

Using CNN's To Detect Mathematical Properties of Strange Attractor Images

Evan Penn
Stanford University
epenn@stanford.edu

Abstract

Strange attractors are subsets of the phase space of a dynamical system. When a strange attractor is present, the state of the system will evolve towards positions within the attractor. However, unlike simple attractors such as limit cycles, the behavior of the system remains unpredictable even after it has entered this pseudo-steady-state. Depictions of these attracting sets have been found to have intrinsic aesthetic value. In fact, research has suggested that human taste naturally converges to depictions of attractors with specific mathematical properties, even though it is difficult to estimate these properties by simple inspection. Convolutional Neural Networks (CNN's) have shown themselves to be capable of almost human level visual classification. Therefore, I explore their ability to learn to judge this subtle property.

1. Introduction

Visualizations of fractal sets have consistently exerted an aesthetic fascination. In contrast to other mathematical visualizations, these have value beyond the ad hoc use of mathematical researchers attempting to understand a technical problem. They constitute a unique category of images, being neither photographs (landscapes, movie stills, x-rays, telescope images, etc.) nor illustrations (diagrams, paintings, animations, etc). Yet, many images from both of these categories have measurable fractal characteristics, including the most famous such property, scale-invariance.

The images used here are called "strange attractors". Their generation and measurement closely follow [1]. The images show a trajectory that is calculated by iterative application of a quadratic equation, that is if $x_n = a$, then $x_{n+1} = f(a)$, $x_{n+2} = f(f(a))$ and so on. There are three basic behaviors that can be observed:

1. The state flies off into infinity
2. The state approaches a limiting orbit
3. Chaotic behavior develops

These three are distinguished in technical contexts by the Lyapunov exponent, a measure of how quickly states with similar starting positions will diverge. A very fast divergence implies that information is being lost quickly because the initial state quickly becomes very difficult to recover. It is possible to measure this quantity empirically by keeping track of nearby trajectories and their divergence. This is one of the two quantities that are predictive of human preference, the other being the dimension.

The fractal dimension manifests the idea that fractal sets have zero measure at some dimension, but infinite measure at the dimension below. For example, a space filling curve may have infinite length (dimension 1), but zero area (dimension 2). A common computational approach to measuring dimension is the correlation dimension [3], which measures the falloff of neighbor points at small scale. Intuitively, neighborhoods must shrink faster in higher dimensional space.

Convolutional Neural Networks have recently gained prominence as a state-of-the-art technique for image classification. Among image classification techniques, CNN's (as well as neural nets generally) are unique in learning a hierarchical representation that starts from raw data, obviating the need to engineer features prior to applying the modeling technique. This makes neural nets particularly suited to image sets that are unusual in some way and thus possibly resistant to summarization via traditional image features developed to suit the needs of more typical image modeling tasks. In this case the images are graphs of chaotic functions, rich in graceful motifs, but rather disparate from everyday images.

2. Background

Convolutional Neural Networks have come to prominence in recent years due to their success in many domains, and particularly image classification. Convolutional networks learn a hierarchy of representations via back propagation. More specifically, they attempt to learn sets of image filters or kernels that are then used to process the images into sets of representations that are fed to multilayer perceptrons [6]. A key breakthrough in classification came

from the ImageNet project [5]. This architecture consists of convolutional layers that are made nonlinear via the ReLu activation function and subject to 2x2 max pooling. Interestingly, the lower level representations learned by this network are so good that they have been successfully transferred to other tasks (see the Flickr Finetuning Example in [4]).

Transfer learning of CNNs or 'Finetuning' is a technique that brings the benefits of very large scale training to smaller problems. The method works by retraining the last several layers while keeping the initial layers constant or nearly so. In effect, this is similar to building a multilayer perceptron on the features learned from the larger dataset. The difficulty of this approach lies in judging how many layers to retrain, and at what rates. The most obvious determinant of this is the amount of available data, but as with all non-convex optimization tasks, it is not predictable *a priori*.

Clinton Sprott was an early proponent of using illustrations of dynamical systems for artistic purposes [7]. Motivated by the prospect of producing beautiful art in a computational manner, he aimed to understand what motivated human aesthetic preference [1]. According to his data, two common metrics were indeed predictive of preference, fractal dimension and Lyapunov exponent. The dimension is a measure of how space-filling the curve is, and is not too hard to estimate with a quick inspection. The Lyapunov exponent is more subtle, and measures how chaotic a process is, understood in the sense of quickly becoming unpredictable. Sprott speculates that the range of Lyapunov exponents that pleases humans corresponds to range that would be observed naturally.

3. Approach

Generating attractor drawings is accomplished randomly choosing a set of exponents for the quadratic dynamical system.

$$x_{n+1} = ax_n^2 + bx_ny_n + cy_n^2 + dx_n + ey_n + f \quad (1)$$

After running the system for 300k iterations, a burn-in period is discarded, and approximations of the correlation dimension and Lyapunov exponents are found computationally. If the Lyapunov exponent is positive, an image is generated and its metrics are saved as metadata. For this task, cython was used. On a i7 machine, an image could be generated every 5 seconds on average. The figures depict 9 examples of the images, arranged to represent the three binned categories.

The CNN that was fine-tuned is known as ImageNet [5], originally trained to recognize 1000 categories. The three major variables investigated were the number of layers to expose to retraining, the objective variable, and the number of bins for the objective. Initially, the experiment was

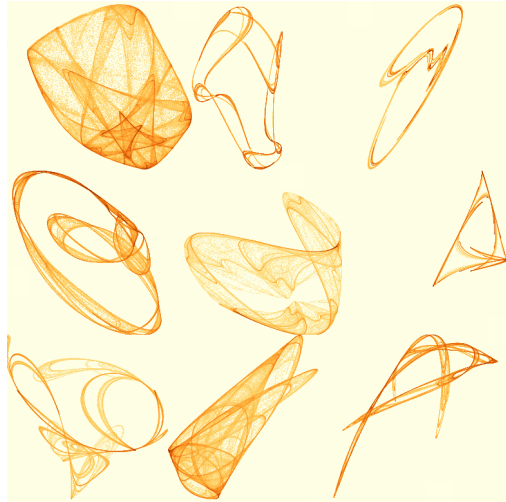


Figure 1. each row shows typical examples from the three different Lyapunov bins, with the smaller (less chaotic) on top

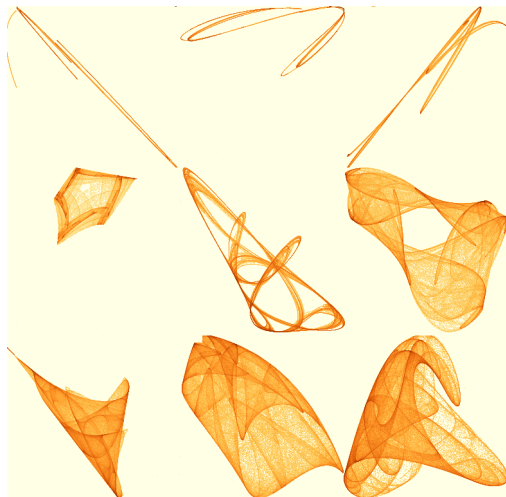


Figure 2. each row shows three typical examples from the three different dimension bins, with the smaller (closer to 1.0 which is a line) on top

run with 20 bins, yielding accuracies around 20%, this was later changed to 3 bins, which gave much better performance. The training was done on EC2 instances through a python script that programatically modified the caffe-included Flickr fine-tuning example [4]. In a further evolution of this project, this infrastructure would allow for a queuing system for cross validation. In the future, this would also be an important feature for integrating with any automated hyper parameter optimizer, such as that in [?].

4. Experiment

The experiment was carried out using the Caffe library. The generated dataset has 30,000 training images and 3,000 test images. Generally, tests were run for 100,000 iterations,

Setting	Test set accuracy	Iterations
Lyapunov exponent (20 bins, single layer fine-tune)	.0904	10,000
Correlation dimension (20 bins, single layer fine-tune)	.2303	15,000
Lyapunov exponent (3 bins, single layer fine-tune)	.8007	100,000
Lyapunov exponent (3 bins, 2 layers fine-tune)	.8068	100,000
Lyapunov exponent (3 bins, 3 layers fine-tune)	.796	100,000
Correlation dimension (3 bins, 2 layers fine-tune)	.871	100,000

but were halted early when progress was evidently stalled. It has been shown elsewhere that early curtailment is a viable strategy [2]. On an amazon GPU EC2 instance, each run took about 6 hours.

The results show that the dimension was much easier to classify than the Lyapunov exponent, as expected. Its possible that the dimension can be guessed simply from the number of illuminated pixels. However, on the 20 bins version, the net still stalled at about 20%. The situation was even worse for the Lyapunov model, which couldn't even break 10%. The range of values for dimension is 1.0 to 2.0, and for Lyapunov it is 0.0 to 1.8. A 20 bin setting amounts to very subtle differences, possibly comparable to margin of error of the empirical measurement techniques. In the future, perhaps some compromise can be found.

There was not an appreciable gain from allowing more layers to be fine-tuned. In fact, further experiments showed that allowing even more layers to learn caused results to decay precipitously. Other experiments with increased learning rates had a similar effect. The overall impression was the fine-tuning approach was able to produce decent results quickly, but was easily pushed into areas of very poor performance. It seems there may be a ceiling to this approach, though its important to keep in mind that this signal was not a terribly strong one and that the natural images that the net was originally trained on differ very substantially from those used here.

5. Conclusion

The results of this investigation show promise, but are not fully conclusive. The three bin setting seems too easy, though in the Lyapunov case human performance may not be a lot better. It was also the case the test performance did not appreciably improve after the first 40,000 iterations, indicating that mere overfitting was kicking in. To prevent this, much more data should be generated.

This research is still in its early stages, though there is clearly a signal being found. Going forward, I think the most important thing is to understand how this dataset is morphing the ImageNet weights. The natural way to do this would be to explore embeddings of the activations of the penultimate fully connected layer, with 4096 units, on these images. Only time constraints and software frustrations kept me from this.

A possible application of this model is for analyzing chaotic time series. The theory of dynamical systems shows that important invariants, such as dimension and Lyapunov exponent are preserved by embeddings under weak conditions. This means that a graph of such a time series could be analyzed 'visually' by a CNN to estimate a measurement. It would be interesting to attempt a more direct deep learning model of dynamical systems with an input layer that was specifically shaped like the dynamical input (2 dimensional in this case). A recurrent network might be a natural choice.

Another possible application would be to use a network trained in this way to examine pictures of naturally occurring fractal sets, such as clouds or plants. Could a CNN trained on these generated images be better able to distinguish, for example, plant species with subtle differences in growth pattern? Moreover, what gains, if any, could be expected by combining a generated set of images such as this with a natural image dataset. Would the training results be different? Would different filters be learned?

References

- [1] D. J. Aks and J. C. Sprott. Quantifying aesthetic preference for chaotic patterns. *Empirical studies of the arts*, 14(1):1–16, 1996.
- [2] T. Domhan, J. T. Springenberg, and F. Hutter. Extrapolating learning curves of deep neural networks.
- [3] P. Grassberger. Generalized dimensions of strange attractors. *Physics Letters A*, 97(6):227–230, 1983.
- [4] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.
- [5] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.
- [6] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [7] J. C. Sprott. *Strange attractors: Creating patterns in chaos*. Citeseer, 1993.