

Adversarial Examiners for Vision

A.L. Yuille

Bloomberg Distinguished Professor

Dept's Cognitive Science and Computer Science

Johns Hopkins University

Abstract

- There has been enormous progress in computer vision using regression-based techniques like deep networks evaluated on finite-sized balanced annotated datasets.
- But in this talk I will argue that there are limitations to this approach. The key problem is that the space of images is infinitely large and visual scenes are combinatorially complicated.
- This means that evaluating algorithms on finite-sized datasets is problematic and alternative challenges and performance measures, such as adversarial examiners, are required in order to probe for the strengths and weaknesses of vision algorithms.
- I will argue that generative approaches are more promising than regression-based methods to address these challenges.

Deep Nets are Very Effective and Innovative

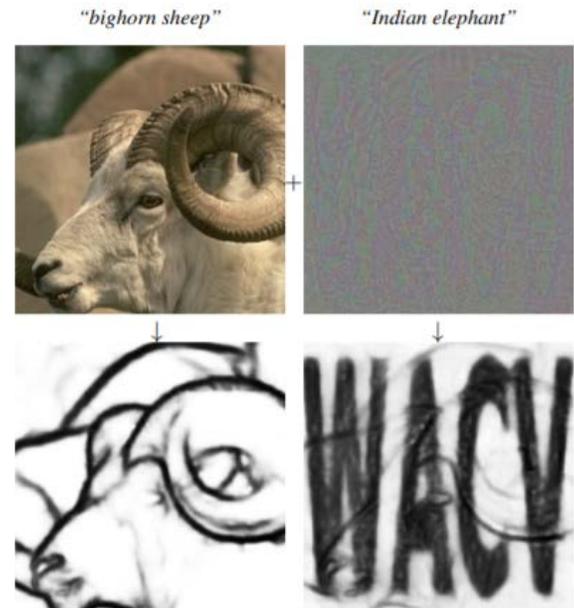
- They are much more *effective* than alternative methods on almost any visual task that can be validated, on a fixed finite-size test dataset.
- They are *innovative*. Recent innovations include:
- Neural Architecture Search (NAS).
- Unsupervised Learning.
- Transformer Networks and Self-Attention Networks.
- Neuro-Modular Networks.

But Deep Networks lack Robustness

- Deep Networks lack robustness to *adversarial attacks* (both *imperceptible* and *perceptible*), to *severe occlusion*, to significant *changes in viewpoint*, to *context*, and to *change of domain*.
- Five Examples:
 1. *Adversarial Attacks – Imperceptible.*
 2. *Changes in Viewpoints and Appearance.*
 3. *Context Changes and Occluders.*
 4. *Patch-Attacks.*
 5. *Domain change.*

1. Adversarial Attacks: Imperceptible

- These attacks exploit the differentiability of the loss function with respect to the input image.
- Imperceptible changes in the image can cause errors for almost all visual tasks: Semantic Segmentation, Pose Estimation, Edge Detection, Classification.





2. Changes in Viewpoint and Appearance

- Sofa detectors trained on ImageNet may not work on unusual viewpoints, or on sofas with unusual colors. Because these viewpoints and colors did not occur in ImageNet.
- We can address this by computer graphics to explore the set of viewpoints and colors. (W. Qiu & A.L. Yuille. ECCV workshop 2016).
- Are these unusual cases rare? They are not rare in the real world.



Fig. 4. Images with different camera height and different sofa color.

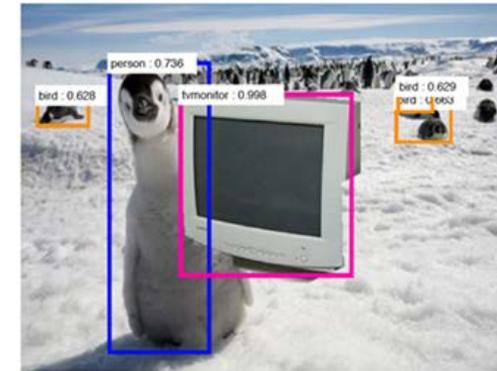
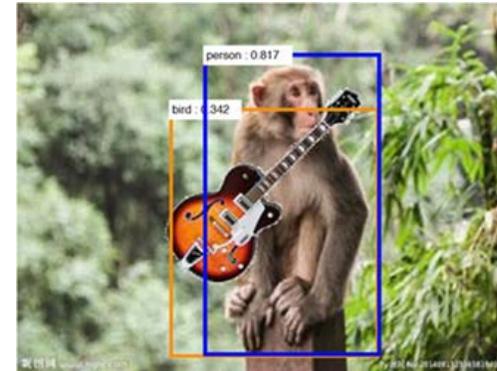
		Azimuth				
		90	135	180	225	270
Elevation	0	-	0.713	0.769	0.930	0.319
	30	0.900	1.000	0.588	1.000	0.710
	60	0.255	0.100	0.148	0.296	0.649

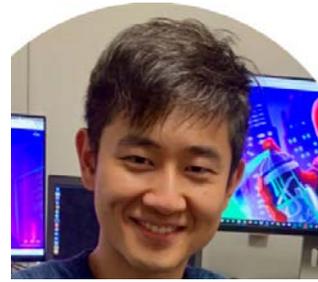
Table 1. The Average Precision (AP) when viewing the sofa from different viewpoints. Observe the AP varies from 0.1 to 1.0 showing the sensitivity to viewpoint. This is perhaps because the biases in the training cause Faster-RCNN to favor specific viewpoints.

3. Context Changes and Occluders



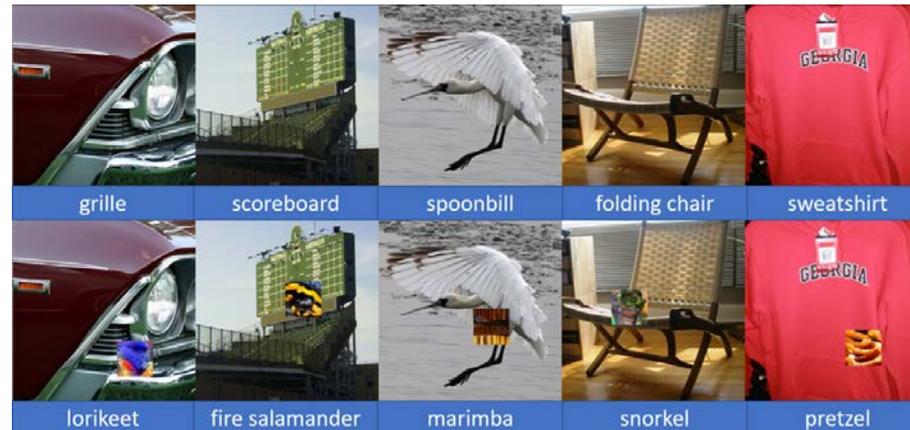
- Context bias. Deep Nets rarely see monkeys carrying guitars or Penguin's holding televisions. This confuses them and makes them misidentify the monkey and penguin as humans (and the guitar as a bird, because the algorithm rarely seeing a guitar with a monkey).
- Jianyu Wang et al. Annals of Mathematical Sciences and Applications. 2018.





4. Patch Attacks.

- Changes to the images which are perceptible but which would not fool a human observer. There is a rapidly growing literature on this topic.
- Here I mention recent work (C. Yang et al. ECCV. 2020).
- This algorithm performs targeted Black Box attacks on Deep Nets. It has a very strong attack rate of over 90% by simply placing small patches in the images.



5. Domain Transfer.

- Humans vision works in many different domains – real images, computer graphics, line-drawings, and art.
- *We can recognize a fish made from bicycle parts, a face made from fruit, and a face made from people.*



- By contrast Deep Networks have difficulties working on different domains. E.g., algorithms trained to detect object boundaries in Images of the countryside may fail to detect them in Indoor Images.

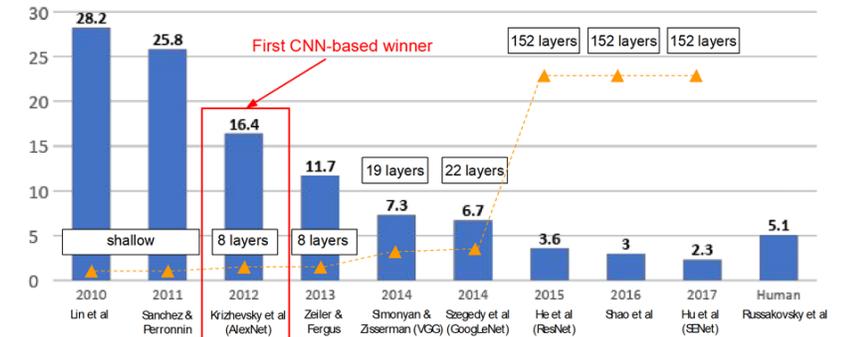
Why? Because the Test Set differs from Training Set

- Deep Nets are regression methods. They assume that the training and testing sets are images which have been randomly sampled from the same underlying distribution. *This relate to PAC theory and is a fundamental assumption of machine learning.*
- This gives a helpfu perspective on non-robustness. *The mistakes arise because the Deep Nets are tested on images that are sampled from a different distribution than the training images.*
- For imperceptible attacks the attacking images are from a very similar distribution. But for other forms of non-robustness the images come from distributions which can be very different.
- *Is this because the datasets are biased?*

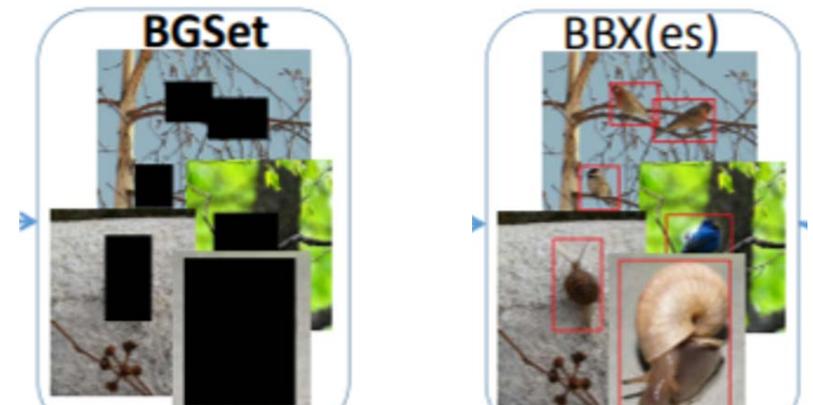
Datasets are Biased. Humans vrs. DNs on ImageNet

- Some people claim that DNs outperform humans on ImageNet.
- *But Deep Nets can classify objects on ImageNet **without even seeing them** much better than humans.* Zhuotun Zhu et al. IJCAI. 2017. BGSet versus BBX(es).
- This suggests that Deep Nets exploit biases in the dataset. *Deep Nets know that the only objects in trees are birds, but humans don't.*
- Humans perform better than Deep Nets on a restricted set of 127 well-known objects on ImageNet.
- *Deep Nets can also do better on objects humans don't care about. They can recognize plants better than me, but not better than an experienced gardener.*
- Comment: classic work on Ideal Observer Models gives many cases where machines can outperform humans.

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) winners



Dataset	AlexNet	Human
OrigSet	58.19%, 80.96%	-, 94.90%*
BGSet	14.41%, 29.62%	-, -
OrigSet-127	73.16%, 93.28%	-, -
FGSet-127	75.32%, 93.87%	81.25%, 95.83%
BGSet-127	41.65%, 73.79%	18.36%, 39.84%



Can we eliminate DataSet bias?

- Consider the five examples we gave.
- *Imperceptible Adversarial Attacks.*
- *Changes in Viewpoints and Appearance.*
- *Context Changes and Occluders.*
- *Patch Attacks.*
- *Domain Transfer.*
- Can we make datasets that are big enough that the biases can be eliminated?

1. Imperceptible Adversarial Attacks.

- The attack images are only slightly different -- small changes to the images in the datasets.
- *We can fix this bias by making our datasets bigger. Replace each datapoint – an image – by a sphere centered on the datapoint.*
- The min-max strategy (Madry et al. 2018): essentially use the images that the algorithm has misclassified as training data.
- How many of them are there? This is unclear, but in practice, there are good defenses which exploit this strategy.
- *So we can probably deal with this.*

2. Viewpoints and Appearances

- Suppose we render the image of an object with 4 viewpoint parameters, 1 material, 4 lighting conditions, and 3 backgrounds.
- Allow 1,000 values for each variable.
- Too many images!

10^{39}



- This gets much worse if we consider an articulated object which has P parts which move semi-independently.
- *We may be able to deal with this. Maybe the set of images is really enormous but a lot of them are very similar to each other?*

3,4 &5. Context, Occluders, Patches, Domains

- But these are exponential.
- An object can be occluded by N possible occluders in M possible locations.



- Or N possible patches in M possible locations.



- *We cannot deal with all of these. We cannot test on all occlusions, or patch attacks, with a fixed finite-sized dataset.*

Finite-sized datasets are not be big enough

- The basic assumption of finite-sized Balanced Annotated Datasets becomes problematic due the combinatorial complexity of the real world. ***We are using the wrong performance measures!***
- *But this yields two big problems.*
- ***(1) How can we test algorithms if finite-sized datasets are not big enough?***
- ***(2) How can we train algorithms if we only have finite-sized training data and know we will have to test on infinite-sized datasets?***

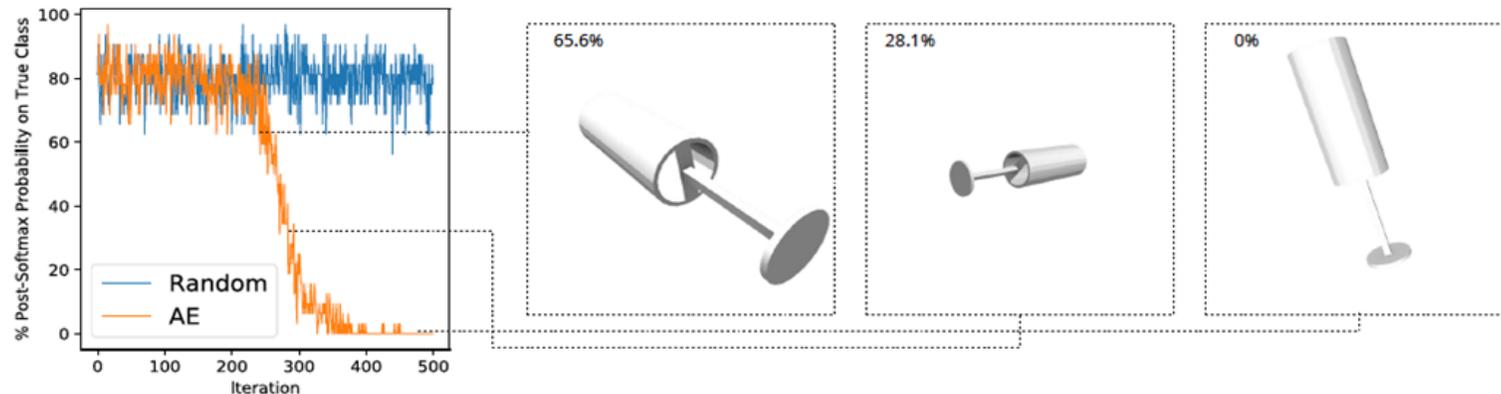
How to test algorithms?



- Computer vision should care about the worst case and not the average case (you don't want your Tesla to detect pedestrians *on average*).
- An *Adversarial Examiner*. Instead of testing an algorithm by its performance on a random sample of images *allow an adversarial examiner to test of a sequence of images, where each image is selected based on the algorithm's performance on earlier images. This enables the examiner to probe the weaknesses of the algorithm. "Let your worst enemy test your algorithm"* .
- ***Would a Professor test his students by asking them a randomly selected set of questions? Better to ask a sequence of questions where each question depends on the student's answer to earlier questions. Like the Game of Twenty Questions.***
- Michelle Shu et al. AAI. 2020.

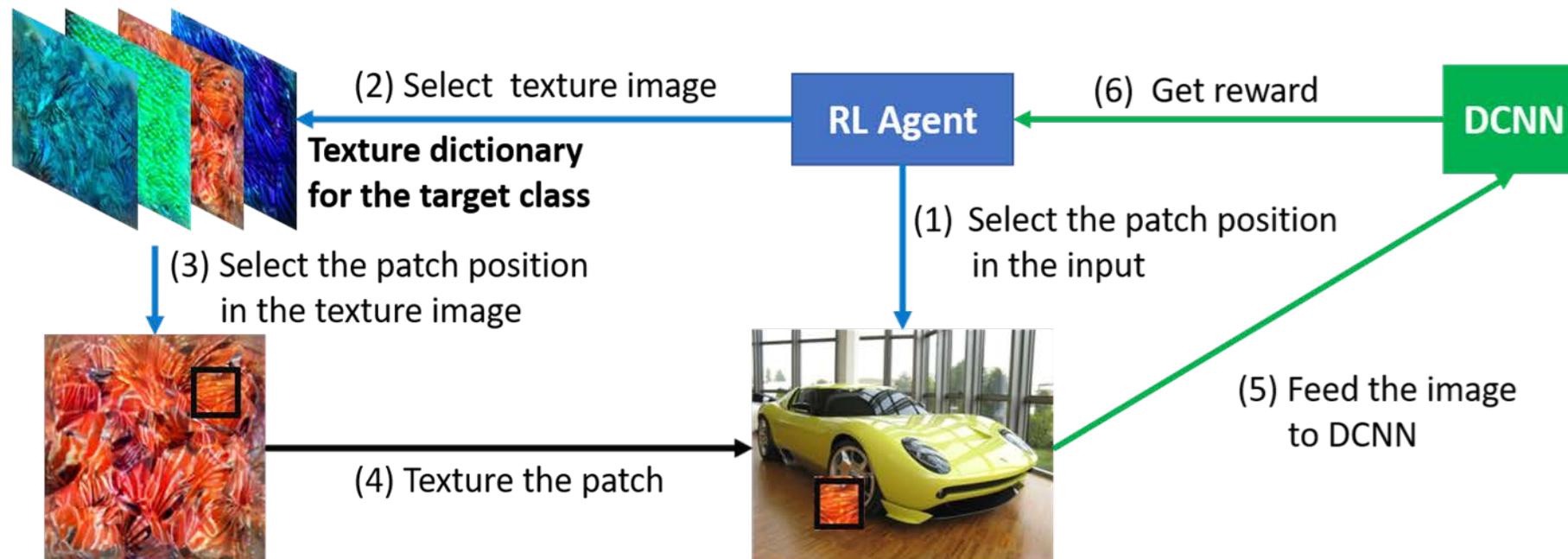
Experiments on ShapeNet: Find Model Weakness

- Test by Adversarial Examiner (AE) instead of by Random testing.
- Search for the worst case using computer graphics to create a *dynamic test set of viewpoints* instead of a (random) fixed test set. Use reinforcement learning to select a series of tests.
- Task: Identify the worst viewpoints where the object can most easily be confused with another object.



Patch Attacks.

- Learn an Attack Policy by reinforcement learning. C. Wang ECCV. 2020.



Adversarial Examiners: Summary

- Balanced Annotated Datasets (BAD) were historically very important and will remain very useful. But they are problematic when faced with the combinatorial complexity of natural images.
- ***No fixed finite-size dataset will be enough.***
- We should introduce more challenging tests for robustness. Our algorithms should be as robust as human observers.
- *We should test our algorithms by adversarial examiners which probe for their weaknesses. Not by testing on random samples.*

Summary

Finite-sized balanced annotated datasets are problematic in face of the combinatorial complexity of the real world. In particular, they are problematic for testing.

For testing – we need to have a dynamic test set – and have a strategy for selecting those images which probe the weaknesses of the algorithms.

For training – generative models enable better generalization outside the training set. In particular, they can be made robust to occluders and adversarial patches.

Some References (1)

- C Xie, J Wang, Z Zhang, Y Zhou, L Xie & A Yuille. “Adversarial Examples for Semantic Segmentation and Object Detection”. ICCV. 2017.
- C Xie, Y Wu, L Maaten, AL Yuille & K He. “Feature Denoising for Improving Adversarial Robustness”. CVPR. 2019.
- C. Cosgrove & A. Yuille. “Adversarial Examples for Edge Detection: They Exist, and They Transfer.” WACV. 2020.
- S. Santurkar, D. Tsipras, B, Tran, A. Ilyas, L. Engstrom & A. Madry. “Computer Vision with a Single (Robust) Classifier”. 2020.
- L. Engstrom, A. Ilyas, S. Santurkar, D. Tsipras, B, Tran & A. Madry. “Adversarial Robustness as a Prior for Learned Representations”. 2020.

Some References (2)

- M. Shu, C. Liu, W. Qiu & A. Yuille. “Identifying Model Weakness with Adversarial Examiner”. AAAI. 2020.
- A. Yuille & C. Liu. “Deep Nets: What have they ever done for Vision?” Arxiv. 2018 (updated 2020).
- Jianyu Wang, Zhishuai Zhang, Cihang Xie, Yuyin Zhou, Vittal Premachandran, Jun Zhu, X Lingxi, Alan Yuille. Visual Concepts and Compositional Voting. Annals of Mathematical Sciences and Applications. Vol. 2. Issue 3. 2018.
- C. Yang, A. Kortylewski, C. Xie, Y. Cao & A. Yuille. “PatchAttack: A Black-box Texture-based Attack with Reinforcement Learning”. ECCV. 2020.
- W. Qiu & A. Yuille. Unrealcv: Connecting computer vision to unreal engine. (Workshop) ECCV. 2016.