Machine Learning

In Robotics

Vinh Ngu

Universität Hamburg
Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachbereich Informatik, Arbeitsbereich TAMS
Masterprojekt SoSe 17

1. Juni 2017

Agenda

- Introduction to Machine Learning
- Object Detection
- Convolutional Neural Networks
- Grasping with the help of CNN's

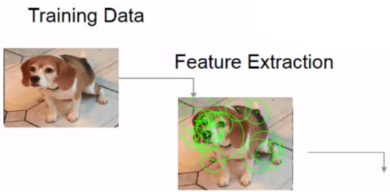
What is machine learning?

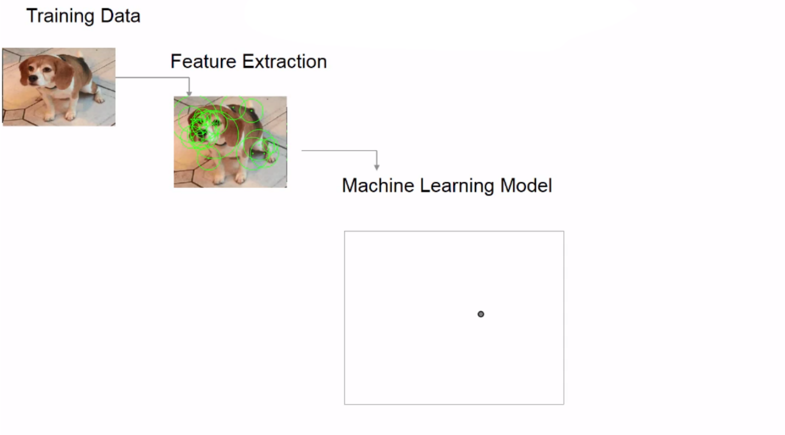
What is machine learning?

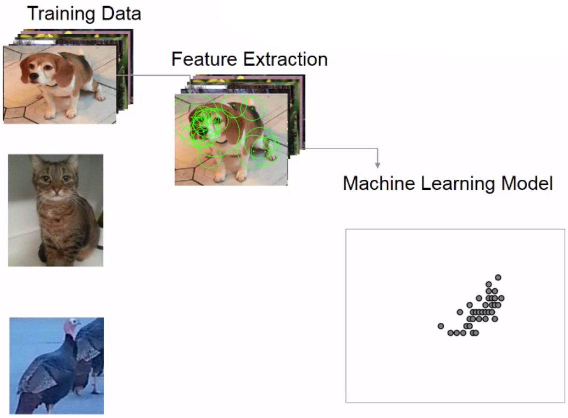
- 1950's
- Arthur Samuel as pioneer
- World's first self-learning program - "checkers"
- Detects hidden patterns
- "Cognitive"functions that humans associate with other human minds
- **Learning** and **Problem-solving**

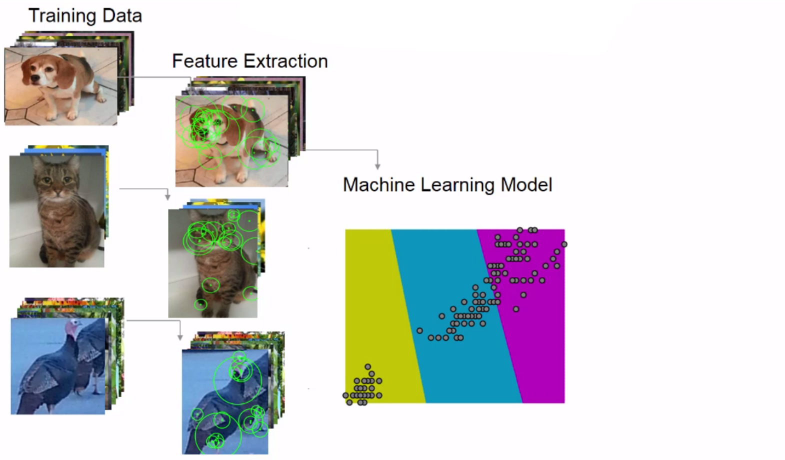
How do machines learn?

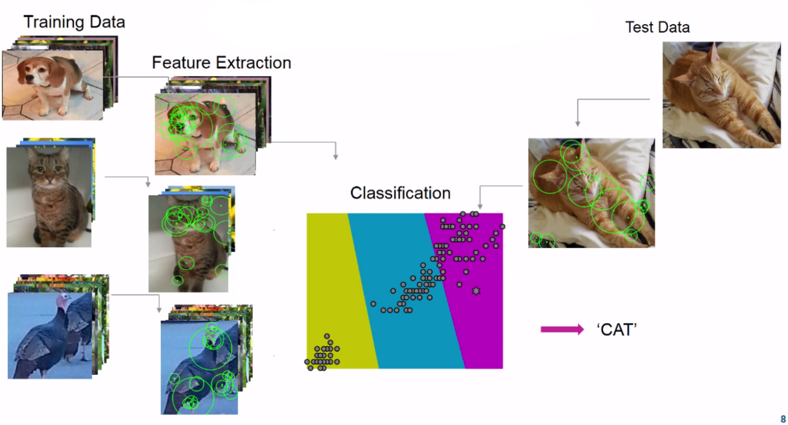
Training Data


Training Data
Feature Extraction


Training Data
Feature Extraction
Machine Learning Model


Training Data
Feature Extraction
Machine Learning Model


Training Data
Feature Extraction
Machine Learning Model


Training Data
Feature Extraction
Classification
Test Data
'CAT'
8

Object Detection

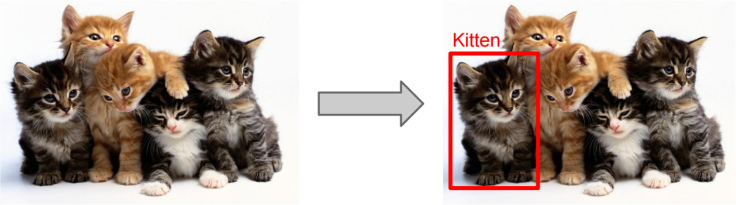
Selective Search

- Published by University of Trento, Italy and University of Amsterdam, the Netherlands, 2012 [1]
- Combines the strength of both an exhaustive search and segmentation.
- Uses use bag-of-words for object recognition.

What we want is...

Object Recognition

Goal:



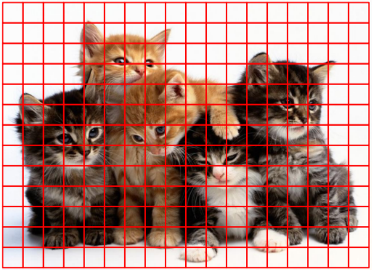
Problem

Where to look at?

One solution

Idea

Let's check everywhere in the image for a possible object!
(Exhaustive-Search in combination for instance "Lampert"[3])



Problem

Extremely slow, must process tens of thousands of candidate objects.

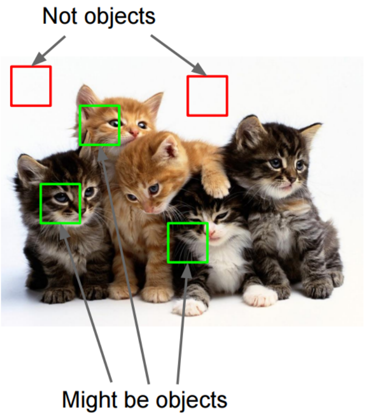
One solution

Idea

Let's use an object detector first!

Problem

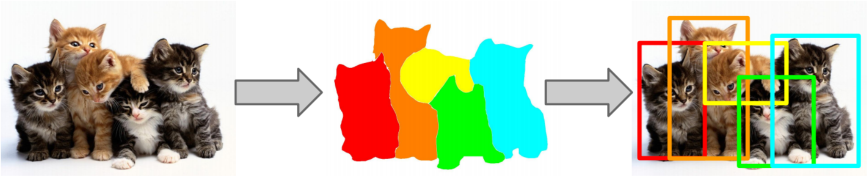
What about oddly-shaped objects?
Will we need to scan with windows of many different shapes?



Final solution

Idea

Let's perform a segmentation first, then run the object recognition. Using the segmentations as candidate for possible objects.



Advantages

Can be efficient, makes no assumptions about object sizes or shapes.

2012

- 2012: Alex Krizhevsky has won the "ILSVRC" (ImageNet Large-Scale Visual Recognition Challenge)
- 2012 first year where a CNN was used to achieve a top 5 test error rate of 15.4%. (2nd 26.2%)
- CNN's grew prominence.

Learning vs Rules

- At the beginning classification uses predefined rules
- The definition of rules becomes impossible by complex images
- Artificial intelligence are used to extract the most relevant characteristics
- Still, modern systems do not learn directly from pixel level

Convolutional Neural Network

What are CNN's?

- A feed-forward artificial neural network
- Inspired by the organization of the animal visual cortex
- Grew prominence in 2012. (Alex Krizhevsky, Classification error: 26% –> 15%.
- Mainly used for image/video recognition and natural language processing.
- Facebook, Google, Amazon.

Region Based Convolutional Neural Networks (R-CNN)

R-CNN's

- Published by the University of California - 2014 [3]
- Combination of CNN and a domain-specific fine-tuning-method.

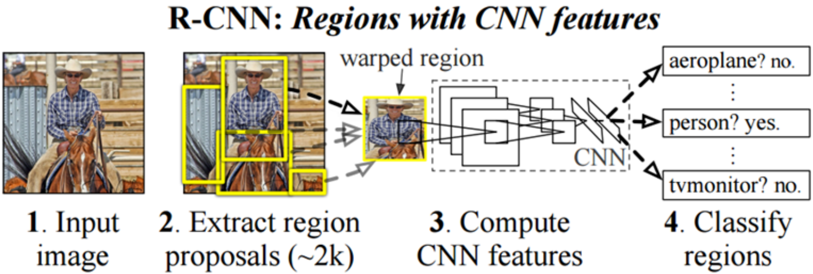
Intresting facts about R-CNN's

Method	**Dataset**	**maP**
Selective Search [1]	PASCAL VOC 2010	35.1%
R-CNN	PASCAL VOC 2010	53.7%
OverFeat [2]	ILSVRC2013	24.3%
R-CNN	ILSVRC2013	31.4%

Definitions

- maP := mean average precision

Workflow of R-CNN



R-CNN workflow

Module design

1. Generates category-independent region proposals.
2. CNN that extracts a fixed-length feature vector from each region.
3. A set of class-specific linear SVMs.

SVM

- **S**upport **V**ector **M**achines.
- Are supervised learning models with associated learning algorithms with the goal of classification and regression analysis.

Module design

1. Generates category-independent region proposals.
   - Makes use of "Selective Search"[1]
2. CNN that extracts a fixed-length feature vector from each region.
   - Convolutional Neural Network predict the object classes.
   - Using the deep-learning framework "Caffe"[1].
3. A set of class-specific linear SVMs.
   - For object recognition.

Further methods based on R-CNN's

- 2013 : R-CNN [3]
- 2015 : Fast-R-CNN [4]
- 2016 : Faster-R-CNN [2]
- 2017 : Mask R-CNN [3]

Learning to Grasp from 50K Tries and 700 Robot Hours

arXiv:1509.06825v1 [cs.LG] 23 Sep 2015

Supersizing Self-supervision: Learning to Grasp from 50K Tries and 700 Robot Hours

Lerrel Pinto and Abhinav Gupta
The Robotics Institute, Carnegie Mellon University
(lerrelp, abhinavg)@cs.cmu.edu

***Abstract*— Current learning-based robot grasping approaches exploit human-labeled datasets for training the models. However, there are two problems with such a methodology: (a) since each object can be grasped in multiple ways, manually labeling grasp locations is not a trivial task; (b) human labeling is biased by semantics. While there have been attempts to train robots using trial-and-error experiments, the amount of data used in such experiments remains substantially low and hence makes the learner prone to over-fitting. In this paper, we take the leap of increasing the available training data to 40 times more than prior work, leading to a dataset size of 50K data points collected over 700 hours of robot grasping attempts. This allows us to train a Convolutional Neural Network (CNN) for the task of predicting grasp locations without severe overfitting. In our formulation, we recast the regression problem to an 18-way binary classification over image patches. We also present a multi-stage learning approach where a CNN trained in one stage is used to collect hard negatives in subsequent stages. Our experiments clearly show the benefit of using large-scale datasets (and multi-stage training) for the task of grasping. We also compare to several baselines and show state-of-the-art performance on generalization to unseen objects for grasping.**

I. INTRODUCTION

Consider the object shown in Fig. 1(a). How do we predict grasp locations for this object? One approach is to fit 3D models to these objects, or to use a 3D depth sensor, and perform analytical 3D reasoning to predict the grasp locations [1]–[4]. However, such an approach has two drawbacks: (a) fitting 3D models is an extremely difficult problem by itself; but more importantly, (b) a geometry based-approach ignores the densities and mass distribution of the object which may be vital in predicting the grasp locations. Therefore, a more practical approach is to use visual recognition to predict grasp locations and configurations, since it does not require explicit modelling of objects. For example, one can create a grasp location training dataset for hundreds and thousands of objects and use standard machine learning algorithms such as CNNs [5], [6] or autoencoders [7] to predict grasp locations in the test data. However, creating a grasp dataset using human labeling can itself be quite challenging for two reasons. First, most objects can be grasped in multiple ways which makes exhaustive labeling impossible (and hence negative data is hard to get; see Fig. 1(b)). Second, human notions of grasping are biased by semantics. For example, humans tend to label handles as the grasp location for objects like cups even though they might be graspable from several other locations and configurations. Hence, a randomly sampled patch cannot be assumed to be

a b

Fig. 1. We present an approach to train robot grasping using 50K trial and error grasps. Some of the sample objects and our setup are shown in (a). Note that each object in the dataset can be grasped in multiple ways (b) and therefore exhaustive human labeling of this task is extremely difficult.

a negative data point, even if it was not marked as a positive grasp location by a human. Due to these challenges, even the biggest vision-based grasping dataset [8] has about only 1K images of objects in isolation (only **one** object visible without any clutter).

In this paper, we break the mold of using manually labeled grasp datasets for training grasp models. We believe such an approach is not scalable. Instead, inspired by reinforcement learning (and human experiential learning), we present a self-supervising algorithm that learns to predict grasp locations via trial and error. But how much training data do we need to train high capacity models such as Convolutional Neural Networks (CNNs) [6] to predict meaningful grasp locations for new unseen objects? Recent approaches have tried to use

General problems in robot's grasping

- Using grasp-data set with human labeling can be quite challenging.
  1. Object can be grasped in multiple ways.
  2. Human notions of grasping are biased by semantics
- Biggest vision-based grasping dataset is only about 1k images. [1]
  1. Objects in isolation.
  2. Could lead to a bad performance under other environments.
  3.

Approaches to deal with the problems.

- Using unlabeled grasping dataset.
    1. Self-supervising algorithm that learns to predict grasp locations via trial and error.
- Created their own dataset for grasping. [1]
    1. 50k items has been collected in 700h of trial and error.

Results

- Test the grasp model both on novel objects and training objects under different pose conditions.
- Still failures even by 700h of “practice”.


Objects
Top Predictions
Executed Prediction
Robot Grasp
Objects
Failed Grasps

Thanks for listening!

References

Lerrel Pinto and Abhinav Gupta. ***The Supersizing Self-supervision: Learning to Grasp from 50K Tries and 700 Robot Hours***. The Robotics Institute, Carnegie Mellon University.

Marco Pedersoli. ***Hierarchical Multiresolution Models for fast Object Detection***. Universitat Autònoma de Barcelona, Bellaterra, April 2012.

Ross Girshick, Jeff Donahue, Trevor Darrell and Jitendra Malik. ***Rich feature hierarchies for accurate object detection and semantic segmentation***. University of California, Berkeley, 2014.

References

- Rodrigo Benenson, Mohamed Omran, Jan Hosang, Bernt Schiele. ***Ten Years of Pedestrian Detection, What Have We Learned?***. Max Planck Institut for Informatics Saarbrücken, Germany.
- Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus and Yann LeCun ***Integrated Recognition, Localization and Detection using Convolutional Networks***. Courant Institute of Mathematical Sciences, New York University
- Ross Girshick, Jeff Donahue, Trevor Darrell and Jitendra Malik. ***Rich feature hierarchies for accurate object detection and semantic segmentation***. UC Berkeley, California, 2014

References

J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. ***Selective search for object recognition.***. IJCV, 2013.

B. Alexe, T. Deselaers, and V. Ferrari. ***Measuring the objectness of image windows.*** TPAMI, 2012.

C. H. Lampert, M. B. Blaschko, and T. Hofmann. ***Efficient subwindow search: A branch and bound framework for object localization.*** TPAMI, 2009.

Ross Girshick. ***Fast R-CNN.*** ICCV, 2015.

References

- Jia, Yangqing and Shelhamer, Evan and Donahue, Jeff and Karayev, Sergey and Long, Jonathan and Girshick, Ross and Guadarrama, Sergio and Darrell, Trevor. ***Caffe: Convolutional Architecture for Fast Feature Embedding.*** 2014.
- Shaoqing Ren, Kaiming He, Ross Girshick and Jian Sun. ***Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.*** 2016.
- Kaiming He, Georgia Gkioxari, Piotr Dollár and Ross Girshick. ***Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.*** Facebook AI Research (FAIR), 2017.

References

Yun Jiang, Stephen Moseson, and Ashutosh Saxena. ***Efficient grasping from rgbd images: Learning using a new rectangle representation.*** 2011.