



Videoların Derin Öğrenme ile Sınıflandırılarak Filtrelenmesi

Murat Kazanç^{1*}, Tolga Ensari², Mustafa Dağtekin³

¹ İstanbul Üniversitesi - Cerrahpaşa, Mühendislik Fakültesi, Bilgisayar Mühendisliği, İstanbul Bölümü, Türkiye (ORCID: 0000-0002-8405-0181)

² Arkansas Tech University, College of Engineering & Applied Science, Dept of Computer and Information Science, Arkansas, USA (ORCID: 0000-0003-0896-3058)

³ İstanbul Üniversitesi - Cerrahpaşa, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, İstanbul, Türkiye (ORCID :0000-0002-0797-9392)

(International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) 2021 – 11-13 June 2021)

(DOI: 10.31590/ejosat.952481)

ATIF/REFERENCE: Kazanç, M., Ensari, T., & Dağtekin, M. (2020). Videoların Derin Öğrenme ile Sınıflandırılarak Filtrelenmesi. *Avrupa Bilim ve Teknoloji Dergisi*, (26), 338-342.

Öz

Bu çalışmada derin öğrenme metodu olan evrişimli sinir ağları (Convolutional Neural Networks - CNN) ve transfer öğrenme metodu kullanılarak tütün mamulleri, alkollü içecek ve silah gibi istenmeyen nesnelerin tespitini ve sınıflamasını yapan bir model geliştirilmiştir. Bu model Tensorflow JS'e dönüştürülerek, internet tarayıcısı için bir eklenti olarak geliştirilmiştir. Bu eklenti ile izlenen videolardan anlık görüntüler alınarak eğitilen model üzerinde sınıflandırmalar yapılmıştır. Videolar üzerinde yapılan sınıflamalarda gerekli görülen sonuçlar, Google tarafından sağlanan bir bulut hizmeti olan Firebase'in RealTimeDatabase'e kaydedilmiştir. Kaydedilen veri tabanı kullanılarak daha önceden kötü içerik tespiti yapılmış videoların engellenmesi sağlanmıştır. Tarayıcıdan yapılan tespitlerin son 25 tanesi bilgilendirme amaçlı kullanıcı tarafından görüntülenebilmektedir. Bu çalışmada, izlenen videolardan anlık görüntüler alınarak model ile sınıflama yapılmıştır. Gerekli hallerde videonun bilgisi veri tabanına eklenebilmektedir ve veri tabanına kayıtlı videoların görüntülenmesi filtre edilebilmektedir. Geliştirilen sistem hem fiziksel cihazlar hem de emülatör aracılığıyla test edilmiştir. CNN ile geliştirilen derin öğrenme modelinin ağ yapısı oluşturulmasında iki yol benimsenmiştir. Birincisinde, tüm ağ modeli tarafımızca oluşturduğumuz modeldir. Bu modelde, parametre sayısı 7.752.707 adettir ve %86,75 eğitim ve %88,02 test doğruluğu elde edilmiştir. İkinci olarak, transfer öğrenme metodu kullanılarak, başarısı literatürde kanıtlanmış modellerden olan MobileNetV2 tercih edilmiştir. Çıkış katmanları düzenlenmiş bu modelde eğitilebilir 593.155 adet ve toplamda 2.852.675 adet parametre ile %65,34 eğitim ve %50,35 test doğruluğu elde edilmiştir. Yapılan çalışma sonucunda, video içeriklerini filtrelemek için CNN modelinin daha verimli olacağı bulgusuna ulaşılmıştır.

Anahtar Kelimeler: Evrişimli sinir ağları (CNN), derin öğrenme, video filtreleme, transfer öğrenme.

Filtering Videos by Classification with Deep Learning

Abstract

In this study, a model that detects and classifies unwanted objects such as tobacco products, alcoholic beverages and weapons was developed using Convolutional Neural Networks (CNN), which is a deep learning methods and transfer learning. This model was converted to Tensorflow JS and developed as an add-on for an Internet Browser. With this add-on, snapshots were taken from the watched videos and classifications were made using the trained model. The results that are deemed necessary in the classification of the videos were recorded to Firebase's RealTimeDatabase, a cloud service provided by Google. By using the recorded database, videos that had previously been detected with harmful content were blocked. The last 25 of the definitions made from the browser can be viewed by the user for informational purposes. In this study, snapshots from watched videos were taken and classified with the model. If necessary, information about the video can be added to the database, and videos saved in the database can be filtered. The developed system has been tested through both physical devices and emulator. Two ways have been adopted in creating the network structure of the model developed with CNN. First, the entire network model is the one that is created by us. In this model, the number of parameters was 7,752,707 and the accuracy of 84.84% training and 79.77% testing was achieved. Second, MobileNetV2, which is one of the models whose success has been proven in the literature, was preferred using the transfer learning method. With 593,155 trainable parameters and 2,852,675 parameters in total, 65.34% training and 50.35% test accuracy was achieved in this model with output layers arranged. As a result of the study, it was found that the CNN model would be more efficient in filtering video content.

Keywords: Convolutional neural networks (CNN), deep learning, video filtering, transfer learning.

* Sorumlu Yazar: İstanbul Üniversitesi - Cerrahpaşa, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, İstanbul, Türkiye, ORCID: 0000-0002-8405-0181, murat.kazanc@ogr.iuc.edu.tr

1. Giriş

Derin öğrenme ham veriden kademeli olarak daha yüksek seviyeli özellikler çıkarmak için çoklu katmanlar kullanan bir makine öğrenmesi alt sınıfı olarak tanımlanmaktadır. Örneğin, görüntü sınıflandırmada alt katmanlar kenarları tanımlarken yüksek katmanlar rakam, harfler veya yüzler gibi kavramları tanımlayabilir. Bu çalışmada, derin öğrenme kullanılarak video sınıflandırma modeli geliştirilmiştir. Sınıflandırma için veri kümesindeki belirgin özelliklerinin ortaya çıkarılması gerekir. Bunun için evrişimli sinir ağları (convolutional neural networks) geliştirilmiştir. Böylece, özellik çıkarımı yapılabilmektedir. Örnek olarak eğriler, kenarlar gibi özelliklerin belirginleşmesi bu ağlar ile sağlanabilir (LeCun, Bengio, & Hinton, 2015).

Tarayıcı eklentileri görüntülenen internet sitesinin içerisine, normalde sitenin içeriğinde olmayan yeni özellikler eklemek için kullanılabileceği gibi tarayıcıda bulunmayan özellikler eklemek için de kullanılmaktadır. Örneğin, tarayıcılar için web sitelerindeki reklamları engelleyen eklentiler yoğun bir şekilde kullanılmaktadır. Bu çalışmada, normalde görüntülenen sitenin içeriğinde olmayan javascript kodları eklenerek işlemler gerçekleştirilmiştir. Tarayıcı eklentisi, html sayfa içerisindeki video etiketini bularak oynatılan video içerisindeki görüntülerde sınıfların bulunmasıdır. Yapılan tespitlerle bir bulut veri tabanında depolanarak bir katalog oluşturulmuştur. Ayrıca, HEVC (Yüksek verimli video kodlaması-High Efficiency Video Coding) kodlamasında video sıkıştırma işlemine evrişimli sinir ağlarının kullanılabileceği gösterilmiştir (Jia, ve diğerleri, 2019). Derin öğrenme ve Markov karar süreçleri kullanılarak IP protokolü üzerinden yayın akışı gerçekleştiren beş yayıncının (YouTube, YouTube TV, Netflix, Amazon Prime ve HBO) veri paketlerinin makine öğrenmesi kullanılarak sınıflandırılması da mümkündür. Benzer çalışmada, farklı hizmetlerden gelen benzer video akış trafiği verilerinin doğru bir şekilde sınıflandırılması sağlanmıştır (Shaout & Crispi, 2020). Google Research tarafından 8 milyon video kullanılarak 200 farklı etiket ile veri kümesi oluşturulmuştur. Bu veri kümesi araştırmacıların erişimine açıktır (Abu-El-Haija, ve diğerleri, 2016).

Videoları sınıflandırmak için meta verileri, arama motoru ve sosyal medya verileri değerlendirilmesi mümkündür. Videoyu sınıflandırırken kare analizi, ses ve metin verileri kullanılmaktadır. Eğer video içerisinde alakasız, nefret, suça teşvik, yetişkin içerik varsa videonun yayınlanmasına izin verilmemektedir. Bunun için bağlam kullanılabilir (Hammam, 2019). Youtube benzeri ortamlarda görüntülenen videonun meta verilerine bakılması ve NLP metotları kullanılması makul görülmektedir. Video yüklendiği zaman otomatik olarak ön tanımlı kategorilerden birisine atanması ve ek açıklamalar oluşturulması önemli bir özellik olabilmektedir. Video sınıflanırken görsel, işitsel, açıklama, etiketler, başlık ve yorumlar dikkate alınmalıdır. Bu sayede, arama sonuçları iyileştirilmesi mümkün olacaktır (Chen, Chang, & Yeh, 2017).

Derin öğrenme video sınıflaması için çözümler sunmaktadır. Son dönemde yapılan çalışmalarda, ikili ağ yapısı önerilmektedir. Birinci ağ yapısı, videonun tamamında hareketlerin algılanmasına odaklanırken, ikinci ağ ile videodan kare görüntüleri üzerinden nesnelere tespit edilmesi sağlanmaktadır. Buradan hareketle, iki ağın sonuçlarından video içeriği anlaşılabilir. Herhangi bir eylemin zaman ve mekânsal olarak anlaşılabilmesi için birkaç hareketin arka arkaya yapılması gerekmektedir. Bu sebeple, ek olarak hafıza özelliği olan model kullanılması performansı

artıracaktır. Bununla ilgili bir diğer öneri ise denetimsiz öğrenme metotlarının video sınıflamada kullanılabileceği olmuştur. Veri kümesindeki videolar, denetimsiz öğrenme metotları ile ön gruplama yapıldıktan sonra asıl sisteme verilmesi ile detaylı sınıflandırma yapılabileceği ifade edilmektedir (Wu, Yao, Fu, & Jiang, 2017). CNN kullanılarak videolar üzerinde sınıflandırma yapılması için model eğitimi uzun süre alabilmektedir (Karpathy, ve diğerleri, 2014). Video sınıflamada iyi bir performans elde etmek için büyük miktarda etiketli veri gerekmektedir. Bu bağımlılığı en aza indirmek için yarı denetimli öğrenme yöntemi önerilmektedir. Sadece %20 ile %50 arasında değişen etiketli örneklerden, önerilen yaklaşım ile denetimli eğitilen CNN'lerden potansiyel olarak daha iyi performans göstermektedir (Jing, Parag, Wu, Tian, & Wang, 2021). UCF101 veri kümesi ile ön eğitilmiş modellerin kullanıldığı referans çalışmada, videoda yapılan spor aktivitesinin sınıflandırılması gerçekleştirilmiştir. Bu sayede, sporcuların kendilerine uygun antrenmanları bulmaları ve performanslarını artırmalarını hedefleyen bir çalışma yapılmıştır (Ramesh & Mahesh, 2020). Bir diğer çalışmada ise videolar içerisinde yetişkin içeriği tespit edilmesine yönelik yapılan çalışmada video süresi boyunca belli periyotlarla insan cildi segmentasyonu yapılmıştır. İnsan cildi yoğunluğuna bakılarak içerik sınıflandırılması gerçekleştirilmiştir (Nugroho, Hardiyanto, & Adji, 2015).

Kamu güvenlik kameralarından anomali (suç, kavga, trafik kazası vb.) tespit etmek için derin öğrenme kullanılabilir. Anomali tespitinde, bir anomaliyi tespit eden algoritma başka bir anomaliyi tespit etmekte başarısız olabilmektedir. Sistemin çevrimiçi çalışabilmesi için donanım ve yazılım (algoritma) birlikte uyumlu olarak tasarlanması gerekmektedir (Nayak, Pati, & Das, 2021). Kapalı devre televizyon kamera sistemleri tarafından üretilen büyük verinin (big data) işlenmesiyle, ilgili sorunları çözmek için kamera konumu ve görüş alanı meta verilerini, mobil nesnelere yönlendirmelerini ve video içerik analizi algoritmalarından (örneğin mobil nesnelere tespiti ve hareketi) meta verileri birleştiren bir yaklaşım tanımlanmıştır. Bu metot ile araştırmacıların insan veya nesne içermeyen kameraların görüntüleri kaldırılarak, arama süresi kısaltılmaktadır (Panta, Qodseya, & Péninou, 2018). Videolar üzerinde duman tespit edilmesi ile ilgili yapılan diğer bir çalışmada transfer öğrenmeden faydalanılmıştır. Videoda sabit kalan kısımlar yerine hareket halinde olan yapıların görüntülerinden duman bilgisinin olup olmadığı derin öğrenme ile tespit edilmeye çalışılmıştır (Jia, ve diğerleri, 2021).

Videodan silah tespiti yapılan bir çalışmada ise başarılı kanıtlanmış modellerden VGG16 (Simonyan & Zisserman, 2014), InceptionV3 (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2015), Inception-ResNetV2 (Szegedy, Ioffe, Vanhoucke, & Alemi, 2016), SSDMobileNetV1 (Howard, ve diğerleri, 2017), Faster RCNN (Ren, He, Girshick, & Sun, 2016), YOLO V3 ve YOLO V4 (Bochkovskiy, Wang, & Liao, 2020) algoritmaları kullanılarak gerçekleştirilen deneylerde, en performanslı modelin YOLO V4 olduğu görülmüştür. Transfer öğrenme için ImageNet (ImageNet, 2021) ve COCO (COCO, 2021) kullanılmıştır. Bu çalışma ile video üzerinde gerçek zamanlı nesne tespiti yapılabileceği gösterilmiştir (Bhatti, Khan, Aslam, & Fiaz, 2021).

2. Derin Öğrenme ve Deneysel Çalışmalar

2.1. Veri Kümesi

Tasarlanan sistemdeki her bir sınıfı belirlemek için fazla sayıda görüntüye ihtiyaç bulunmaktadır. Yine her bir sınıf için yaklaşık olarak aynı sayıda görüntü olması en ideal durumdur. Bu

çalışmada, üç adet sınıf olacak şekilde tasarım yapılmıştır. Bu sınıflar tütün mamulleri, silah ve alkollü içeceklerdir. Yapılan araştırmalar sonucunda öncelikle Kaggle (Kaggle, 2021) internet sitesinde tütün mamulleri (Lee, 2019), silah (Sasank, 2019) ve alkollü içecekler (Nouisser, 2019) için veri kümeleri temin edilmiştir. Görsel olarak veri seti üzerinde yapılan incelemelerden sonra sınıflardaki resim sayıları şu şekilde olmuştur: tütün mamulleri sınıfı için 2147 resim, silah sınıfı için 728 resim ve alkollü içecekler sınıfı için 2928 resim. Özellikle, silah sınıfı için veri kümesinde görüntü sayısının yetersiz olmasından dolayı veri kümesini geliştirmeye çalışılmıştır. Bu amaçla öncelikle arama motorlarının görsel arama sonuçlarından faydalanılmıştır. Ayrıca Python programlama dilinde yazılmış “Bulk Bing Image Downloader” (Ostrolucky, 2020) uygulaması ile Bing arama motorunda verilen anahtar kelimelerimiz silah ve gun arama sonuçlarında gelen görüntülerin kaynak dosyaları elde edilmiştir.



Şekil 1. Veri kümesinden örnek tütün mamulleri, silah ve alkollü içecek görüntüleri

2.2. CNN Modeli ve Eğitimi

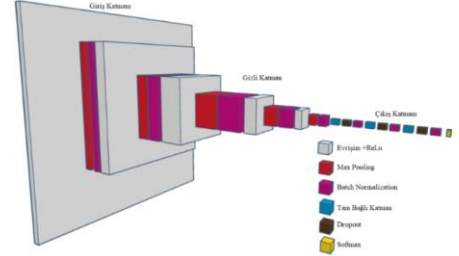
Bu çalışmada, ilk olarak ağ yapısını bizim oluşturduğumuz model için Sıralı (Sequential) metodu ile başlanmıştır. Modelimizin giriş katmanını oluşturacak ilk Conv2D (evrişim) katmanımızda 256 adet nöron ve 2x2'lik bir filtre tanımladıktan sonra 224x224x3 olan giriş biçimimizi belirttik. Bu parametre ile sistemin eğitiminde ve eğitim sonrasında tahmin için modele verilecek resimlerin pikselleri bu tensöre dönüştürülmelidir. Bu katman için aktivasyon fonksiyonu “Relu” seçilmiştir. Sonrasında tanımlanan “Max Pooling2D” ile 2x2 ortaklama (pooling) işlemi yapılmakta ve girişe uygulanan resim verisi yarılanarak 111x111x256 boyutuna indirilmektedir. Son olarak, batch normalizasyonu işlemi ile bu ara katman tamamlanmıştır.

Sonraki ara katman Conv2D' da 512 adet nöron ve 2x2'lik bir filtre tanımladıktan sonra aktivasyon fonksiyonu yine “relu” seçilmiştir. Sonrasında, tanımlanan “Max Pooling2D” ile 2x2 ortaklama işlemi yapılmakta ve girişe uygulanan resim verisi yarılanarak 55x55x512 boyutuna indirilmektedir. Tekrar batch normalizasyonu işlemi ile bu ara katman tamamlanmış olur. Bundan sonraki 3 ara katmanda, sadece nöron sayısı değiştirilerek özellik çıkarma işlemine devam edilmiştir. Son aşamada ise 6x6x256' lük bir tensör kalmıştır. Flatten katmanı ile tensör bir vektöre dönüştürülerek özellik çıkarma işlemi tamamlanmıştır. Buradan sonraki ağ yapısı, sınıfları öğrenmek için geliştirilmektedir.

Dense katmanında (tam bağlı katman-fully connected layer) 256 nöron tanımlanmıştır. Aktivasyon fonksiyonu evrişim katmanlarında olduğu gibi “relu” seçilmiştir. Modelin, aşırı öğrenme (overfitting) sorununa takılmaması için 0,2 değerinde bir “Dropout” katmanı eklenmiştir. Son olarak, batch normalizasyonu işlemi ile bu katman da tamamlanmış olmaktadır.

Son olarak bir dense katmanı daha eklenmiştir ve bu katman 3 nörondan oluşur. Bu katmanın aktivasyon fonksiyonu

“softmax” seçilerek modelin ağ yapısı tamamlanmış olur. Tasarlanan modelin eğitim işlemleri gerçekleştirilmiştir. Bu çalışma için Keras ve Tensorflow yapıları (frameworks) kullanılmıştır. Model olarak bir tane ağ yapısını oluşturduğumuz model ve transfer öğrenme metodu kullanılarak farklı modellerin başarımları ile çalışma gerçekleştirilmiştir.



Şekil 2. Oluşturulan CNN modelinin ağ katman yapısı

Ağ modelimiz üç ana kısımdan oluşmaktadır. İlk katman giriş katmanı modele eğitim ve tahmin için dışarıdan görüntü bilgisinin verildiği katmandır. İkinci katman ise gizli katman olarak isimlendirilmekte ve verilen görüntünün öznelikleri bu katmanda çıkarılmaktadır. Son olarak, çıkış katmanında ise sınıfın öğrenildiği katman olarak görev yapmaktadır. Şekil 2' de gösterilen modelin dışında başarımları kanıtlanmış olarak ifade edilen modellerin giriş ve gizli katmanları dondurularak sadece çıkış katmanı ağına tekrar düzenlenmesi metodu ile transfer öğrenme gerçekleştirilmektedir. Transfer öğrenme bu başarımları kanıtlanmış modellerin özellik çıkartma katmanını kendi çalışmamızda kullanmamızı sağlamaktadır. Diğer bir fayda ise eğitilebilir parametre sayısının azalması ile eğitim süresi kısalmıştır.

3. Deneysel Sonuçlar

3.1. CNN Model Eğitiminin Sonuçları

Doğruluk model eğitimi sırasında modelin yaptığı doğru tahminlerin doğru ve yanlış tüm tahminlerin toplamına oranıdır. Kayıp ise modelin doğru çıkış değerinden ne kadar farklı tahmin değerinde bulunduğunu gösteren parametredir. Kayıp değeri ne kadar küçük olursa model gerçek değere o kadar yakındır.

Ağımızı bizim oluşturduğumuz model %86,75 doğruluk başarımları göstermiştir. Transfer öğrenme, veri kümesinin yetersiz olduğu durumlarda daha önceden çok büyük veri kümeleri ile eğitilmiş modellerin ağırlıklarını yani özellik çıkartma becerilerini kullanmamıza imkân vermektedir. Ön eğitilmiş modellerden MobileNetV2, DenseNet121, NasnetMobile, Resnet50V2 kullanılarak yazılımsal deneyler gerçekleştirilmiştir. Transfer öğrenmeden faydalanırken kullanılacak önceden eğitilmiş modelin çıkış katmanları model tasarımından çıkarılır ve katmanlar dondurularak üzerine bizim çıkış katmanlarımız eklenir. Eğitim az miktarda eğitilebilir parametre ile gerçekleştirilir. Bu durum, eğitimin daha kısa sürmesinin yanı sıra daha başarılı modeller elde edilmesini sağlamaktadır. İlk olarak, çıkış katmanı için ağ yapısı ile ilgili denemeler yapılmıştır. Çıkış katmanında kullanılan katman sayısı ve nöron sayısı artırıldıkça başarımlar artmaktadır. Ancak eğitilmiş modelin boyutunun çok büyümesi tercih edilmemektedir. Ayrıca, bir noktadan sonra çıkış katmanlarının büyütülmesi iyi yönde değişime sebep olmamaktadır. Tablo 1' de bu durum kolayca görülmektedir.

Tablo 1. Çıkış Katmanına Ağ Büyüklüğünün Etkisi

Model Adı	Eğitilebilir Parametre	Eğitim Başarım	Doğrulama Başarım	Model Boyutu (MB)
	Sayısı	(%)	(%)	
MobileNetV2	395.011	65,06	58,00	12
	593.155	65,34	50,35	14
	2.762.243	65,11	55,03	30
DenseNet121	329.475	73,62	74,74	30
	527.619	73,60	76,22	32
	2.631.171	74,42	73,18	48

Model eğitimi sırasında seçilen optimizasyon algoritmasının, kaç döngü (epoch) sayısında en iyi öğrenme düzeyine ulaşılacağını belirlemede önemli rol oynamaktadır. En iyi başarıma ulaşmak için optimize edici algoritmaların bazıları kayıp (loss) değerini az devir sayısında gürültü oluşturarak düşürebilirken, bazıları da daha fazla devir sayısı ile daha az gürültü oluşturarak düşürebilirler. Ancak bilinmelidir ki devir sayısının fazla olması, başarıyı sürekli yukarı doğru yükselten bir durum oluşturmamaktadır. Bir seviyeden sonra başarıyı düzeyi yatay eğri haline gelecektir.

Yapılan deneylerde farklı optimizasyon algoritmaları denenerek doğruluk ve kayıp değerleri birlikte değerlendirildiğinde “Adam” ve “Rmsprop” optimizasyon algoritmalarının veri kümesi için uygun olduğuna karar verilmiştir.

Transfer öğrenme kullanılırken, modeller arasında ağ yapısının oluşturulması ve parametre sayılarında farklılıklar bulunmaktadır. Genellikle, daha fazla parametreye sahip modellerin daha başarılı olacağı düşünülse de bu durum her zaman geçerli değildir. Tablo 1’den bu durum görülebilmektedir. Modeldeki hesaplanacak parametre sayısının artması, daha fazla işlem gücü gerektirecektir. Geliştirme yapılacak platformların işlem gücü göz önünde bulundurularak seçim yapılmalıdır. Yine, model dosyasının büyüklüğü önem arz etmektedir. Çünkü modelin çalışabilmesi için bulut ortamından veya yerel ortamdan hafızaya yüklenmesi gerekecektir. Büyük boyuttaki model dosyalarının yüklenmesi sistemin başlatılmasında gecikmelere sebep olabilmektedir. Bundan dolayı, küçük dosya boyutuna sahip modeller tercih edilmiştir. Bu çalışmada, veri kümemiz ile yapılan deneylerde ise Tablo 2’deki sonuçlar elde edilmiştir.

Tablo 2: Ön eğitilmiş modellerin karşılaştırılması

Model	Eğitim		Doğrulama		Boyut (MB)
	Doğruluk (%)	Kayıp	Doğruluk (%)	Kayıp	
DenseNet121	0,74	0,62	0,64	0,73	48
NasnetMobile	0,74	0,64	0,70	0,73	39
MobileNetV2	0,80	0,65	0,55	1,14	30
ResNet50V2	0,68	0,74	0,73	0,67	115
EfficientNetB0	0,90	0,26	0,13	0,95	32

Test edilen modellerden sadece “Efficient NetB0” farklı bir ağ yapısına sahiptir. Diğer modeller özellik çıkartma katmanları dondurularak aynı çıkış katmanı ve hiper parametreler kullanılarak eğitimler gerçekleştirilmiştir. Efficient NetB0 modeli en iyi başarıyı göstermektedir. Aşırı öğrenmeyi (overfitting) önlemek için ağda ki bazı nöronların unutulmasını sağlamayan metot olan “drop out” yaklaşımının kullanılmasına rağmen sonuçlarda bir iyileşme olmamıştır. “ResNet50V2” modeli büyük

4. Sonuç

Bu çalışmada, derin öğrenme yaklaşımlarından CNN modeli geliştirilerek, iki farklı uygulamada çalıştırılmıştır. Modelin geliştirilmesi aşamasında iki metot benimsenmiştir. Birincisi, ağ yapısını oluşturduğumuz modelde farklı parametreler ile yapılan denemelerde ağ büyüklüğünün özellik çıkarma konusunda avantaj sağladığı görülmüştür. Yine çıkış katmanı için ağ

dosya boyutuna sahip olmasından dolayı tercih edilmemiştir. Modellerin geliştirilen uygulamalarda çalıştırılabilmesi için dönüştürme işlemleri yapılması gerekmektedir. Bu dönüşümler sonucu hem tarayıcı hemde mobil uygulama için model dönüştürme işlemi sadece MobileNetV2 ile sorunsuz gerçekleştirildiği için uygulamalarda kullanılmak üzere seçilen model olmuştur.

büyüklüğünün artırılması olumlu sonuçlar vermiştir. Modelin geliştirilmesi için benimsenen diğer metot ise transfer öğrenmedir. Transfer öğrenme ile kullanılan başarıları kanıtlanmış modellerin özellik çıkarımı (feature extraction) konusunda başarılı olduğu gözlenmiştir. Hazırlanan farklı çıkış katmanları ile yapılan deneylerde, ağ yapısını oluşturduğumuz modelden daha başarılı olduğu görülmüştür. Geliştirilen tarayıcı eklentisinde modelin depolanmasının, Google Firebase “Storage” bulut

hizmetinde yapılması kullanım ve güncelleme kolaylığı sağlamaktadır. Bu çalışmanın devamı olarak, videoların görüntüden sınıflandırması yapılmadan ilk olarak videonun meta verilerine bakarak bir sınıflandırma işlemi uygulanması ek bir önlem olarak sistemin performansını artıracığı düşünülmektedir.

Kaynakça

- (2021). Kaggle: <https://www.kaggle.com/> adresinden alındı
- (2021). ImageNet: <https://image-net.org/> adresinden alındı
- (2021). COCO: <https://cocodataset.org/> adresinden alındı
- Abu-El-Haija, S., Kothari, N., Lee, J., Natsev, P., Toderici, G., Varadarajan, B., & Vijayanarasimhan, S. (2016, 9 27). *YouTube-8M: A Large-Scale Video Classification Benchmark*. arxiv.org: <https://arxiv.org/abs/1609.08675> adresinden alındı
- Bhatti, M. T., Khan, M. G., Aslam, M., & Fiaz, M. J. (2021, 2 12). Weapon Detection in Real-Time CCTV Videos Using Deep Learning. *IEEE Access*, s. 34366 - 34382. doi:10.1109/ACCESS.2021.3059170
- Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020, 4 23). *YOLOv4: Optimal Speed and Accuracy of Object Detection*. arxiv.org: <https://arxiv.org/abs/2004.10934> adresinden alındı
- Chen, Y.-L., Chang, C.-L., & Yeh, C.-S. (2017, Eylül). Emotion classification of YouTube videos. *Decision Support Systems*, s. 40-50. doi:10.1016/j.dss.2017.05.014
- Hamman, A. M. (2019). An Extensible, Modular Framework for Classifying YouTube Videos Using Web and Social Media. *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE. doi:10.1109/MIPR.2019.00092
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., . . . Adam, H. (2017, 4 17). *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. arxiv.org: <https://arxiv.org/pdf/1704.04861.pdf> adresinden alındı
- Jia, C., Wang, S., Zhang, X., Wang, S., Liu, J., Pu, S., & Ma, S. (2019). Content-Aware Convolutional Neural Network for In-Loop Filtering in High Efficiency Video Coding. *IEEE Transactions on Image Processing*, 3343-3356. doi:10.1109/TIP.2019.2896489
- Jia, Y., Chen, W., Yang, M., Wang, L., Liu, D., & Zhang, Q. (2021). Video smoke detection with domain knowledge and transfer learning from deep convolutional neural networks. *Optik*(240). doi:10.1016/j.ijleo.2021.166947
- Jing, L., Parag, T., Wu, Z., Tian, Y., & Wang, H. (2021). VideoSSL: Semi-Supervised Learning for Video Classification. *Winter Conference on Applications of Computer Vision (WACV)* (s. 1110-1119). IEEE/CVF.
- Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. (2014). Large-Scale Video Classification with Convolutional Neural Networks. *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. doi:10.1109/CVPR.2014.223
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 436-444. doi:10.1038/nature14539
- Lee, H. (2019). *Cigarette Smoker Detection*. 04 24, 2021 tarihinde <https://www.kaggle.com/vitaminc/cigarette-smoker-detection> adresinden alındı
- Nayak, R., Pati, U. C., & Das, S. K. (2021). A comprehensive review on deep learning-based methods for video anomaly detection. *Image and Vision Computing*. doi:10.1016/j.imavis.2020.104078
- Nouisser, K. (2019). *Alcoholic Drinks*. 04 24, 2021 tarihinde <https://www.kaggle.com/khalilnouisser/alcoholicdrinks5> adresinden alındı
- Nugroho, H. A., Hardiyanto, D., & Adji, T. B. (2015). Negative content filtering for video application. *7th International Conference on Information Technology and Electrical Engineering (ICITEE)*. IEEE. doi:10.1109/ICITEED.2015.7408912
- Ostrolucky, G. (2020). *Bulk Bing Image Downloader*. 04 24, 2021 tarihinde <https://github.com/ostrolucky/Bulk-Bing-Image-downloader> adresinden alındı
- Panta, F. J., Qodseya, M., & Péninou, A. (2018). Management of Mobile Objects Location for Video Content Filtering. *MoMM2018: Proceedings of the 16th International Conference on Advances in Mobile Computing and Multimedia* (s. 44-52). New York: Association for Computing Machinery. doi:10.1145/3282353.3282368
- Ramesh, M., & Mahesh, K. (2020). A Performance Analysis of Pre-trained Neural Network and Design of CNN for Sports Video Classification. *International Conference on Communication and Signal Processing*. Chennai: IEEE. doi:10.1109/ICCSP48568.2020.9182113
- Ren, S., He, K., Girshick, R., & Sun, J. (2016, 1 6). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. arxiv.org: <https://arxiv.org/pdf/1506.01497v3.pdf> adresinden alındı
- Sasank, S. (2019). *Guns Object Detection*. 04 24, 2021 tarihinde <https://www.kaggle.com/issaisasank/guns-object-detection> adresinden alındı
- Shaout, A., & Crispi, B. (2020). Streaming Video Classification Using Machine Learning. *The International Arab Journal of Information Technology*, 667-682. doi:10.34028/iajit/17/4A/13
- Simonyan, K., & Zisserman, A. (2014, 9 4). *Very Deep Convolutional Networks for Large-Scale Image Recognition*. arXiv.org: <https://arxiv.org/abs/1409.1556> adresinden alındı
- Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. (2016, 2 23). *Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning*. arXiv.org: <https://arxiv.org/abs/1602.07261v2> adresinden alındı
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2015, 12 5). *Rethinking the Inception Architecture for Computer Vision*. arXiv.org: <https://arxiv.org/abs/1512.00567v3> adresinden alındı
- Wu, Z., Yao, T., Fu, Y., & Jiang, Y.-G. (2017). Deep learning for video classification and captioning. *Frontiers of Multimedia Research*, 3-29. doi:10.1145/3122865.3122867