Paper Summary BACON

Abstract

BACON: Band-limited Coordinate Networks for Multiscale Scene Representation. Coordinate-based network are trained to map continuous input coordinate to the value of a signal at each point. **Training at a single scale** will result in artifacts when naive downsampling and upsampling. BACON can be designed based on the **spectral characteristic of the represented signal** at unsupervised signal. Demonstrate BACON for

1. Multiscale neural representation of images.
2. radiance fields.
3. 3D scenes using **signed distance functions**

1. Introduction

Neural representations approximate signals using a continuous function that is embedded in the learned weights of a fully-connected NN. Since it is designed to represent signals at a single scale, the behavior of the NN at unsupervised coordinate is difficult to predict.

The **key properties** of BACON architecture are:

1. the maximum frequency at each layer can be manipulated analytically.
2. The behavior of a trained network is entirely characterized by its **Fourier spectrum**.

Contribution:

1. Introducing BACON for representing and optimizing
2. Developing methods for spectral analysis of the architecture and proposing initialization scheme

BACON is suited to **multiscale signal representation** because band-limited output layers can be designed with an inductive bias towards a particular resolution or scale.

2. Related Work

Architectures for Scene Representation

NN architectures for scene representation networks can be classified as **feature-based**, **coordinate-based**, and **hybrid**.

Feature-based: quickly evaluated, but have a large memory footprint.

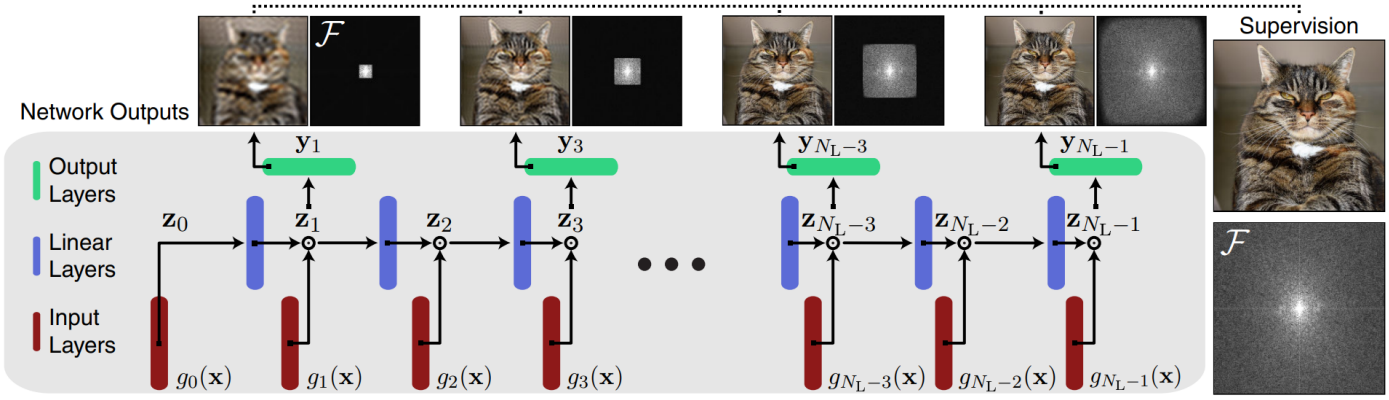
Coordinate-based: Map from an input coordinate to a signal value, and the proposed one in this paper is coordinate-based. Didn't use MLP but **multiplicative filter networks**(MFNs, recently proposed).

3. Method

3.1 Band-limited Coordinate Networks

MLP employ a Hadamard product b/t linear layers and sine activation functions. Extending the theoretical understanding and practicality of MFN by:

1. Architecture change to achieve multi-scale, band-limited outputs
2. Deriving formulas to quantify the expected frequencies in the representation
3. Deriving initialization scheme preventing vanishing activations in deep networks



$x \sim U(-0.5, 0.5)$

$g_0(x) = \sin(w_0 x + \phi_0)$

$z_0 = g_0(x)$

$g_1(x) = \sin(w_1 x + \phi_1)$

$z_1 = g_1(x) \circ (W_1 z_0 + b_1)$ linear layer

$y_1 = W_1^{out} z_1 + b_1^{out}$

$g_2(x) = \sin(w_2 x + \phi_2)$

$z_2 = g_2(x) \circ (W_2 z_1 + b_2)$

$y_2 = W_2^{out} z_2 + b_2^{out}$

All these intermediate steps can be transferred to two single expressions:

$$y_k = \sum_{j=0}^{N_{sine}^{(k)}-1} \bar{\alpha}_j \sin(\bar{\omega}_j x + \bar{\phi}_j)$$

where $N_{sine}^{(k)} =:$

$$N_{sine}^{(k)} = \sum_{i=0}^{k-1} 2^i d_h^{i+1}$$

3.2 Frequency Spectrum

Designing the architecture so that the frequency of all represented sines never exceeds a desired threshold. Setting ω_i's to a bandwidth in $[-B_i, B_i]$ using random uniform initialization.

Introducing linear layers at intermediate stages throughout the network to extract band-limited outputs.

Because the outputs are band-limited, BACON can be trained in a semi-supervised fashion where the bandwidth of the **supervisory signal** need **not match** the **desired bandwidth** of the output of the network.

$*$ Products of sine result in summed frequencies. Why do we need this? Recall:

$$\begin{aligned} sin(a)sin(b) &= \frac{1}{2}(sin(a+b-\frac{\pi}{2}) + sin(a-b+\frac{\pi}{2})) \\ &= \frac{1}{2}(cos(a-b) - cos(a+b)) \end{aligned}$$

suppose $a, b \in [-\pi, \pi] \Rightarrow a-b, a+b \in [-2\pi, 2\pi] \Rightarrow$ summed freq

$$y_1 = W_1^{out} z_1 + b_1^{out}$$

$$z_1 = g_1(x) \circ (W_2 z_0 + b_1)$$

$$= \sin(W_1 x + \phi_1) \circ (W_2 \sin(W_0 x + \phi_0) + b_1)$$

$$\in d^h \qquad \in d^h$$

$$= \sin(\ )\sin(\ ) \in d^h$$

So suppose the bandwidth of W_0 is $[-B_0, B_0]$, and W_1 is also $[-B_0, B_0]$, then bandwidth of z_1 is $[-2B_0, 2B_0]$

3.3 Initialization Scheme

Assume the input to the network is uniformly distributed $x \sim U(-0.5, 0.5)$ with $\omega_i \sim U(-B_i, B_i)$ and $\phi_i \sim U(-\pi, \pi)$. Then, $\omega_i x + \phi_i$ is distributed uniformly as:

$$\begin{cases} 1/B_i \; log(B_i / min(|2x|, B_i)), & -B/2 \le x \le B/2 \\ 0 & \text{else} \end{cases}$$

$g_0(x)$ will be distributed uniformly, and $g_i(x)$ will be approximately arcsine distributed with variance of 0.5. Now let $W_i \sim U[-\sqrt{6/d_h}, \sqrt{6/d_h}]$. Then we have that $W_1 g_0(x) + b_1$ converges to std normal distribution with increasing d_h. Finally the Hadamard product $g_1(x) \circ (W_1 z_0 + b_1)$ is the product of **arcsine**

distributed and std normal random variable, which again has a variance of 0.5. Applying the next linear layers results in another standard normal distribution, which is also the case after all subsequent linear layers.

4. Experiments for **Neural Radiance Field** (4.2)

Adaption for BACON:

1. Setting the maximum bandwidth B to 64 cycles per unit interval allows fitting high frequency image detail.

2. Evaluate all methods without the viewing direction input originally used for NeRF => the input to all networks is a **3D coordinate corresponding to the position along the ray** $r(t)$.

3. BACON produces 4 outputs using B_i to constrain each output to $1/8$, $1/4$, $1/2$, and full resolution.

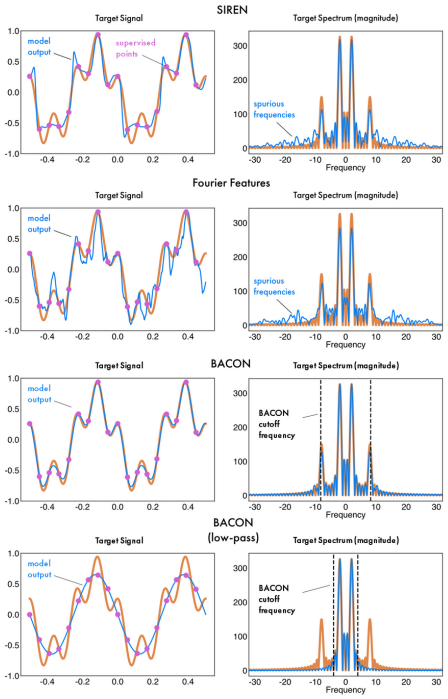
4. Hierarchical Sampling procedure of NeRF: the alpha compositing weights ω_j from an initial forward pass are used to resample the ray in regions of non-zero-opacity. To improve efficiency, using the lowest-resolution output of the network for this initial forward pass.

In BACON, y_i is actually the output which we can treat it as $C(r)$ in NeRF.

Question: The behavior of the network at unsupervised coordinates is difficult to predict?

A: They do not build consistent relationship between target signal and model output. BACON build it by **Fourier spectrum**. It is actually entirely characterized by Fourier spectrum.

1D Fitting Example



From Figure, we can find SIREN and Fourier Features do not work well with low frequency in Fourier

spectrum. For high frequency part, which most represent detail part in reconstruction, they both produce spurious frequency. Therefore, we can expect they won't work well with detail representation.

Question: How to predict behavior at unsupervised point?

A: Using the consistency b/t original (RGB) domain and Fourier domain. After training by Fourier spectrum, we can use inverse Fourier transform to recover the Fourier signal to original domain.

Recall **3.2 Frequency Spectrum**, to this end, freezing the frequency ω_i, and set them to a bandwidth in $[-B_i, B_i]$ using random uniform initialization. The total bandwidth of an output layer i of the network is less than $\sum_{j=0}^{i} B_j$, and the maximum bandwidth is $B = \sum_{i=0}^{N_L-1} B_i$