PUNE INSTITUTE OF COMPUTER TECHNOLOGY,
DHANKAWADI PUNE-43.

A Seminar Report

On

Visualization of Convolutional Neural Network(CNN) and effects of Adversarial Examples

SUBMITTED BY

NAME: Atharva Bhagwat
ROLL NO: 3109
CLASS: TE-1

GUIDED BY
PROF. R.A.Kulkarni

COMPUTER ENGINEERING DEPARTMENT
Academic Year: 2018-19

PUNE INSTITUTE OF COMPUTER TECHNOLOGY,
DHANKAWADI PUNE-43.

CERTIFICATE

This is to certify that Mr. **Atharva Bhagwat** , *R*oll No**. 3109** a student of T.E. (Computer Engineering Department) Batch 2018-2019, has satisfactorily completed a seminar report on "**Visualization of Convolutional Neural Network(CNN) and effects of Adversarial Examples**" under the guidance of Prof. R.A.Kulkarni towards the partial fulfillment of the third year Computer Engineering Semester II of SPPU.

Prof. R.A.Kulkarni
Internal Guide

Dr. R.B.Ingle
Head of Department,
Computer Engineering

Date:

Place:

Abstract:

With rapid progress and significant success in a various applications, deep learning is being applied in many safety-critical environments, like self driving cars, face recognition for security etc. Hence it is important for us to learn how it reaches to a conclusion. For this reason we try to visualize the features in deep layers of a neural network. Recently, it is found that deep neural networks are vulnerable to well-designed input samples, called adversarial examples. Adversarial examples are imperceptible to human but can easily fool DNNs in the testing/deploying stage. This vulnerability to adversarial examples becomes one of the major risks for applying deep neural networks in life-critical environments. Therefore, we try to explore for possible solutions from these attacks.

Keywords: Deep Neural Network, Adversarial Examples, Interpreting Neural Networks.

Visualization of Convolutional Neural Network(CNN) and effects of Adversarial Examples

Contents

List of Figures

1 INTRODUCTION

With rapid progress and significant success in a wide spectrum of applications, deep learning is being applied in many safety-critical environments. Therefore, it is crucial to have a model with higher accuracy. In spite of good performance, a deep Convolutional Neural Network is considered as a black box. Boosting the feature interpretability of CNN has attracted a lot of attention recently. A good performing model cannot always ensure correct feature representation. Various methodologies can be used to better understand the neural networks.

Recently it is found that neural networks are vulnerable to carefully generated inputs called adversarial examples. These examples can be generated using various methods. Many solutions are proposed to addressed this problem.

1.1 Motivation

In recent years, convolutional neural networks has achieved superior performance in many visual tasks, sch as object detection and classification. In spite of good performance, a deep CNN has been considered a black-box model with weak feature interpretability for decades. Without clear understanding of how they work, development is reduced to trial and error. Understanding their working will help us in hyperparameter tuning, finding out the failure of the model and understanding why they fail and also we will be able to explain the working to the end user.

Another need to visualize CNN is their failure on some carefully designed inputs known as adversarial examples. Most existing machine learning classifiers are vulnerable to adversarial examples. An adversarial example is a sample input data which has been modified very slightly in a way that is intended to cause a model to misclassify it. Day by day deep neural networks are used in safety critical applications, it becomes very crucial that we address this problem.

1.2 History

1.3 Literature Survey:

1.3.1 Interpretable CNNs

This paper talks about various method by which we can visualize CNNs. It focuses on learning an interpretable CNN where filters of high conv-layers represent specific object parts.

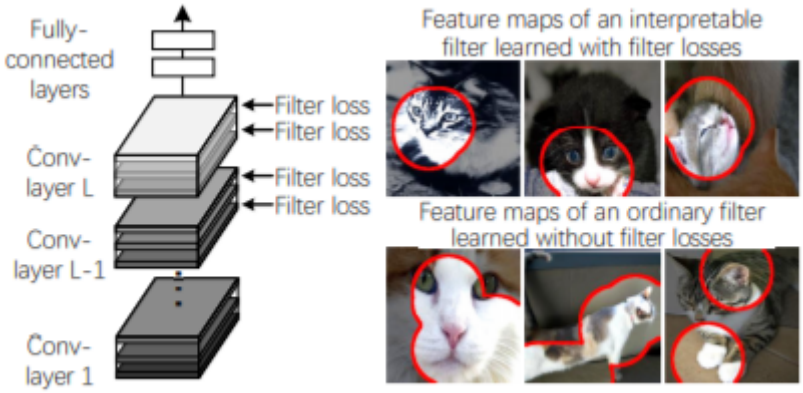


Figure 1: Comparison of an filters

Algorithm

For given input I, the filter f computes a feature map x after ReLU operation. As shown in fig.2, mask layer is added above the interpretable conv-layer.

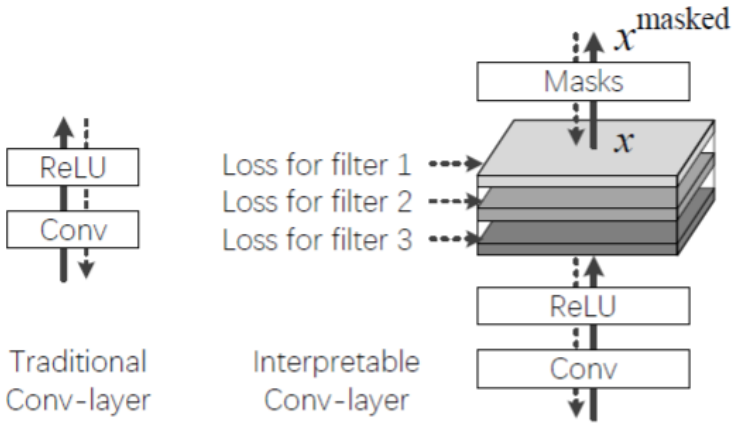


Figure 2: Structure of conv-layers

During forward propagation process, each filter in the CNN passes its information in a bottom-up approach. During back propagation, each filter receives gradients w.r.t its feature map from final task loss on k^{th} sample and filter loss.

Visualization of filters

It is observed that interpretable CNNs usually encode head patterns of animals, although no part annotations were used to train the CNN.

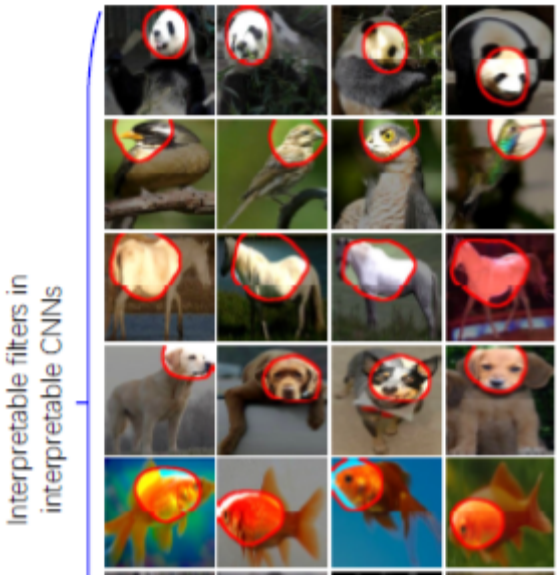


Figure 3: Filter visualization

1.3.2 Adversarial Examples: Attacks and Defenses for Deep Learning

This paper talks about various ways to generate adversarial examples and possible defenses against such attacks.

Fast Gradient Sign Method(FGSM)

Some other attacks such as L-BFGS use an expensive linear search method which is time-consuming. In FGSM, only one step gradient update is done along the direction of the sign of gradient at each pixel.

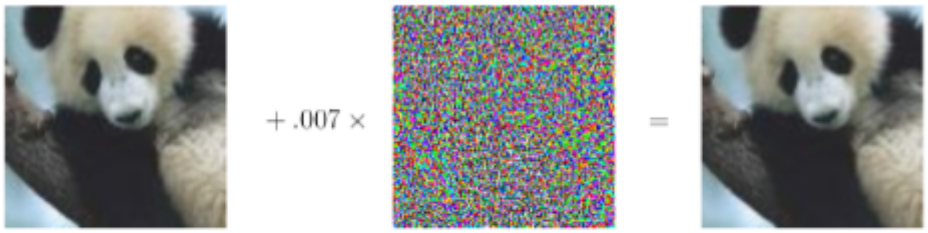


Figure 4: FGSM

Countermeasures for adversarial examples:

There are two types of defense strategies:

- Reactive:
  - Adversarial Detecting: A binary classifier can be trained to detect if the input is adversarial or not.
  - Input Reconstruction: Adversarial examples can be transformed to clean data via reconstruction. After transformation, these examples will not affect the prediction.

- Proactive:
  - Adversarial (Re)Training: Training with the adversarial examples in one of the countermeasures to make neural networks robust. Here, the input sample contains such examples.

1.4 Applications

1.4.1 Automation industry

Day-to-day the application of deep neural networks in automation is increasing. This is a safety critical industry, a small mistake can cost human lives. This requires robust models with high accuracy.

1.5 Challenges

1.5.1 Transferability

It is found that adversarial examples generated against a neural network can fool neural networks with different architectures. This is critical for black-box-attacks, as attacker can train a substitute model and generate adversarial examples w.r.t this model and still victim's model is vulnerable to the attack.

1.5.2 Robustness Evaluation

Most attacks and defenses described their methods without publicly available code, the parameters used in their methods. This brings difficulties for other researchers to reproduce their solutions and provide the corresponding attacks/defenses.

2 DESIGN AND ANALYSIS OF SYSTEM

2.1 Visualization of CNN

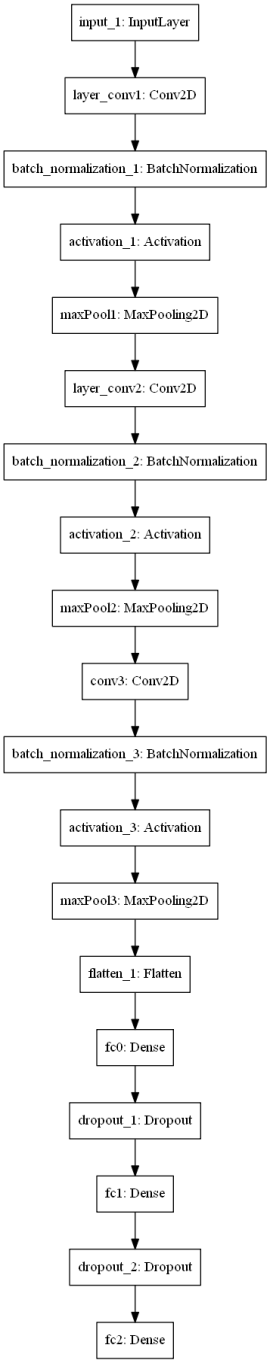
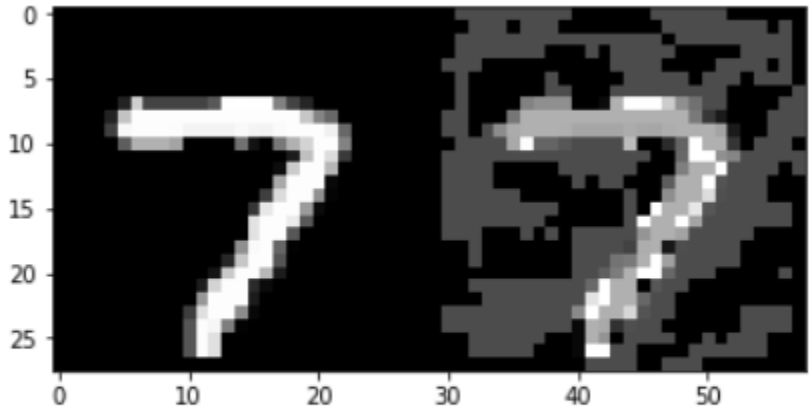


Figure 5: CNN structure

Above model is used for implementation. The model consists of 3 convolutional layers, 3 pooling layers, 2 fully connected layers and the last layer with softmax activation. Softmax activation squashes the outputs of each unit to be between 0 and 1.

3 DISCUSSION ON IMPLEMENTATION RESULTS

Above model was attacked with cleverhans toolbox. The generated image was fed to the model, giving us misclassification.




```
The normal digit is predicted to be a [7]
The adversarial example digit is predicted to be an [2]
```

Figure 6: Input image vs Generated image

4 CONCLUSION AND FUTURE ENHANCEMENT

4.1 Conclusion

We have seen various methods to visualize a CNN and the effect of adversarial examples on its output. We have also seen few defense strategies to prevent our models from such attacks and make the model more robust.

4.2 Future Enhancements

These methods can be used to improve accuracy in safety critical applications, such as:

- Self Driving cars
- Medical applications

References

[1] Interpretable Convolutional Neural Networks(10.1109/CVPR.2018.00920), 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, By: Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu, 2018.

[2] Adversarial Examples: Attacks and Defenses for Deep Learning(10.1109/TNNLS.2018.2886017), IEEE Transactions on Neural Networks and Learning Systems, By: Xiaoyong Yuan, Pan He, Qile Zhu, Xiaolin Li, 2019.

[3] Cleverhans Documentation.