DÜMF Mühendislik Dergisi
web: http://dergipark.gov.tr/dumf

Araştırma Makalesi / Research Article

Gömülü Derin Öğrenme ile Tehdit İçeren Nesnelerin Gerçek Zamanda Tespiti

İlhan AYDIN
Fırat Üniversitesi, Bilgisayar Mühendisliği Bölümü, Elazığ
iaydin@firat.edu.tr ORCID: 0000-0001-6880-4935

Mehmet Umut SALUR*
Harran Üniversitesi, Bilgisayar Mühendisliği Bölümü, Şanlıurfa
umutsalur@harran.edu.tr ORCID: 0000-0003-0296-6266, Tel: (414) 318 00 00 (1261)

Mehmet KARAKÖSE
Fırat Üniversitesi, Bilgisayar Mühendisliği Bölümü, Elazığ
mkarakose@firat.edu.tr ORCID: 0000-0002-3276-3788

Geliş: 05.12.2018, Kabul Tarihi: 31.03.2019

Öz

Derin öğrenme metotları bilgisayarlı görme ve görüntü işlemede özellikle de görüntü sınıflandırma probleminde önemli bir teknoloji haline gelmiştir. Bunun en önemli nedenlerinden biri farklı problemler üzerinde derin öğrenmenin göstermiş olduğu üstün başarıdır. İnternetin gelişimi ile çok büyük veri kümeleri toplanmakta ve yüksek güçlü grafik işlemci kartlar ile bu veriler gerçek zamanlı olarak işlenebilmektedir. Fakat her problemler için bu şekilde büyük ölçekli veri toplamak oldukça maliyetli bir işlemdir. Bu amaçla ön eğitilmiş derin öğrenme modelleri transfer öğrenme yöntemi ile daha düşük boyuttaki verileri sınıflandırmak için kullanılabilir. Bu çalışmada X-ray cihazlarından alınan görüntülerde tehdit unsuru içeren nesneleri sınıflandırmak için transfer öğrenme yöntemi ile gömülü ve gerçek zamanlı çalışabilen bir sistem geliştirilmiştir. Bu sistem Nvidia Jetson TX2 geliştirme kartı üzerinde bir evrişimsel sinir ağı olan Alexnet derin öğrenme modeli kullanmaktadır. Bu model ile X-ray bagaj güvenlik görüntüleri içerisindeki bıçak, silah, jilet ve Ninja yıldızı gibi tehdit unsuru içeren nesneler sınıflandırılmıştır. Oluşturulan deney ortamında Alexnet 12.000 görüntü ile eğitilmiş ve gerçek ortamda test edilmiştir. Önerilen yöntemin performansı aynı veri kümesi üzerinde daha önce yapılan farklı bilgisayarlı görme teknikleri ile karşılaştırılmış ve daha başarılı sonuçlar elde edilmiştir.

Anahtar Kelimeler: *Derin öğrenme; Jetson Tx2 Geliştirme kartı; Gömülü sistemler; Evrişimsel sinir ağları; Nesne sınıflandırma;*

* Yazışmaların yapılacağı yazar

DOI: 10.24012/dumf.492433

Giriş

İnsan güvenliğinin sağlanması kuşaklar boyunca sürekli önemli konular arasında yer almıştır. İnsan güvenliğinin sağlanması çoğu zaman topluluk halindeki ortamlarda hem daha zorlaşmakta hem de daha kritik bir öneme sahip olmaktadır. İnsanların topluluk halinde bulundukları ortamların başında havalimanı, otobüs terminalleri, müzeler, alış veriş merkezleri ve hastaneler gelmektedir. Bu merkezlerin hepsinin girişinde X-ray cihazları ile insan güvenliğinin sağlanması çalışılmaktadır (Parliament, 2012). X-ray cihazlarının kontrolü manuel bir şekilde ilgili noktadaki görevliler tarafından yapılmaktadır. Bu hem zaman alan bir süreç hem de görevlinin dikkatsizliği sonucunda kötü niyetli kişiler tarafından içeriye tehdit unsuru içeren bıçak, jilet, silah gibi nesnelerin koymasında bir güvenlik zafiyeti oluşturmaktadır. Bilgisayar biliminin gelişmesiyle güvenlik konusunda bilgisayar tabanlı birçok yöntem geliştirilmiştir. Bu yöntemlerinin başında yüksek başarımlı görüntü tespit etme ve sınıflandırma algoritmaları gelmektedir. Bu algoritmaların başında son yıllarda göstermiş olduğu başarımlarla birçok probleme çözüm getiren derin öğrenme algoritmaları gelmektedir.

Derin öğrenme algoritmalarının başında çok katmanlı mimarisi ile Evrişimsel Sinir Ağı (ESA) gelmektedir. 2012 yılında yapılan ImageNet yarışmasında göstermiş olduğu başarımdan sonra birçok alanda ESA kullanılmıştır (Krizhevsky ve Hinton, 2012). ESA insanların algılama sistemini kullanarak öğrenen ve iyi bilinen bir derin öğrenme mimarisidir (Gu vd., 2018). ESA'nın ilk ortaya çıkışı 1990'lı yıllara dayanmaktadır. Geliştirilen ilk çok katmanlı ESA Lenet-5 olarak isimlendirilmiş ve bu ağ el yazısı tanıma için kullanılmıştır (LeCun vd., 1998). Bu ağın eğitimi için geriye yayılım öğrenme algoritması kullanılmıştır. Geliştirilen ağ orijinal görüntüyü doğru bir şekilde temsil etmektedir.

Derin öğrenme algoritmalarındaki temel problemlerden biri eğitim işleminin uzun zaman almasıdır. 2012'den beri derin evrişimsel ağların eğitimi için birçok yöntem önerilmiştir (Niu ve Suen, 2012). İlk olarak geliştirilen Alexnet, Lenet 5 ağına benzemesine rağmen daha derin bir yapıya sahiptir (Russakovsky vd., 2015). Alexnet'in eğitimdeki başarısı ile birlikte VGGNet (Simonyan ve Zisserman, 2014), GoogleNet (Szegedy vd., 2015) ve Resnet (He vd., 2016) gibi ağ yapıları geliştirilmiştir. Geliştirilen derin evrişimsel ağ yapıları daha öncekilere göre daha fazla katmana sahip olmaktadır. Bu özellik giriş verisinin doğrusal olmayan özelliğini ortaya çıkarmakta ve ağ başarımını arttırmaktadır. Fakat bu özellik aynı zamanda ağın karmaşıklığını arttırmakta ve eğitim sürecini uzatmaktadır.

Derin ESA'nın iyi başarım sağlaması için eğitim aşamasında büyük miktarda veri ile eğitilmesi gerekmektedir. Veri boyutunun büyümesi ve ESA'nın yapısındaki karmaşıklık eğitim sürecini klasik işlemcilerde yapılması önündeki en önemli engellerden birini oluşturmaktadır (Ponti vd., 2017). Bu durum eğitim aşamasında paralel hesaplama yapan grafiksel işlemci kartlarının kullanılması gerekli kılmıştır. Bu karta sahip bir bilgisayar üzerinde yapılan eğitim hem daha hızlı olmakta hem de gerçek zamanlı test işlemi kolaylıkla yapılabilmektedir.

Derin ESA birçok uygulama alanında kullanılmaktadır. En önemli uygulama alanlarından biri ESA'nın görüntü sınıflandırma probleminde kullanılmasıdır. Görüntü içerisinde nesne tespiti ve tanıma amacıyla bölgesel ESA'lar geliştirilmiştir (Ren vd., 2017). Bu tür bir ağda görüntü içerisinde aday nesneler belirlenerek ESA ile eğitilmektedir. Eğitim aşamasında bu nesnelerin konumları ve görüntünün tamamı kullanılmaktadır. Test aşamasında ise verilen görüntüden nesnenin olup olmadığı ve hangi nesne olduğu belirlenmektedir. X-ray cihazlarının görüntülerinden nesnelerin tespit edilmesinde ESA'nın diğer yöntemlere göre daha yüksek başarım sağladığı görülmüştür (Mery vd., 2017). Bunların dışında ESA ses işleme, doğal dil

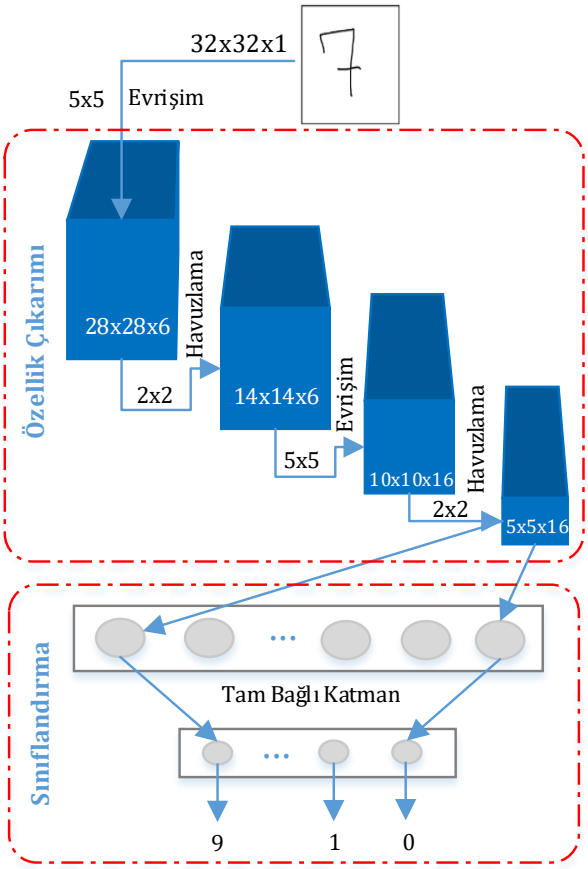
işleme, tahmin ve metin sınıflandırma gibi alanlarda da kullanılmıştır (Chen, 2015) (Salur ve Aydın, 2018). Ses ve diğer sinyallerin sınıflandırılması amacıyla genellikle uzun kısa süreli hafıza (LSTM) ağları kullanılmaktadır (Hochreiter ve Schmidhuber, 1997). Bir LSTM ağı tek veya çok boyutlu zaman serilerini bir ağa giriş olarak verir ve serilerin bireysel zaman adımlarına dayanarak tahminler yapmamızı sağlar.

Günümüzde farklı amaçlar için birçok derin ESA yapısı önerilmesine rağmen, geliştirilen algoritmaların çoğu yüksek hızlı GPU kartlarına sahip bilgisayarlarda eğitilmekte ve test işlemi de aynı platformlarda yapılmaktadır. Bu çalışmada özellikle otonom araçlar ve insansız hava araçlarında kullanılabilen bir platform olan GPU geliştirme kartı üzerinde derin öğrenme uygulaması geliştirilmiştir. Bu amaçla NVIDIA Jetson TX2 geliştirme kartı üzerinde Caffe derin öğrenme platformu kurulmuştur. Önerilen yöntemde hem eğitim hem de test işlemi bu kart üzerinde geliştirilmiş ve kart üzerinde bulunan kamera kullanılarak gerçek zamanlı olarak alınan görüntüler sınıflandırılmıştır. X-ray görüntülerinden oluşturulan eğitim kümesinde bıçak, jilet, Ninja yıldızı ve silah gibi nesnelerin görüntü içerisinde tespit edilmesi sağlanmıştır. Bu çalışmayla NVIDIA Jetson TX2 geliştirme kartının toplumsal güvenlik gerektiren noktalarda kolay bir şekilde uygulanabilir bir prototip geliştirilmesi hedeflenmiştir. Çalışmanın sonucunda elde edilen yönteme farklı işlevler eklenerek otonom araçlar için kolaylıkla kullanılabilir modüler bir model elde edilebilir.

Gömülü derin öğrenme gerçekleştiren bu çalışmanın ikinci bölümünde ESA'nın yapısı ve çalışma şekli anlatılmıştır. Çalışmanın üçüncü bölümünde geliştirilen sistemin bileşenleri verilmiştir. Dördüncü bölümde ise sistem üzerinde gerçekleştirilen deneysel işlemler anlatılmıştır. Ayrıca yine dördüncü bölümde sistemin başarım metrikleri değerlendirilmiştir. Beşinci bölümde çalışmanın sonucu ve gelecek çalışmalardan bahsedilmiştir.

Evrişimsel Sinir Ağları

ESA görüntü sınıflandırma gibi bilgisayarlı görme uygulamaları için en çok kullanılan derin öğrenme algoritmalarından biridir. Bir ESA'nın temel bileşenleri evrişimler, havuzlama, aktivasyon fonksiyonu ve tam bağlı katmanıdır. Son bileşen olan tam bağlı ağ ileri beslemeli yapay sinir ağındaki gizli katmana benzer bir yapıya sahiptir. ESA'nın farklı varyasyonları olmasına rağmen temel bileşenler birbirine benzemektedir. Dijit sınıflandırma için kullanılan Lenet-5 ağının yapısı Şekil 1'de verilmiştir (LeCun, 2015).

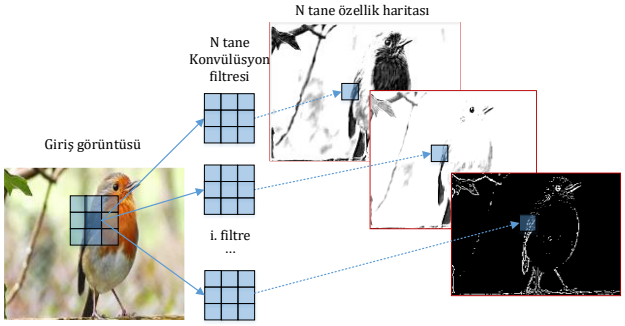


Şekil 1. Lenet-5 ağının yapısı

Şekil 1'de verildiği gibi görüntü üzerinde gezdirilen evrişim filtreleri ile özellik haritaları oluşturulmaktadır. Havuzlama katmanında ise görüntü boyutu verilen alt örnekleme boyutuna göre düşürülmektedir. Tam bağlı katmanda ise en son aşamada vektör olarak elde edilen özelliklere göre görüntü sınıflandırması yapılmaktadır.

Evrişim Katmanı

Evrişim katmanı görüntü üzerinde belirli filtrelerin kaydırılarak görüntünün yeni piksel değerlerinin komşu piksellere göre hesaplanması işlemidir. Her filtre kxk boyutunda bir matrisi temsil etmektedir. Her filtre kendi boyutu ile belirlenen bir komşulukta bütün piksel değerlerinin bir doğrusal kombinasyonunu üretir. Şekil 2'de basit bir evrişim işlemi gösterilmiştir.

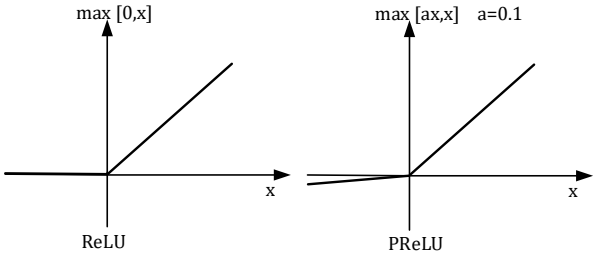


Şekil 2. Giriş görüntüsü üzerinde evrişim işlemi

Evrişim işlemi bir renkli görüntü üzerinde 5x5 boyutunda 4 adet filtreyle uygulanmak istenirse elimizde 5x5x3 boyutunda filtreler olmalıdır. Çünkü renkli görüntüler 3 kanala sahiptirler. Eğer görüntümüz 32x32x3 boyutunda ise 4 filtre ile elde edilecek çıkış 32x32x4 boyutunda 4 matristen oluşan bir tensordur. En çok kullanılan filtreler 5x5xd, 3x3xd ve 1x1xd olup buradaki d parametresi derinliği göstermektedir.

Aktivasyon Fonksiyonu

Aktivasyon fonksiyonu yapay nöron hücresine girdi olarak gelen değere karşılık bir çıktı değeri üreten fonksiyonlardır. İleri beslemeli yapay sinir ağlarında kullanılan tanjant, sigmoid ve doğrusal gibi aktivasyon fonksiyonları yerine ESA'da evrişim işleminden sonra düzeltilmiş bir doğrusal fonksiyon kullanılır. Şekil 3'te düzeltilmiş doğrusal fonksiyonun iki biçimi gösterilmektedir.

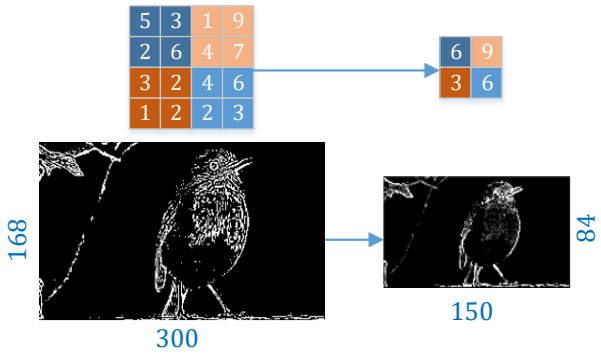


Şekil 3. ReLU ve PReLU aktivasyon fonksiyonları gösterimi

Şekil 3'te verilen ReLU fonksiyonu görüntü işleme algoritmaları sonrasında elde edilen özellik haritasında negatif değerleri düzeltmek için kullanılır. PReLU fonksiyonu ise küçük negatif değerlere izin vermektedir. Her evrişim nöronu aktivasyon fonksiyonundan geçen ve özellik haritası olarak adlandırılan yeni bir matris üretmektedir. Oluşturulan bu matris bir sonraki katmanın girdi değerini oluşturmaktadır.

Havuz Katmanı

Havuzlama katmanı evrişim işleminden sonra görüntü boyutunu düşürmek için kullanılır. En çok kullanılan havuzlama işlemi maksimum havuzlamadır. Bu işlemin temel iki amacı veri boyutunu düşürmek ve görüntü boyutunu azaltmaktır. Basit bir havuzlama işlemi Şekil 4'te gösterilmiştir.



Şekil 4. Havuz katmanı

Tam Bağlı Katman

Verilen bir görüntü birçok evrişim katmanından geçtikten sonra elde edilen çıkış değerleri tam bağlı katmana verilir. Tam bağlı katman aslında ileri beslemeli sinir ağlarındaki gizli katmana benzemektedir. Verilen görüntünün sınıflandırılması için ağın ağırlıklarının

öğrenilmesi gerekmektedir. En son evrişim işleminden sonra elde edilen matrisler bir vektör olarak birleştirilir. Örneğin en son katmanda elde edilen tensor 4x4x20 ise elde edilen vektör 4x4x20=1x320 boyutunda olur. Dolayısıyla toplamda 320 adet ağırlığın eğitilmesi gerekmektedir. Bir ESA'nın son katmanı ise sınıf etiketlerini göstermektedir. Sınıf etiketleri tahmin etmek için Denklem 1'in sonucunda elde edilen olasılık değeri kullanılır.

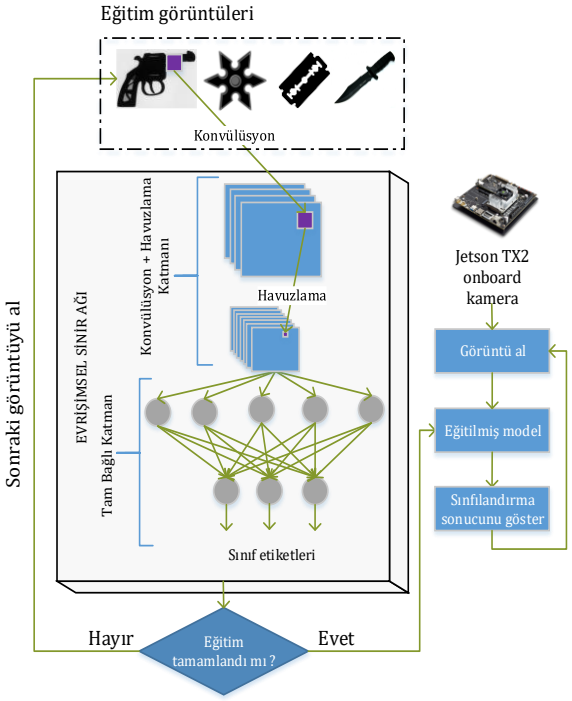
$$P(y = c \mid x; w; b) = \frac{e^{x^T w_c + b_c}}{\sum_j e^{x^T w_j + b_j}} \qquad (1)$$

Denklem (1)'de y tahmin edilen sınıfın olasılığını, x giriş vektörünü, w ve b ise ağın ağırlıklarını göstermektedir.

Gömülü Derin Öğrenme için Önerilen Yöntem

Otonom araçlar, yardımcı ev robotları, silahsız hava araçları vb. birçok uygulama alanında gömülü derin öğrenme yöntemleri uygulanmaktadır. Bu çalışmada gömülü derin öğrenme algoritmalarını gerçekleştirmek için Nvidia Jetson Tx2 GPU geliştirme kartı üzerinde Caffe derin öğrenme paketi kurulmuştur. Eğitim aşamasında bilgisayarda kayıtlı olan görüntüler kullanılarak ESA eğitilmiştir. Test aşamasında ise Nvidia Jetson TX2 üzerinde gömülü bulunan kameradan alınan gerçek zamanlı görüntüler kullanılmıştır. Şekil 5'te önerilen yöntemin blok diyagramı verilmiştir.

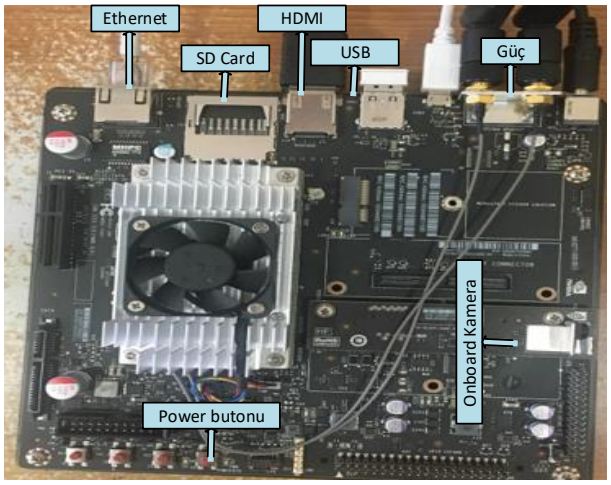
Şekil 5'te önerilen sistemde Jetson TX2 geliştirme kartı üzerinde kayıtlı eğitim veri kümesi kullanılarak ESA eğitilmektedir. Test işlemi ise eğitim aşamasında elde edilen modele göre gerçekleştirilmektedir. Jetson TX2 kartının kamerasından alınan görüntüler modele verilmekte ve sınıflandırma işlemi yapılmaktadır.



Şekil 5. Önerilen sistemin blok diyagramı

Nvidia Jetson Tx2 GPU geliştirme kartı

Jetson TX2 gömülü sistemi NVIDIA tarafından üretilmiştir. Bu gömülü sistem önceden düzenlenip uyarlanmış olan bir Linux işletim sistemi kurulu bir şekilde gelmektedir. Yapay zekâ destekli sistemlerde düşük güç tüketimi ile yüksek hesaplama performansı gerektiren tüm işlemleri hızlı bir şekilde yerine getiren gömülü sistemdir. Geliştirme kartı NVIDIA'nın ücretsiz olarak sunduğu tüm geliştirme API'lerini (Application Programming Interface) ve yaygın olan diğer API'leri de desteklemektedir. Ayrıca geliştirme seti olduğu için çeşitli donanım ara yüzü sunarak, oldukça esnek ve genişletilebilir bir platforma olanak sağlamaktadır. Şekil 6'da Jetson TX2 geliştirme kartının bir görseli verilmiştir.



Şekil 6. Jetson Tx2 bağlantıları

Şekil 6'da verilen Jetson TX2 geliştirme kartının teknik özellikleri Tablo 1'de verilmiştir.

Tablo 1. *Jetson TX2 teknik özellikleri*

Özellik	Açıklama
GPU	NVIDIA Pascal™, 256 CUDA Çekirdeği
CPU	HMP Dual Denver 2/2 MB L2 + Quad ARM® A57/2 MB L2
Hafıza	8 GB 128 bit LPDDR4
PCIE	58.3 GB/s
Disk kapasite	32 GB eMMC, SDIO, SATA
Protokoller	CAN, UART, SPI, I2C, I2S, GPIOs
USB	USB 3.0 + USB 2.0
Internet Bağlantısı	1 Gigabit Ethernet, 802.11ac WLAN, Bluetooth

Geliştirme kartına gerekli programları yüklemek için Ubuntu 16.04.4 LTS sürümlü işletim sistemine sahip olan bir bilgisayar gerekmektedir. İşletim sistemini güncellemeden önce JetPack dosyası gerekmektedir. Bu yazılım geliştirme kiti içerisinde görüntü işleme için gerekli uygulamaları içermektedir. Kurulacak olan JetPack içinde TensorRT, cuDNN, CUDA Toolkit, VisionWorks, GStreamer ve OpenCV dâhil olmak üzere tüm Jetson platform yazılımlarını içermektedir.

Caffe derin öğrenme kütüphanesi ile görüntü sınıflandırma

Caffe Berkeley üniversitesinde yapay zekâ araştırma grubu tarafından geliştirilen bir derin öğrenme platformudur (Jia, 2014). Caffe'de kullanılan ifade yapısı yeni uygulamaların gerçekleştirilmesini kolaylaştırmaktadır. Model üzerinde yapılacak değişiklikler kod yazmadan ayar dosyaları ile yapılmaktadır. Hem CPU hem de GPU tabanlı eğitimi desteklemektedir. Caffe'nin temel özellikleri aşağıdaki gibi sıralanabilir;

- CPU ve GPU desteği mevcuttur.
- Çoklu GPU ile eğitim yapılabilir.
- Python'da katman tanımına izin verir.
- Yeni katmanlar ve ağ oluştururken kod yazmaya gerek yoktur.
- MATLAB ve Python desteği vermektedir.

Hızlı işlem yapma yeteneği, Caffe'yi araştırma deneyleri ve endüstri uygulamaları için mükemmel bir hale getirmiştir. Caffe tek bir NVİDİA K50 GPU ile günde 60M görüntüden daha fazlasını işleyebilir. Uygulama için ilk olarak Caffe derin öğrenme paketi Nvidia Jetson Tx2 kartına yüklenmiştir. Yükleme tamamlandıktan sonra Makefile.config dosyasında USE_CuDNN=1 ayarlaması yapılarak CUDA aktifleştirilmiştir. NVIDIA cuDNN, derin sinir ağları için GPU hızlandırmalı ilkel bir kütüphanedir. cuDNN, ileri ve geri dönüş, normalleştirme ve etkinleştirme katmanları gibi standart işler için yüksek düzeyde ayarlanmış uygulamalar sağlar. NVIDIA derin öğrenme SDK'sinin (Software Development Kit) bir parçasıdır. Ayrıca Opencv 3'ü kullanabilmek için OPENCV_VERSION = 3 olarak ayarlanmıştır.

Şekil 7'de verilen ağ yapısı ilgili problemin modelini oluşturmak için Caffe klasöründe bulunan Models klasöründeki .prottxt uzantılı dosyanın çizdirilmesidir. Models klasöründe yazılan .prottxt uzantılı dosyada bulunan herhangi bir evrişim ve havuzlama katmanı yapısı Şekil 7'de verilmiştir.

Evrişim katmanı

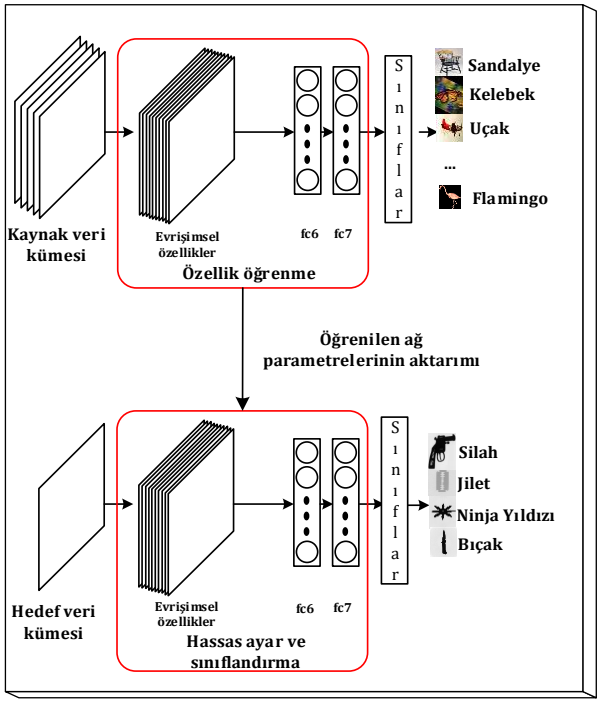
```
layer {
  name: "conv1"
  type: "Convolution"
  bottom: "data"
  top: "conv1"
  # learning rate for the filters
  param: { lr_mult: 1 }
  # learning rate for the biases
  param: { lr_mult: 2 }
  convolution_param: {
    num_output: 20
    kernel_size: 5
    stride: 1
    weight_filler: { type: "xavier" }
    bias_filler: { type: "constant" }
  }
}
```

Havuzlama katmanı

```
layer {
  name: "pool1"
  type: "Pooling"
  bottom: "conv1"
  top: "pool1"
  pooling_param: {
    pool: MAX
    kernel_size: 2
    stride: 2
  }
}
```

Şekil 7. Havuz ve evrişim katmanlarının .prottxt dosyası görünümü

Şekil 7'de verilen görselde evrişim katmanında filtre fonksiyonun sayısı ve boyutu verilmektedir. Caffe'de görüntü sınıflandırma için kurulan ağ yapısı Şekil 8'de verilmiştir.

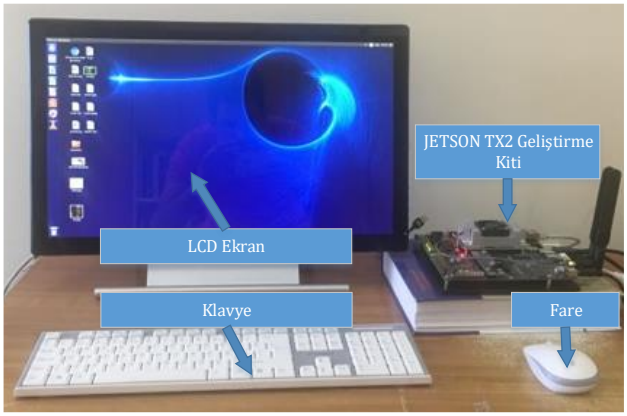


Şekil 8. Transfer öğrenme algoritması

Şekil 8'de ağ yapısını çizdirmek draw_net.py dosyasına .prototxt dosyasını göndermek gerekmektedir. Burada elde edilen çıktı görüntü formatında kaydedilmektedir.

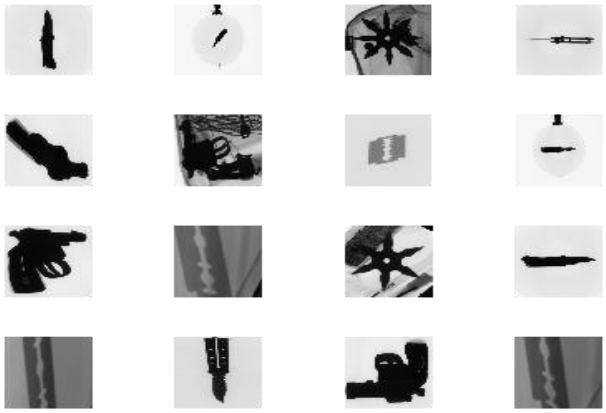
Deneysel Sonuçlar

Bu çalışmada önerilen derin öğrenme algoritmasının gerçek zamanlı gerçekleştirimi için Nvidia Jetson TX2 kartı kullanılmıştır. Bu kart üzerinde uygulamaların yapılabilmesi için bir deney ortamı oluşturulmuştur. Bu deney ortamında Nvidia Jetson TX2 kartına bir adet ekran, kablosuz klavye ve fare bağlanmıştır. Kurulan gerçek zamanlı sistem Şekil 9'daki görselde verilmiştir.



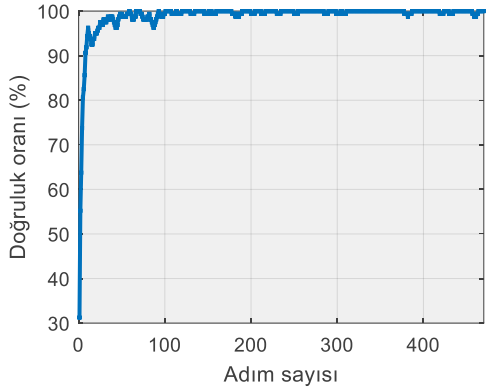
Şekil 9. Gömülü derin öğrenme için kurulan sistem

Şekil 9'da verilen sistem gerçek zamanlı ve gömülü çalışma özelliklerine sahiptir. Jetson Tx2 geliştirme kartı üzerinde bulunan 5 mega piksel kamera ile alınan görüntülerden önerilen sistemin gerçek zamanlı testi yapılabilmektedir. Her bir sınıf için 4.000 tane örnek olmak üzere toplam 12.000 tane görüntü eğitim aşamasında kullanılmıştır. Şekil 10'da kullanılan görüntü kümesindeki görüntülerden bir örnek verilmiştir. Şekil 10'da görüldüğü üzere eğitim kümesinde aynı nesnenin farklı açı ve yönlerdeki görüntüleri kullanılmıştır. Örneğin; bıçak nesnesi için farklı şekil, model, biçimde birden fazla görüntü kullanılmıştır.

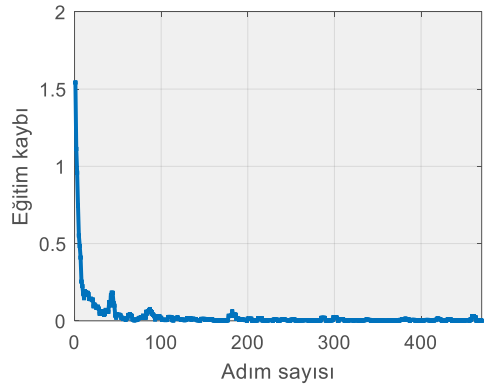


Şekil 10. Kullanılan veri kümesi örneği

Şekil 10'daki görüntüler farklı açılardan alınarak elde edilen bir veri kümesi kullanılmıştır. Veriler sadece farklı açılardan alınmamış olup aynı zamanda karmaşık arka plana sahip veriler de eğitim aşamasında kullanılmıştır. Önerilen derin öğrenme yönteminin eğitim başarımı Şekil 11'de verilmiştir.



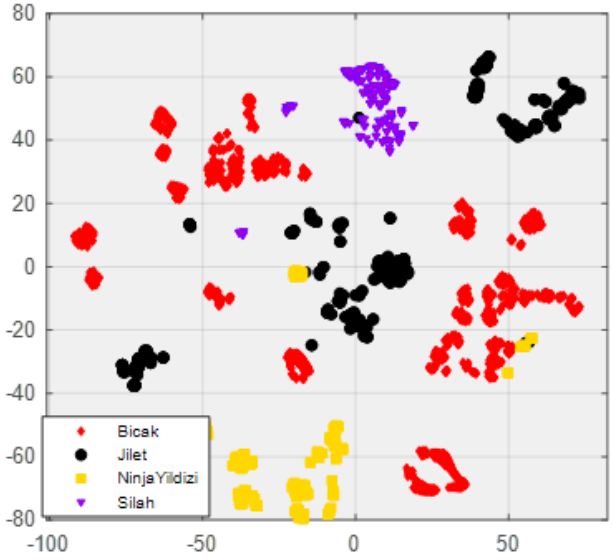
(a) Adım sayısına göre doğruluk oranı



(b) Adım sayısına göre eğitim kaybı

Şekil 11. Derin öğrenme ağının eğitim başarımı

Şekil 11'de görüldüğü gibi eğitim aşamasında, çok kısa bir adımından sonra eğitim başarımının maksimum olduğu görülmektedir. Her bir eğitim aşaması için algoritma 94 adım çalışmaktadır. Eğitim 5 aşamada yapıldığından toplam 470 adım eğitim yapılmaktadır. Deneyler sonucunda eğitim başarımı %99.6 olarak elde edilmiştir. ESA'nın doğru bir şekilde sınıfları ayırt edip etmediğini belirlemek için tam bağlı katman görselleştirilmiştir. Bu amaçla yüksek boyutlu veriyi görselleştiren bir boyut azaltma algoritması olarak t-dağılımlı stokastik komşu gömülme (t-SKE) yöntemi kullanılmıştır. Bu yöntem yüksek boyutlu noktalar arasındaki benzerliğe göre daha düşük boyuta dönüştürmektedir. Yöntem ilk olarak yüksek boyutlu nokta çiftleri arasındaki mesafeyi hesaplamaktadır. Bir sonraki adımda benzerlik matrisi oluşturmaktadır. Daha sonrasında düşük boyutlu noktalar için bir başlangıç kümesi oluşturulmaktadır. Son aşamada ise interaktif bir şekilde düşük boyutlu uzaydaki t dağılımı ve yüksek boyutlu uzaydaki Gauss dağılımı arasındaki mesafeyi azaltmak için düşük boyutlu uzaydaki noktalar güncellenmektedir. Tam bağlı katmanın eğitim kümesi üzerinde t-SKE yöntemi ile elde edilen görselleştirilme sonucu Şekil 12'de verilmiştir.



Şekil 12. ESA'nın tam bağlı katmanından elde edilen özelliklerin t-SKE ile görselleştirilmesi

Şekil 12'de verilen t-SKE yöntemi ile bu sınıflandırma problemi için ESA ile elde edilen

özelliklerin başarılı bir şekilde ayırt edildiği görülmektedir.

Önerilen yöntemin doğruluğunu ispatlamak için karmaşıklık matrisinden elde edilen ölçümler kullanılmaktadır. Karmaşıklık matrisinden kesinlik, hassasiyet ve F1 ölçüm değerleri elde edilmektedir. İki sınıflı bir sınıflandırıcı için basit karmaşıklık matrisi Tablo 2'de verilmiştir.

Tablo 2. *İki sınıflı bir problem için karmaşıklık matrisi*

		Gerçek Etiket		
		1	**2**	
Tahmin Edilen Etiket	**1**	DP_{11}	YP_{12}	TS_1
	2	YP_{21}	DP_{22}	TS_2
		ES_1	ES_2	Toplam

Tablo 2'de verilen karmaşıklık matrisine göre doğruluk oranı, kesinlik, hassasiyet ve F1 ölçümü hesaplanmaktadır. Doğruluk oranının hesaplanması denklem (2)'de verilmiştir.

$$Dogruluk = \frac{\sum_{i=1}^{C} DP_{ii}}{Toplam} \quad (2)$$

Hassasiyet her bir sınıf için doğru bir şekilde tahmin edilen sınıf etiketlerinin toplam tahmine oranıdır. Aşağıda verilen denklem (3)'e göre hesaplanır.

$$Kesinlik = \sum_{i=1}^{C} \frac{DP_{ii}}{TS_i} \quad (3)$$

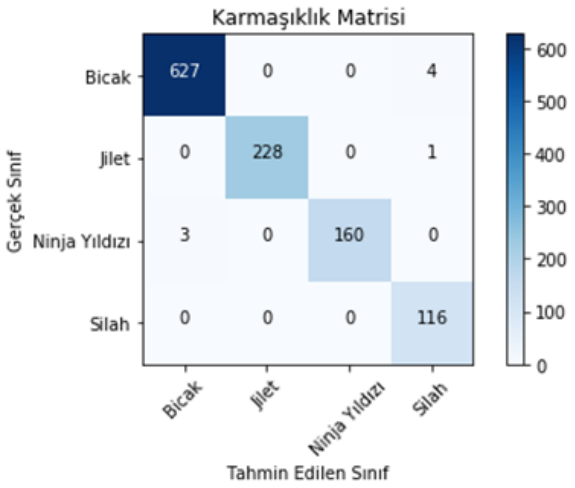
Denklemde TSi değeri i etiketi ile sınıflandırılan toplam test örneklerinin sayısıdır. DPii ise i sınıf etiketi için doğru bir şekilde sınıflandırılan örnek sayısıdır. Hatırlama değeri ise her bir sınıf için her bir sınıf için doğru bir şekilde sınıflandırılan doğru etiketlerin ağırlıklı ortalamasıdır. Bu değer denklem (4)'e göre hesaplanır.

$$Hatirlama = \frac{1}{C} \sum_{i=1}^{C} \frac{DP_{ii}}{ES_i} \quad (4)$$

Denklemde ESi değeri i gerçek etiketi ile sınıflandırılan test örneklerinin sayısını gösterir ve ilgili sütunun toplamı ile hesaplanır. Son ölçüm olarak F1 ölçümü ise kesinlik ve hatırlama değerlerini tek bir ölçümde birleştirmek için kullanılır. Bu ölçümün değeri 0 ile 1 arasında değer alır ve sınıflandırıcı bütün örnekleri doğru sınıflandırırsa 1 değerini alır. F1 ölçümü denklem (5)'te verilmiştir.

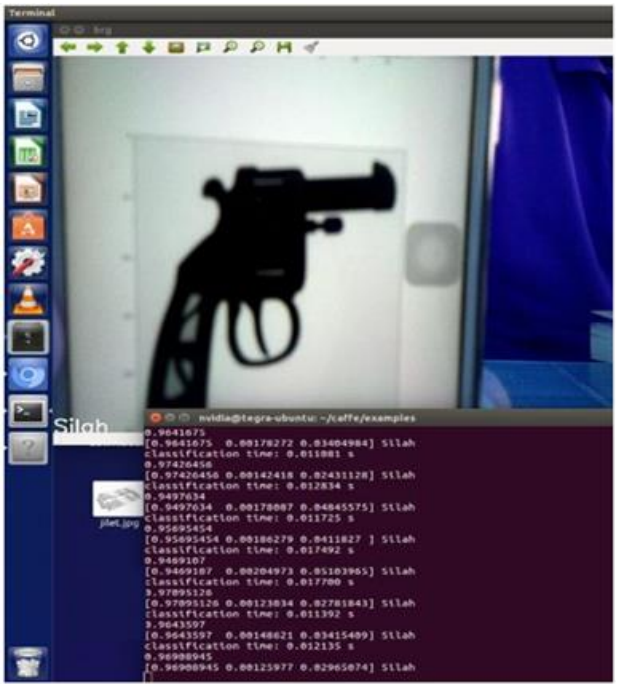
$$F_1 = \frac{2 * Kesinlik * Hatirlama}{Kesinlik + Hatirlama} \quad (5)$$

Eğitim işlemi tamamlandıktan sonra test aşamasında toplam 1239 adet test örneği ile eğitilen model test edilmiştir. Test sonucu elde edilen karmaşıklık matrisi Şekil 13'te verilmiştir.



Şekil 13. ESA test sonucu karmaşıklık matrisi

Şekil 13'te gösterildiği gibi karmaşıklık matrisinde yanlış sınıflandırılan örnek sayısının oldukça düşük olduğu görülmektedir. Önerilen yaklaşım aynı zamanda Jetson TX2 kartının kamerasından alınan görüntüler ile de test edilmiştir. Bu amaçla bir cep telefonundan kameraya gösterilen görüntüler kullanılmıştır. Şekil 14'teki örnekte giriş olarak bir silah görüntüsünü alan algoritmanın sınıflandırma sonucu verilmiştir.



Şekil 14. Jetson Tx2 kamerasından alınan görüntü ile ESA'nın test edilmesi

Şekil 14'teki sistemin çalışması sırasında her defasında yeni bir görüntü alınmakta ve öğretilen model test edilerek alınan görüntünün sınıfı belirlenmektedir. Bir sonraki adımda tekrar yeni görüntüler alınarak aynı işlemler tekrar yapılmaktadır. Dolayısıyla sistem gerçek zamanlı olarak çalışmaktadır. Şekil 14'te verilen görseldeki örnekte kartın kamerasından alınan görüntü % 96.9 silah olarak sınıflandırılmaktadır. Test aşamasında her bir görüntünün işlenmesi yaklaşık olarak 12 milisaniye sürmektedir. Dolayısıyla bir saniyede 80 görüntü işlenebilmektedir. Önerilen yöntem doğruluk oranı, kesinlik ve hatırlama parametrelerine göre farklı yöntemler ile karşılaştırılmıştır. Tablo 3'te Kesinlik (K) ve Hatırlama (H) performans ölçütlerine göre önerilen yöntem ile literatürde önerilen yöntemleri başarımları verilmiştir. Önerilen yöntem aynı veri seti üzerinde daha önce yapılan çalışmalardan daha iyi bir sınıflandırma başarımı sağladığı görülmüştür.

Tablo 3. *Her bir sınıflandırıcı için kesinlik ve hatırlama değerleri*

Yöntem	Silah		Ninja Yıldızı		Jilet		Doğruluk
	K	***H***	***K***	***H***	***K***	***H***	**(%)**
BoWs (Baştan vd., 2011)	0.65	0.84	1.0	0.92	1.0	0.97	89.4
Seyrek KNN (Mery vd., 2016)	0.99	0.97	1.0	0.99	0.97	0.94	97.7
AISM (Riffo ve Mery, 2016)	0.97	0.97	0.95	0.96	0.99	0.88	95.2
XASR+ (Mery vd., 2015).	0.92	0.88	0.69	1.0	0.78	0.97	86.7
DVM (Mery vd., 2017)	0.9	0.99	1.0	0.85	0.51	1.0	86.0
Önerilen Yöntem	**1.0**	**0.96**	**0.98**	**1.00**	**1.0**	**1.0**	**99.00**

Tablo 3'te BoWs kelimeler çantası yöntemi olup SIFT yöntemine göre görüntülerden anahtar noktalar çıkarmakta ve rastgele orman yöntemine göre sınıflandırma işlemi yapılmaktadır (Baştan vd., 2011).

Seyrek KNN K-ortalamalar kümeleme algoritması ile bölütleme yapmakta ve SIFT anahtar noktalarını elde etmektedir. Daha sonra özellik seçimi yapılarak sınıflandırma işlemi yapılmaktadır (Mery vd., 2016).

Uyarlamalı örtülü şekil modelinde (AISM) görüntünün benzer parçaları elde edilen SIFT anahtar noktalarına göre bir kümeleme yöntemi ile kümelenmektedir. Test aşamasında ise gelen test örneğine en benzer parçalara göre nesne tanıma işlemi yapılmaktadır (Riffo ve Mery, 2016).

Uyarlamalı seyrek gösterim (XASR+) x-ray görüntülerinden nesne tanıma için geliştirilmiştir. Bu yöntem bazı rastgele parçalar görüntüden çıkarılarak bir gösterim modeli oluşturulur. Ayrıca birbirine benzeyen parçalar modelden çıkarılır. Test aşamasında ise test görüntüsünden çıkarılan modeli en yakın temsil eden eğitim verisine göre sınıflandırma yapılmaktadır (Mery vd., 2015).

Destek vektör makinalar ise görüntüden elde edilen SIFT anahtar kelimelerine göre sınıflandırma işlemi yapmaktadır. Tablo 3'te edilen sonuçlar göz önüne alındığında önerilen

derin öğrenme tabanlı yöntemin hem kesinlik ve hatırlama hem de doğruluk oranı açısından daha iyi sonuçlar elde ettiği görülmektedir (Mery vd., 2017).

Sonuç ve Gelecek Çalışmalar

Derin öğrenme yöntemlerinin güçlü grafik kartları üzerinde sağlamış olduğu sınıflandırma başarımları, bu yöntemlerin hızlı bir şekilde birçok alanda uygulanmaya başlanmasına neden olmuştur. Bu çalışmada da gömülü bir kart üzerinde ESA ile güvenlik açısından tehdit unsuru taşıyan görüntüleri sınıflandırmak için bir yöntem geliştirilmiştir. Bu amaçla çalışmada Nvidia Jetson Tx2 yapay zekâ geliştirme kartı kullanılmıştır.

Kartın temel özelliği, üzerinde gömülü bir Linux işletim sisteminin bulunmasıdır. Kart üzerine Caffe derin öğrenme kütüphanesi yüklenmiş ve transfer öğrenme algoritması ile eğitim işlemi yapılmıştır. Geliştirme kartı üzerinde tasarlanan LeNet ağı ile görüntü sınıflandırma işlemi yapılmıştır. Bu doğrultuda bıçak, silah, jilet ve Ninja yıldızı gibi görüntüler başarılı bir şekilde sınıflandırılmıştır. Her bir sınıf için nesnelerin, farklı açılardan ve arka plan gürültüleri eklenmiş görüntüler kullanılmıştır. Yapılan testlerde nesne sınıflandırmada başarım oranının (% 99) oldukça yüksek olduğu görülmüştür.

Seçilen uygulama alanı ile derin öğrenme algoritmalarını kullanarak havaalanları, otogar, müze ve büyük alışveriş merkezlerinde görev yapan güvenlik görevlilerinin kontrol görevi için otomatik bir asistan tasarlanması mümkündür. Gelecek çalışmalarda önerilen yöntem, otonom araçlarda tehdit oluşturan durumların tespit edilmesinde, düşük donanım gereksinimleriyle yüksek başarım sağlaması yönünde iyileştirmeler yapılması hedeflenmektedir.

Kaynaklar

Baştan, M., Yousefi, M. R., & Breuel, T. M. (2011). Visual words on baggage X-ray images. *Computer analysis of images and patterns*, 360-368, Springer, Berlin, Heidelberg.

Chen, Y. (2015). Convolutional Neural Network for Sentence Classification, *Yüksek lisans tezi*, University of Waterloo, Kanada.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. (2018). Recent advances in convolutional neural networks, *Pattern Recognition*, **77**, 354-377.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition, *Proceedings*, IEEE Conference on Computer Vision and Pattern Recognition, 770-778.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory, *Neural Computation*, **9**, 8, 1735-1780.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., ... & Darrell, T. (2014, November). Caffe: Convolutional architecture for fast feature embedding, *Proceedings*, 22nd ACM international conference on Multimedia, 675-678.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, **25**, 1097-1105.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition, *Proceedings of the IEEE*, **86**, 11, 2278-2324.

LeCun, Y. 2015. LeNet-5, Convolutional Neural Networks. URL: http://yann.lecun.com/exdb/lenet/. Erişim tarihi: 12 Ekim 2018.

Mery, D., Svec, E., Arias, M., Riffo, V., Saavedra, J. M., & Banerjee, S. (2017). Modern computer vision techniques for x-ray testing in baggage inspection, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, **47**, 4, 682-692.

Mery, D., Svec, E., & Arias, M. (2016). Object recognition in X-ray testing using adaptive sparse representations, *Journal of Nondestructive Evaluation*, **35**, 3, 45.

Mery, D., Svec, E., & Arias, M. (2015, November). Object recognition in baggage inspection using adaptive sparse representations of X-ray images, *Proceedings*, Pacific-Rim Symposium on Image and Video Technology, 709-720, Springer, Cham.

Niu, X. X., & Suen, C. Y. (2012). A novel hybrid CNN–SVM classifier for recognizing handwritten digits, *Pattern Recognition*, **45**, 4, 1318-1325.

Parliament, E. (2012). Aviation security with a special focus on security scanners. European Parliament Resolution (2010/2154 (INI)), 1-10.

Ponti, M. A., Ribeiro, L. S. F., Nazare, T. S., Bui, T., & Collomosse, J. (2017, October). Everything you wanted to know about Deep Learning for Computer Vision but were afraid to ask. In Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2017 30th SIBGRAPI Conference, 17-41.

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: towards real-time object detection with region proposal networks, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **6**, 1137-1149.

Riffo, V., & Mery, D. (2016). Automated detection of threat objects using adapted implicit shape model, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, **46**, 4, 472-482.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). Imagenet large scale visual recognition challenge, *International Journal of Computer Vision*, **115**, 3, 211-252.

Salur, M. U., & Aydin, İ. (2018, May). Sentiment classification based on deep learning, *Proceedings*, 2018 26th Signal Processing and Communications Applications Conference (SIU), 1-4, IEEE.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions, *Proceedings* IEEE Conference on Computer Vision and Pattern Recognition, 1-9.

Real-Time Detection of Threatening Objects with Embedded Deep Learning

Extended abstract

Deep learning methods have become an important technology in computer vision and image processing, especially image classification problems. One of the most important reasons for this is the superior achievement of deep learning on different problems. With the thanks to development of the Internet, very big data sets are collected and these data can be processed in real time with the high power graphics processor. However, for many problems, it is a very costly large-scale operation to collect data in this way. For this purpose, pre-trained deep learning models can be used to classify the data of a lower size with the transfer learning method. In this study, a system which is embedded and real time can be developed by transfer learning method to classify objects containing threat elements in the images taken from X-ray devices. This system utilizes the Alexnet deep learning model, which is a convolutional neural network on the Nvidia Jetson TX2 development kit. The X-ray baggage security images including objects such as knives, weapons, razors, and ninja stars have been successfully classified. The performance of the proposed method was compared with the different computer vision techniques previously applied to the same data set and more successful results were obtained.

For the test the proposed method, we set the experimental environment which consists of Nvidia Jetson TX2 development kit. Firstly, we have used 12.000 images for training the Alexnet. Secondly, we used fine-tuning hyper parameters with the help of transfer learning to test the system.

Table 4. *Accuracy and recall values for each classifier*

Methods	Weapon		Ninja Stars		Razor Blade		Accuracy
	P	***R***	***P***	***R***	***P***	***R***	**(%)**
BoWs (Baştan vd., 2011)	0.65	0.84	1.0	0.92	1.0	0.97	89.4
Sparse KNN (Mery vd., 2016)	0.99	0.97	1.0	0.99	0.97	0.94	97.7
AISM (Riffo ve Mery, 2016)	0.97	0.97	0.95	0.96	0.99	0.88	95.2
XASR+ (Mery vd., 2015).	0.92	0.88	0.69	1.0	0.78	0.97	86.7
SVM (Mery vd., 2017)	0.9	0.99	1.0	0.85	0.51	1.0	86.0
Proposed Method	**1.0**	**0.96**	**0.98**	**1.00**	**1.0**	**1.0**	**99.00**

The proposed method was compared with the different methods according to the accuracy rate, precision and recall parameters. Table 3 shows the performance of the methods proposed in the literature by the proposed method according to the criteria of precision (P) and recall (R). The table clearly shows that the proposed method has achieved higher performance than the methods in the literature.

Keywords: *Deep learning, Jetson Tx2 Development card, Embedded systems, Convolutional neural networks, Object classification*