Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images

1 Citation

Nguyen, Anh, Jason Yosinski, and Jeff Clune. "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

`https://arxiv.org/pdf/1412.1897.pdf`

2 Abstract

We show how to generate undecipherable (by humans) images that CNNs classify with high confidence. This shows how CNNs can be fooled.

3 Introduction

Using evolutionary algorithms or gradient descent, we can fool an MNIST (i.e. handwritten digit recognition dataset) CNNs and ImageNet CNNs to think that something that looks like noise belongs to some class.

4 Methods

For ImageNet, we consider the AlexNet model. For MNIST, we consider LeNet.

Evolutionary algorithms create organisms (images), score them with a fitness function (DNN score), applies crossover and random mutation to generate a new generation of organisms, and repeats.

We make an algorithm called MAP-elites which keeps one champion image for each category. On each iteration, we pick one image, mutate it, and replaces an existing champion if appropriate. To represent an image, we consider simply using the actual pixels and we mutate each pixel with probability 10% (we decrease this every 1000 generations) using the polynomial mutation operator - this is direct encoding. We also consider representing images with a compositional pattern-producing network (CPPN), which produces images that humans can also recognized (it is trained with a crowdsourced fitness function from humans).

5 Results

With 50 generations, we can produce white-noise like images that LeNet thinks are MNIST digits with probability 99.99%. We can also start with a digit and use the CPPN to achieve the same result (here

we get strange patterns that definitely don't resemble a digit). Direct encoding was less successful for ImageNet, only being able to generate something that fooled the network for 45 of the 1000 classes. 5000 generations of the CPPN fools AlexNet so that it gets the wrong class (with median confidence 81%). These images have some aspects of their "fooling" class (e.g. blue ocean color for starfish class, stitching pattern for baseball class). Re-running the algorithm comes up with different ways to fool the DNN. Many fooling images had repeated patterns, indicating that DNNs learn low/mid-level features and don't care where they appear in the image. Repeating the spatial patterns increases the DNN confidence (covering some of them up hurts confidence). It's harder to generate fake cat and dog images because ImageNet has a ton of cat and dog images, so AlexNet has learned those classes well.

Do images that fool one DNN fool another? We trained two AlexNets (each with different random initializations). Some images (not all) could fool both of them.

To try and help our DNNs avoid getting fooled. We added a "fooling class". We trained on images on the original classes for several iterations. Then we generated a bunch of fooling images and threw those in the training set as well. We repeated this process for a while. After doing this, it was still pretty easy for our evolutionary algorithm to find images to fool LeNet. This technique worked better for ImageNet - when the network was fooled, it had a median confidence of 12% instead of 81%.

In addition to evolutionary algorithms, we also tried creating fooling images with gradient descent. Basically, we hold the weights fixed use backpropagation to adjust the input image pixels to maximize some incorrect class's probability in the softmax output. This works pretty well for both ImageNet and MNIST.

6 Discussion

Evolutionary algorithms produce a great diversity of fooling images, but mostly unrecognizable images. We think this is because DNNs are discriminative models, so synthetic images simply need to lie in the right region of space, rather than being something that is a natural image. A generative model might be able to avoid this problem, but generative models are hard to scale to datasets the size of ImageNet. We submitted our fake images to an art contest, along with a sample image from the fooling class, and they got accepted. It might be interesting to see how you can make a DNN creative using the kind of approaches described here.

Malicious actors might leverage this property of DNNs and generate images to fool DNNs.

7 Conclusion

We presented two evolutionary algorithms and gradient descent to generate fooling images for DNNs.

The supplementary materials in this paper discuss each of the topics above in a little more depth.