

GAZİ ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ

BM495 Bilgisayar Projesi I  
Araştırma Raporu

Derin Öğrenmeye Dair Bilgi Edinilmesi  
ve  
Bu Alanda Yapılmış Projelerin İncelenmesi

141180001 · Abdullah Akalın · abduallahakalin@gmail.com  
141180400 · Karim El Guermai · karimgreek@gmail.com  
131180027 · Muhammed Emre Emrah · emre95@gmail.com

Danışman:  
Dr. Oktay YILDIZ

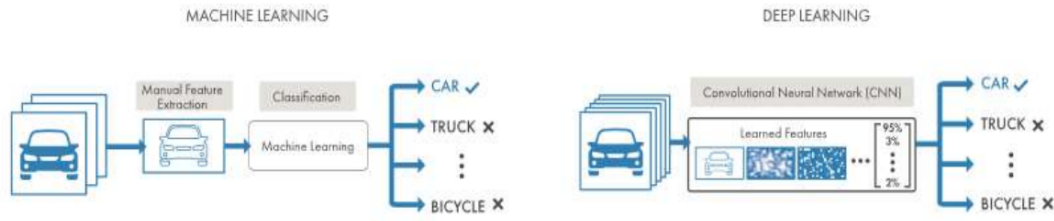
19.10.2017

# İçindekiler

<b>1</b>	<b>Derin Öğrenme: Tanımlar ve Yöntemler</b>	<b>1</b>
1.1	Evrişimli Yapay Sinir Ağları (CNN) . . . . .	1
1.2	Regresyon Yapay Sinir Ağları (RNN) . . . . .	1
1.3	Pekiştirmeli Öğrenme . . . . .	1
<b>2</b>	<b>Proje İncelemeleri</b>	<b>2</b>
2.1	Görüntü Renklendirme . . . . .	2
2.2	Resimlerde Stil Transferi . . . . .	2
2.3	El Yazısıyla Yazılmış Rakamları Tanıma . . . . .	3
2.4	Google Street View Apartman Numaraları . . . . .	4
2.5	Drivers, Focusing on Your Driving! . . . . .	4
2.6	Stack Overflow Query Outcome Prediction . . . . .	4
	<b>Kaynaklar</b>	<b>6</b>

# 1 Derin Öğrenme: Tanımlar ve Yöntemler

Derin öğrenme, yapay öğrenme konusunun bir alt dalıdır. Adını, neredeyse tüm derin öğrenme algoritmalarının dayandığı yapay sinir ağlarındaki "saklı katman" denilen, girdi ve çıktı arasındaki verinin işlenip bir sonraki katmana geçtiği derin yapıdan alır. İnsan beynindeki nöronların çalışma prensibini taklit etmeye çalışır. En iyi avantajı, kendi kendine feature'lar oluşturup kullanabilmesidir. Dezavantajı ise, çok fazla veriye ihtiyaç duymasındır.



## 1.1 Evrişimli Yapay Sinir Ağları (CNN)

Yaygın olarak görüntü işlemede sahne tanıma (scene recog.) ve obje konumlama, tanıma kullanılır. Bir resimdeki ana yapıları (vektörler, arklar vb.), küçük objelerin dahi küçük yapıtaşlarını öğrenir. Bunları kendi kendine bir feature haline getirebilir. Bu sayede daha büyük yapıları tanır. Şaşırtıcı derecede başarılı sonuçlar elde edilebilir.

## 1.2 Regresyon Yapay Sinir Ağları (RNN)

Yapay sinir ağlarının regresyon ile kullanılmasıdır.

## 1.3 Pekiştirmeli Öğrenme

Başka alanlarda da kullanılıyor olup, sistemin davranışı sırasında istenilen şeyleri yapınca ödül, istenmeyen şeyleri yapınca ceza puanı almasını sağlayarak; ve puana etki eden inputlara ağırlık verilerek, sonuçta sistemin istenmeyen davranışları yapmamasını sağlamak amaçlıdır.

## 2 Proje İncelemeleri

### 2.1 Görüntü Renklendirme

**Çalışmanın özgün adı:** Colorful Image Colorization [1].

**Yazarlar:** Richard Zhang, Phillip Isola, Alexei A. Efros

**Kısa Özet:** Bu çalışmanın amacı gri tonlamalı bir fotoğrafın girdi olarak alınıp, çıktı olarak aynı fotoğrafın renklendirilmiş halini üretmektir (Şekil 1). Çıkan görüntünün asıl görüntüyle aynı renklerde olması beklenmemektedir. Esas amaç insan gözü tarafından ayırt edilemeyecek uygunlukta görüntüler çıkartmaktır.

**Kullanılan Yöntemler:** Çalışmada temel olarak CNN (Convolutional Neural Network)'ler ve "Self-Supervised Feature Learning" yöntemleri kullanılmıştır.

**Veri Seti:** ImageNet[2], SUN[3] görüntü verileri.

**Sonuç:** Yapılan deneylerde insan gözlemcilerin %32 oranında, üretilmiş renkli görüntüleri gerçek görüntüler zannettikleri tespit edilmiştir<sup>1</sup>.



Şekil 1: Başarılı renklendirme örnekleri.[1]

### 2.2 Resimlerde Stil Transferi

**Çalışmanın özgün adı:** Semantic Style Transfer and Turning Two-Bit Doodles into Fine Artwork [4].

**Yazar:** Alex J. Champandard

**Kısa Özet:** Bu çalışmanın amacı bir sanat eserindeki karakteristikleri derin öğrenme

<sup>1</sup>Örnek görüntüler için: <http://richzhang.github.io/colorization/>



Şekil 2: Artistik transfer örneği. Asıl görüntü sol-üstte, bilgisayar tarafından çıkartılan taslak sağ-üstte, insan tarafından çizilen taslak sol-altta, bilgisayar tarafından oluşturulan çıktı sağ-altta bulunmaktadır.[4]

yöntemleriyle öğrenip, basit bir karalamayı bu sanat eserine benzer karakteristikte bir görüntüye dönüştürmektir (Şekil 2). Amaç, gerçekleştirilmesi çok zor olan birebir taklit olmayıp, insan gözlemci tarafından, esas alınan esere karakteristik olarak benzeyen görüntü üretmektir.

**Kullanılan Yöntemler:** Çalışmada temel olarak "VGG Convolutional Neural Networks"[5] yöntemleri ve verileri kullanılmıştır.

**Sonuç:** Birebir örtüşme aranmadığı sürece kabul edilebilir sonuçlar gözlemlenmiştir<sup>2</sup>.

### 2.3 El Yazısıyla Yazılmış Rakamları Tanıma

**Çalışmanın özgün adı:** Handwritten Digits Recognition [6].

**Yazarlar:** Yann LeCun, Courant Institute (NYU), Corinna Cortes (Google Labs, New York), Christopher J.C. Burges (Microsoft Research, Redmond)

**Kısa Özet:** Çalışmanın amacı, girdi olarak verilen el yazısı ile yazılmış rakamın hangi rakam olduğunu tanımadır [6].

**Kullanılan Yöntemler:** Deep learning yöntemleri dahil olmak üzere bir çok yöntem ve algoritma kullanılmıştır.

<sup>2</sup>Örnek görüntüler için: <https://github.com/alexjc/neural-doodle>

**Veri Seti:** MNIST SD-1 ve SD-3 veri setleri kullanılmıştır. Toplam 60,000 training images, 60,000 training labels; 10,000 test images ve 10,000 test labels kullanılmıştır. Test setlerinin yarısı güzel yazı yazan kişiler, diğer yarısı da anlaşılması daha zor yazan kişiler tarafından alınmıştır.

## 2.4 Google Street View Apartman Numaraları

**Çalışmanın özgün adı:** The Street View House Numbers (SVHN) Dataset [7].

**Yazarlar:** Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, Andrew Y.

**Kısa Özet:** Apartman tabelalarında yer alan (genelde mavi renkli) apartman numaralarını tanıma ve üzerinde yazan numarayı elde etme.

**Veri Seti:** Training için 73257 rakam, test için ise 26032 rakam bulunuyor. Ek olarak ise, okunması biraz daha kolay olan 531131 rakam bulunmaktadır. Veriler iki formattadır. Birincisinde, orijinal çekilen fotoğraflar, ikincisinde ise 32x32 olacak şekilde, sayıların ortalandığı görüntüler şeklindedir.

## 2.5 Drivers, Focusing on Your Driving!

**Author:** Yundong Zhang [8]

**Summary:** The aim of this project is to build a computer vision system that detects the distracted behaviors of drivers and alarms them when they perform one. The distracted behaviors can be texting, drinking, talking on the phone and etc.

**Methods:** Two methods were used to classify drivers behaviors based on the input images: Support Vectors Machines (SVM) and phone and Convolutional neural network (CNN).

**Data Set:** The data is available on Kaggle Dataset, It is named: Stated Farm Distracted Driver. It consists of 10 classes of image data taken in a car for training purposes. It has 22424 training images and 79727 test images each of size 640x280 pixels.

**Result:** Please refer to Şekil 3.

## 2.6 Stack Overflow Query Outcome Prediction

**Author:** Robbie M. Jones, David Lin [9]

**Summary:** The project basically builds a classifier that predicts whether a user's question will be closed along with the reason for closure. This will reduce the work of moderators and will give better insights for users to ask valid questions.

**Methods:** In this project, the models were trained using logistic regression, Support Vector Machines (SVMs) and boosting.

Model	local accuracy	local log-loss	Kaggle Score
Pure SVC	18.3%	1.94	1.98
SVC + PCA	34.8%	1.73	1.69
SVC + PCA + HOG	40.7%	1.47	1.53
Scratched CNN	63.3%	0.98	1.02
VGG-16	90.2%	0.28	0.34
VGG-CAM	91.3%	0.27	0.32
<b>Ensemble VGG-16 &amp; VGG-CAM</b>	<b>92.6%</b>	<b>0.23</b>	<b>0.24</b>

Şekil 3: Model score.

Model	ACC
Lasso Logistic (L1)	.607
Ridge Logistic (L2)	.610
Adaboost	.552
Linear SVM	.763
Polynomial SVM	.618
Radial SVM	.748

Şekil 4: Classification summary.

**Data Set:** The data used for this project was found on StackExchange Data Explorer and then was preprocessed. Preprocessing included separating closed and non-closed questions, restricting time-sensitive attributes and applying various transformations to textual data.

**Result:** The most accurate models were the linear SVM followed by the radical SVM. The classifiers did a good job separating non-closed questions from closed questions, but not as good when it comes to differentiate between the different reasons for closure (Şekil 4).

## Kaynaklar

- [1] R. Zhang, P. Isola, and A. A. Efros, “Colorful image colorization,” *CoRR*, vol. abs/1603.08511, 2016.
- [2] “Imagenet.” <http://www.image-net.org/>. Eriřim tarihi: 19.10.2017.
- [3] “Sun database.” <https://groups.csail.mit.edu/vision/SUN/>. Eriřim tarihi: 19.10.2017.
- [4] A. J. Champandard, “Semantic style transfer and turning two-bit doodles into fine artworks,” *CoRR*, vol. abs/1603.01768, 2016.
- [5] “Vgg practical.” <http://www.robots.ox.ac.uk/~vgg/practicals/cnn/>. Eriřim tarihi: 19.10.2017.
- [6] “The minst database of handwritten digits.” <http://yann.lecun.com/exdb/mnist/>. Eriřim tarihi: 22.10.2017.
- [7] “The street view house numbers (svhn) dataset.” <http://ufldl.stanford.edu/housenumbers>. Eriřim tarihi: 22.10.2017.
- [8] “Drivers, focusing on your driving!” <http://cs229.stanford.edu/proj2016/poster/SamCenLuo-ClassificationOfDriverDistraction-poster.pdf>. Eriřim tarihi: 22.10.2017.
- [9] “Stack overflow query outcome prediction.” <http://cs229.stanford.edu/proj2016/poster/LinJones-Stack%20overflow%20Query%20Outcome%20Prediction-poster.pdf>. Eriřim tarihi: 22.10.2017.