Enhanced robustness of convolutional networks with a push–pull inhibition layer

Presented by Zubia Mansoor

November 9, 2020

Overview

1 Motivation

2 Methods

3 Experiments

4 Summary

5 Questions?

Motivation

Review from last JC

Consider a motivating scenario in radiology

- What happens when the model encounters something it hasn't seen before?
  - For instance, if the X-Ray copies are blurry and noisy
- Changes in the training and test distribution pose a serious challenge to deep learning vision systems

Review from last JC Contd.

- Geirhos et al. "Generalisation in humans and deep neural networks"
- Trained CNNs on different types of image distortions to make them more robust
  - Human visual system appears to be more robust than DNNs for the most part
  - DNNs surpass human performance only when trained on the exact distortions type they are later tested on
- Motivation for today's paper
  - CNNs lack robustness to test image corruptions that are not seen during training
  - Need to improve robustness to classification of corrupted test samples

A Brief Background on Model Robustness

- Data augmentation
  - Included data augmentation schemes such as rotations, cropping to avoid overfitting by CNNs
  - Acquired robustness only to the classes of perturbations used for training
- Adversarial attacks
  - Slightly distorting an input sample for the purpose of confusing a classifier
  - Possibly the worst case of input corruption that networks can be subjected to
- Biologically inspired models
  - Network architecture modelled using simple and complex cells in the visual system of the brain

Goals of this paper

- Overcome the drawbacks in data augmentation methods that requires robustness is learned
- Incorporate mechanisms in network architecture that intrinsically increase their robustness to corruption of input data
- Propose a new layer for CNNs that increases their robustness to several types of corruptions of the input images

Methods

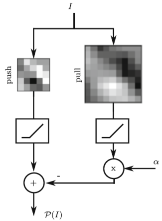
Background

- Inspired by the push–pull inhibition exhibited by neurons in area V1 of the visual system
- Tuned to respond to visual stimuli even when they are heavily corrupted by noise
- Benefits
  - No increase in the number of parameters
  - Only a negligible increase in computation
  - Scalable: can be used in any CNN architecture

Implementation

- Design the push–pull layer $P(I)$ using two convolutional kernels: push, pull kernels

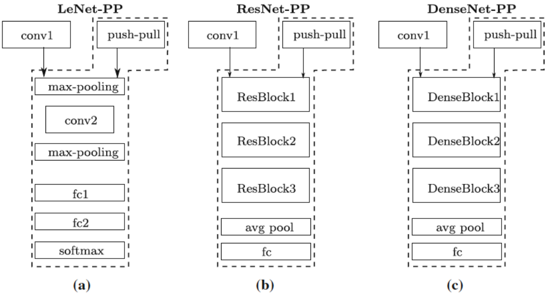
$$P(I) = \theta(k * I) - \alpha.\theta(-k_{\uparrow h} * I)$$



- Pull kernel
  - larger support region
  - weights are computed by inverting and upsampling the push kernel
- Mimic push–pull inhibition by subtracting a fraction of the response of the pull component from that of the push component
- Use ReLU activation for the nonlinear behavior of the push–pull neurons

Implementation Contd.

- Substituting the first convolutional layer of existing CNN architectures



- Do we need to train models from scratch?
  - can replace the first layer of convolutions of an already trained model with the push–pull layer
  - Needs fine-tuning for succeeding layers to adapt to the new responses
- Does it have to be the first layer?
  - can be used at any depth level
  - related to the functions of neurons in early stages of the visual system of the brain

Experiments

Experiments

Switch to paper

Findings

- Classification accuracy on the original test set (without corruption) is not affected by the use of the push–pull layer
- Need models with adequately large capacity to substantially benefit from the effect of the push–pull layer

Summary

Summary

- Proposed a novel push–pull layer for CNN architectures to increase the robustness of existing networks
- Results using LeNet on MNIST and ResNet and DenseNet on CIFAR demonstrate that the push–pull layer considerably increase robustness
- Guarantees a systematic improvement of generalization capabilities of the network measured by the relative corruption error

Questions?