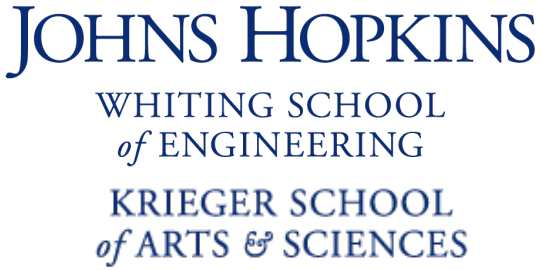


Beyond IID Testing.

Alan Yuille

Departments of Computer Science and Cognitive Science

Johns Hopkins University

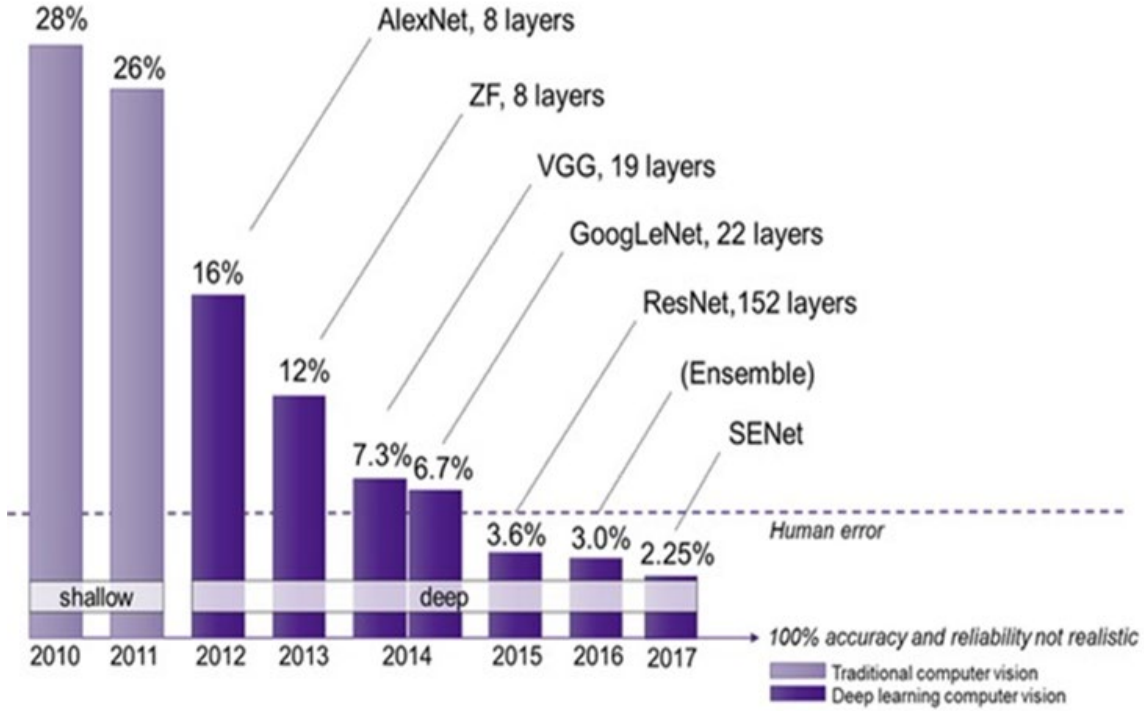
Plan of Talk

- The Mismeasure of AI.
- *Current performance benchmarks are problematic. The need for tougher performance benchmarks like out-of-distribution testing and adversarial examiners. (Yuille and Liu IJCV 2021).*

- *Computer Vision uses a standard ML paradigm to evaluate algorithms. Evaluate algorithms by performance on a random set of samples from the same source as the training data. This is called IID testing.*

Part 1. The Mismeasurement of AI

- AI vision algorithms are very successful when measured on standard academic performance benchmarks.
- Their performance appears to be superhuman.



But they are less successful in the Real World

- In a recent talk A. Karpathy (Vision Group Tesla) reported that AI algorithms could not detect stop signs.
- But stop signs are designed to be easy to detect!

- ***What is the problem? AI algorithms do not generalize from their training set to the real world.***

Balanced Annotated Datasets and I.I.D.testing.

- AI algorithms are benchmarked by average case performance on balanced annotated datasets (BADs).
- The training and testing sets in a BAD are independently and identically distributed (iid) samples from a data source domain.
- Good performance on the testing set guarantees good performance on other data *from the same source.*

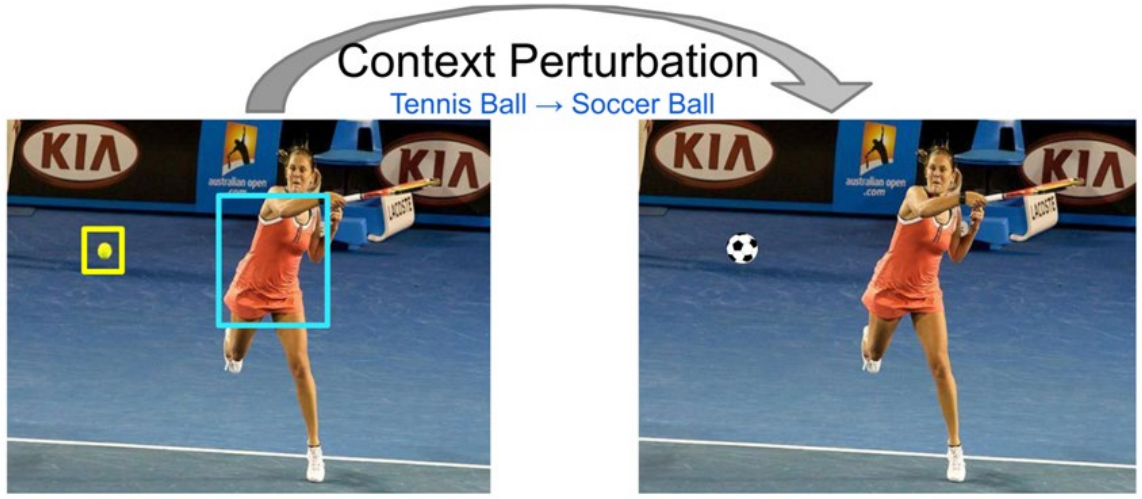
- *These assumptions seem very natural and have a solid mathematical theory supporting them (e.g., Probably Approximately Correct – Vapnik, Valiant, Smale and Poggio…)*
- ***So what is the problem?***

The Problem: Generalizing to the Real World

- *We want results on these BADs to generalize to the real world.*
- But good results on a BAD benchmark only guarantees that the AI algorithms perform well on data that comes from the same source domain.
- To ensure generalization to the real world, the BADs should be big enough to be representative of the real world.
- *But this is not possible. The real world is combinatorially complex. It is impossible for any finite-size BAD to be representative of the real world. They inevitably contain biases which AI algorithms can exploit.*

Problem: Bias example Over-reliance on Context

- There is a growing literature on this.
- Here is a recent examples from (Gupta et al. CVPR 2022).
- *Changing the tennis ball to a football changes the color of the woman's dress!*



Question: What color is the woman's dress?
Ground-truth answer: Orange.
Model prediction: Orange. ✓ **Model prediction:** White. ✗

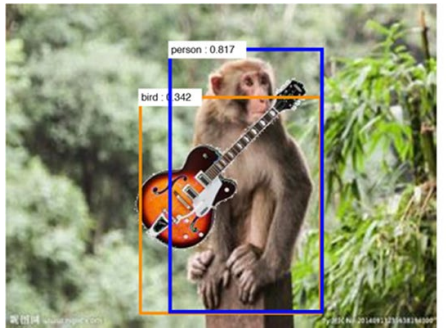
More Examples:

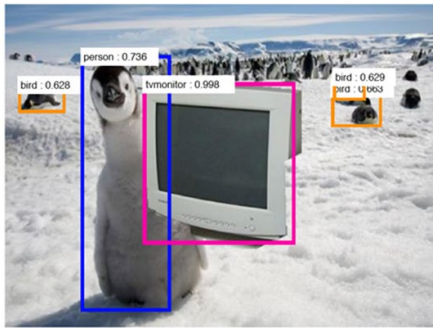
- Counting the number of people in images. Three datasets. Fairly good performance on all datasets. But algorithms trained on one dataset do not transfer to other datasets.
- MIT Technology Article: The AI developed by Google Health can identify signs of diabetic retinopathy from an eye scan with more than 90% accuracy—which the team calls “human specialist level”—and, in principle, give a result in less than 10 minutes. The system analyzes images for telltale indicators of the condition, such as blocked or leaking blood vessels. *The system worked well on datasets in Google, but mostly failed on real patients in Thailand.*

Why don't results transfer?

- Case Study: Edge Detection – Sowerby and South Florida datasets.
- Sowerby Images – English Country Scenes: significant texture in background, edges not very sharp. (figure top left: left panel)
- South Florida dataset – Mostly indoor images: texture regions removed, edges very sharp (step like). (figure top left: right panel).
- Different types of image statistics.
- But transfer is possible with generative methods (Konishi et al.) P(f|on-edge) P(f|off-edge). The statistics P(f|off-edge) differ between datasets but can assume that P(f|on-edge) is similar. Exploit fact that most image pixels are off-edge.
- See Konishi et al. 2003. Handout: edge detection lecture in this course.

Why Don't Results Transfer?





- Bias of context, viewing conditions.
- Rare events. Corner cases. "In the real world everything is a corner case" (Anonymous annotator at a DARPA presentation).
- Background/context bias (see figure top right).
- Caltech 101: fish are the only objects which occur in water (i.e. can detect fish by detecting water).
- UFC activity classification dataset: boxing occurs in a boxing ring – detecting the boxing ring detects boxing (see later).
- Many rockstars are photographed with guitars. But a guitar is not a Rockstar (see presentation later in this lecture).
- A. Torrabla & A. Efros. Unbiased Look at Dataset Bias. CVPR. 2011.

Virtual Data: Making Controlled Datasets

- Tools like UnrealCV enable us to generate datasets which have many annotations and which test algorithms systematically.
- This enables us to stress test algorithms in challenging conditions.

UnrealCV: Weichao Qiu

- UnrealCV: http://unrealcv.org/
- ***Weichao Qiu***

- UnrealCV is a project to help computer vision researchers build virtual worlds using Unreal Engine 4 (UE4). It extends UE4 with a plugin by providing:
- (i) A set of UnrealCV commands to interact with the virtual world.
- (ii) Communication between UE4 and external programs like Caffe.

Using Virtual Stimuli to Stress-Test Algorithms.

- Object detection algorithms (W. Qiu & A.L. Yuille. ECCV workshop 2016).
- E.g., Sofa detectors trained on ImageNet may not work on other data.

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

Elevation \ Azimuth	90	135	180	225	270
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.

- Stress-test binocular stereo. Yi Zhang et al. UnrealStereo. 3DV. 2018.

(a) Specularity (b) No texture

(c) Disparity jumps (d) Transparency

Synthetic Data: Activity Recognition

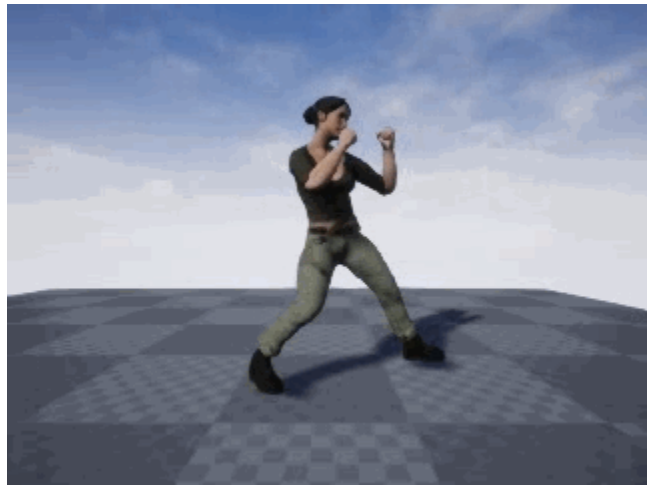
- Activity Recognition is a visual task which is at big risk for combinatorial complexity. Synthetic Data can be used to explore this.
- We render some synthetic videos of humans punching. Train state-of-the art activity recognition methods (TSN and I3D) on these tasks using the USC101 activity dataset.

Model	Class Name	Top-1 accuracy	Top-5 accuracy
TSN	Punching	0.00	0.00
I3D	Punching bag	6.25	41.67
I3D	Punching person	6.25	31.25

- Why are the Deep Nets (TSN and I3D) so bad at generalizing to the synthetic data?
- (There are problems for algorithms trained on real to generalize to synthetic, but they are not usually as bad as this).

Why TSN fail to recognize synthetic punching ?

- Conjecture: TSN model trained on UCF101 (right) may have overfit to background and are unable to localize punching action. Synthetic data consists of a single boxer (left).
- Videos from this class in UCF101 are mostly boxing games and punching sandbags.



Can the TSN correctly localize the punching action ?

- Class Activation Maps (CAM) are a standard technique to detect the discriminative image regions used by a CNN to identify a specific activity class.
- CAMs of punching videos from UCF101 test set – detecting ropes.




Understanding Scenes

- Volleyball Spiking:

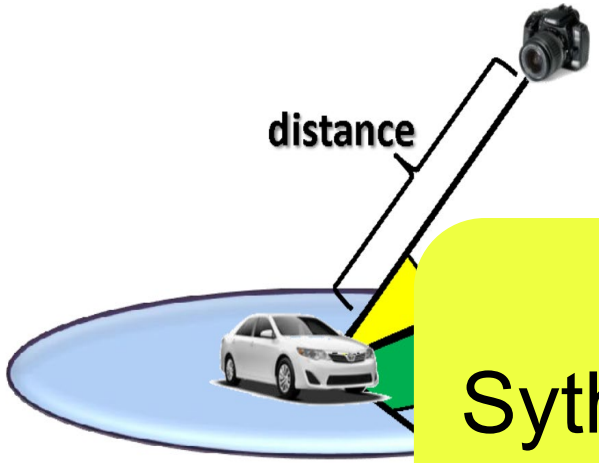
- The model is unable to localize the spiking action in time. It relies on context, i.e. the ability to recognize the scene of volleyball games.

Can you simply make the dataset bigger?

- Thought experiment: for object recognition (unoccluded) the dataset should take into account all the possible viewing conditions (lighting, viewpoint, material, local background).
- From a computer graphic perspective. A model for rendering a 3D virtual scene into an image will have several parameters: e.g.,. camera pose, lighting, texture, material and scene layout. If we have 13 parameters, see next slide, and they take 1,000 values each then we have a dataset of 10^39 images.

- This is a very large number. (Need a model that understands how the data was generated -- i.e. the underlying 13 dimensional space).

Images from synthesized computer graphics model.



Sythesized data: INFINITE image space

Camera Pose(4):
azimuth
elevation
tilt(in-plane rotation)
distance

#light source
type(point, dire
omni)
position
color
...

cene Layout(3):
Background
Foreground
Position(Occlusion)

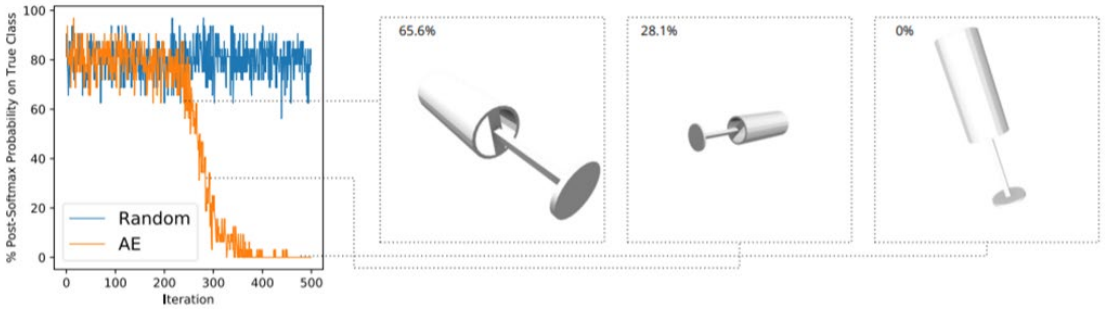
Suppose we simply sample 10^3 possibilities of each parameter listed...

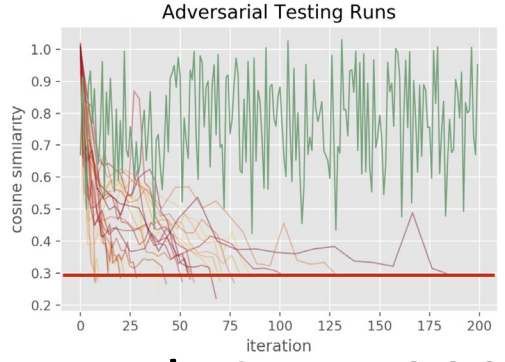
What can we do? Out-of-distribution testing.

- We can supplement performance measures on BADs with tougher tests.
- *We can train the AI algorithm from data on a source domain and test it on data from a different target domain.*
- This is known as out-of-distribution testing in some research communities. By contrast with i.i.d. testing on BAD. I gave examples of this earlier in the course.

What can we do? Adversarial Examiners

- Testing AI on random samples may fail to detect the weak points of algorithms.
- Instead we should search for the worst case(s) and pinpoint the failure modes. Like evaluating mature technologies, e.g., software or cars. This also enables us to find worst case performance instead of average case.
- Red/Orange – adversarial testing. Blue/green – random testing.





- Example: (M. Shu et al. AAAI 2020, N. Ruiz et al. CVPR 2022).