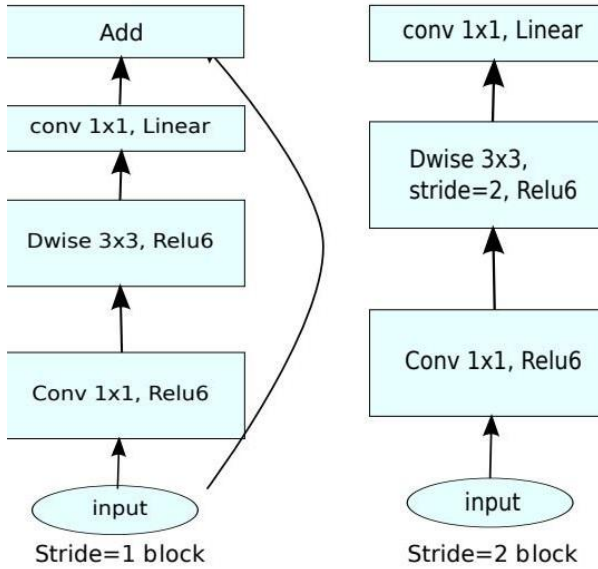
düzeyi de o kadar yükselir [16]. Filtre bankası katmanı, her biri belirli bir özelliği tanıyan birkaç sürüklenbilir çekirdekten yapılmıştır. Bu özellik çıkarımı, her giriş konumunda gerçekleştirilir ve bu nedenle evrim katmanı, özelliğin giriş görüntüsündeki konumuna duyarlı değildir [16]. Aktivasyon katmanı, filtre bankası katmanının tüm çıktısı boyunca doğrusal olmayan bir transfer işlevi uygulayacaktır [16]. Havuzlama katmanı, evrim katmanının alt örneklerini oluşturan katmandır. Bu alt örnekleme, her bir özellik haritası için gerçekleştirilir ve haritanın daha düşük çözünürlüklü bir temsiliyle sonuçlanır, ancak gelişmiş gürültü direnci ile. Örneğin, özellik haritasındaki her bir değeri belirli bir aralıktaki komşularının ortalaması olarak tanımlayarak alt örnekleme yapılabilir [16]. Son evrim katmanı tamamlandığında çıktı, sınıflandırma modülüne beslenecektir. Bu modül, tamamen bağlı birkaç katman içerir. Bu katmanlar, görüntünün ait olduğu sınıfı veya sınıfları belirten bir çıktı ile sonuçlanır [16].

2.3. MobileNetV2

MobileNet mimarisi görüntülerden konvolüsyonel filtrelerle özellik çıkarımı aşamasında, standart konvolüsyonel operasyonu Derinlemesine Ayrılabilir Konvolüsyon (Depthwise Separable Convolutions) tekniğini kullanmaktadır. Bu teknik sayesinde standart konvolüsyon işlemine nazaran sekiz veya dokuz kat daha az parametreyle özellik çıkarımı yapılabilmektedir. 2019 yılında ise MobileNet mimarisi

geliştirilerek, daha hızlı ve daha verimli olacak şekilde güncellemeler yapılmıştır. 1*1 konvolüsyonlar kullanılarak özellik haritalarının boyutu daraltılmıştır. Ayrıca, ResNET (Szegedy et al., 2017) modellerinde de yer alan bağlantı atlama (skip connection) tekniği sayesinde daha hızlı bir hesaplama süreci sağlanmıştır. MobileNetV2 mimarisi kullanılarak ImageNet veri seti üzerinde gerçekleştirilen testlerde, kendisinden çok daha fazla parametreye sahip derin mimarilere oldukça yakın performans gösterdiği saptanmıştır.

MobileNetV2 ağı kendi içinde 16 blok katmanına sahiptir. Her bir katmanda her blok birbiriyle aynı değil ancak bloklarda Batch Normalization, Conv2D, Depthwise Conv2D, Zero Padding2D, Expansion Layer, Projection Layer gibi katmanlar bulunmaktadır. Ayrıca katmanlarda residual connection bulunmaktadır ve her bir blok Bottleneck Residual Block olarak isimlendirilir. MobileNetV2 ana blok yapısı aşağıdaki şekil 3’de gösterilmiştir.



Şekil 3. MobileNetV2 mimarisi

2.4. Veri Üretimi

Veri üretiminin otomatik öğrenme modellerinin performansını iyileştirmeye katkıda bulunduğu kabul edilmektedir [18]. Ayrıca, sinir ağlarına ve derin öğrenme modellerine dayalı araç sınıflandırma gibi sınıflandırma görevleriyle çalışırken veri hazırlama gereklidir. Şekil 4, uygulamamızın amacı için gerçekleştirilen bazı Veri Üretme örneklerini göstermektedir.

3. Uygulama ve Sonuçlar

Veri üretiminde çeşitli geometrik dönüşümler ile veri üretimi adımını tamamladıktan sonra model için veri seti eğitim, validasyon ve test setlerine bölünmüştür. Çalışmada 7018 görüntü eğitim için, 1126 görüntü validasyon için ve 8041 görüntü test için ayrılmıştır. İlk olarak 3 katmanlı CNN uygulanmıştır. Bu modelin parametre sayısı 4.591,876’dır. Bu model sonucunda başarımlar 0.04 olarak elde edilmiştir. İkinci aşamada 4 katmanlı CNN uygulanmıştır ve bu model sonucunda başarımlar 0.10’dur. 4 katmanlı modelin parametre sayısı 1.088,804’dür. Üçüncü aşamada 5 katmanlı CNN uygulanmıştır ve bu model sonucunda başarımlar 0.09’dur. 5 katmanlı modelin parametre sayısı 263,876’dır. Son olarak MobileNetV2 veri setine uygulanmıştır. Bu modelin başarımlarını 0.64’dür. MobileNetV2 parametre sayısı 957,300’dir. Tablo 1’de modellerin başarımlar değerleri gösterilmiştir.

Tablo 1. Başarımlar Değerleri

| Model | Başarımlar |
|-------------|------------|
| 3-Layer | 0.04 |
| 4-Layer | 0.10 |
| 5-Layer | 0.09 |
| MobileNetV2 | 0.64 |

İlk üç model 30 epoch ile eğitilmiştir. MobileNetV2 modeli ise 80 epoch eğitilmiştir. Modeller için learning rate 0.0001 ve batch size 32 ortak olarak kullanılmıştır. Modellerde optimizer olarak Adam, loss parametresi olarak ise categorical crossentropy kullanılmıştır.

Modellerde ariyeten Dropout, Maxpool2D gibi katmanlar kullanılmıştır. Maxpool2D, ağırlık azaltımına yaramaktadır. Dropout ise rastgele olarak belli bir oranda bilgiyi bırakmaktır. Aşırı uydurmayı önlemek için kullanılır. İlk 3 adımdaki basit CNN modellerine nazaran MobileNetV2’de farklı olarak Depthwise Conv2D, Expansion Layer, Projection Layer, Batch Normalization ve Global Average Pooling2D gibi katmanlar kullanılmıştır. Bu katmanları açıklamak gerekirse ; DepthWise Conv2D, Konvolüsyon işleminde gelen veri fotoğraf formatında olduğu için bu verinin 3 kanalı vardır. Bunlar da RGB olarak adlandırılan fotoğraftaki kırmızı tonları gösteren R bileşeni yeşil tonları gösteren G bileşeni ve mavi tonları gösteren B bileşenidir. Conv2D’den farklı olarak bu üç kanalın her biri ayrılarak her birine uygun olduğu kanalla eşleşen çekirdek(kernel) matrisi uygulanır.



Şekil 4. Veri üretimi sonucu oluşan görüntüler

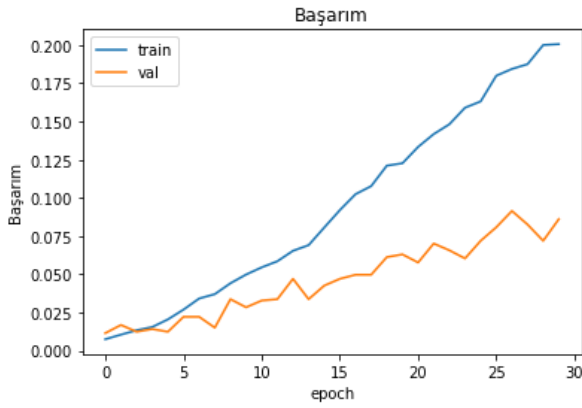
Global Average Pooling2D, ilgilenilen 2 boyutlu bir matrisin her bir elemanının toplanıp eleman sayısına bölünmesi işlemi olarak adlandırılır. Pooling işlemlerinin genel amacı boyut küçültmedir ve küçülen boyuta karşılık matrisin nasıl doldurulacağı pooling işleminin tipi ile belirlenir. Eğer “average pooling” kullanılıyorsa ortalama “max pooling” kullanılıyorsa pool_size kadarlık bir matristeki maksimum değer alınır. Expansion Layer , temel amacı gelen verinin kanal sayısını büyütmektir. Kanal sayısı expansion factor parametresinde verilen değerle çarpılarak büyür. Batch Normalization, genel anlamıyla bir yapay sinir ağının gizli katmanlarındaki parametreleri normalizasyon işlemine tabi tutar. Projection Layer, temel amacı gelen verinin kanal sayısını küçültmektir. Aktivasyon fonksiyonu olarak ise softmax kullanılmıştır. Softmax çoklu sınıflandırma problemlerinde tercih edilmektedir.

Modellerin eğitim ve validasyon başarımları grafikleri Şekil 5’de verilmiştir. Tüm modeller için grafiklere bakıldığında aşırı uyum problemi gözükmemektedir.

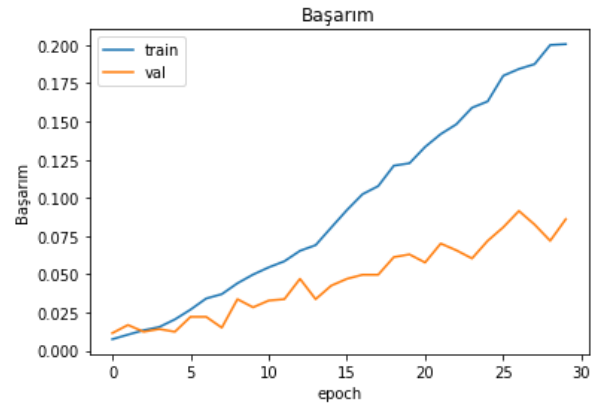
4. Gelecek Çalışmalar

Bu çalışmada Stanford Üniversitesi’nin araç verilerini kullanarak evrişim aktivasyon fonksiyonuna dayalı bir araç tipi sınıflandırma sistemi uygulanmıştır. Nesne algılama ve tanımadaki etkinliği ile tanınan evrişimli sinir ağını kullanarak özellikler çıkarılmıştır. Çalışma başarımının artırılması için dengesiz veri seti dengelenebilir, veri çeşitliliği veri üretimi yöntemleri ile daha fazla artırılabilir ve ResNet, VGG gibi daha fazla

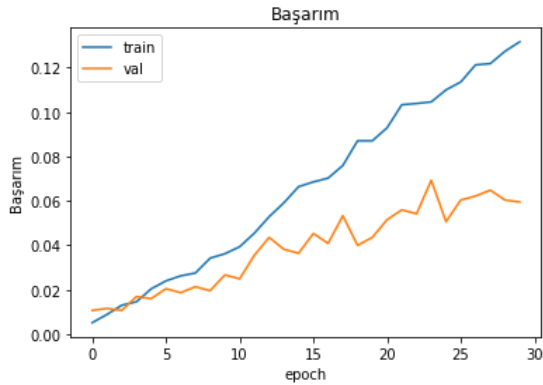
parametre içeren yüksek genelleme kapasitesine sahip modeller uygulanabilir.



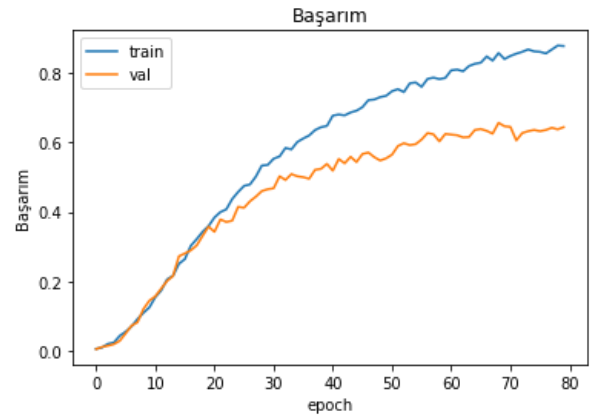
(a) 3 Layer



(b) 4 Layer



(c) 5 Layer



(d) MobileNet V2

Şekil 5. Model eğitim grafikleri

Kaynakçalar

- [1] A. H. S. Lai, G. S. K. Fung, and N. H. C. Yung, Vehicle type classification from visual-based dimension estimation, in Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE, 2001, pp. 201- 206.
- [2] Y. C. Wang, C. C. Han, C. T. Hsieh, and K. C. Fan, Vehicle type classification from surveillance videos on urban roads, in Ubi-Media Computing and Workshops (UMEDIA), 2014 7th International Conference on, 2014, pp. 266-270.
- [3] D. F. Llorca, R. Arroyo, and M. A. Sotelo, Vehicle logo recognition in traffic images using HOG features and SVM, in Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on, 2013, pp. 2229-2234.
- [4] N. Dalal and B. Triggs, Histograms of oriented gradients for human detection, in Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, 2005, vol. 1, pp. 886- 893.
- [5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, in Advances in neural information processing systems, 2012, pp. 1097-1105.
- [6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, Rich feature hierarchies for accurate object detection and semantic

segmentation, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580-587.

- [7] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, Large-scale video classification with convolutional neural networks, in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2014, pp. 1725-1732.
- [8] J. Long, E. Shelhamer, and T. Darrell, Fully convolutional networks for semantic segmentation, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431-3440.
- [9] K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv Prepr. arXiv1409.1556, 2014.
- [10] C. Szegedy et al., Going deeper with convolutions, 2015.
- [11] G. E. Hinton and R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science (80-.),2006, vol. 313, no. 5786, pp. 504-507.
- [12] Convolutional Neural Networks (LeNet). [Online]. Available: <http://deeplearning.net/tutorial/lenet.html>.

- [13] M. D. Zeiler and R. Fergus, Visualizing and understanding convolutional networks, in European conference on computer vision, 2014, pp. 818–833.
- [14] D. C. Ciresan, U. Meier, J. Masci, L. Maria Gambardella, and J. Schmidhuber, Flexible, high performance convolutional neural networks for image classification, in IJCAI Proceedings-International Joint Conference on Artificial Intelligence, 2011, vol. 22, no. 1, p. 1237.
- [15] N. Le Roux and Y. Bengio, Representational power of restricted Boltzmann machines and deep belief networks, Neural Comput., 2008, vol. 20, no. 6, pp. 1631–1649.
- [16] Y. LeCun, K. Kavukcuoglu, and C. Farabet, Convolutional networks and applications in vision, in Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on, 2010, pp. 253–256.
- [17] Convolutional Neural Network. [Online]. Available: <http://ufldl.stanford.edu/tutorial/supervised/ConvolutionalNeuralNetwork/>. [Accessed: 15-Dec-2017].
- [18] P. Y. Simard, D. Steinkraus, J. C. Platt, and others, Best practices for convolutional neural networks applied to visual document analysis., in ICDAR, 2003, vol. 3, pp. 958–962.