Deep Learning for Object Recognition on a Mobile Robot

Written by Team: Ice Station Zebra Associates

October 27, 2017
Oregon State University
CS 461 Fall 2017

Michael Rodriguez, Julian Weisbord, Miles McCall

Abstract

This paper goes in-depth into the requirements that we as a team must accomplish. We will be working to build an image classifier to work with our autonomous robot to help recognize basic objects in different environmental settings.

1 1. Introduction

1.1 Purpose/Scope

For our senior design capstone project, we will build an image classifier on top of an autonomous robot. By leveraging ROS (Robot Operating System) and the existing mobile robot platform, we can devote all of our resources to sequentially training a Convolutional Neural Network (CNN) with online learning. In this project, we propose a plan of action and several potential solutions to the issues brought about from sequential learning. Additionally, there are multiple environmental variables that must be addressed during training in order to classify everyday objects in a wide variety of settings. To build a robust classifier, we will pursue three different methods to improve on the current system at the Personal Robotics Lab of OSU. These are: new data capture methods, overfitting to data in different environmental contexts using multiple classifiers, and testing different online learning models to achieve the best classification rate.

1.2 Definitions, acronyms, and abbreviations

Overfitting: Trained model only makes predictions for data set that you have trained and not new data.

Backpropagation: Used to calculate the error contribution of each neuron after a batch of data is processed. Calculates gradient of loss function (gradient descent). Backprogagation is a generalization of the Delta Rule.

Forward Propagation: We apply a set of weights to the input data and calculate an output. For the first forward propagation, the set of weights is selected randomly.

Train Neural Network: Forward Prop (start with random weights) sum products of the inputs with their corresponding set of weights to arrive at first values of hidden layer-> apply activation function to hidden layers -> sum product of hidden layer results with the second set of weights to determine output sum -> take activation function at that output sum to get the final output result -> Back Propagation [Mil15]
-Output sum margin of error = target -calculated

Hyperparameters: Model values set before training on a dataset. Cannot be directly learned from training process. Ex: number of hidden layers in a neural network, learning rate for logistic regression.

Online Machine Learning: A method of machine learning in which data becomes available sequentially. The opposite of batch learning, online learning updates its predictor for future data continuously with each new piece of data.

Catastrophic Interference: Catastrophic Interference becomes present when learning is sequential. Sequential training involves the network learning an input-output pattern until the error is reduced below a specific criterion, then training the network on another set of input-output patterns. A backpropogation network will forget information A if it learns input A and then input B. Catastrophic forgetting occurs because when many of the weights, where knowledge is stored, are changed, it is impossible for prior knowledge about past data to be kept intact. During sequential learning, the inputs become mixed with the new input being superimposed over top of the old input. To recognize multiple sets of patterns, the network must find a place in weight space that can represent both the new and the old output.One way to do this is by connecting a hidden unit to only a subset of the input units. [Kar]

1.3 References

See references at the end of document

1.4 Overview

- Analyze the benefits of at least 4 different lifelong learning techniques added to the pipeline such as: Dual Memory Architecture, Functional Approximation, etc.
- Test the difference in classification success of using Stochastic Gradient Descent and Deep Mind's Synthetic Gradient Descent technique.
- Classify at least 3 different object classes with a minimum of 80% accuracy.

2 2. Overall description

2.1 Product perspective

The product we aim to deliver consists of a code base to be ran on a preexisting operating system with strict hardware boundaries. As a component of this larger existing system, our code base will append to, or replace current models running on the robots.

2.2 Product functions

The software package encapsulates a few major functionalities. The software will control the robot's data capturing system, image tagging, and image recognition software. All of these functionalities currently exist within the running system, we simply aim to replace and improve these functions.

2.3 User characteristics

The intended user of our software package primarily consists of individuals already involved in the robotics lab. These users have a high degree of knowledge on the major topics covered within our project, and already have training on how to interact with the robot and integrated systems. Beyond the lab, however, users expand into a much broader range of individuals. With goals of deploying these robot models in house settings, labs, or workplace environments, the end goal user will have a large spectrum of knowledge and specialzied education regarding the robot and ecompassing software.

2.4 Constraints

Our project is largely framed by the technologies provided to us. We will have access to two models of mobile robot, the Fetch and the PR2. These are running on ROS platforms, which we also have little control over. In regards to training, testing, and designing our machine learning algorithms, we will be utilizing the robotics lab's server environments. Our local or personal machines will likely lack the performance needed to effectively train our models.

2.5 Assumptions and dependencies

Our overarching project depends entirely on the described and agreed upon robot models and operating system environments. The different robots have different functionality and features which will affect the resulting software we create. The operating system, however, should remain more consistent across different platforms. Luckily the ROS platform handles all communications with the underlying hardware and robot accessories, so API's will make our code more portable and modular.

3 Specific Requirements

Analyze the benefits of at least 4 different lifelong/online learning techniques added to the pipeline such as: Dual Memory Architecture, Functional Approximation, etc. Use the most accurate(remembers the most information) Online Learning technique in the final classifiers.

Test the difference in classification success of using Stochastic Gradient Descent and Deep Mind's Synthetic Gradient Descent technique. Whichever gives us the best classification rate will be used as the final backpropagation algorithm.

Classify at least 3 different object classes with a minimum average of 80

4 Appendixes

5 Index

References

[Kar] Andrej Karpathy. Image recognition lectures. `http://cs231n.github.io/`. (Accessed on 10/26/2017).

[Mil15] Steven Miller. Neural nets. https://stevenmiller888.github.io/mind-how-to-build-a-neural-network/, 08 2015. (Accessed on 10/25/2017).