Research Methods

Effectiveness of adversarial example generation methods in image recognition deep learning framework

R. Possas
Supervisor: Y. Zhou

THE UNIVERSITY OF SYDNEY

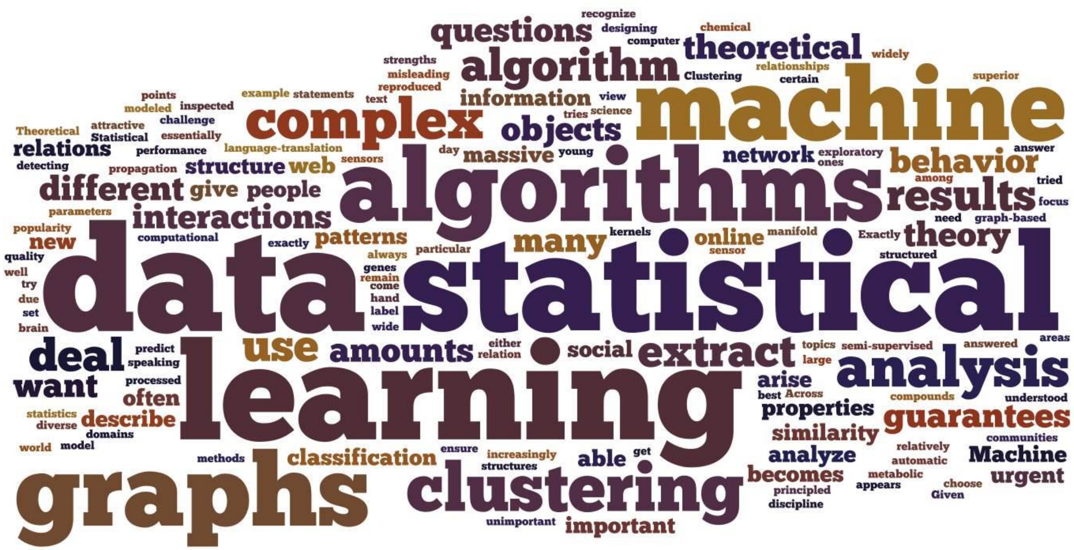
Context

› Smartphones and Computers have become increasingly powerful

› Cameras sensors are ubiquitous

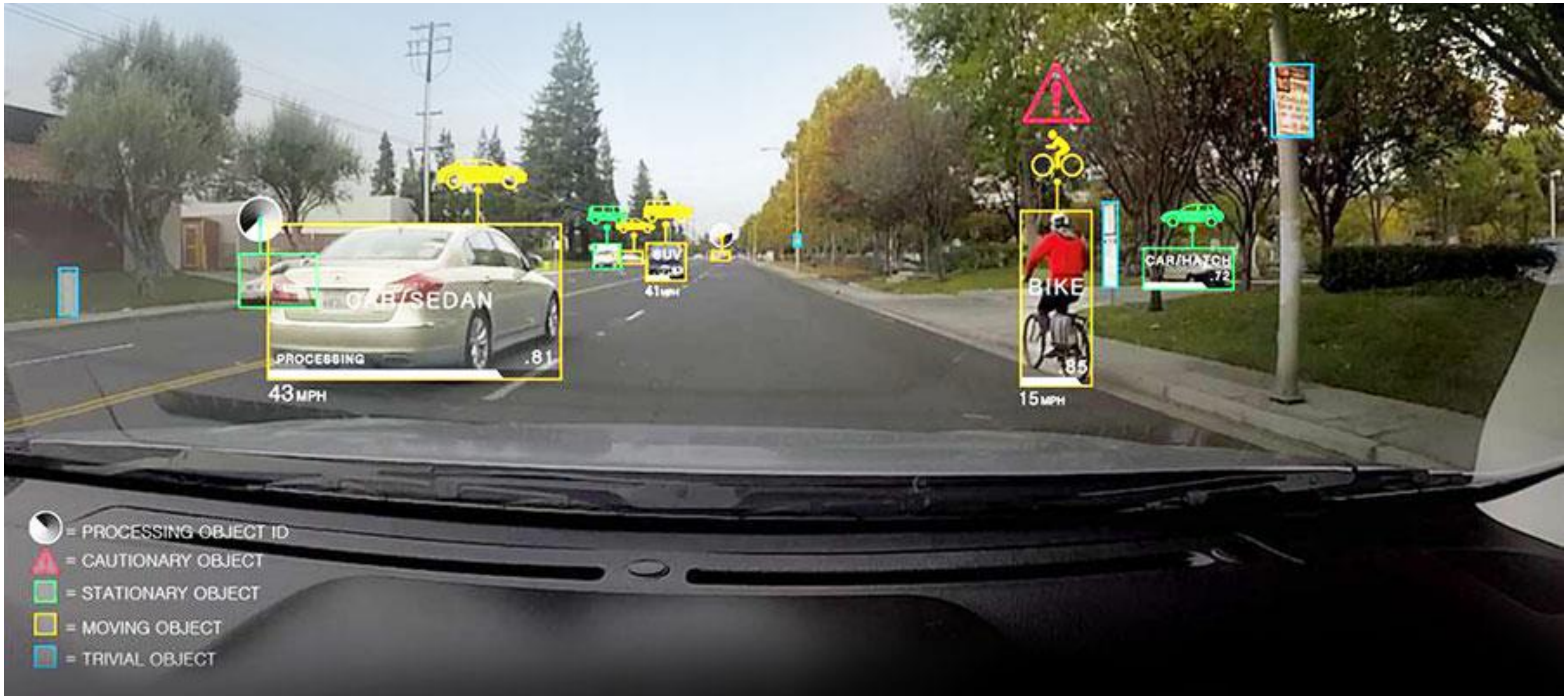
› Machine Learning algorithms are able to run on portable devices

› Data is growing exponentially



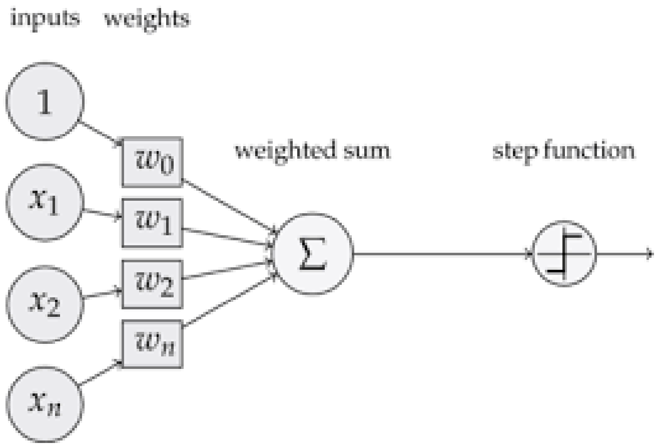
What is Computer Vision?

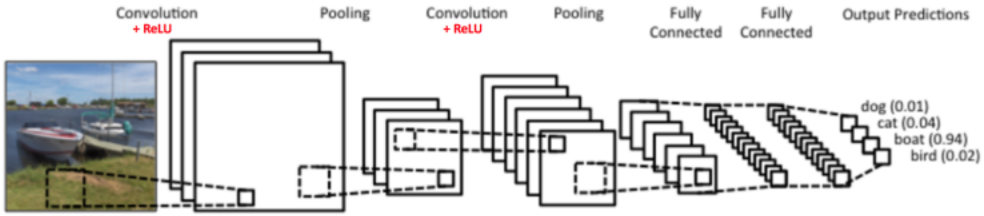
› *" Computer vision is an interdisciplinary field that deals with how computers can be made to gain high-level understanding from digital images or videos"* - Wikipedia



The Evolution of Neural Networks

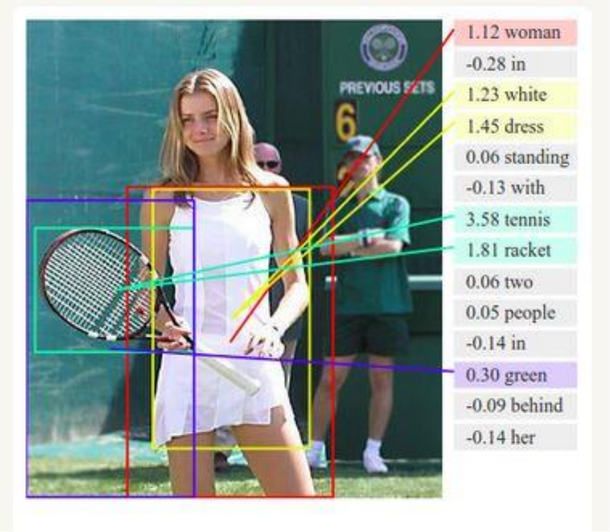
- Networks evolved from single perceptron to multi-layered networks (Convolutional Neural Nets) [2]
- Cloud Computing along with GPU power made it feasible to train complex algorithms
- Highly-non linear models capable of generalizing well [3][6]





Deep Learning

- Uses deep networks instead of shallow models with many layers not always fully connected.
- Multiple processing layers, composed of multiple linear and non-linear transformations
- Learns multiple levels of representation that correspond to different levels of abstraction; the levels from a hierarchy of concepts



Fooling a Deep Learning System

Are those systems safe?

- Wide spread use of computer vision systems can be a threat if algorithms can be “fooled” [12]
- Deep Learning techniques are still a black box and generalization is not fully understood [9][15]
- Methods developed range from black box attacks to ones that require understanding of underlying algorithm structure. [3][6][20]
- Image and Speech recognition being used as “identification” methods in lots of fields
- Deep Neural Networks, can classify images with different levels of confidence.[2]

THE UNIVERSITY OF SYDNEY

Questions and Contributions

Questions:

› How likely images with different levels of classification confidence can be misclassified when perturbed by adversarial methods.

› Which adversarial method yields the higher error rate on a state of the art convolutional neural network?

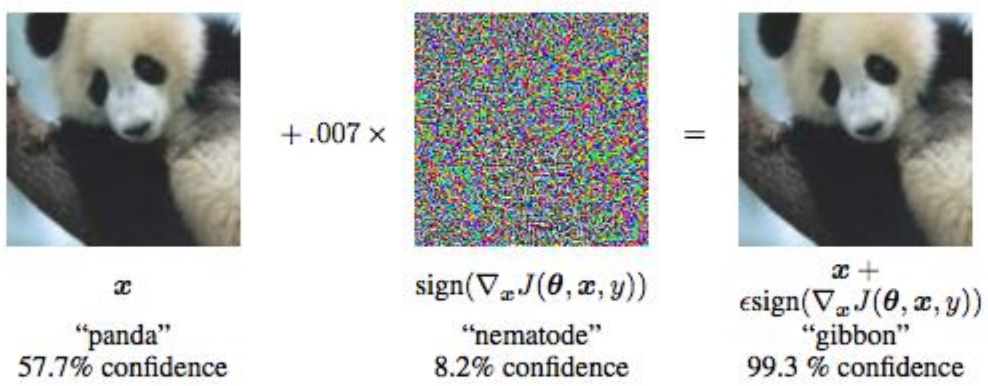
Contributions:

› Identify the relationship between high, medium and low confidence intervals of computer vision classification and the robustness of the output when an image is submitted to different adversarial perturbation methods (fast gradient sign and iterative gradient sign).

Adversarial Examples

What is an Adversarial?

- Small Perturbations to images, usually not recognizable by humans, that completely changes the output label on a classification task [3]
- Methods rely on transfer learning [4]
- Uses gradient information from the targeted system to optimize pixel perturbation within images



Related Work

- Fast Gradient Sign first developed by Goodfellow et al (2014) and has been used on most of the adversarial research until now
- Adversarials have been recently tested in the physical world
- Studies have shown that deep networks have highly linear behavior
- Billovits et al (2014) has categorized adversarial results in four different categories: True Adversarial, Re-Focused, Conaturally and Benign

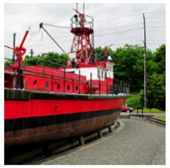
(a) Original Label: Carbonara Adversarial label: Swab, Mop

(b) Original Label: Bassinet Adversarial label: Running shoe

(c) Original Label: Bicycle for two Adversarial label: Plough

(a) Original Label: Fountain - 53% Adversarial label: Fireboat - 99%

(b) Original Label: Dishwasher - 67% Adversarial label: Plate rack - 61% shoe



(c) Original Label: Dock - 63% Adversarial label: Cargo Ship - 69%

(a) Original Label: Keeshond - 53% Adversarial label: Shetland Sheepdog - 82%

(b) Original Label: American Chameleon - 53% Adversarial label: African Chameleon - 52%

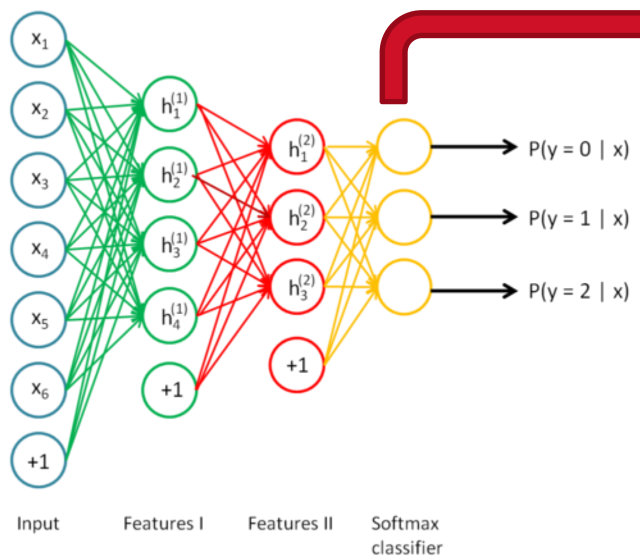
(a) Original Label: Assault Rifle - 40% Adversarial label: Military Uniform - 54%

(b) Original Label: Car Mirror - 72% Adversarial label: Toaster - 62% shoe

(c) Original Label: Lawn Mower - 50% Adversarial label: Go-kart - 75%

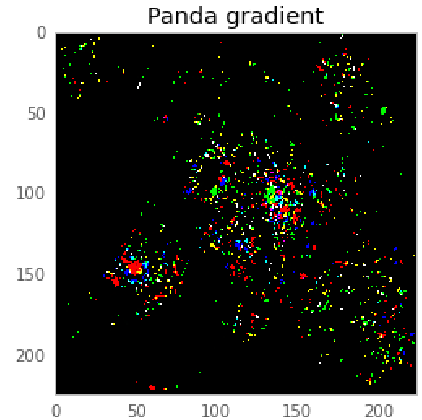
Network Structure and Softmax Layer

How a neural network with softmax looks like?



Probability of each output

Setting an entry to 1 and the others to 0 and running backpropagation yields the gradient for the specific image



Levels of Confidence

- [output >= 30%] on the 1st Softmax Output = High Confidence
- [15% <= output < 30%] on the 1st Softmax Ouput = Medium Confidence
- [output < 15%] on the 1st Softmax Ouput = Medium Confidence

High Confidence

[35%] Panda
[25%] Bear
[15%] Cat
[15%] Dog
[10%] Others entries

Medium Confidence

[15%] Panda
[13%] Bear
[12%] Cat
[10%] Dog
[50%] Other entries

Low Confidence

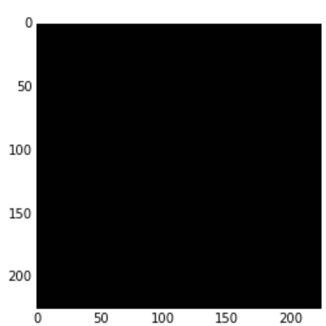
[10%] Panda
[8%] Bear
[7%] Cat
[5%] Dog
[70%] Other entries

Gradient Sign Method: Adversarial Crafting

- Adding noise that emphasizes the pixels in the image with the highest importance, so the resulting perturbation can likely lead to a misclassified result.
- Maximizing the direction of the gradient calculation by multiplying each pixel of the image by the sign of the gradient vector of the desired image used as a perturbation example.
- Fast Method : **Runs once with big delta**
- Iterative Method: **Runs more than once with very small delta**

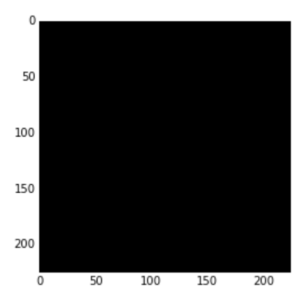
$$C(x+\delta) \approx C(x) + \delta * sign(\nabla C)$$

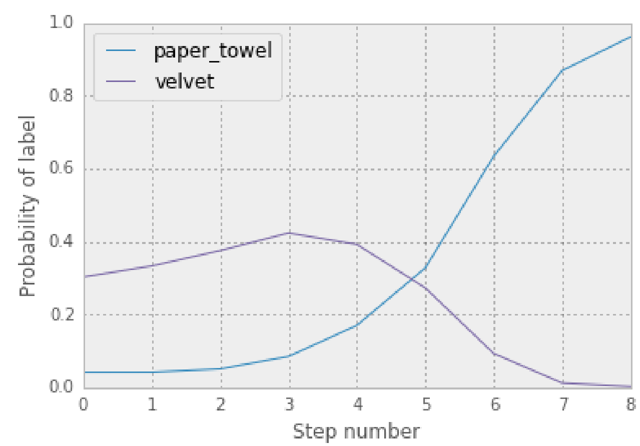
```
label: 885 (velvet), certainty: 27.38%
label: 794 (shower curtain), certainty: 6.4%
label: 911 (wool, woolen), certainty: 6.19%
label: 700 (paper towel), certainty: 4.67%
label: 904 (window screen), certainty: 4.39%
```




```
_ = predict(black + 0.9*delta, n_preds=5)
```

```
label: 885 (velvet), certainty: 54.75%
label: 700 (paper towel), certainty: 16.03%
label: 911 (wool, woolen), certainty: 12.4%
label: 533 (dishrag, dishcloth), certainty: 2.65%
label: 794 (shower curtain), certainty: 2.11%
```





Types of perturbation

Which gradients were used as perturbation factor?

› Least Likely Class Perturbation:

- Sets the prediction of the lowest output to 1 (100%) and retrieve the gradient using backpropagation
- Results on making an image to be more like its weakest top 5 likelihood

› Inverse Perturbation:

- Sets the prediction of the actual prediction to 1 (100%) and retrieve the gradient using backpropagation
- Results on making an image to be less like itself.

Methods and Design

- Use of pre-trained ImageNet deep neural network available on Tensorflow software package (Inception V3 Network)
- Set of 1.500 total images with 500 for each level of confidence (High, Medium, Low)
- Evaluation of results for Inverse and Least Likely Method for both Fast Gradient Sign and Iterative Gradient Sign perturbation technique on all the 3 subset of images.

Results

Top 5 error rate: Image correct label, previously showing on top 5 results, is no longer displayed in top 5 after being perturbed

Top 1 error rate: Image correct label is no longer the one in the first position (higher confidence) after being perturbed

Least Likely Method

	Iterative top 5 error rate	Iterative top 1 error rate	Fast top 5 error rate	Fast top 1 error rate
High	35%	75%	12%	73%
Medium	43%	84%	20%	80%
Low	60%	92%	40%	88%

Inverse Method

	Iterative top 5 error rate	Iterative top 1 error rate	Fast top 5 error rate	Fast top 1 error rate
High	52%	83%	48%	82%
Medium	63%	88%	55%	85%
Low	72%	91%	67%	90%

Conclusions

- A set of small perturbations (Iterative method) are usually more efficient than one big step (Fast Method)
- Inverse gradient perturbation yields higher errors than least likely methods
- Least likely perturbation has higher variance between fast gradient and iterative gradient sign.
- Images with low confidence classification are generally more susceptible to gradient perturbations

References

- [1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in neural information processing systems, 2012, pp. 1097–1105.
- [2] I. G. Y. Bengio and A. Courville, “Deep learning,” 2016, book in preparation for MIT Press. [Online]. Available: http://www.deeplearningbook.org
- [3] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” arXiv preprint arXiv:1412.6572, 2014.
- [4] N. Papernot, P. McDaniel, and I. Goodfellow, “Transferability in machine learning: from phenomena to blackbox attacks using adversarial samples,” arXiv preprint arXiv:1605.07277, 2016.
- [5] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” arXiv, Tech. Rep.,
- [6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, “Intriguing properties of neural networks,” arXiv preprint arXiv:1312.6199, 2013.
- [7] Y. LeCun and C. Cortes, “The mnist database of handwritten digits,” 1998.
- [8] M. Nielsen, “Neural networks and deep learning,” 2016. [Online]. Available: http://www.neuralnetworksanddeeplearning.com
- [9] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in European Conference on Computer Vision. Springer, 2014, pp. 818–833.
- [10] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improvi2016. [Online]. Available: http://arxiv.org/abs/1607.02533

THE UNIVERSITY OF SYDNEY

References (2)

- [11] S. Gu and L. Rigazio, “Towards deep neural network architectures robust to adversarial examples,” arXiv preprint arXiv:1412.5068, 2014.
- [12] A. Nguyen, J. Yosinski, and J. Clune, “Deep neural networks are easily fooled: High confidence predictions for unrecognizable images,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).IEEE, 2015, pp. 427–436.
- [13] N. Dalvi, P. Domingos, S. Sanghai, D. Verma et al., “Adversarial classification,” in Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2004, pp. 99–108.
- [14] D. Lowd and C. Meek, “Adversarial learning,” in Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. ACM, 2005, pp. 641–647.
- [15] C. Billovits, M. Eric, and N. Agarwala, “Hitting depth: Investigating robustness to adversarial examples in deep convolutional neural networks.”
- [16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.
- [17] D. Floreano and C. Mattiussi, Bio-inspired artificial intelligence: theories, methods, and technologies. MIT press, 2008.
- [18] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” in Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014, pp. 675–678.
- [19] K. O. Stanley, “Compositional pattern producing networks: A novel abstraction
- [20] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. Berkay Celik, and A. Swami, “Practical black-box attacks against deep learning systems using adversarial examples,” arXiv preprint arXiv:1602.02697, 2016.
- [21] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.