Derin Öğrenme ile Göz Tespiti
Eye Detection by Using Deep Learning

Şamil Karahan, Yusuf Sinan Akgül
Bilgisayar Mühendisliği Bölümü
Gebze Teknik Üniversitesi, Kocaeli, 41400, Türkiye
samilkarahan@gmail.com, akgul@bilmuh.gtu.edu.tr

***Özetçe*—Son yıllarda makine öğrenmesi alanında adından sıklıkla söz ettiren derin öğrenme, literatürde yer alan önemli problemlerin başarım oranında kayda değer iyileştirmeler sağlamıştır. Bu çalışmamızda derin öğrenme kullanarak bir göz tespiti yöntemi gerçekleştirilmiştir. Geliştirilen yöntemde, Caffe kütüphanesinin girdi olarak kabul ettiği konvolüsyon sinir ağı tasarımı yapılmıştır. 3 konvolüsyon ve 3 bileştirme operasyonu içeren bu ağ yaklaşık 50K negatif 16K pozitif imge ile eğitilmiştir. Bu ağın son katmanındaki göz ve göz değil sınıflandırmasına göre verilen girdinin göz olup olmadığına karar verilmektedir. Eğitilen model FDDB ve CACD veri kümelerinden seçilen görüntülerle test edilip Haar göz tespiti yöntemi ile karşılaştırılmıştır. Geliştirilen yöntemin Haar yöntemine göre çağrışım sonucunun daha iyi olduğu, kesinlik sonucunun da CACD veri kümesinde daha iyi olmasına karşın FDDB kümesinde biraz daha düşük çıkmıştır.**

Anahtar Kelimeler — Derin öğrenme; Caffe; Digits; Göz tespiti.

***Abstract*—In recent years, deep learning algorithm has been one of the most used method in machine learning. Success rate of the most popular machine learning problems has been increased by using it. In this work, we develop an eye detection method by using a deep neural network. The designed network, which is accepted as an input by Caffe, has 3 convolution layers and 3 max pooling layers. This model has been trained with 16K positive and 52K negative image patches. The last layer of the network is the classification layer which operates a softmax algorithm. The trained model has been tested with images, which were provided on FDDB and CACD datasets, and also compared with Haar eye detection algorithm. Recall value of the method is higher than the Haar algorithm on the two datasets. However, according to the precision rate, the Haar algorithm is successful on FDDB dataset.**

Keywords — Deep Learning; Caffe; Digits; Eye Detection.

I. GİRİŞ

Son yılların popüler konusu olan ve birçok alanda başarı oranını ciddi oranda artıran derin öğrenme, makinelerin dünyayı algılama ve anlamasına yönelik yapay zekâ geliştirmede en popüler yaklaşım olmuştur. Özellikle önemli bilimsel konferanslara kabul edilen yayınların büyük bir kısmının derin öğrenme yöntemlerini içermesi bu konuda çok sayıda araştırma gruplarının çalıştığını göstermektedir. Bugün, görüntü sınıflandırma, video analizi, konuşma tanıma ve doğal dil öğrenme gibi alanlarda dünyanın önde gelen araştırma grupları -gerek üniversite gerekse özel firmalar- üzerinde çalıştıkları problemleri derin öğrenme yöntemleriyle çözmektedir [1].

Geleneksel öğrenme yöntemlerinde, belli bir matematiksel yönteme dayanarak öznitelik çıkartan algoritma, ilgili veriye uygulanması sonucu elde edilen öznitelik vektörü çeşitli algoritmalarla sınıflandırılmakta veya regresyon değeri hesaplanmaktadır (Tablo-1). Fakat derin öğrenmenin sağladığı en büyük avantaj, önceden oluşturulmuş herhangi bir matematiksel modele dayalı öznitelik çıkarıcıya gerek duymamasıdır. Öznitelik çıkartıcı otomatik olarak kullanılan veri için uygun olacak şekilde öğrenilmektedir. Bu sayede veri için hangi öznitelik tipinin uygun olduğuna karar vermekten ziyade derin öğreneme de kullanılacak olan ağ ve her bir katmanda uygulanacak olan operasyonlar önem kazanmaktadır.

Derin öğrenme algoritmaları sayesinde sınıflandırma problemlerinin yanında regresyon problemleri için de eğitilebilmektedir. Son katmanda basit bir öğrenme yöntemi kullanılmaktadır. Daha karmaşık bir öğrenme algoritması büyük veriler üzerinde işlem süresini uzatacaktır. Buradaki amaçlardan birisi hızlı bir şekilde kullanılan büyük veriyi temsil edecek olan öznitelik çıkartıcıyı öğrenip, verilen imgelere öğrenilen modelin uygulanması ile istenilen katmandan öznitelik vektörünü elde edip SVM, karar ağaçları vb. makine öğrenme yöntemleriyle eğitilebilmesidir.

Derin öğrenme yöntemlerinin en önemli avantajlarından birisi de genelleştirilebilir olmasıdır. Öğrenilen bir sinir ağı yaklaşımı başka uygulamalar ve veri yapıları için kullanılabilir. Elimizdeki veri kümesinin yetersiz olması durumunda, ilgilendiğimiz veri ile alakalı kamuya açık olan ve büyük veri içeren veri kümeleri üzerinde öğrenilen öznitelik çıkartıcıları direkt olarak elimizdeki imgelere uygulanabilmektedir. Bazı veri kümelerinde eğitilmiş derin öğrenme algoritmasının çıktısı olan model dosyaları paylaşılmaktadır. İlgilendiğimiz problem ile ilgili olan model dosyasını kullanarak elimizdeki veriden öznitelik vektörü elde edebilir veya

978-1-5090-1679-2/16/$31.00 ©2016 IEEE

Problem	Girdi Verisi	Öznitelik Çıkarımı	Sınıflandırıcı	Sonuç
İmgeden Nesne Tespiti	Piksel Dizisi	HOG, SIFT, SURF, Gabor…	SVM, Yapay Sinir Ağları, Karar Ağaçları, ..	Tespit edilen nesnenin konumu.
Ses Tanıma	Ses Sinyali	FFT, ..	HMM, Yüzeysel Sinir Ağları, …	Konuşan kişi, ses transkripsiyonu, ...
Doküman sınıflandırma	Karakter Dizisi	Ngrams, ..	Sınıflandırma, HMM, LDA	Başlık Sınıflandırma, Makine Çevirisi,…

Tablo 1 Geleneksel yöntemlerin kullanılmasıyla yapılan sınıflandırma işlemlerinin aşamaları.

elimizdeki veriye daha uygun hale getirilmesi için ince ayar yapılabilir. Bu ince ayar aşamasında modeldeki sınıf sayısına dikkat etmek gerekmektedir. İnce ayar var olan bir modeli hemen hemen daha az parametre ile daha kısa sürede öğrenmeyi amaçlamaktadır. Yapılan çalışmalarda daha fazla verinin kullanılmasının başarımı genellikle artırdığı gözlemlenmiştir. Bu nedenle derin öğrenme ile yapılan eğitimlerin üst düzey iş istasyonlarında günler haftalar sürebilmektedir.

En yaygın kullanılan derin öğrenme yöntemi konvolüsyon sinir ağları algoritmasıdır. Bu algoritma insan görme korteksinden etkilenmiştir. Görsel özniteliklerin hiyerarşini öğrenmektedir. İlk katmanlarda daha düşük seviyeli öznitelikler (basit filtreler) öğrenilirken, ilerleyen katmanlarda yüksek seviyeli (nesnenin bazı parçaları) öznitelikler öğrenilmiştir.

Derin öğrenme algoritmaları literatürde yer alan bazı problemlerin başarısını ciddi oranda artırmıştır. Bilgisayarla görme alanındaki en klasik problemlerden birisi olan sayı tanıma problemini yaklaşık olarak %0.23

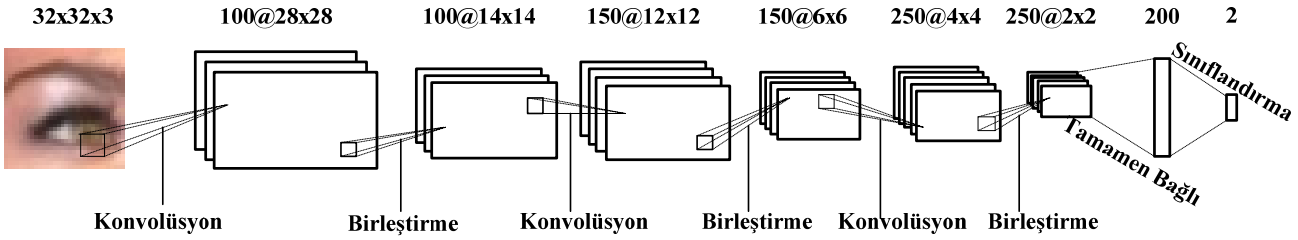
hata (MNIST veri kümesinde) ile çözebilmektedir [2]. Bu problemin çözümünde birden fazla ağ kullanılarak sonuçları birleştirilmiştir. Bir başka problem olan yüz tanımada da son yıllarda ciddi başarımlar elde edilmiştir. Yüz tanımada denektaşı testi olarak kabul edilen LFW (Label Face in Wild) [3] veri kümesindeki imgelerle oluşturulan yüz doğrulama probleminde insan başarısının üstüne çıkılmıştır. İnsanın bu veri kümesindeki başarısı %97.53 olarak verilmektedir. Google araştırma grubunun 2015 yılındaki yayında doğrulama başarısı %99.63 olarak raporlanmıştır [4]. Kullanılan konvolüsyon sinir ağı yaklaşık 8 milyon kişinin 100-200 milyon imgesiyle eğitilmiştir. Eğitim sonucunda elde edilen öznitelik vektörünün Öklid uzaklığına bakılarak doğrulama işlemi gerçekleştirilmiştir. Aynı veri kümesindeki aynileştirme probleminde, verilen bir test imgesine en yakın getirilen kişinin etiketinin test ile aynı olup olmadığına bakılmaktadır. Literatürde bu probleme en iyi sonuç üreten çalışmada da birden fazla konvolüsyon sinir ağı modeli öğrenilmiştir [5]. Sistemin başarısı %96 olarak raporlanmıştır.

Literatürdeki önemli problemlerden bir diğeri de nesne tanıma problemidir. Bu alandaki çalışmalar, ImageNET [6] veri kümesindeki imgeleri kullanarak her yıl başlatılan ILSVRC (Large Scale Visual Recognition Challenge) yarışmasına katılarak sonuçlarını raporlamıştır. Şu anda 21841 nesne kategorisinin yaklaşık 14.2 milyon tane imgesi yer almaktadır. Her yıl bu veri kümesindeki nesne ve görüntü sayısı artmakta ve gittikçe daha da zor hale getirilmektedir. Şekle bakıldığında, 2010 ve 2011 yıllarında geleneksel çözümlerin döndüğü en iyi 5 sonuç içeresinde olmaması durumu %25.8 olduğu gözlemlenmiştir. 2012 yılındaki AlexNET [7] çalışması %9 oranında bir iyileşme sağlamıştır. 2015 yılındaki yarışmanın sonucuna göre, Google tarafından yapılan çalışmada, bu hata oranının %4.82'ye kadar indirildiği raporlanmıştır [8]. 2012 sonrasında derin öğrenme ve ekran kartlarının işin içine girmesi nesne tanıma yarışmalarının her sene zorlaşmasına rağmen başarımın gittikçe artığı gözlemlenmektedir.

Bu çalışmada, göz tespiti problemine derin öğrenme algoritması olan konvolüsyon sinir ağlarıyla çözüm sunulmuştur. Negatif ve pozitif olarak elde edilen imgeler Şekil 1'de gösterilen ağ ile eğitilerek göz ve göz değil ikili sınıflandırması yapılmıştır. Daha sonra eğitimde yer almayan iki farklı veri kümesindeki imgelerle ilgili testler çalıştırılmış ve sonuçlar raporlanmıştır. Aynı görüntüler OpenCV'de yer alan Haar [9] algoritması ile de çalıştırılarak sonuçlar karşılaştırılmıştır. Geliştirilen modelin çağrışım değerinin test edilen veri kümelerinde daha üstün olmasına rağmen, sadece kesinlik değerinin FDDB veri kümesindeki sonucu biraz düşük çıkmıştır.

II. GÖZ TESPİTİ İÇİN VERİ HAZIRLAMA

Göz tespit yönteminin eğitilmesi için öncelikle farklı yüz veri kümelerinden negatif ve pozitif imgeleri içeren veri kümesi hazırlanmıştır. Bu işlem için öncelikle OpenCV kütüphanesinde yer alan göz tespit yöntemiyle elde edilen göz imgeleri içerisinden yanlış doğrular seçilerek negatif kümeye, geri kalanlar da pozitif kümeye eklenmiştir. Yine yüz alanından alınan ve tespit edilen göz bölgesi ile kesişmeyen imgeler de negatif kümesine eklendi. Fakat bu durum Haar tarafından tespit edilemeyen gözleri de içereceğinden imgeler seçilerek pozitif ve negatif kümeye ekleme yapıldı. Bu işlem sonucunda negatif veri kümesinde yaklaşık 52K adet negatif imge, pozitif veri kümesinde ise yaklaşık 16K adet göz imgesi elde edilmiştir. Bu görüntülerin test aşamasında kullanılan göz imgeleriyle aynı olmamasına dikkat edilmiştir.



Şekil 1 Kullanılan ağın yapısı ve uygulanan operasyonların gösterimi.

III. YÖNTEM

Bu çalışmada derin öğrenme konusunda sıklıkla kullanılan Caffe [10] kütüphanesinden yararlanılmıştır. Caffe kütüphanesini web ara yüzünden çalıştırma imkânı sunan DIGITS [11] programı sayesinde sistem eğitilmiştir. Eldeki görüntülerin %20'si doğrulama kümesi olacak şekilde ayarlanarak sistem eğitimi yapılmıştır. Eğitim için Şekil-1'de gösterilen ağ kullanılmıştır. Bu ağda öncelikle 32x32 olarak girdi kabul edilen görüntüye 5x5 boyutunda 100 adet farklı filtre -eğitim sırasında sürekli güncellenen filtreler- uygulanarak konvolüsyon işlemi gerçekleştirilmiştir. Filtre sonucunda elde edilen görüntülerin her birinden 2x2'lik alanlardan elde edilen maksimum değer seçilerek birleştirme işlemi yapılmıştır. Bir sonraki aşamada kullanılan filtre sayısı 150 olacak şekilde 3x3'lük filtreler 14x14 boyutundaki birleştirme sonuçlarına uygulanmıştır. Elde edilen 2B sonuçlar üzerinden tekrar bir önceki aşamadaki birleştirme işlemi gerçekleştirildi. Son konvolüsyon katmanında ise 3x3'lük 250 adet filtre sonucuna yine aynı birleştirme gerçekleştirildi. Elde edilen bu 2B sonuçlar 200 elemanlı tamamen bağlı bir vektöre eşleştirilecek sınıflandırmadan önceki katman oluşturuldu. Bu vektördeki değerler de sınıflandırmanın yapıldığı 2 elemanlı vektöre eşleştirilecek şekilde son katman oluşturuldu. Girdi imgelerinin değerlerini [0-1] aralığına normalize edecek şekilde bir ön işlem uygulandı.

Caffe kütüphanesiyle eğitim sırasında LevelDB veri formatı kullanıldı. Bu veri formatı oluşturulurken toplu eğitim sırasında sürekli aynı sınıfa ait verilerin kullanılmasını engellemek için veri karıştırılarak oluşturuldu. Sistem eğitim aşamasında 64 imge kullanarak toplu işleri gerçekleştirmesi sağlandı. Toplamda her bir imge için 200 tur atacak şekilde eğitim parametresi ayarlandı. Öğrenme katsayısının değerini belli tur aralıklarında düşürerek veriyi daha iyi öğrenmesi sağlanmıştır. Öğrenilen model belli aralıklarla kaydedilerek, doğrulama kümesinin en iyi sonucu ürettiği modelin test olarak kullanılması sağlanmıştır. Bu model kullanılarak C++ ortamında test amaçlı oluşturduğumuz kümelerin test sonuçları hesaplanmıştır.

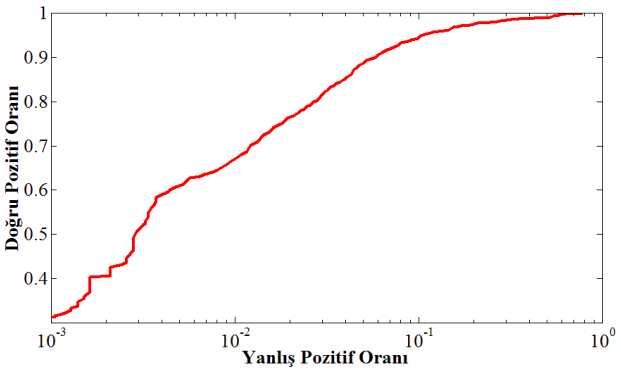
IV. DENEYLER

Eğitim sırasında kaydedilen modellerden doğrulama kümesinde en iyi sonucu üreten model test sırasında kullanılmak üzere nihai model olarak seçilmiştir. Göz tespiti için standart bir veri kümesi bulunmadığından, deney için FDDB [12] ve CACD [13] veri kümelerinden tespit edilen yüz imgeleriyle algoritmalar karşılaştırılmıştır. CACD veri kümesinden seçilen 100 imgenin kapanma, kapalı göz, profil, gözlük gibi zor durumları içermesi sağlanırken, FDDB veri kümesinden ise rastgele 340 imge alınmıştır. Eğitim sırasında kullanılan 32x32'lik küçük imge parçacıklarıyla yüz üzerindeki gözü tespit etmek için kayan pencere yöntemi uygulanmıştır. Sistemin test edilebilmesi için Caffe'nin C++ ara yüzünden yararlanılmıştır. Kayan pencere ile elde edilen test imge parçacıkları 100'lük iş bloğu şeklinde C++ ara yüzüne yüklenen eğitim modeline girdi olarak verildi. Tasarlanan ağda öğrenilen parametrelerin uygulanması ile her bir parçacığın göz olup olmadığı bilgisi elde edildi. Genellikle gözler etrafında birden fazla tespit edilen alanlardan en büyük skoru üreten pencere göz olarak seçilmiştir.

Geliştirilen model ile Haar algoritması, CACD ve FDDB veri kümelerinden elde edilen imgelerle test edilerek kesinlik ve çağrışım değerleri hesaplanmıştır.

Method/Veri Kümesi	Kesinlik (%)	Çağrışım (%)	F-Skor (%)
Deep /CACD	**91.96**	**91.04**	**91.49**
Haar/CACD	86.76	58.71	70.06
Deep /FDDB	81.29	**92.12**	**86.37**
Haar/FDDB	**85.76**	81.95	83.81

Tablo 2 CACD ve FDDB veri kümelerinde geliştirilen yöntem ile Haar algoritmasının kesinlik, çağrışım ve f-skor sonuçları.

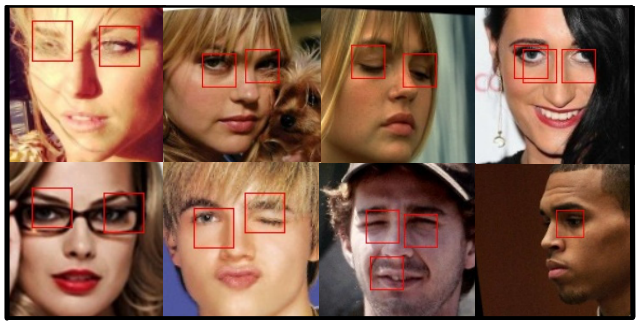


Şekil 2 Geliştirilen yöntemin FDDB veri kümesindeki ROC eğrisi.

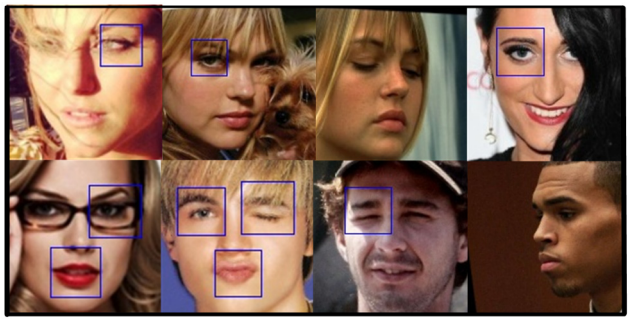
Tablo 2'de gösterilen sonuçlara göre, geliştirilen model çağrışım sonucuna göre her iki veri kümesinde başarılı iken kesinlik sonucuna göre FDDB veri kümesinde Haar sonucunun %3.5 daha iyi çıktığı sonucuna varılmıştır. Bu durum geliştirilen yöntemin bu veri kümesinde daha fazla göz olmayan bölgelere göz olarak sınıflandırmasından kaynaklanmaktadır. Ağız ve burundan gelen bu yanlış pozitif değerlerinin azaltılması için bu bölgelerden daha fazla negatif imgeyle eğitime ihtiyaç duyulduğu sonucuna varılabilir. Şekil 2'de FDDB veri kümesinde tespit skoruna göre doğru pozitif ve yanlış pozitif oranları hesaplanarak ROC eğrisi çizdirilmiştir. ROC eğrisine bakıldığında 1/10 oranında yanlış olarak göz tespit ettiğimiz eşik değerinde sistemin başarısı % 94'ün üzerinde olduğu görünmektedir. Şekil 3'te CACD veri kümesinden seçilen zor durumlara ait bazı imgelerden geliştirilen yöntemin tespit ettiği göz görüntüleri yer alırken, Şekil 4'de ise aynı imgelerin Haar ile olan sonuçları gösterilmiştir. Şekil 5'te ise FDDB veri kümesinde her iki algoritmanın da bazı imgelerde göz olarak tespit ettiği sonuçlar gösterilmiştir.

V. SONUÇLAR VE GELECEK ÇALIŞMALAR

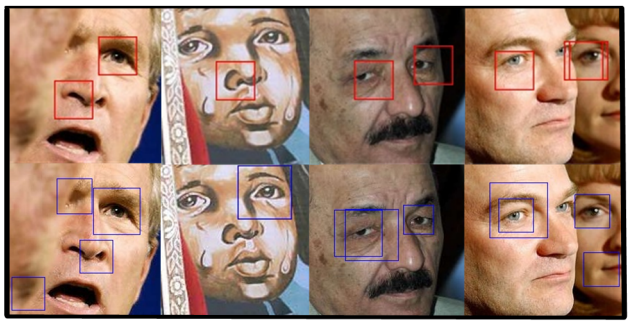
Geliştirilen yöntemin çağırışım değeri her iki veri kümesinden iyi sonuçlar verirken kesinlik değeri de Haar algoritmasına göre FDDB veri kümesinde biraz düşük çıkmıştır. Geliştirilen sistemin zor durumlarda Haar'a göre daha gürbüz olduğu sonuçlardan rahatlıkla görünmektedir. Haar ve geliştirilen yöntemin genellikle ağız ve burundan yanlış pozitif tespit ettiği sonucuna varılmıştır. Bu yanlışın



Şekil 3 CACD kümesinden geliştirilen modelin test sonuçları.



Şekil 4 CACD kümesinden Haar algoritmasının test sonuçları.



Şekil 5 FDDB veri kümesindeki test sonuçları. a) Üst tarafta geliştirilen model b) alt kısımda ise Haar sonuçları yer almaktadır.

giderilmesi için yapılabilecek işler listesine özellikle ağız ve burun etrafından daha fazla negatif görüntünün kullanılmasıyla sistemin eğitilmesi eklenebilir. Burada sistemin tespit ettiği yanlış pozitiflerden yararlanılabilir. Aynı zamanda sistem daha farklı ağ modelleriyle test edilerek başarımın artması sağlanabilir.

KAYNAKÇA

[1] http://www.deeplearning.net.

[2] Ciresan, Dan, Ueli Meier, and Jürgen Schmidhuber, "Multi-column deep neural networks for image classification.", CVPR, 2012.

[3] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments.", University of Massachusetts, Amherst, Technical Report 07-49, October, 2007.

[4] Florian Schroff, Dmitry Kalenichenko, and James Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering.", (CVPR), 2015.

[5] Sun, Yi, Ding Liang, Xiaogang Wang, and Xiaoou Tang, "Deepid3: Face recognition with very deep neural networks.", arXiv preprint arXiv:1502.00873 (2015).

[6] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks.", In Advances in neural information processing systems, pp. 1097-1105. 2012.

[7] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions.", arXiv preprint arXiv:1409.4842 (2014).

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database.", IEEE Computer Vision and Pattern Recognition (CVPR), 2009.

[9] Viola, Paul, and Michael Jones, "Rapid object detection using a boosted cascade of simple features.", Computer Vision and Pattern Recognition, 2001. CVPR 2001.

[10] Jia, Yangqing, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding.", In Proceedings of the ACM International Conference on Multimedia, pp. 675-678. ACM, 2014.

[11] https://developer.nvidia.com/digits.

[12] Jain, Vidit, and Erik G. Learned-Miller, "Fddb: A benchmark for face detection in unconstrained settings.", UMass Amherst Technical Report (2010).

[13] Bor-Chun Chen, Chu-Song Chen, Winston H. Hsu, "Cross-Age Reference Coding for Age-Invariant Face Recognition and Retrieval.", ECCV 2014.