AI: Dangers

CPSC 501: Advanced Programming Techniques
Fall 2020

Jonathan Hudson, Ph.D
Instructor
Department of Computer Science
University of Calgary

Wednesday, August 12, 2020

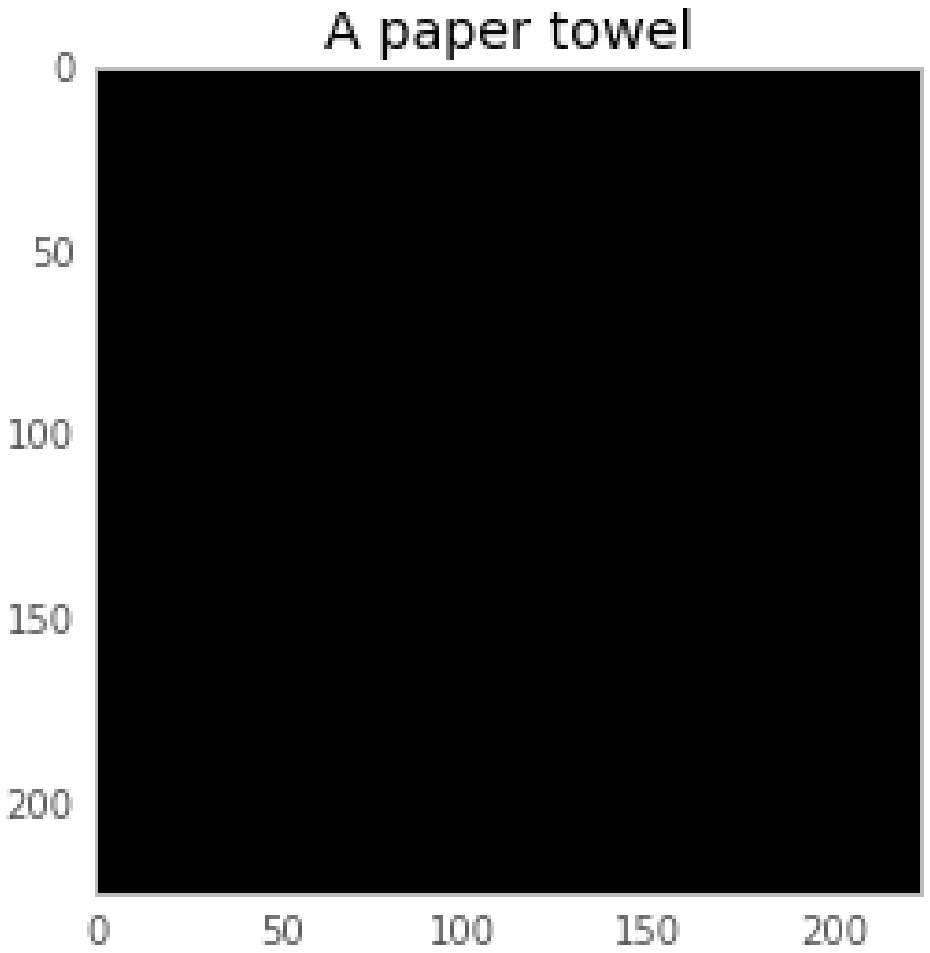
UNIVERSITY OF CALGARY

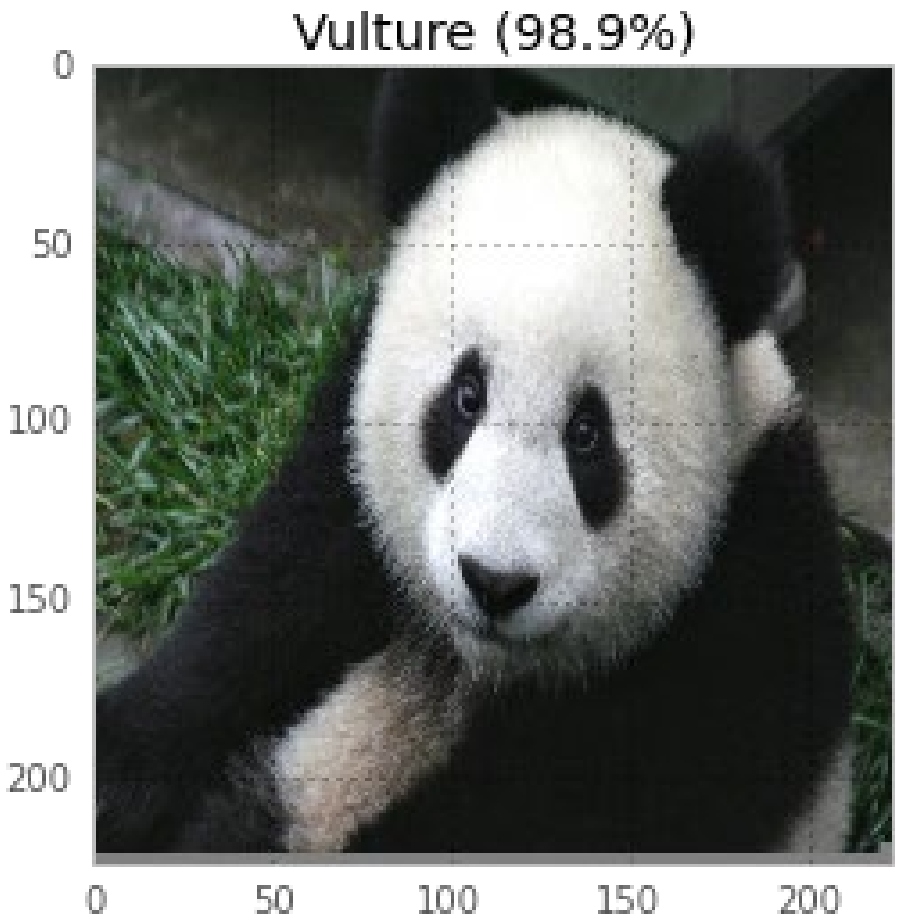
Neural Network Specific Adversarial Attacks

Adversarial

- We are usually nice to neural networks
- We feed it data like it has seen before and get back positive results
- Instead what if we are malicious and exploit how they work
- Main point (neural networks are basically very complicated functions which we can back solve and exploit)

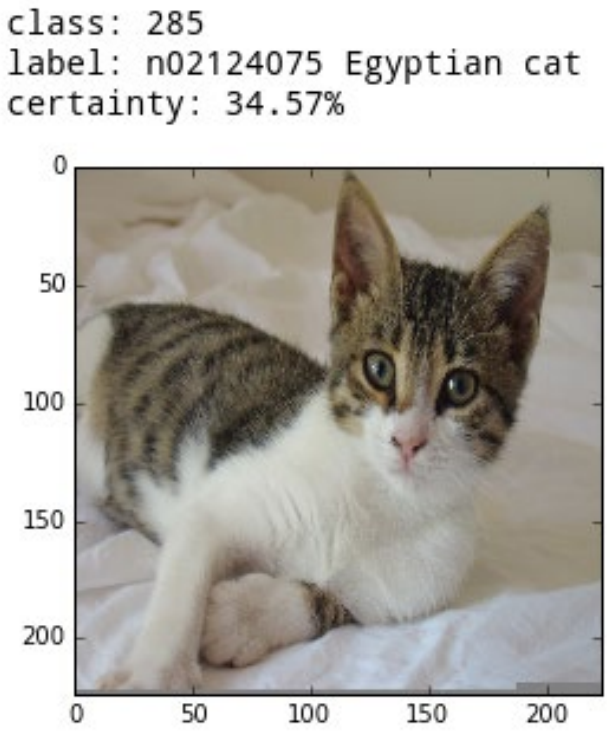
What!

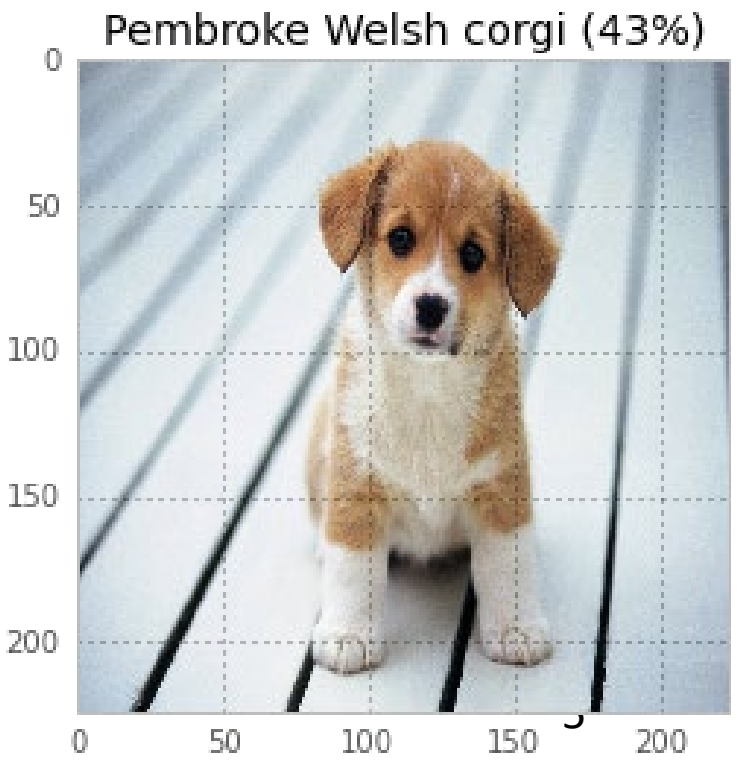




GoogLeNet

- These examples are GoogLeNet from the ImageNet competition
- So a large CNN, that should be (slightly) more robust against exploitation

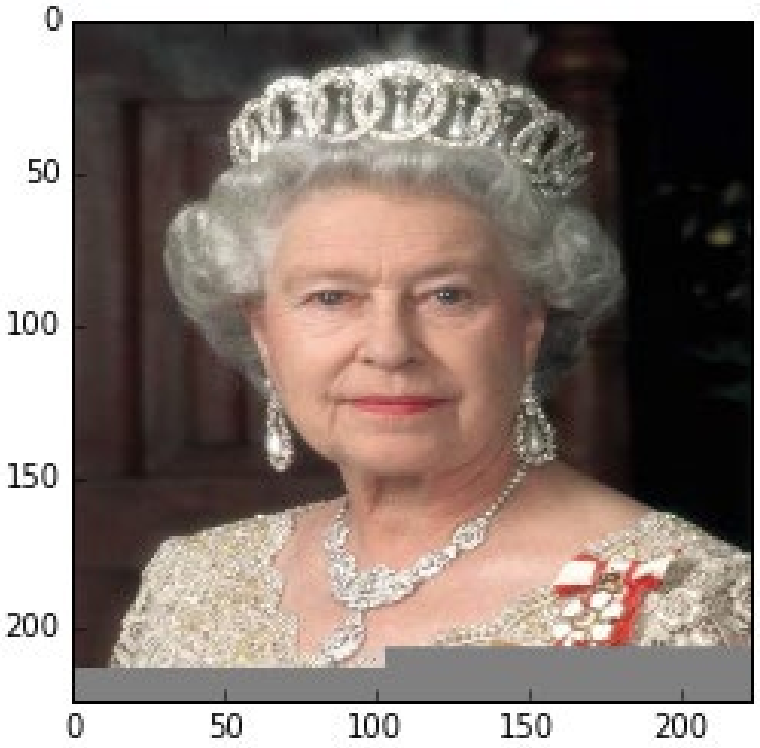




GoogLeNet

- Even good CNNs can suck without our help

```
class: 793
label: n04209133 shower cap
certainty: 99.7%
```

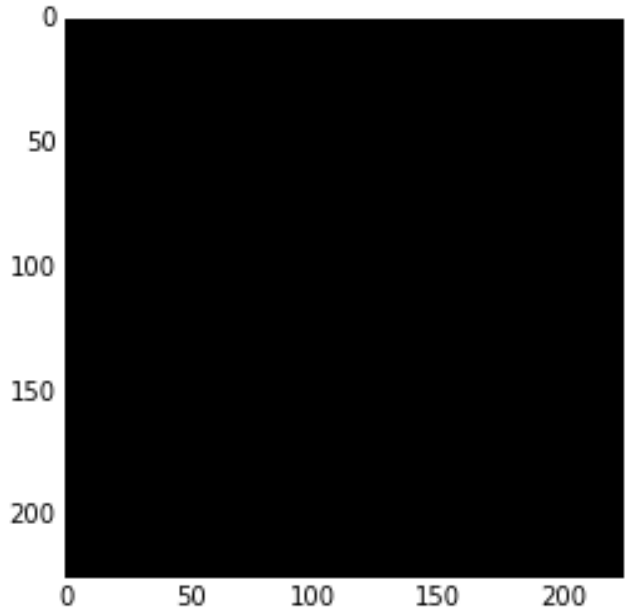


Let’s dig around inside

- Make all black input
- Look at labels
- Even non-data has classification
- We are going to play with gradients

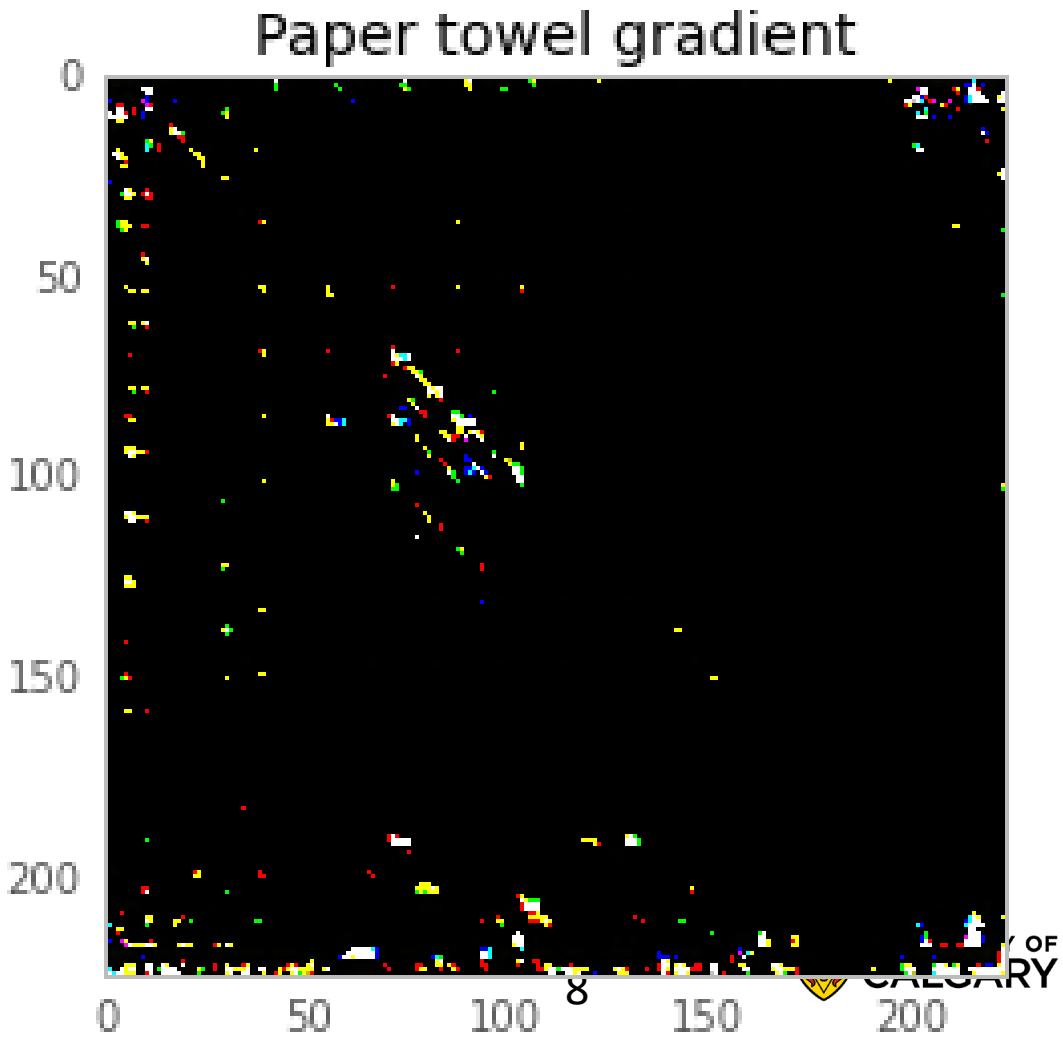
```
black = np.zeros_like(grad) * 255
_ = predict(black, n_preds=5)
```

```
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%
```



Reverse back-propogation

- Take paper towel as a label
- Set it to a full 1
- And back propagate the neurons

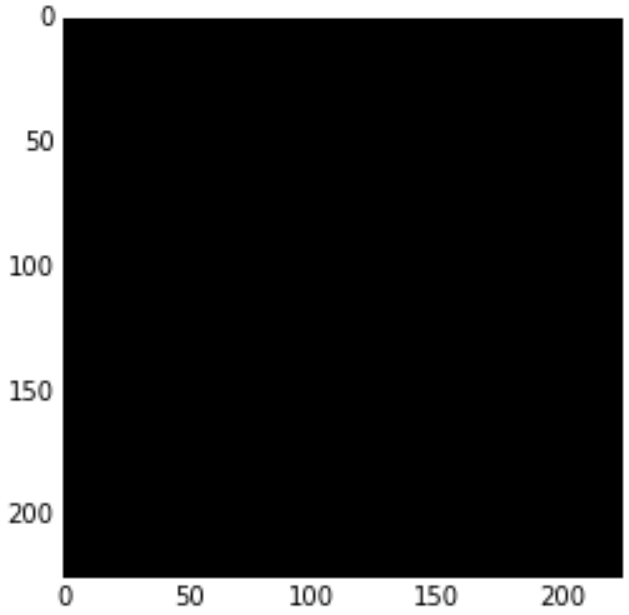


Reverse back-propogation

- We can see the garbage input ourselves
- So let’s drop the ratio to 1/256
- We went from 4.67 to 16.03 %
- On something that still looks black to us

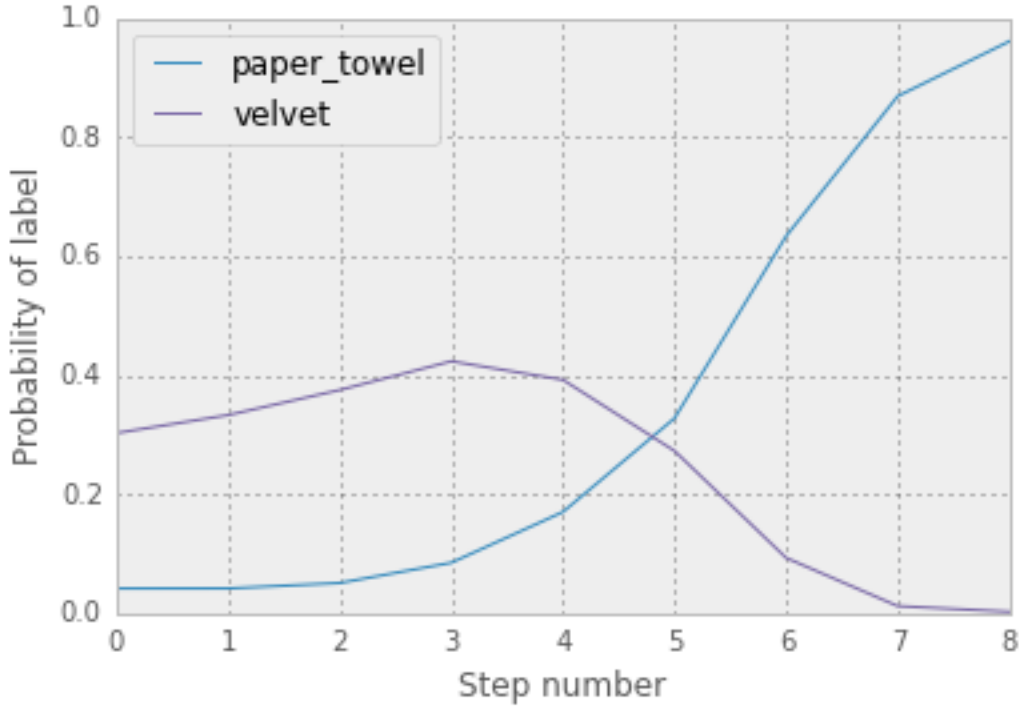
```
_ = predict(black + 0.9*delta, n_preds=5)
```

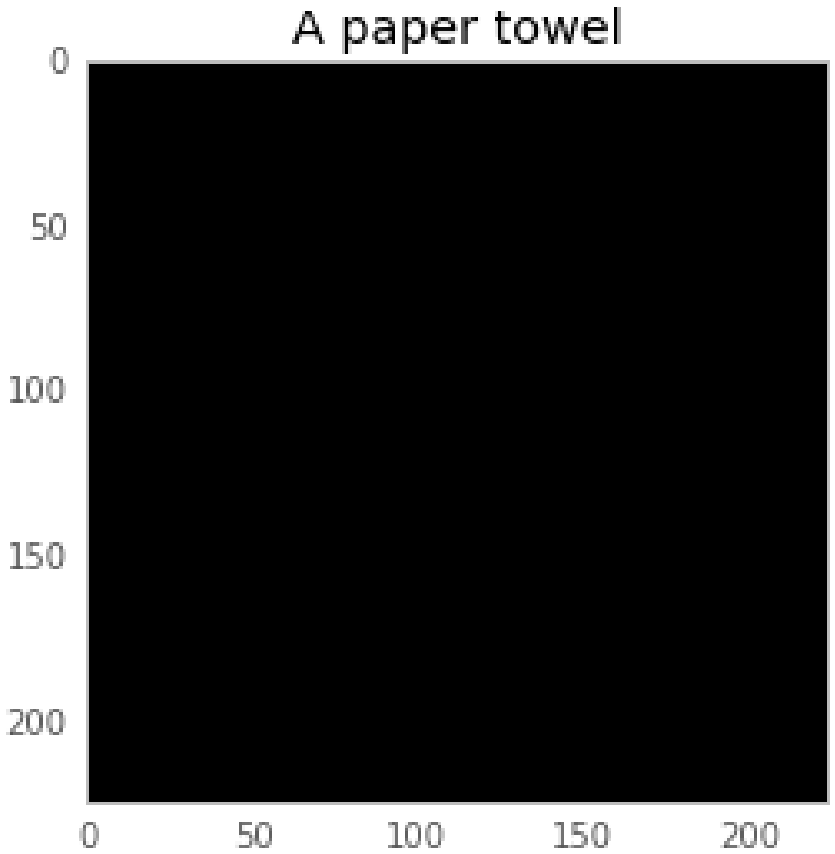
```
label: 885 (velvet), certainty: 54.75%
label: 700 (paper towel), certainty: 16.03%
label: 911 (wool, woolen), certainty: 12.4%
label: 533 (dishrag, dishcloth), certainty: 2.65%
label: 794 (shower curtain), certainty: 2.11%
```



Reverse back-propogation

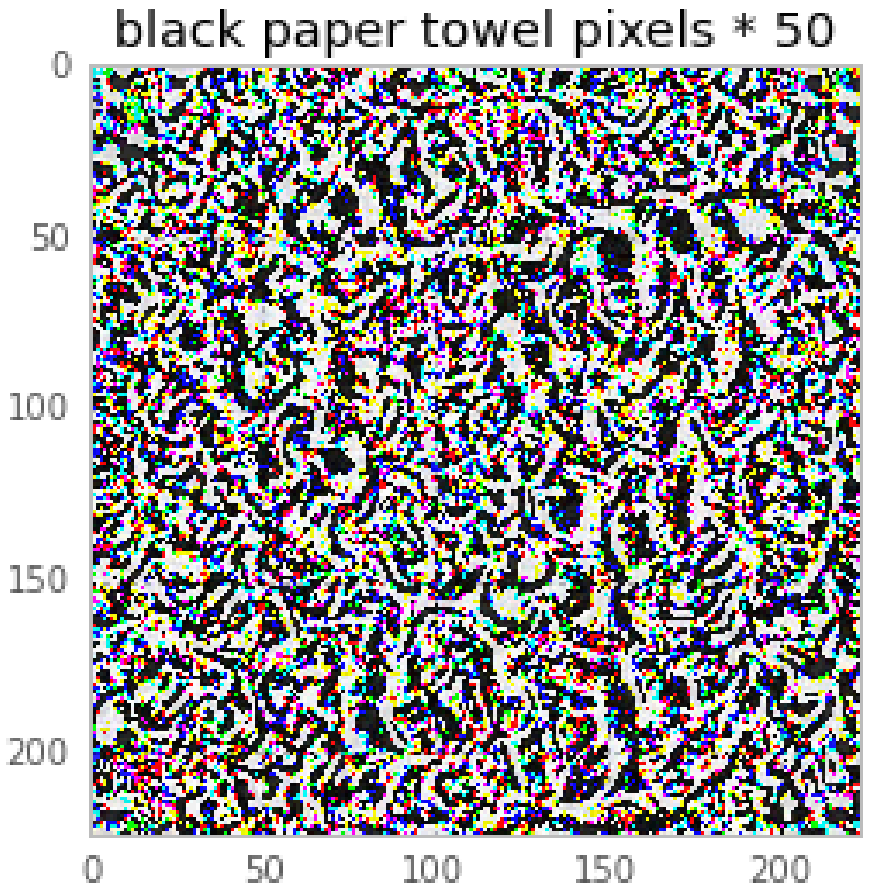
- Looping back propogation



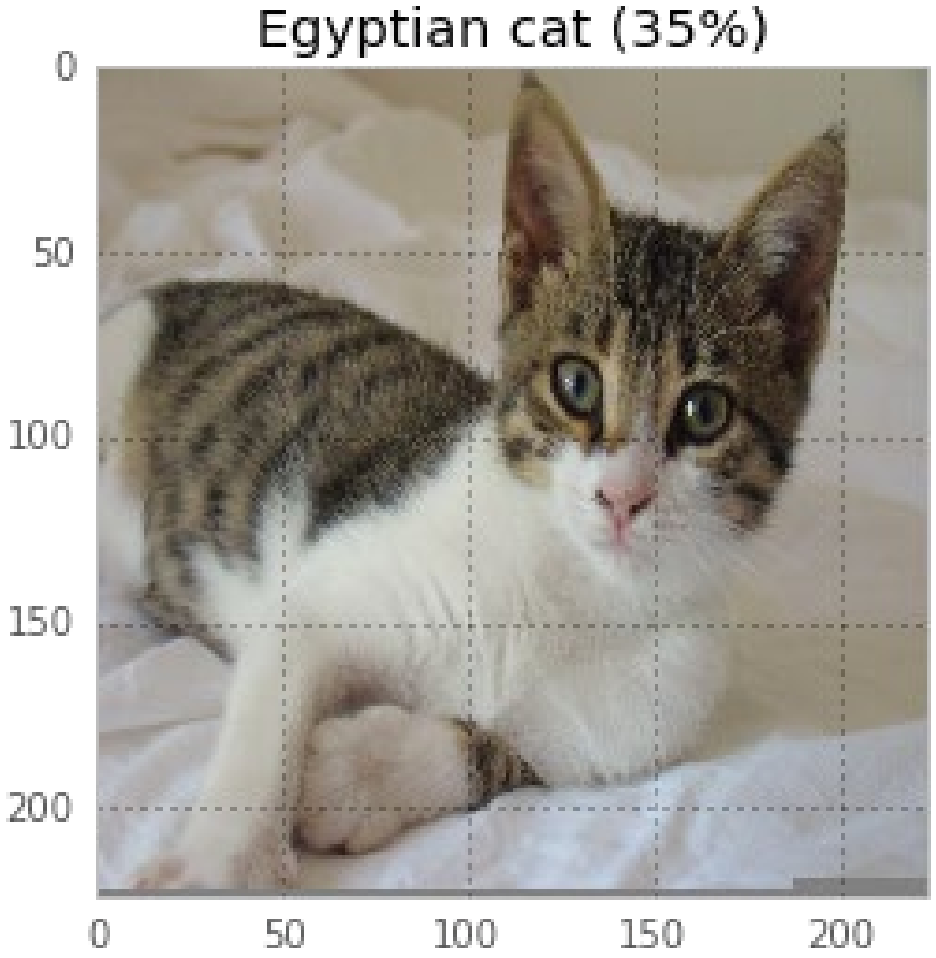


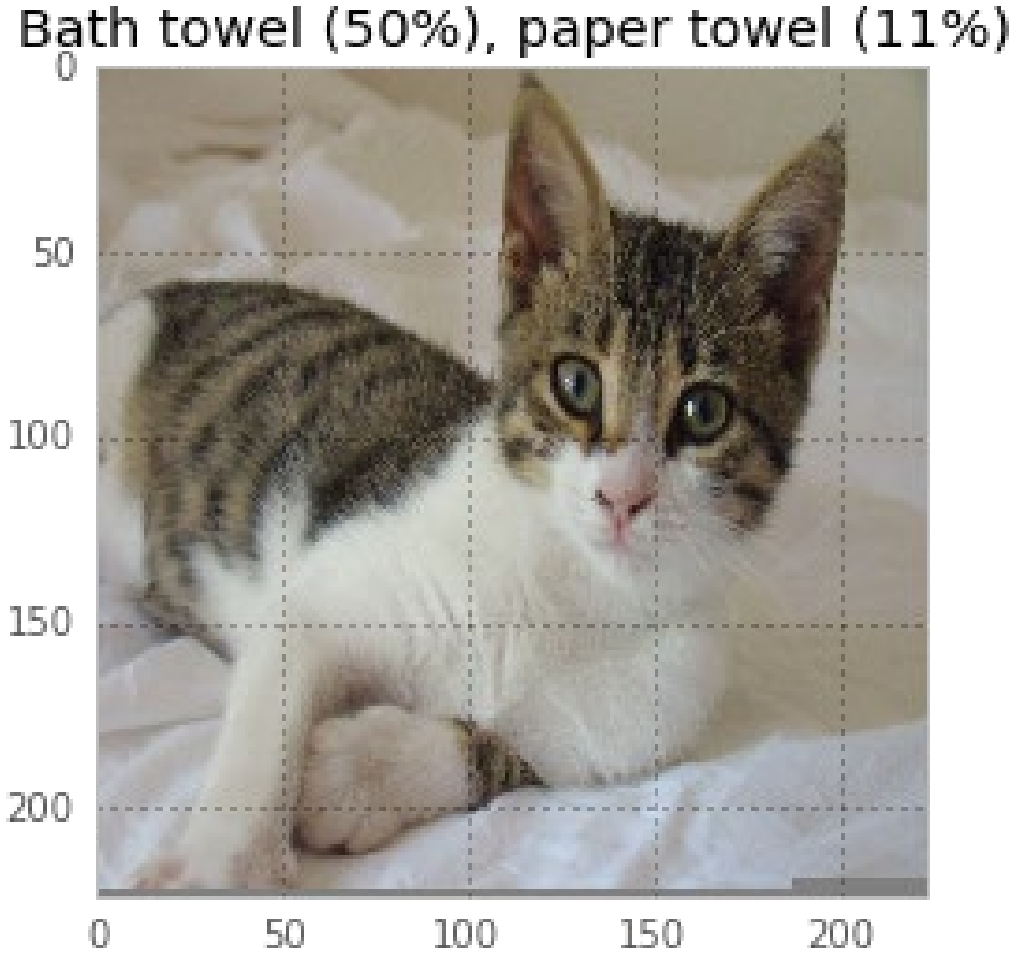
Reverse back-propogation

- Force the pixel values larger so we can see underlying structure

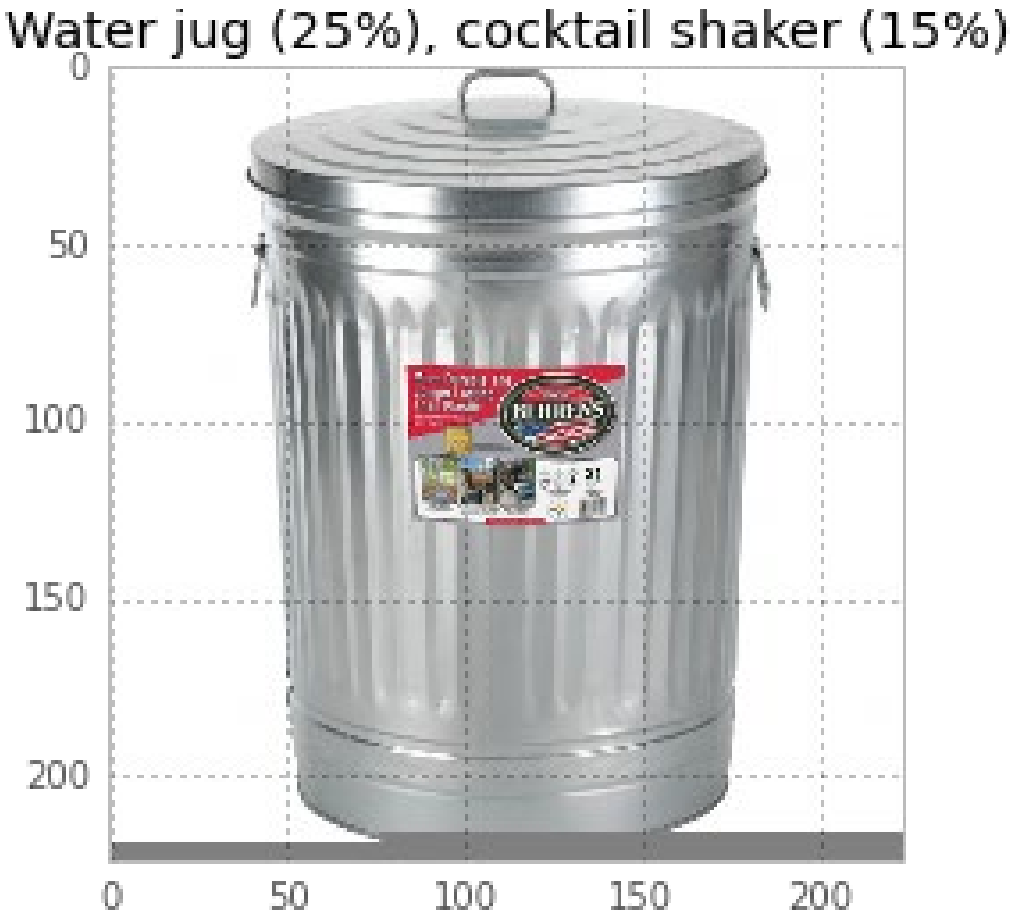


Now push this data over top of other images



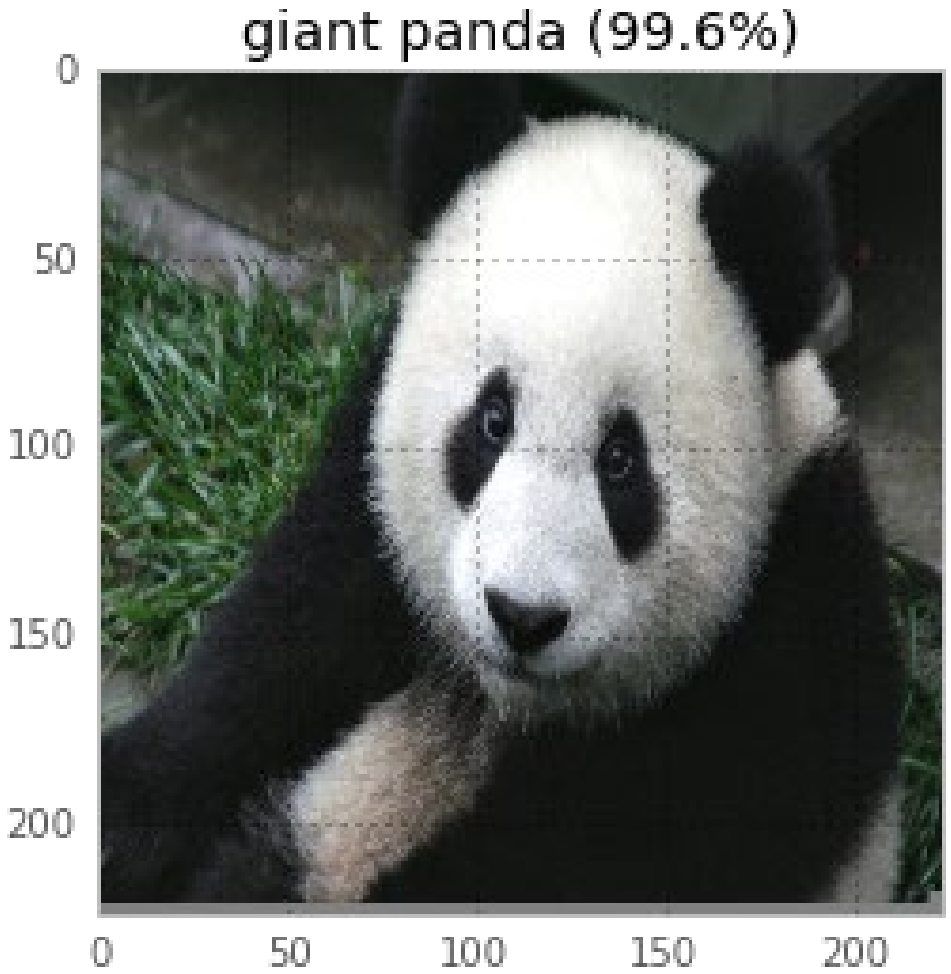


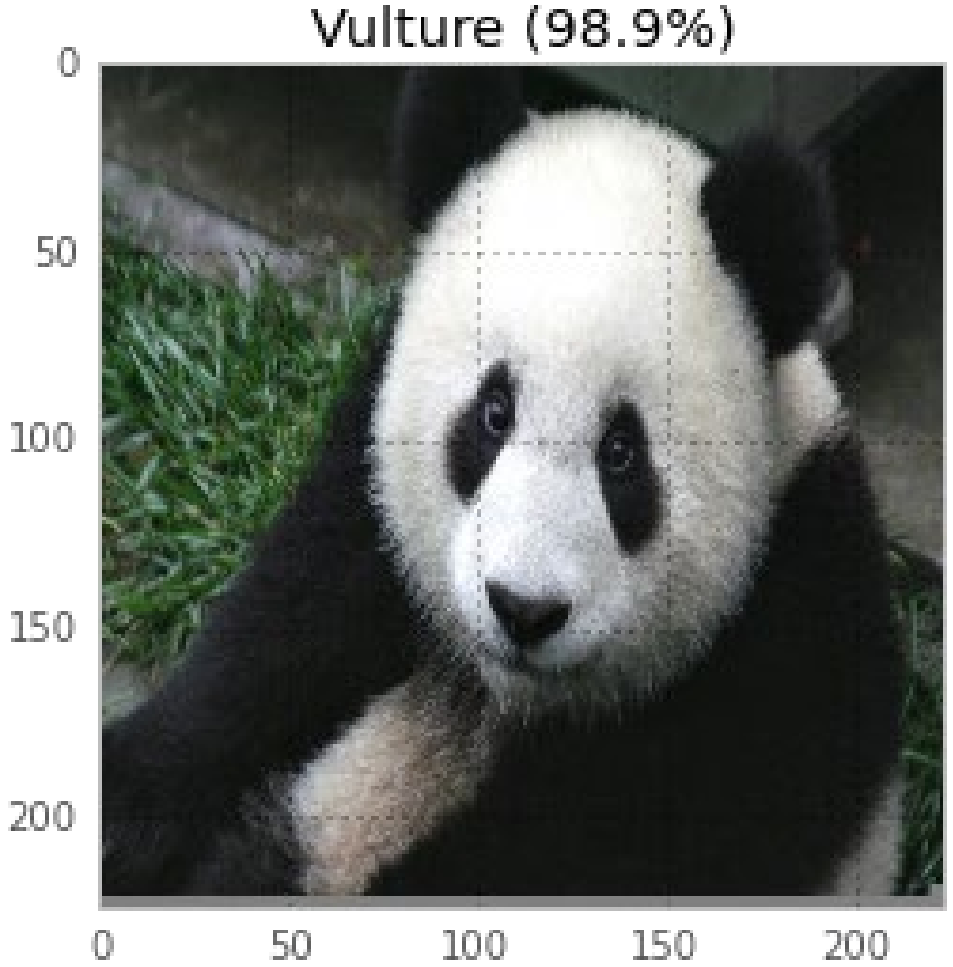
Now push this data over top of other images



13

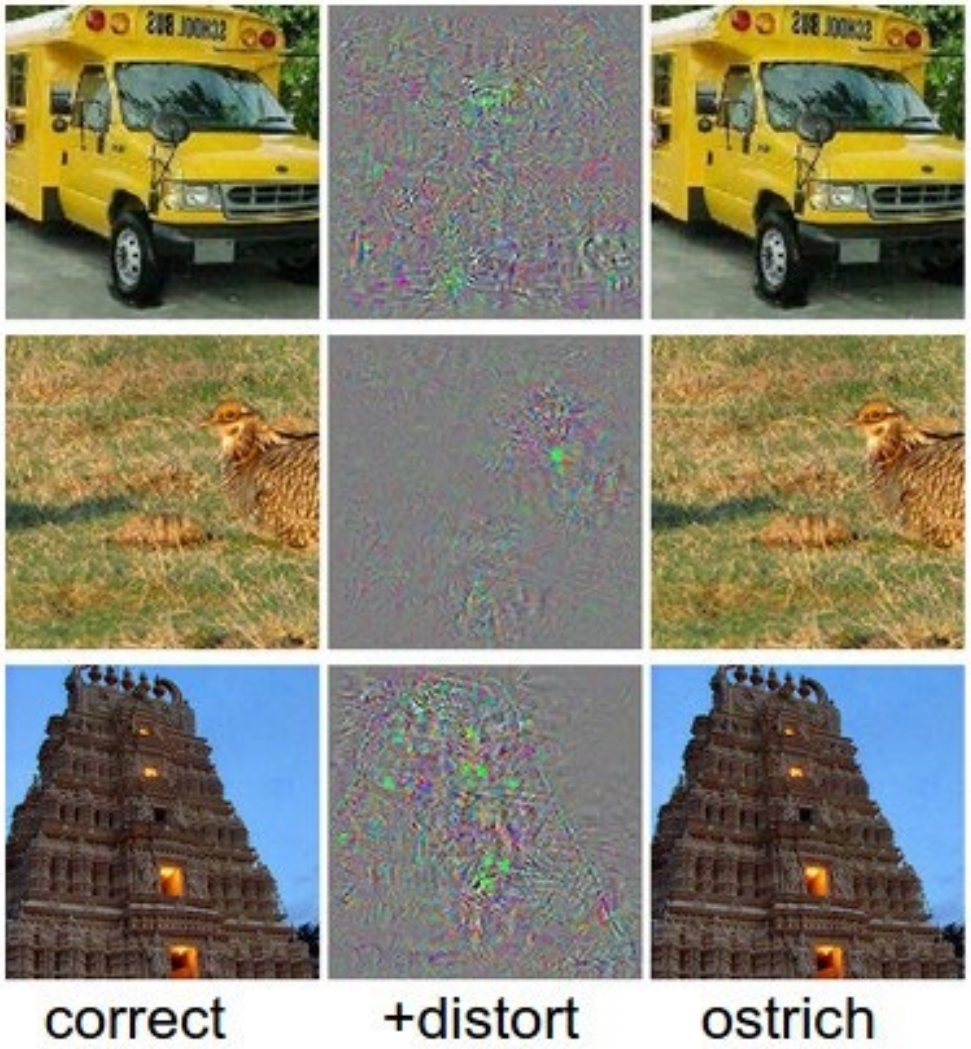
Now push this data over top of other images

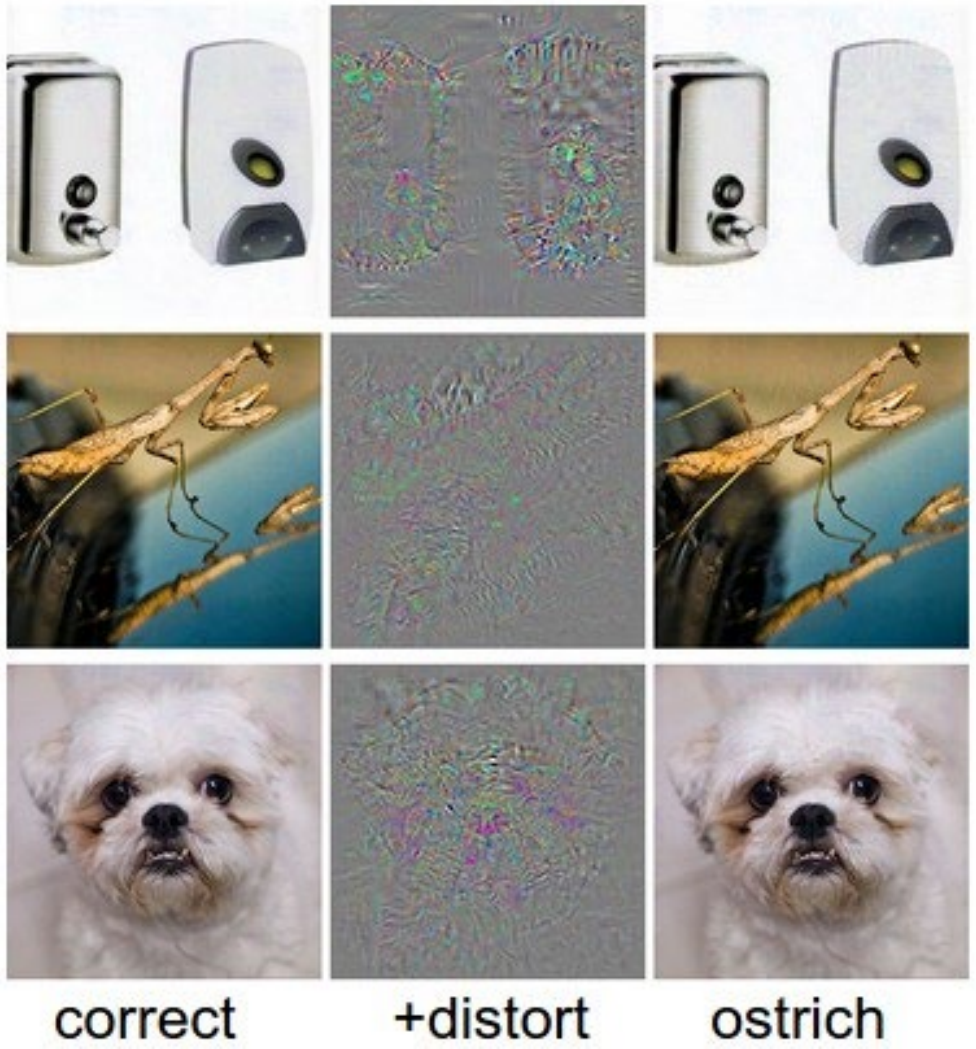




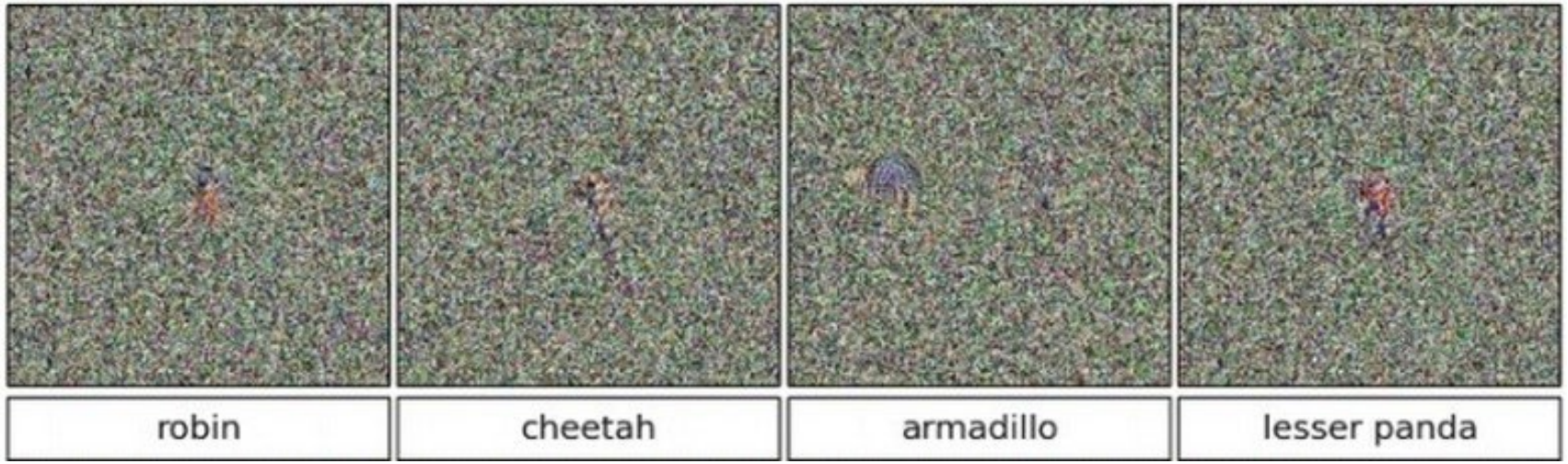
14

Now push this data over top of other images



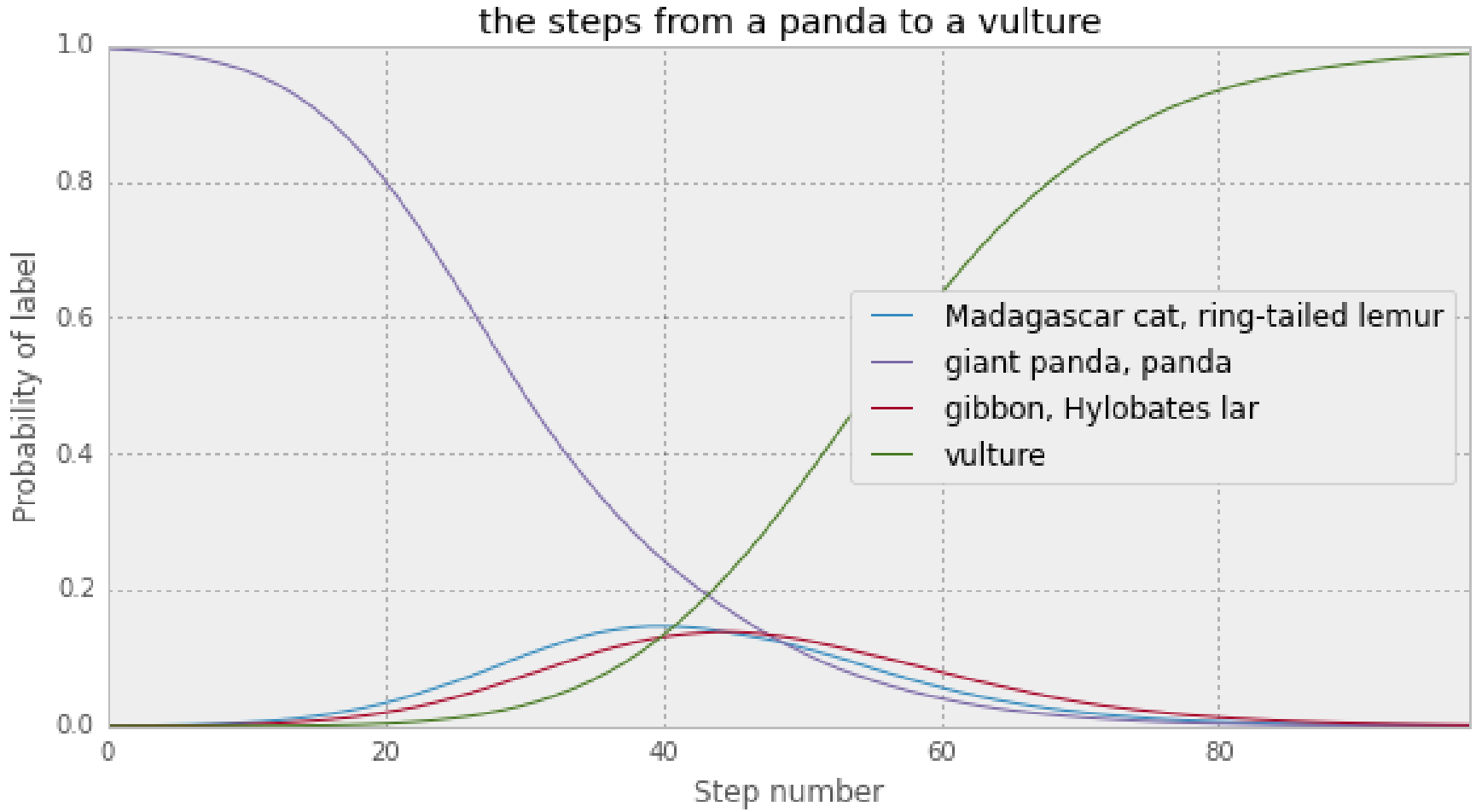


99.6% +



16

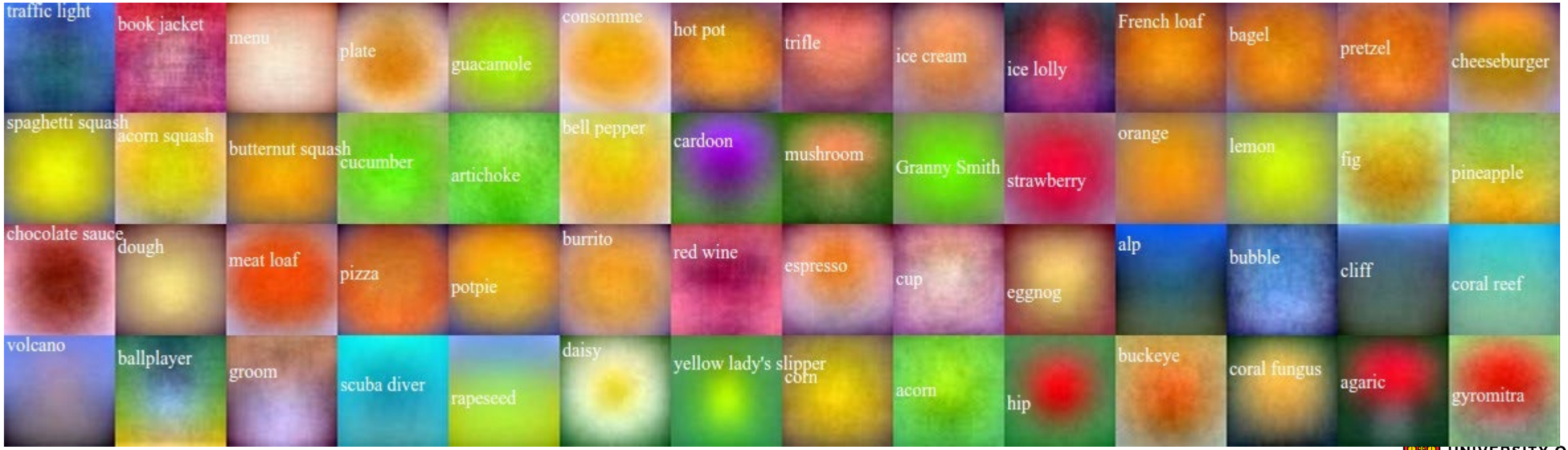
Now push this data over top of other images



17

Breaking Linear Regression

- Linear classifiers that takes every input pixel and maps to labels
- Take some food ones and back reverse to find image colours



18

Breaking Linear Regression

- For example, Granny Smith apples are green, so the linear classifier has positive weights in the green color channel and negative weights in blue and red channels, across all spatial positions. It is hence effectively counting the amount of green stuff in the middle.

- trick the Granny Smith classifier

1. figure out which pixels it cares about being green the most
2. tint those green
3. profit!

19

Neural Networks are Logistic Regressions

- We are basically training a large function
- Finding its weights
- A fundamental struggle will always be the exploitability of this exact back-relationship of input and output

- Doesn’t actually ‘think’ on an abstract conceptual level at this time
- We can find reverse engineer mistakes based on trivial signals

20

Take-away

- If we are aware of this issue we can make neural networks better
- The main techniques are essentially trying to break the ability of a neural network to make direct connections between input/output

- there is a struggle between models that are easy to train (e.g. models that use linear functions) and models that resist adversarial perturbations.
- CNN are quite good at expected images, but anything around edges they often are very indeterminate

21

Other AI Failures

AI is Easy to Mis-Use

- First: There is a non-ending list of these.
- As long as AI exists it will be used either naively, actively negligently, or maliciously to bad ends.
  - Facial Recognition: being declared illegal in numerous cities, numerous non-white lawmakers in US mis-identified as criminals
  - Neural Network hiring recommendations for video interviews: simply should be illegal
  - Resume screening: good at patterns, horrendous at unique
  - Legal sentencing AI recommendations: repeats social biases
  - ImageNet: embedded biases
    - MIT ‘80 Million Tiny Images’ had same issue
  - Microsoft Tay: under 24 hours twitter corrupted it
  - AI trained on copyrighted art to create ‘furry’ avatars for others (stealing?)
  - To name only a few

AI is Easy to Mis-Use

- Your responsibility is for honest use
- AI methods rely on bias
  - In fact many are just ways to learn bias
- It could be in your data you start with, or your methods on the data

- Naïve usage of AI likely to trend towards being ‘illegal’
  - Right of accuser to see your algorithm and data (been cases already)
  - Properly fit into existing laws (employment law, sentencing laws)
  - Or new laws (right to own data in EU, facial recognition rights)

AI is Easy to Mis-Use

1. Just because you ‘can’ do it, doesn’t mean you ‘should’ do it
2. Should be honest about limitations
   - As valuable as showing your NN is good at identifying X image 99% accurate, it is maybe more valuable to know it fails at Y image
   - Is a person tracking system really a good system if a person with darker skin isn’t identified?
3. Diversity is a key component.
   - Either domain experts that can tell social/economic/race/age/etc. biases in your data
   - Or minorities:
     - Minorities can represent data cases that don’t have enough for a pattern (too few)
     - Or those where your/algorithm assumptions are wrong

Onward to …
nothing.

Jonathan Hudson
jwhudson@ucalgary.ca
https://pages.cpsc.ucalgary.ca/~hudsonj/

UNIVERSITY OF CALGARY