GENERATIVE SIGNAL PROCESSING THROUGH MULTILAYER MULTISCALE WAVELET MODELS

By

Jieqian He

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computational Mathematics, Science and Engineering – Doctor of Philosophy
Statistics – Dual Major

2021

ABSTRACT

GENERATIVE SIGNAL PROCESSING THROUGH MULTILAYER MULTISCALE WAVELET MODELS

By

Jieqian He

Wavelet analysis and deep learning are two popular fields for signal processing. The scattering transform from wavelet analysis is a recently proposed mathematical model for convolution neural networks. Signals with repeated patterns can be analyzed using the statistics from such models. Specifically, signals from certain classes can be recovered from related statistics. We first focus on recovering 1D deterministic dirac signals from multiscale statistics. We prove a dirac signal can be recovered from multiscale statistics up to a translation and reflection. Then we switch to a stochastic version, modeled using Poisson point processes, and prove wavelet statistics at small scales capture the intensity parameter of Poisson point processes. We also design a scattering generative adversarial network (GAN) to generate new Poisson point samples from statistics of multiple given samples. Next we consider texture images. We successfully synthesize new textures given one sample from the texture class through multiscale, multilayer wavelet models. Finally, we analyze and prove why the multiscale multilayer model is essential for signal recovery, especially natural texture images.

ACKNOWLEDGEMENTS

I would like to express the deepest appreciation to my advisor and committee chair Professor Matthew Hirn. He led me into the area of signal processing and provided lots of guidance during my PhD research. He has always been supportive and kind to me during the past five years. This dissertation work would not have been possible without his guidance and encouragement.

I would also like to thank my committee mumbers, Professor Yuying Xie, Professor Mark Iwen, Professor Yimin Xiao and Professor Vishnu Boddeti, for their insightful comments and support during my research.

Thanks to my dear friends, Binbin, Hongnan, Ningyu, Yuning, Yuanyi, Zheng for their consistent help and company. I greatly value their friendship and they are the treasures I got from MSU during the past five years. Thanks to my old friends Yilang and Yanjun for their long distance support and life-long friendships.

Most importantly, I would like to thank my parents Xiaoqiang He and Li Lin for loving me and supporting me at every stage of my life. Many thanks to my fiance Longhao Jin for always caring for me and standing by my side. And thanks to my little friends Kiwi and Simba for the warmest accompany.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ALGORITHMS

CHAPTER 1

INTRODUCTION

Wavelet analysis [4] and deep learning are two popular fields for signal processing. Wavelet transforms can be used to extract information from many different kinds of signal data and are related to harmonic analysis. In the 90's, people used wavelets to analyze 2D signals and do classification on texture images [5]. In [2], directional even and odd wavelets are used to extract statistical features of texture images and reconstruct the images from such features. In recent years, image processing has achieved great progress with the development of deep learning, which uses deep neural networks (DNN) to process signals. One popular model is the convolutional neural network (CNN), which achieves great performance on image classification [1, 6, 7]. DNNs are not only used to do classification on high dimensional data, but also to generate new data. A very deep neural network called VGG [1], which is pre-trained over Imagenet [8], successfully synthesizes texture images from only one sample [3]. Another recent model is the Generative Adversarial Network (GAN) [9], which generates signals that are from the same distribution of given data by training two networks in an adversarial way. However, there is not enough theory to explain why deep learning works, so these types of methods remain black boxes. Recently, theory was developed on the scattering transform [10], which combines wavelet transforms and nonlinear operators to capture features of signals. The process is similar to a CNN and it connects wavelet theory to deep neural nets. So a new task is to understand deep neural networks through theory in wavelet analysis.

The research contained in this thesis involves wavelet analysis and deep learning in signal processing. Within this field, an interesting and difficult task is signal reconstruction. It requires to extract all important features of signals and to develop efficient algorithms to reconstruct the signals from such features. This thesis describes the use of modified scattering transforms to synthesize 1D sparse signals and 2D texture images. It also reports a new

algorithm that combines GANs and scattering networks to generate 1D stochastic processes. The relationship between filter size, model depth and signals is also discussed. Finally, wavelet analysis on random fields is explored.

The remainder of this thesis is organized as follows. Chapter 2 introduces background material. Chapter 3 describes work on 1D sparse signal analysis and synthesis. Chapter 4 explores multilayer scattering features on Poisson point processes, especially at small scales. Chapter 5 presents a multiscale, multilayer model for texture image synthesis. Chapter 6 provides theoretical analysis on filter size, model depth and signal reconstruction. Chapter 7 discusses wavelet analysis on random fields.

CHAPTER 2

BACKGROUND

This chapter introduces the background of stochastic processes, CNNs, GANs, wavelets, and the scattering transform. In section 2.1, stochastic processes will be defined and properties such as being stationary and ergodic will be discussed. Section 2.2 covers the background of CNNs and a specific architecture VGG, to explain the basic operators needed to define a CNN. Section 2.3 reviews the intuition and architecture of GANs. Section 2.5 introduces the 1D and 2D wavelet family and wavelet transforms. The intuition of wavelet analysis in signal processing will be briefly described. Section 2.6 reviews the definition and properties of the scattering transform, which was recently proposed as a mathematical model for CNNs that uses wavelet theory.

2.1 Stochastic processes

Stochastic processes are used to model signals from real life that exhibit randomness. A stochastic process $\{X_t : t \in \mathbb{R}^d\}$ is a collection of random variables indexed by t. When $d > 1$, the stochastic process is referred to as a random field. For $d = 2$, the process is often used to model grey level texture images. The mean and covariance function of X_t is defined as:

$$\mu_X(t) = \mathbb{E}X_t,\ \sigma_X(s,t) = \mathrm{Cov}(X_s, X_t)$$

Two processes $X = \{X_t : t \in \mathbb{R}^d\}$ and $Y = \{Y_t : t \in \mathbb{R}^d\}$ are said to be equal in the sense of distribution if for any index set $\{t_1, t_2, ..., t_k\} \subset \mathbb{R}^d$,

$$(X_{t_1}, X_{t_2}, ..., X_{t_k}) \stackrel{d}{=} (Y_{t_1}, Y_{t_2}, ..., Y_{t_k})$$

A stochastic process is stationary if,

$$\{X_{t+h} : t \in \mathbb{R}^d\} \stackrel{d}{=} \{X_t : t \in \mathbb{R}^d\},\ \forall h \in \mathbb{R}^d.$$

Stationarity implies the mean function $\mu_X(t) = \mu$ is a constant that does not depend on t and the covariance function $\sigma_X(s,t) = \sigma(s-t)$ only depends on the difference between s and t. A stochastic process has stationary increments if,

$$\{X_{t+h} - X_h : t \in \mathbb{R}^d\} \stackrel{d}{=} \{X_t - X_0 : t \in \mathbb{R}^d\}, \forall h \in \mathbb{R}^d.$$

Define the time average of X to be,

$$\widehat{\mu}_{X,T} = \frac{1}{2T} \int_{-T}^{T} X_t \, dt,$$

and let $\sigma^2_{X,T}$ be the variance of $\widehat{\mu}_{X,T}$. A stationary process is said to be mean-ergodic if

$$\lim_{T \to \infty} \sigma^2_{X,T} = 0,$$

which implies

$$\lim_{T \to \infty} \mathbb{E}|\widehat{\mu}_{X,T} - \mu|^2 = 0.$$

From a practical perspective, if x_t is a realization of X_t over a large interval $[-T, T]$, then with high probability

$$\frac{1}{2T} \int_{-T}^{T} x_t \, dt \approx \mu.$$

The ergodic property for d moments is defined in a similar way. If X is stationary and ergodic, the statistics of the process can be derived from a single, sufficiently long, random realization of this process. These properties become useful when we discuss the wavelet statistics of texture images in the next few chapters.

2.2 Convolutional Neural Networks

CNNs are a special class of deep neural networks for dealing with data that has geometric structure. A deep neural network is usually constructed with an input layer, an output layer and many hidden layers. Each layer consists of a linear or affine operator and a nonlinear operator. Training a network usually corresponds to learning the weights of the linear operators. CNNs use convolution as the linear operator to detect local information and

use a pooling function for nonlinear downsampling. Figure 2.1 shows a specially designed and pre-trained CNN called VGG. We now give an explicit definition of the operators in 2D CNNs.

Given a discrete input signal x and a filter K in $\mathbb{R}^2$, the ordinary discrete convolution operator is defined as:

$$(x * K)(i, i') = \sum_m \sum_{m'} x(m, m') K(i - m, i' - m').$$

In a CNN, an image is always regarded as a three dimensional data, where the third dimension is referred to as the channel dimension. For example, a grey scale image is regarded of size $N \times N \times 1$ and an RGB image is regarded as $N \times N \times 3$. For a convolution layer in a CNN, suppose we have the input x with n_1 channels denoted as $(x_i)_{1 \le i \le n_1}$ and we would like the output to have n_2 channels; we need $n_1 \times n_2$ filters which are denoted as $(K_{i,j})_{1 \le i \le n_1, 1 \le j \le n_2}$. The convolution layer specifically computes:

$$C(x)_j = \sum_{i=1}^{n_1} x_i * K_{i,j},\ 1 \le j \le n_2$$

for each output channel j. In practice, the convolution filters to be learned are usually of small size, such as 3×3 or 5×5. Good things about it include sparse interactions (each pixel at the current layer is computed from localized output from the last layer), parameter sharing (pixels are transformed with the same filters), and equivariance to translation (this operation is commutative with translation). It is also efficient to detect the singularities of a signal, such as the edges in an image.

This linear operation $C(x)$ is usually followed by a nonlinear activation function in a CNN. For example, a ReLU (Rectified linear unit) function $g(x) = \max\{0, x\}$ is used in the VGG network. Other CNN networks also use logistic sigmoid and hyperbolic tangent as the activation function. For classification tasks, a softmax function

$$\text{softmax}(x)_{i_0} = \frac{\exp(x_{i_0})}{\sum_i \exp(x_i)}$$

is used at the output layer to predict the probability of a data that belongs to class i_0.

A typical CNN layer may also have a pooling function. A max pooling returns the maximum output of the corresponding rectangular neighborhood. The average pooling is another popular pooling function. Pooling functions are used to represent the data with localized summary statistics which are approximately invariant to local translations. Pooling also increases the receptive fields, i.e., each output pixel contains information of more pixels from the last layer. These properties are important for image processing.

Since each operator is defined explicitly, the gradient of the loss function over the weights can be computed explicitly, which makes the training efficient with gradient descent. To compute the gradient over weights and update the weights in gradient descent is known as the back-propagation, where we need the computation of the gradient to flow backwards in the network through the chain rule.

CNNs have shown great efficiency and accuracy in practical applications, such as natural language processing, image classification and video recognition. Nowadays, people also use it to reconstruct and generate high dimensional data from an unknown distribution. However, there is not too much theory on why CNNs work so well. We try to interpret CNNs through scattering transforms [10] in Section 2.6.

2.3 Generative Adversarial Networks

A very widely used model to learn the distribution of high-dimensional data is the Generative Adversarial Network (GAN) [9] from deep learning. Figure 2.2 shows the structure of the GAN which consists of two models: a generator and a discriminator. The two models are trained in an adversarial way in which the generator generates fake data that looks like real data and the discriminator tries to distinguish between the real data and generated data. If trained successfully, the generator learns the real data distribution.

Suppose we have a discriminator D and a generator G. The generator G takes in a random vector z that comes from a prior random distribution p_z and outputs data $G(z)$. The discriminator D takes in real data x that comes from the real distribution p_{data} and fake

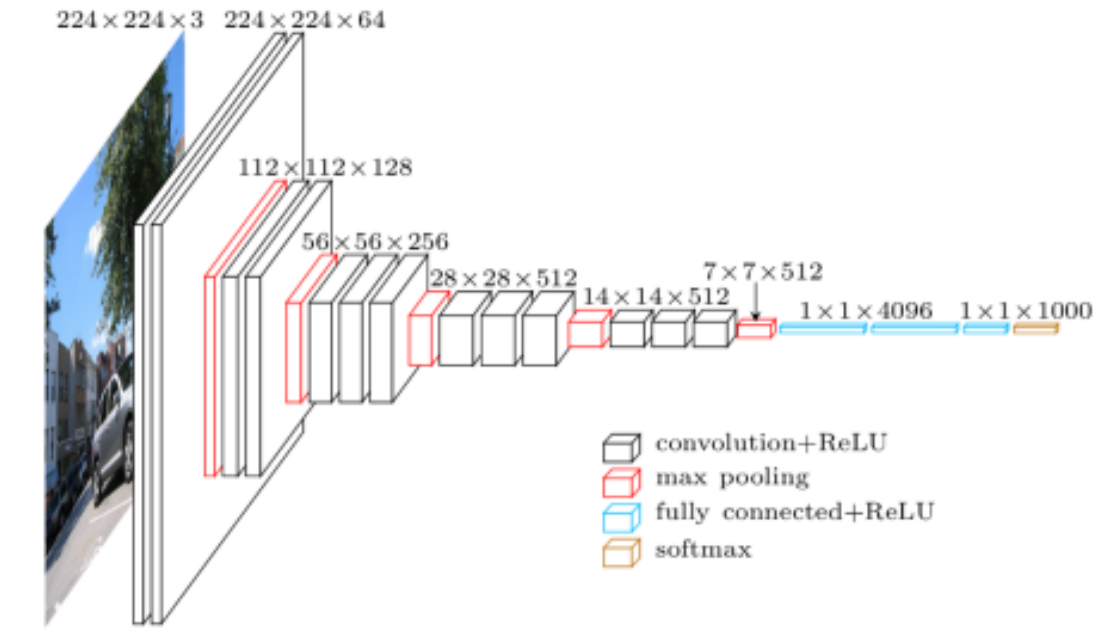


Figure 2.1 A special pre-trained very deep CNN: VGG[1]

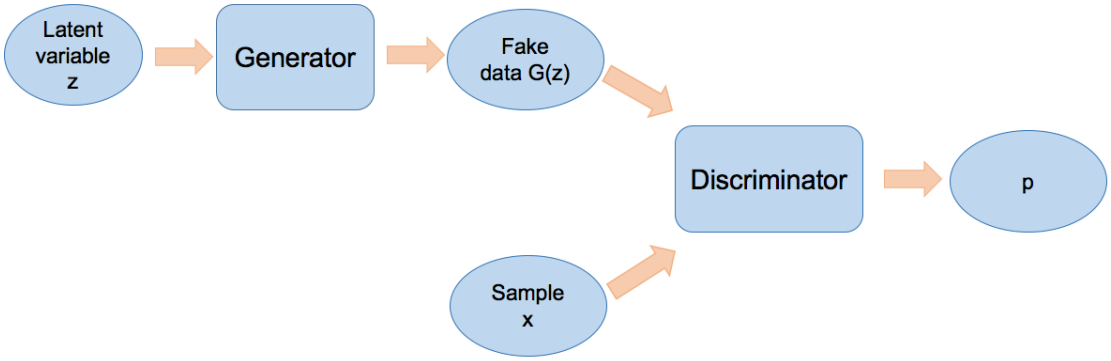


Figure 2.2 Structure of the GAN. The generator takes in a random vector z with lower dimensions and tries to generate a new signal $G(z)$ that comes from the same distribution as the given data x. The discriminator takes in a signal $G(z)$ or x and tries to output a probability p of the signal coming from the real distribution.

data $G(z)$ that comes from the generated distribution p_g and outputs $D(x)$ and $D(G(z))$, which represents the probability the data comes from the real distribution. The loss function is defined through the binary cross entropy:

$$V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log(D(x))] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$

The goal is to find the D and G that solve the following minimax problem:

$$\min_G \max_D V(D,G)$$

The problem is solved by using gradient descent to update the parameters in the deep neural networks D and G. The authors prove that the global optimal solution is achieved when $p_g = p_{data}$ and $D(x) = \frac{1}{2}$. The GAN model does not learn the data distribution explicitly but generates signals that come from the real data distribution, that is, the generated data is not a real data but is similar to the real data. GANs have been widely used to generate natural images. In [11], the authors train a GAN in a progressive manner by gradually adding more layers in the training process and generates high resolution images. In [12], the model takes in sentences and generates images that fit the description. GANs can also be used to repair images with missing parts [13] or generate images with certain artistic style [14]. GANs are useful for many applications.

2.4 Texture Synthesis using VGG and Style Transfer

In the previous section, we introduced GANs for generating new images. Another interesting topic from image processing is style transfer, which aims to transfer the style of one image to another. This is motivated by [3], where the authors generate a new texture image by matching the multi-layer statistics of image features computed through VGG [1]. They show the statistics from multiple deep layers is necessary to generate a new realization of the texture. And the loss between such statistics is referred to as the style loss in later research work.

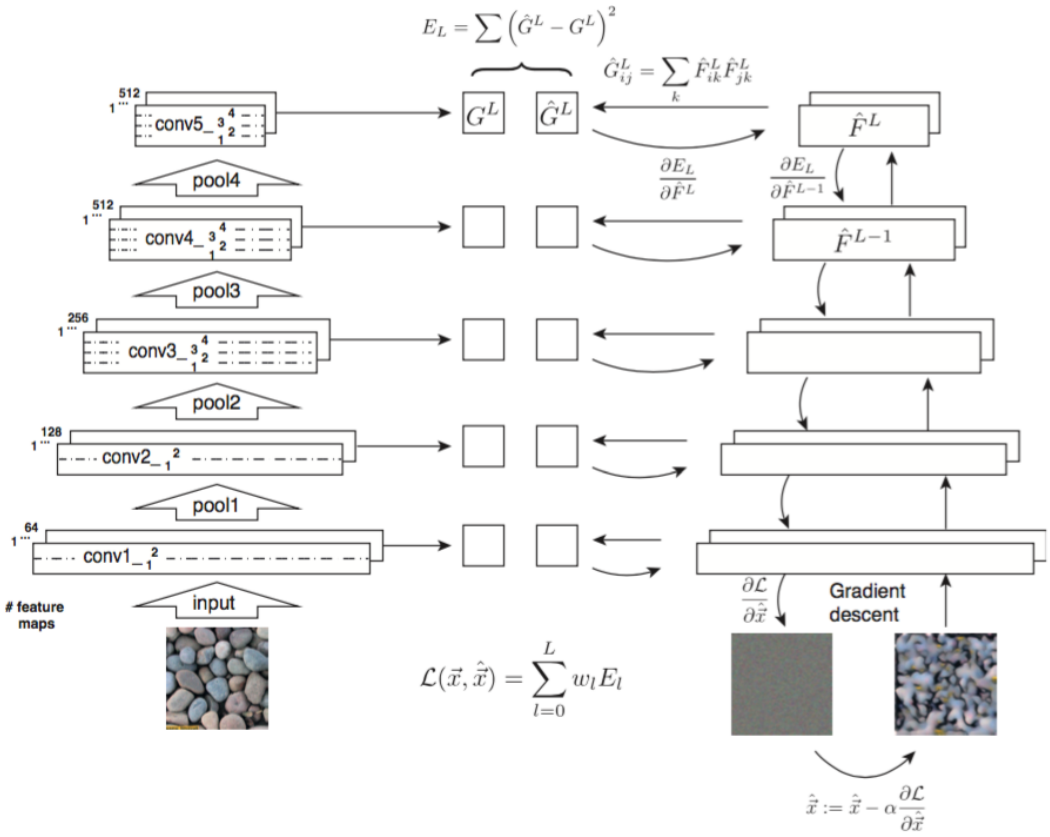


Figure 2.3 Process of using VGG features to synthesize texture images [1]

VGG is a very deep CNN with 16-19 layers. Figure 2.1 shows the architecture and the definition for each layer is described in Section 2.2 when we reviewed the CNN. This network has been pre-trained and showed great performance on image classification. Figure 2.3 shows the process of texture synthesis in [3] and the details will be described in the following paragraph.

Suppose we have an input texture x_0. The output at layer l is a matrix $H^l(x_0) \in \mathcal{R}^{N_l \times M_l}$ where N_l is the number of output channels in layer l and M_l is the size of output from each channel. The authors use the gram matrix between channels as the summary statistics:

$$G^l(x_0)_{ij} = \sum_k H^l(x_0)_{ik} H^l(x_0)_{jk}$$

Features $\{G^1(x_0), G^2(x_0), \ldots\ G^L(x_0)\}$ from multiple layers are used to represent the texture

image x_0. Define the loss function between two images x_0 and x as:

$$\text{loss}(x_0, x) = \sum_l ||G^l(x_0) - G^l(x)||_2^2$$

which is the summation of the l_2 distance between gram matrices from each layer l. The goal is to find x^* that minimizes the loss:

$$x^* = \arg\min_x \text{loss}(x_0, x).$$

By solving the problem above using gradient descent, they get back a new image which has the same statistics but is different from the input. This can be thought of as generating a new realization of one texture class from a given realization. Figure 2.4 shows the synthesis results. The good performance of matching the gram matrix from multiple layers explains that multi-layer statistics from the VGG network might capture general features of a class of texture. This work motivates the work on style transfer and has shown great performance [15]. It also inspired our work on texture synthesis in Chapter 5. However, it has poor performance on reproducing natural images which have more localized geometric structure and are not stationary.

2.5 Wavelets

Wavelet theory is an important field developed for signal processing [4]. The Fourier transform of a function $x \in L^2(\mathbb{R}^d)$ is defined as,

$$\widehat{x}(\omega) = \int_{\mathbb{R}^d} x(u) e^{-iu\cdot\omega} du, \, \forall \omega \in \mathbb{R}^d$$

and has the property

$$\widehat{x * y} = \widehat{x} \odot \widehat{y}\,. \tag{2.1}$$

One can recover the function x through the inverse Fourier transform:

$$x(u) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} \widehat{x}(\omega) e^{iu\cdot\omega} d\omega, \, \forall \omega \in \mathbb{R}^d.$$

Figure 2.4 Synthesis results with different layers. The last row shows the original images. The first row shows synthesized image by matching statistics from only "conv1-1" layer and the second shows that matching statistics from "conv1-1" and "pool1" layers and so on. With only lower layer statistics, the synthesis lose information of the texture. Adding statistics from deeper layers improves the performance, which proves higher layer statistics are necessary to capture texture information. The last column shows result of a natural image, which has more localized structure and is not stationary.

A wavelet $\psi \in L^2(\mathbb{R}^d)$ is defined as a function that is localized in space and frequency and has zero average [4]:

$$\int_{\mathbb{R}^d} \psi(u)du = 0$$

It is normalized so that $\|\psi\| = 1$. We focus on the cases $d = 1$ and $d = 2$ in the following. For $d = 1$, a family of wavelets is obtained by dilating ψ:

$$\psi_j(u) = 2^{-j}\psi(2^{-j}u),\ j \in \mathbb{Z}.$$

The 1D wavelet transform of a signal $x \in L^2(\mathbb{R})$ is defined as:

$$Wx = (x * \psi_j(u))_{u\in\mathbb{R},j\in\mathbb{Z}}.$$

For $d = 2$, a family of wavelets is obtained by dilations and rotations:

$$\psi_{j,\theta}(u) = 2^{-2j}\psi(2^{-j}R_\theta^{-1}u),\ R_\theta = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

A 2D wavelet transform is defined as

$$Wx = (x * \psi_{j,\theta}(u))_{u\in\mathbb{R}^2,j\in\mathbb{Z},\theta\in\Theta\subset[0,\pi]}$$

Usaully $\Theta = \{\frac{k\pi}{K} : k = 0, \ldots, K-1\}$. Figure 2.5 shows a texture image and its wavelet coefficients computed through Morlet wavelets. Figure 2.6 shows a family of 2D Morlet wavelets in space and frequency. A 2D mother Morlet wavelet is defined as:

$$\psi(u_1, u_2) = \frac{1}{2\pi\sigma_1\sigma_2}\exp\Big(-\frac{1}{2}(\frac{u_1^2}{\sigma_1^2} + \frac{u_2^2}{\sigma_2^2})\Big) \cdot \exp(2\pi i\xi u_1 - C),\ (u_1, u_2) \in \mathbb{R}^2$$

where σ_1 and σ_2 determine the scale of the mother wavelet at two directions and $(\xi, 0)$ determines the center frequency of the wavelet. The constant C ensures $\int \psi(u_1, u_2)du_1du_2 = 0$. A family of Morlet wavelets is obtained by dilation and rotation described above. As can be seen, the Fourier transform of Morlet wavelets are essentially supported over different bounded regions, which means wavelets are localized in the frequency field. With Equation 2.1, the wavelet transform in frequency can be written as:

$$\widehat{Wx} = (\widehat{x} \odot \widehat{\psi_{j,\theta}})_{j,\theta}$$

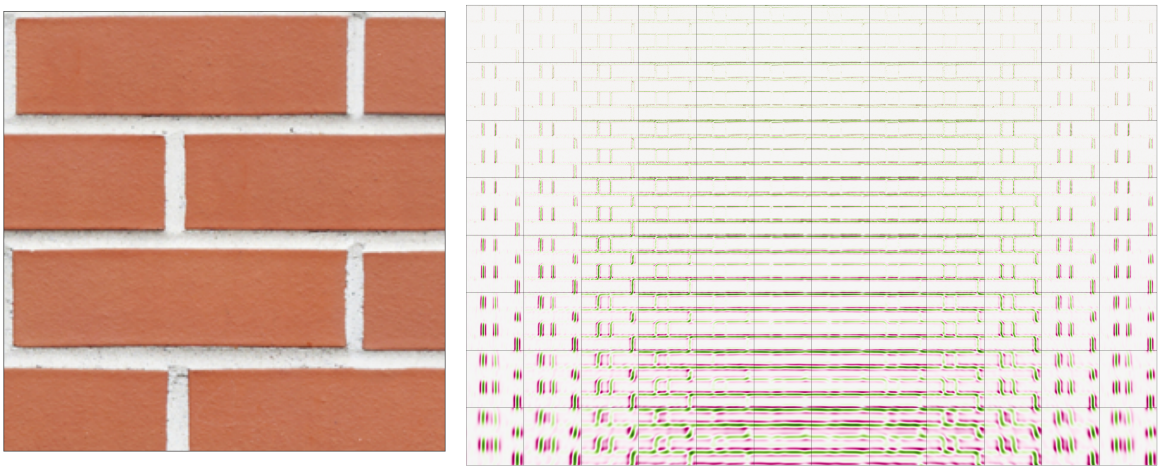


Figure 2.5 **Left:** A texture image of bricks with edges at direction $0, \pi/2$. **Right:** The wavelets of the texture computed through Morlet wavelets, shown in Figure 2.6. It shows the texture has large response at $\theta = 0$ and $\theta = \pi/2$, tiny response at other angles, which match the directions of the edges in the texture. Also, as the wavelet scale goes larger, the response gets smoother.

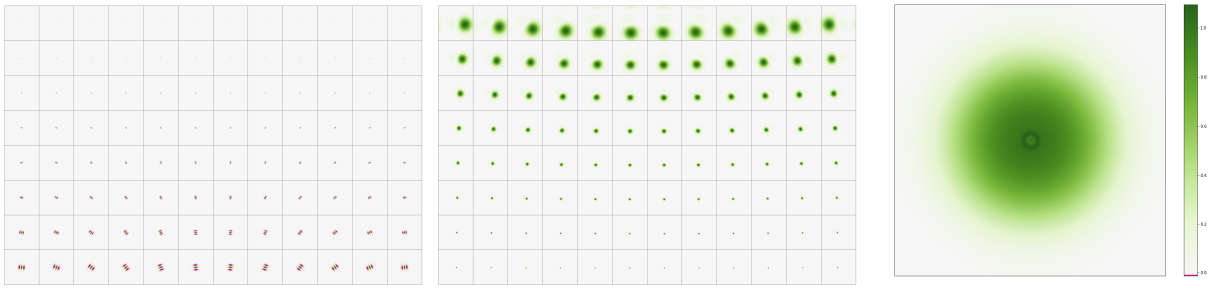


Figure 2.6 Morlet wavelets (real part) with 8 different scales and 12 different rotations in space and frequency. **Left:** 2D Morlet wavelet family $(\psi_{j,\theta})_{j,\theta}$ in space. **Middle:** 2D Morlet wavelet family in frequency, which is the Fourier transform of 2D Morlet wavelet family $(\widehat{\psi}_{j,\theta})_{j,\theta}$. **Right:** The summation $|\widehat{\phi_J}(\omega)|^2 + \sum_{\lambda\in\Lambda} |\widehat{\psi_\lambda}(\omega)|^2$ which is almost a constant over all frequencies, making an approximate Littlewood-Paley frame. This figure shows this group of wavelets plus a low pass satisfy the Littlewood-Paley condition on a half plane approximately, except for the boundary.

By multiplying $\widehat{x}$ with $(\widehat{\psi_{j,\theta}})_{j,\theta}$, the wavelet transform can capture localized information of the signal x in the frequency field.

Let $\lambda = (j, \theta) \in \Lambda_J = \{(j, \theta) : j \leq J, \theta \in \Theta \subset [0, 2\pi]\}$ where Λ_J is the index set for wavelets and use ψ_λ to denote $\psi_{j,\theta}$. Let ϕ be a low pass filter whose Fourier transform is concentrated at low frequencies. Set $\phi_J(u) = 2^{-2J}\phi(2^{-J}u)$. The filters $\{\phi_J, (\psi_\lambda)_\lambda\}$ are said

to be a Littlewood-Paley tight frame if:

$$|\widehat{\phi_J}(\omega)|^2 + \sum_{\lambda \in \Lambda_J} |\widehat{\psi_\lambda}(\omega)|^2 = 1, \, \forall \omega \in \mathbb{R}^2$$

Define $A_J x = x * \phi_J$ and $W[\lambda]x = x * \psi_\lambda$. The wavelet transform can be written as $W_J x = \{A_J x, (W[\lambda]x)_{\lambda \in \Lambda_J}\}$ and its norm is:

$$\|W_J x\|^2 = \|A_J x\|^2 + \sum_{\lambda \in \Lambda_J} \|W[\lambda]x\|^2$$

If $\{\phi_J, (\psi_\lambda)_\lambda\}$ is a Littlewood-Paley tight frame, then $\|Wx\|^2 = \|x\|^2$. This shows the wavelet transform preserves the norm of a signal x and the property can be proved using Plancherel formula. We can define the dual wavelets as

$$\widehat{\widetilde{\phi_J}}(\omega) := \overline{\widehat{\phi_J}(\omega)}, \quad \widehat{\widetilde{\psi_\lambda}}(\omega) := \overline{\widehat{\psi_\lambda}(\omega)}.$$

A signal x can be reconstructed from its wavelet transforms using the dual filters and the formula

$$x = x * \phi_J * \widetilde{\phi}_J + \sum_{\lambda \in \Lambda_J} x * \psi_\lambda * \widetilde{\psi}_\lambda \, . \tag{2.2}$$

2D Morlet wavelets are used in image processing since they have a strong response to edges. Texture images always have edges distributed in a repeated pattern. Therefore the statistics (especially mean and variance) of wavelet coefficients, which summarize global edge information, may be used to analyze texture images.

2.6 Scattering transform

Convolutional neural networks work very well for signal processing, especially image processing. However, why they work is not well understood and theory still remains to be developed. In [10], the author comes up with a scattering transform, which can be viewed as a mathematical model for CNNs. It has been used to achieve near state of the art results in the fields of audio signal processing [16, 17, 18, 19, 20], computer vision [21, 22, 23, 24, 25, 26], and quantum chemistry [27, 28, 29, 30], amongst others. The scattering transform is provably

invariant to local (or global) translations of the input signal and is also Lipschitz stable to the actions of diffeomorphisms on the input. These properties are motivated by the fact that signals with translation or small deformation usually contain similar information (e.g., they are in the same class). On the other hand, high frequency information is needed to distinguish one signal class from another, which motivated the use of nonlinear activation functions and a deep architecture. This non-paramatric model helps us to understand CNNs in a mathematical way.

Assume we have a real valued function $x \in \mathbf{L}^2(\mathbb{R}^d)$. Let $L_c : \mathbf{L}^2(\mathbb{R}^d) \to \mathbf{L}^2(\mathbb{R}^d)$, for $c \in \mathbb{R}^d$, be a translation operator:

$$L_c x(u) = x(u - c)$$

A map Φ on $\mathbf{L}^2(\mathbb{R}^d)$ is translation invariant if:

$$\Phi(L_c x) = \Phi(x),\ \forall x \in \mathbf{L}^2(\mathbb{R}^d), c \in \mathbb{R}^d$$

Let $\tau : \mathbb{R}^d \to \mathbb{R}^d$ be a displacement function and $L_\tau : \mathbf{L}^2(\mathbb{R}^d) \to \mathbf{L}^2(\mathbb{R}^d)$ be a diffeomorphism operator:

$$L_\tau x(u) = x(u - \tau(u))$$

Let $\nabla\tau(u)$ and $H\tau(u)$ be the Jacobian and Hessian of τ. Define:

$$\|\tau\|_\infty = \sup_{u\in\mathbb{R}^d} |\tau(u)|,\ \|\nabla\tau\|_\infty = \sup_{u\in\mathbb{R}^d} \|\nabla\tau(u)\|,\ \|H\tau\|_\infty = \sup_{u\in\mathbb{R}^d} \|H\tau(u)\|$$

The map Φ is Lipschitz continuous to diffeomorphisms if there exists a constant C such that

$$\|\Phi(x) - \Phi(L_\tau x)\| \le C\|x\|_2(\|\tau\|_\infty + \|\nabla\tau\|_\infty + \|H\tau\|_\infty),\ \forall x \in \mathbf{L}^2(\mathbb{R}^d)$$

The modulus of the Fourier transform $\Phi(x) = |\widehat{x}|$ is proved to be translation invariant but not stable to diffeomorphisms. So we turn to wavelet transforms.

In this paragraph, we will explain the intuition of how the scattering transform is developed to satisfy the translation invariance property while preserving high frequency information. The paper [10] proves it is also stable to diffeomorphism but we will omit the details.

Recall we defined $\lambda = (j, \theta) \in \Lambda_J$, as well as the operators $A_J x = x * \phi_J$ and $W[\lambda]x = x * \psi_\lambda$ in section 2.2. Since $A_J x$ and $W[\lambda]x$ are convolution operators, they commute with translations:

$$A_J(L_c x) = L_c(A_J x),\ W[\lambda](L_c x) = L_c(W[\lambda]x)$$

Then translation invariance can be obtained by integrating $x * \phi_J$ or $x * \psi_\lambda$. Also note that the operator A is locally translation invariant in the sense that $\|A(L_c x) - A(x)\| \leq \frac{C \cdot |c| \cdot \|x\|}{2^J}$ since ϕ_J is a low pass filter. This is essential because sometimes we want local translation invariance instead of full translation invariance. Then instead of $\int x * \psi_\lambda$, we could try to use $x * \psi_\lambda * \phi_J$ to obtain local translation invariance. However, since ψ_λ is a wavelet with zero average, the integral $\int x * \psi_\lambda = 0$. Also since the support of $\widehat{\psi_\lambda}$ and $\widehat{\phi_J}$ have small overlap, $x * \psi_\lambda * \phi_J \approx 0$. Therefore we need a nonlinear operator. The scattering transform uses the modulus since it is non-expansive. Then nontrivial global and local translation invariance are obtained by

$$\int |x * \psi_\lambda| \text{ or } |x * \psi_\lambda| * \phi_J.$$

Note that the modulus operator nonlinearly projects $x * \psi_\lambda$ from the high frequency field to the low frequency field. Also note that $\int |x * \psi_\lambda| = \widehat{|x * \psi_\lambda|}(0)$ and $|x * \psi_\lambda| * \phi_J = \widehat{|x * \psi_\lambda|}\widehat{\phi_J}$ and $\widehat{\phi_J}$ is supported in low frequency field. However, the support of $\widehat{\phi_J}$ is smaller than the support of $\widehat{|x * \psi_\lambda|}$. Therefore, part of the information in $\widehat{|x * \psi_\lambda|}$ gets lost in the above representation. To recover it, we apply another wavelet transform over $|x * \psi_\lambda|$. The information in $|x * \psi_\lambda|$ is preserved by $\{|x * \psi_\lambda| * \phi_J, (|x * \psi_\lambda| * \psi_{\lambda'})_{\lambda'}\}$. The term $|x * \psi_\lambda| * \phi_J$ is invariant to translation while $|x * \psi_\lambda| * \psi_{\lambda'}$ recovers the lost high frequency information. By applying another nonlinear operator and a low pass to $|x * \psi_\lambda| * \psi_{\lambda'}$, it also gets translation invariance. We give the standard definition of scattering transform in the next paragraph.

Let p be a path sequence $p = \{\lambda_1, \lambda_2, ..., \lambda_m\} \in \Lambda_J^m$. Define $U[\lambda]x = |x * \psi_\lambda|$ for $x \in L^2(\mathbb{R}^d)$. A scattering propagator is defined as the path-ordered product:

$$U[p] = U[\lambda_m]...U[\lambda_2]U[\lambda_1]$$

Then the windowed scattering transform of $x \in L^2(\mathbb{R}^d)$ for a path p is defined as:

$$S_J[p]x(u) = U[p]x * \phi_J(u)$$

and is invariant to local translation. The scattering transform is defined as:

$$\bar{S}[p]x = \frac{1}{\mu_p} \int U[p]x(u)du \quad \text{with} \quad \mu_p = \int U[p]\delta(u)du \tag{2.3}$$

and is invariant to global translation. If the infinite set of finite paths p is defined as $\mathcal{P}_J$, we can denote:

$$S_J[\mathcal{P}_J]x = \{S_J[p]x\}_{p\in\mathcal{P}_J} \quad \text{or} \quad \bar{S}[\mathcal{P}_J]x = \{\bar{S}[p]x\}_{p\in\mathcal{P}_J}$$

The multi-layer structure preserves high frequency information. The author also proves the scattering transform over $\mathcal{P}_J$ is stable to diffeomorphisms, i.e., for all $\tau \in \mathbf{C}^2(\mathbb{R}^d)$ with $\|\nabla\tau\|_\infty \leq \frac{1}{2}$, there exists a constant C such that

$$\|S_J[P_J]x - S_J[P_J](L_\tau x)\| \leq C\|U[P_J]x\|_1(\|\tau\|_\infty + \|\nabla\tau\|_\infty + \|H\tau\|_\infty), \forall x \in \mathbf{L}^2(\mathbb{R}^d)$$

where $\|U[P_J]x\|_1 = \sum_{m=0}^{+\infty} \|U[\Lambda_J^m]x\|$. The key point to the proof is the stability of the wavelet transform which we omit here.

The scattering transform in $\mathbf{L}^2(\mathbb{R}^d)$ can be extended to stationary processes. If $X(u)$ is a stationary process, for all $p = \{\lambda_1, \lambda_2, ..., \lambda_m\} \in \Lambda_J^m$, $U[p]X(u)$ is also stationary, and its expected value does not depend on u. The expected scattering transform is defined as:

$$\bar{S}[p]X = \mathbb{E}(U[p]X) = \mathbb{E}(||X * \psi_{\lambda_1} * \cdots| * \psi_{\lambda_m}|).$$

This definition replaces the normalized integral of the scattering transform in Equation 2.3 by an expected value. If X is also ergodic, the expected scattering transform is estimated by computing the scattering transform of a realization $x_0(u)$ of $X(u)$:

$$\bar{S}[p]X \approx \frac{1}{\mu_p} \int U[p]x_0(u)du$$

2.7 Signal recovery through statistical features

This section introduces the general types of problems we will consider in this thesis.

Given a deterministic signal $x \in \mathbf{L}^2(\mathbb{R}^d)$ or a stochastic process $X(u) : u \in \mathbb{R}^d$, our goal is to find a representation $\Phi(x)$ or $\Phi(X)$ that is translation invariant, i.e.,

$$\Phi(L_c x) = \Phi(x), \quad \text{or} \quad \Phi(L_c X) = \Phi(X), \, \forall c \in \mathbb{R}^d$$

where $L_c x(u) = x(u - c)$ or $L_c X(u) = X(u - c)$. Typically we are interested in when Φ is a CNN representation or a scattering representation.

For a deterministic signal x, we explore under what conditions can we recover x from Φ up to translation, i.e.,

$$\text{if } \Phi(x) = \Phi(x'), \text{ then } x' = L_c x.$$

This aims to recover the exact signal up to translation. Chapter 3 explores this issue for 1D deterministic sparse signals. Similarly, Chapter 6 explores 2D deterministic line texture images and frame-like texture images.

For a stochastic process X, we think of Φ as statistics, i.e.,

$$\Phi(X) = \mathbb{E}[\tilde{\Phi}(X)]$$

for some function $\tilde{\Phi}$. When X is ergodic, $\Phi(X)$ is usually estimated from a sample $x \sim X$:

$$\Phi(X) \approx \frac{1}{(2T)^d} \int_{[-T,T]^d} \tilde{\Phi}(x)$$

Given a sample x of X, we explore under what conditions can we generate another sample x' of X from $\tilde{\Phi}$, i.e.,

$$\text{if } \frac{1}{(2T)^d} \int_{[-T,T]^d} \tilde{\Phi}(x') = \frac{1}{(2T)^d} \int_{[-T,T]^d} \tilde{\Phi}(x), \text{ then } x' \sim X.$$

Chapter 4 explores the problem for X being a Poisson point process and we try to generate new samples by learning the distribution of $\Phi(X)$. Chapter 5 suppose X to be a class of texture images and we try to generate new samples $x' \sim X$ given an estimate of $\Phi(X)$ from one sample texture image $x \sim X$.

CHAPTER 3

CHARACTERIZING SPARSE SIGNALS THROUGH A HYBRID SCATTERING TRANSFORM

3.1 Introduction

As reviewed in Section 2.6, the scattering transform is a mathematical model for CNNs that is invariant to translation, stable to diffeomorphism and captures high frequency information. These properties are essential for signal analysis as signals with translations and small deformation are usually from the same class and high frequency information is important to distinguish signals from one class to another. On the other hand, filters learned in the early layers of CNNs typically resemble wavelets. But it usually requires a large dataset to learn a CNN and it remains unclear why such models work well for different types of signals, especially natural images.

The scattering transform and CNN both show great performance in image classification, proving their ability to learn essential representations and distinguish different types of signals. Another interesting task from machine learning is to understand how different these representations are with dissimilar signals. This is referred to as the completeness of a model in machine learning. A more complicated task from this topic is to find representations that can be used to recover a signal. Intuitively this requires to learn all important information contained in a signal.

Motivated by the structure of the scattering transform and CNNs, we propose a two-layer hybrid scattering model to capture signals with isolated singularities in this chapter. In the first layer, we apply a wavelet transform to sparsify the signal, while in the second layer, we use a Gabor type filter to leverage this sparsity. More specifically,

- We propose a new model to capture the singularities of the input signal.
- We prove the Gabor measurements used in the second layer determine the locations

and heights of a sparse signal.

- We provide an algorithm that successfully synthesizes 1D sparse signals using these measurements up to translation and reflection.

In Section 3.2 we provide our model in detail, while in Section 3.3 we show our main theorems and proofs. Section 3.4 describes our algorithm for sparse signal synthesis and Section 3.5 shows synthesis results. Section 3.6 summarizes our conclusions.

3.2 Model

Let $y(t)$ with $t \in \mathbb{R}$ be a piecewise polynomial whose knots $\{u_i\}_{i=1}^k$, $u_i < u_{i+1}$, are located on the grid $h\mathbb{Z} = \{hn : n \in \mathbb{Z}\}$ for some $h > 0$:

$$y(t) = \sum_{i=0}^{k} y_i(t), \text{ where } y_i(t) = \begin{cases} p_i(t) & t \in [u_i, u_{i+1}] \\ 0 & \text{elsewhere} \end{cases}$$

where $\{p_i\}_{i=0}^k$ are a group of polynomials satisfying $p_i(u_{i+1}) = p_{i+1}(u_{i+1}), \forall i \in \{0, 1, \cdots k-1\}$. We also assume that each of the piecewise polynomial components $y_i(t)$ has degree m_i at most m, i.e., $0 \leq m_i \leq m$. Let ψ be a mother wavelet with $\text{supp}(\psi) \subseteq [-1, 1]$ that has $m+1$ vanishing moments, i.e.,

$$\int_{-\infty}^{+\infty} t^k \psi(t) dt = 0, \, \forall 0 \leq k \leq m.$$

Since ψ has $m + 1$ vanishing moments, one can show $p_i * \psi(t) = 0$. Let $\psi_\ell(t)$ be the dilated wavelet:

$$\psi_\ell(t) = \frac{1}{2^\ell} \psi\left(\frac{t}{2^\ell}\right).$$

We have $\text{supp}(\psi_\ell) \subseteq [-2^\ell, 2^\ell]$. Figure 3.1 gives an example of a wavelet with 4 vanishing moments. Therefore we have $y * \psi_\ell(t) = 0$ for $t \notin \bigcup_{i=1}^k [u_i - 2^\ell, u_i + 2^\ell]$ as $\{u_i\}_i$ are the singularities of y. Equivalently $\text{supp}(\psi_\ell * y)$ is contained in $\bigcup_{i=1}^k [u_i - 2^\ell, u_i + 2^\ell]$. To further promote sparsity, we next apply a max-pooling operator:

$$MP_\ell z(t) = \begin{cases} z(t) & \text{if } z(t) = \max_{t' \in [t_i - 2^\ell, t_i + 2^\ell] \cap h\mathbb{Z}} z(t') \\ 0 & \text{otherwise} \end{cases}.$$

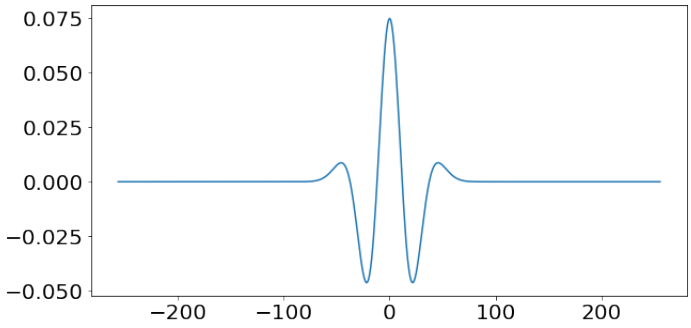
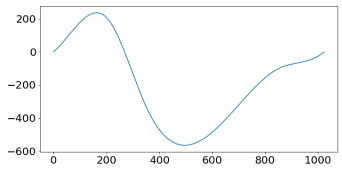
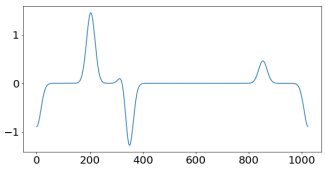


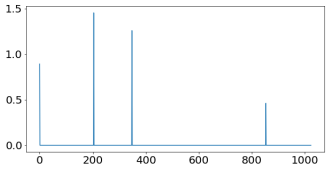
Figure 3.1 A wavelet ψ of 4 vanishing moments.



(a) piecewise polynomial $y(t)$ with highest order of 3



(b) Convolution against wavelet ψ_ℓ



(c) Sparse Signal from Max Pooling

Figure 3.2 Wavelets sparsify piecewise polynomials on the interval $[0, 1024]$.

Figure 3.2 shows the process of sparsifying a piecewise polynomial into a sparse signal with wavelet ψ_ℓ. As summarized in the following theorem, this yields a linear combination of Dirac delta functions.

Theorem 1. *Assume that* $2^{\ell+1} \leq \min_{1\leq i\leq k-1} |u_i - u_{i+1}|$. *Then,*

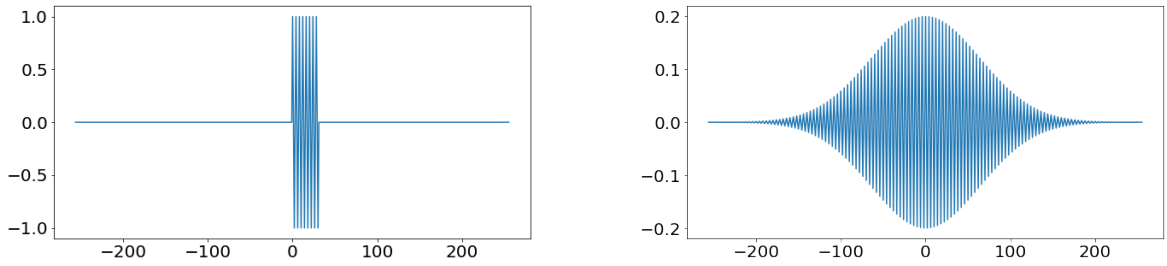
$$MP_\ell(|\psi_\ell * y|)(t) = \sum_{j=1}^{k} a_j \delta_{v_j}(t).$$

for some $a_1, \ldots, a_k > 0$, $v_j \in [u_j - 2^\ell, u_j + 2^\ell], 1 \leq j \leq k$.

In our second layer, rather than use another wavelet, we use a Gabor filter

$$g_{s,\xi}(t) = w\left(\frac{t}{s}\right) e^{i\xi t}, \tag{3.1}$$

where the parameters s and ξ determine the scale and central frequency of the filter and the window function w is supported on an interval of unit length. It differs from a wavelet in that the scale and frequency are separated in a gabor filter, giving more flexibility to adjust



(a) Indicator function window

(b) Gaussian window

Figure 3.3 Gabor filters used in the second layer (real parts)

these parameters independently. Note that with an appropriately chosen window function w, Equation 3.1 includes dyadic wavelet families in the case that one selects $s = 2^\ell$ and $|\xi| = C/s$. However, it also includes many other families of filters, including Gabor filters used in the windowed Fourier transform. Next, we take the L^p norm for some integer $p \geq 1$. As a result, we obtain translation invariant *hybrid scattering coefficients*

$$\{\|g_{s,\xi} * MP_\ell(|\psi_\ell * y|)\|_p\}_{s,\xi}.$$

By design, these measurements are invariant to translations, reflections, and global sign changes. We aim to investigate the ability of our measurements to characterize y up to these natural ambiguities. The wavelet-modulus is known to be a powerful signal descriptor [31]. Therefore, in light of Theorem 1, we shall analyze the ability of the measurements

$$\{\|g_{s,\xi} * x\|_p\}_{s,\xi} \tag{3.2}$$

to characterize sparse signals of the form

$$x(t) = \sum_{j=1}^{k} a_j \delta_{v_j}(t). \tag{3.3}$$

Furthermore, to supplement our theory, we show that the measurements (3.2) can be used to reconstruct a sparse signal of the form (3.3) up to translations and reflections (see Figure 3.2).

3.3 Theory and proofs

Let

$$x(t) = \sum_{j=1}^{k} a_j \delta_{v_j}(t), \quad v_1 < v_2 < \cdots < v_k. \tag{3.4}$$

We first consider the support set $\{v_j\}_{j=1}^k$, and let $\mathcal{D}(x) = \{\Delta_{i,j} = v_j - v_i : i < j\}$ denote the difference set as the pairwise distances of the Dirac locations $\{v_j\}_{j=1}^k$. Let

$$f_{x,\xi}(s) = \|g_{s,\xi} * x\|_p^p$$

be the gabor measurements. The following theorem shows the singularities of the measurements as a function of the scale s are contained in the pairwise distance set $\mathcal{D}(x)$.

Theorem 2. *Let $p \geq 1$ be an integer, and assume $w(t) = 1_{[0,1]}(t)$. For $i \leq j$, let*

$$\beta_{i,j}(\xi) = \sum_{\ell=i}^{j} a_\ell e^{i\xi\Delta_{i,\ell}} \tag{3.5}$$

Then, for every fixed $\xi \in \mathbb{R}$, the function $f_{x,\xi}(s)$ is piecewise linear, and $\partial_s^2 f_{x,\xi}(s)$ is a grid-free sparse signal whose support is contained in $\mathcal{D}(x)$. Specifically,

$$\partial_s^2 f_{x,\xi}(s) = \sum_{d \in \mathcal{D}(x)} \left(\sum_{\Delta_{i,j}=d} c_{i,j}(\xi) \right) \delta_d,$$

where

$$c_{i,i+1}(\xi) = |\beta_{i,i+1}(\xi)|^p - |\beta_{i+1,i+1}(\xi)|^p - |\beta_{i,i}(\xi)|^p \tag{3.6}$$

and

$$c_{i,j}(\xi) = |\beta_{i,j}(\xi)|^p + |\beta_{i+1,j-1}(\xi)|^p - |\beta_{i+1,j}(\xi)|^p - |\beta_{i,j-1}(\xi)|^p \quad \textit{for} \quad j \geq i+2. \tag{3.7}$$

Proof. We first note that

$$\begin{aligned} |(g_{s,\xi} * x)(t)| &= \left| \sum_{i=1}^{k} a_i g_{s,\xi}(t - v_i) \right| \\ &= \left| \sum_{i=1}^{k} a_i e^{i\xi(t-v_i)} 1_{[v_i, v_i+s]}(t) \right| \\ &= \left| \sum_{i=1}^{k} a_i e^{-i\xi v_i} 1_{[v_i, v_i+s]}(t) \right|. \end{aligned}$$

For every subset $I \subseteq \{1, \ldots, k\}$, let

$$R_I(s) = \{t : t \in [v_i, v_i + s] \text{ for all } i \in I, t \notin [v_i, v_i + s] \text{ for all } i \notin I\},$$

i.e. let $R_I(s)$ be the set of t for which $a_i e^{-i\xi v_i} 1_{[v_i, v_i+s]}(t)$ is nonzero if and only if $i \in I$. Then, since $w(t) = 1_{[0,1]}(t)$ it is clear that for $t \in R_I$,

$$|(g_{s,\xi} * x)(t)| = \left| \sum_{i \in I} a_i e^{-i\xi v_i} \right| \coloneqq y_I(\xi).$$

Therefore,

$$f_{x,\xi}(s) = \|(g_{s,\xi} * x)(t)\|_p^p = \sum_{I \subseteq \{1,\ldots k\}} |y_I(\xi)|^p |R_I(s)|. \tag{3.8}$$

We will show that for all $I \subseteq \{1, \ldots, k\}$, $|R_I(s)|$ is a piecewise linear function as a function whose Dirac locations are contained in $\mathcal{D}(x)$.

First, we note that $R_I(s) = \emptyset$ unless I has the form $\{i, i+1, \ldots, j-1, j\}$ for some $i \leq j$. Therefore,

$$f_s(\xi) = \sum_{i=1}^{k} \sum_{j=i}^{k} |\beta_{i,j}(\xi)|^p |R_{i,j}(s)|, \tag{3.9}$$

where, as in (3.5), $\beta_{i,j}(\xi)$ is given by

$$|\beta_{i,j}(\xi)| = \left| \sum_{\ell=i}^{j} a_\ell e^{i\xi\Delta_{i,\ell}} \right| = \left| \sum_{\ell=i}^{j} a_\ell e^{i\xi v_\ell} \right|,$$

and $R_{i,j} \coloneqq R_{\{i,\ldots,j\}}$. Now, turning our attention to $R_{i,j}(s)$, we observe by definition that a point t is in $R_{i,j}(s)$ if and only if it satisfies the following three conditions:

$$\begin{aligned} v_\ell \leq & t \leq v_\ell + s \quad \text{for all } i \leq \ell \leq j, \\ & t > v_{i-1} + s, \text{ and} \\ & t < v_{j+1}. \end{aligned}$$

Therefore, letting $(a \wedge b) \coloneqq \max\{a, b\}$ and $(a \vee b) \coloneqq \min\{a, b\}$, we see

$$R_{i,j}(s) = [x_j, x_i + s] \cap [x_{i-1} + s, x_{j+1}] = [x_j \vee (x_{i-1} + s), (x_i + s) \wedge x_{j+1}], \tag{3.10}$$

and

$$|R_{i,j}(s)| = ((x_i + s) \wedge x_{j+1}) - (x_j \vee (x_{i-1} + s)),$$

if the above quantity is positive and $|R_{i,j}(s)| = 0$ otherwise. It follows from (3.10) that $|R_{i,j}(s)|$ is a piecewise linear function, and that $\partial_s^2 |R_{i,j}(s)|$ is given by

$$\partial_s^2 |R_{i,j(S)}| = \delta_{\Delta_{i,j}}(s) + \delta_{\Delta_{i-1,j+1}}(s) - \delta_{\Delta_{i-1,j}}(s) - \delta_{\Delta_{i,j+1}}(s). \tag{3.11}$$

In order for this equation to be valid for all $1 \leq i < j \leq k$, we identify x_0 and x_{k+1} with $-\infty$ and ∞, and therefore, $\delta_{\Delta_{0,j}}$ $\delta_{\Delta_{i-1,k+1}}$ are interpreted as being the zero function since the domain of f is $(0, \infty)$. Likewise $\delta_{\Delta_{i,i}} = \delta_0$ is interpreted as the zero function in the above equation.

Combining (3.11) with (3.9) implies that $\partial_s^2 f_{x,\xi}(s)$ is a sparse signal with support contained in $\mathcal{D}(x)$, and for $d \in \mathcal{D}(x)$,

$$\partial_s^2 f_{x,\xi}(d) = \sum_{\Delta_{i,j}=d} c_{i,j}(\xi)$$

as desired. □

The following example shows that, in general, the support of $\partial_s^2 f_{x,\xi}(s)$ may be a proper subset of $\mathcal{D}(x)$.

Example 1. *If* $p = 2$ *and*

$$x(t) = \delta_1(t) + \delta_2(t) + \delta_3(t) - \delta_4(t),$$

then $2 \in \mathcal{D}(x)$, *but*

$$\partial_s^2 f_\xi(2) = 0.$$

Proof. For this choice of x, there are two pairs (i, j) such that $\Delta_{i,j} = 2$, namely $(1, 3)$ and $(2, 4)$. Therefore, by Theorem 2,

$$\begin{aligned}\partial_s^2 f_\xi(2) = &\left(|\beta_{1,3}(\xi)|^2 + |\beta_{2,2}(\xi)|^2 - |\beta_{1,2}(\xi)|^2 - |\beta_{2,3}(\xi)|^2\right) + \\ &\left(|\beta_{2,4}(\xi)|^2 + |\beta_{3,3}(\xi)|^2 - |\beta_{2,3}(\xi)|^2 - |\beta_{3,4}(\xi)|^2\right).\end{aligned}$$

Inserting $(a_1, a_2, a_3, a_4) = (1, 1, 1, -1)$, $\Delta_{i,i+1} = 1$, and $\Delta_{i,i+2} = 2$ into (3.5) implies that

$$\begin{aligned}\partial_s^2 f_\xi(2) =& \left(|1 + e^{i\xi} + e^{2i\xi}|^2 + 1 - |1 + e^{i\xi}|^2 - |1 + e^{i\xi}|^2\right) + \\ & \left(|1 + e^{i\xi} - e^{2i\xi}|^2 + 1 - |1 + e^{i\xi}|^2 - |1 - e^{i\xi}|^2\right) \\ =& |1 + e^{i\xi} + e^{2i\xi}|^2 + |1 + e^{i\xi} - e^{2i\xi}|^2 + 2 - 3|1 + e^{i\xi}|^2 - |1 - e^{i\xi}|^2 \\ =& 0.\end{aligned}$$

The last inequality follows from repeatedly applying the the trigonometric identities $\sin^2(\theta) + \cos^2(\theta) = 1$ and $\cos(\theta) = \cos(2\theta)\cos(\theta) + \sin(2\theta)\sin(\theta)$. □

As illustrated by Example 1, the reason why $\partial_s^2 f_{x,\xi}(2)$ is equal to zero is because x is not collision free, which is defined as follows. A signal x is *collision free* if $|v_i - v_j| \neq |v_{i'} - v_{j'}|$ unless $(i, j) = (i', j')$. With this assumption, it is known [32] that the support set $\{v_j\}_{j=1}^k$ is determined (up to reflection and translation) by $\mathcal{D}(x)$. Specifically in the above example, there are two different pairs of points $(1, 3)$ and $(2, 4)$ in the support set of x that are both distance two from each other and $c_{1,3}(\xi) = -c_{2,4}(\xi)$. When x is collision free, this cancellation cannot occur, and as guaranteed by the following theorem the support set of $\partial_s^2 f_{x,\xi}(s)$ will exactly be $\mathcal{D}(x)$. Therefore, our measurements with sufficiently many scales and only one frequency completely characterize the support set of a sparse signal up to translation and reflection.

Theorem 3. *Assume that x is a collision-free k-sparse signal as in Equation* (3.3) *and that $p \geq 1$ is an integer. Then, for almost every ξ, $\partial_s^2 f_{x,\xi}$ is a sparse signal whose support is exactly equal to $\mathcal{D}(x)$.*

In order to prove Theorem 3, we will introduce a class of functions which we call Generalized Exponential Laurent Polynomials and state several lemmas about these functions. For the proof of the lemmas in this section, please see the appendix.

We call a function $q(\theta)$ a *Generalized Exponential Laurent Polynomial* if it can be written as

$$q(\theta) = \sum_{k=1}^{N} \alpha_k e^{i\gamma_k \theta}, \quad \theta \in [0, 2\pi) \tag{3.12}$$

where $N \geq 1$, $\alpha_k, \gamma_k \in \mathbb{R}$, $\alpha_k \neq 0$, and $\gamma_1 < \gamma_2 < \ldots < \gamma_N$. We let $\mathcal{E}$ be set of all such functions. For $q \in \mathcal{E}$, we refer to γ_N as the degree of q and let $\mathcal{E}(d)$ refer to the set of all $q \in \mathcal{E}$ with degree d. Note that we do not assume that the γ_k's are nonnegative or even rational. Therefore, the degree of q may be negative. We let $\mathcal{E}_0(d)$ denote the set of $q \in \mathcal{E}(d)$ such that $\gamma_1 \geq 0$.

Lemma 1. *Let $q, q' \in \mathcal{E}$,*

$$q(\theta) = \sum_{k=1}^{N} \alpha_k e^{i\gamma_k \theta} \text{ and } q'(\theta) = \sum_{k=1}^{N'} \beta_k e^{i\eta_k \theta}.$$

Then $q = q'$ if and only if $N = N'$ and for all $k = 1, \ldots, N$, $\alpha_k = \beta_k$ and $\gamma_k = \eta_k$.

Lemma 1 implies that if $q \in \mathcal{E}(d_1)$ and $q' \in \mathcal{E}(d_2)$ then

$$qq' \in \mathcal{E}(d_1 + d_2). \tag{3.13}$$

In particular, if $q \in \mathcal{E}_0(d)$

$$|q|^2 = q\bar{q} \in \mathcal{E}(d + 0) = \mathcal{E}(d). \tag{3.14}$$

Furthermore, if $d_2 \leq d_1$ then

$$(q + q') \in \mathcal{E}(d_1), \tag{3.15}$$

except, of course, if $d_1 = d_2$ and the lead coefficients of q and q' are negatives of one another.

Lemma 2. *Let p be an integer. For $i = 1, 2, 3, 4$, let $q_i \in \mathcal{E}_0(d_i)$ assume that $d_1 > d_2, d_3, d_4$. Then the set of points such that*

$$|q_1|^p + |q_2|^p = |q_3|^p + |q_4|^p \tag{3.16}$$

has measure zero.

Proof of Theorem 3. Under the assumption that $x(t)$ is collision free, it suffices to show that for all $i \leq j$, $c_{i,j}(\xi) \neq 0$ for almost every $\xi \in \mathbb{R}$, where as in (3.6) and (3.7),

$$c_{i,i+1}(\xi) = |\beta_{i,i+1}(\xi)|^p - |\beta_{i+1,i+1}(\xi)|^p - |\beta_{i,i}(\xi)|^p,$$

$$c_{i,j}(\xi) = |\beta_{i,j}(\xi)|^p + |\beta_{i+1,j-1}(\xi)|^p - |\beta_{i+1,j}(\xi)|^p - |\beta_{i,j-1}(\xi)|^p$$

for $j \geq i+2$, and

$$\beta_{i,j}(\xi) = \sum_{l=i}^{j} a_l e^{i\xi\Delta_{i,l}}.$$

Observe that $\beta_{i,j}$ is a generalized exponential Laurent polynomial of the form introduced in Equation 3.12, with degree $\Delta_{i,j}$. Therefore, when $j \geq i+2$, it follows from Lemma 2 that $c_{i,j}$ vanishes on a set of measure zero since $c_{i,j}(\xi) = 0$ implies

$$|\beta_{i,j}(\xi)|^p + |\beta_{i+1,j-1}(\xi)|^p = |\beta_{i+1,j}(\xi)|^p + |\beta_{i,j-1}(\xi)|^p.$$

In the case where $j = i+1$, we see that

$$c_{i,i+1}(\xi) = |a_i + a_{i+1}e^{-i\xi\Delta_{i,i+1}}|^p - |a_i|^p - |a_{i+1}|^p,$$

For any ξ such that $c_{i,i+1}(\xi) = 0$, we see that $\xi\Delta_{i,i+1}$ is a solution to

$$\left|a_i + a_{i+1}e^{i\theta}\right|^2 - (|a_i|^p + |a_{i+1}|^p)^{2/p} = 0.$$

Therefore, $c_{i,i+1}(\xi)$ vanishes on a set of measure zero because the left-hand side of the above equation is a trigonometric polynomial. □

Theorem 3 proves our measurements on a sparse signal x with one frequency and enough scales locate the positions of diracs, i.e., the support of x, up to translation and reflection. The strategy of selecting the scales is described in Section 3.4.

The following theorems shows with enough frequencies, our measurements also characterize the heights $\{a_i\}_i$ of x. Moreover, if the moments p is even, the number of frequencies needed can be reduced. The proofs of the following theorems are described in appendix.

Theorem 4. *Let $p \geq 1$ be an odd integer, let*

$$x(t) = \sum_{j=1}^{k} a_j \delta_{v_j}(t)$$

be a collision-free sparse signal, and let $\vec{a} = (a_1, \ldots, a_k)$. Let $\xi_1, \ldots, \xi_L$ be L frequencies chosen independently at random from some probability distribution which is absolutely continuous with respect to the Lebesgue measure for some $L \geq 4p + 2$. Then, with probability one, the following uniqueness result is true. Let

$$y(t) = \sum_{j=1}^{k} b_j \delta_{u_j}(t) \tag{3.17}$$

be any other sparse signal such that $\mathcal{D}(y) = \mathcal{D}(x)$, and let $\vec{b} = (b_1, \ldots, b_k)$. If $\partial_s^2 f_{x,\xi_\ell}(d) = \partial_s^2 f_{y,\xi_\ell}(d)$ for all $d \in \mathcal{D}(x)$ and all $1 \leq \ell \leq L$ and $\sum_{i=1}^{k} |b_i|^p = \sum_{i=1}^{k} |a_i|^p$, then $\vec{b} = \pm\vec{a}$ and therefore $y(t)$ is equivalent to $\pm x(t)$ up to translation and reflection.

Theorem 5. *Let $p = 2p'$ be an even integer, and let*

$$x(t) = \sum_{j=1}^{k} a_j \delta_{v_j}(t)$$

be a collision-free sparse signal, and let $\vec{a} = (a_1, \ldots, a_k)$. Let $\xi_1, \ldots, \xi_L$ be L frequencies chosen independently at random from some probability distribution which is absolutely continuous with respect to the Lebesgue measure for some $L \geq p + 2$. Then, with probability one, the following uniqueness result is true. Let

$$y(t) = \sum_{j=1}^{k} b_j \delta_{u_j}(t) \tag{3.18}$$

be any other sparse signal such that $\mathcal{D}(y) = \mathcal{D}(x)$, and let $\vec{b} = (b_1, \ldots, b_k)$. If $\partial_s^2 f_{x,\xi_\ell}(d) = \partial_s^2 f_{y,\xi_\ell}(d)$ for all $d \in \mathcal{D}(x)$ and all $1 \leq \ell \leq L$ and $\sum_{i=1}^{k} |b_i|^p = \sum_{i=1}^{k} |a_i|^p$, then $\vec{b} = \pm\vec{a}$ and therefore $y(t)$ is equivalent to $\pm x(t)$ up to translation and reflection.

3.4 Algorithm

Theorems 3, 4 and 5 shows our measurements characterize the support of a sparse signal x and the heights of the diracs. In this section, we describe our algorithm that uses such

measurements to recover the signal (up to translation and reflection). Let x be a sparse signal that is collision free,

$$x = \sum_{j=1}^{k} a_j \delta_{v_j}(t)$$

Let

$$f_{x,\xi}(s) = \|g_{s,\xi} * x\|_p^p$$

be the measurements. Let $\Lambda = \{(\xi_j, s_j)\}_j$ be the set of frequencies and scales we choose to define the gabor filters. The strategy to choose different scales is discussed in the next paragraph. With two signals x and x', we define the loss between them as:

$$\ell(x, x') = \sum_{(\xi,s)\in\Lambda} \left(f_{x,\xi}(s) - f_{x',\xi}(s)\right)^2$$

Given a target signal x, our goal is to find a new signal such that

$$x^* = \arg\min_{x'} \ell(x, x') \tag{3.19}$$

For simplicity, we choose $p = 1$ in our algorithm for synthesis.

Now we present a strategy of choosing the right set of scales. Let $\mathcal{D}(x) = \{\Delta_{ij} = |v_i - v_j|\}$ be the pairwise distance between the spikes of x. Since x is collision free, we know $\Delta_{ij} \neq \Delta_{i'j'}$ unless $(i, j) = (i', j')$ except for $\Delta_{ii} = 0, \forall i$. Therefore, there are $k(k-1)/2+1$ unique elements in $\mathcal{D}(x)$. Without loss of generality, we suppose $0 = d_0 < d_1 < \cdots < d_{k(k-1)/2}, d_i \in \mathcal{D}(x), \forall i$ in the following context. We also assume x is periodic in numerical experiments. Therefore $d_{k(k-1)/2} \leq \frac{n}{2}$. As stated in Theorem 3, $f_{x,\xi}(s)$ is a piecewise linear function of s and the singularities locate exactly at $\mathcal{D}(x)$. When choosing the scales to compute the scattering statistics, we need to ensure there is at least one scale between d_i and $d_{i+1}, \forall i$. Therefore, we compute the minimum pairwise distance of $\mathcal{D}(x)$, i.e.,

$$\min_{DD} = \min_{0\leq i<k(k-1)/2} |d_i - d_{i+1}|,$$

and sample scales every $\min_{DD}$ from 1 to $\frac{n}{2}$. We also insert three scales between $\frac{n}{2}$ to n to capture $f_{x,\xi}(s)$ in this interval. With this strategy, the scales are sampled and the relative positions of the Diracs are captured.

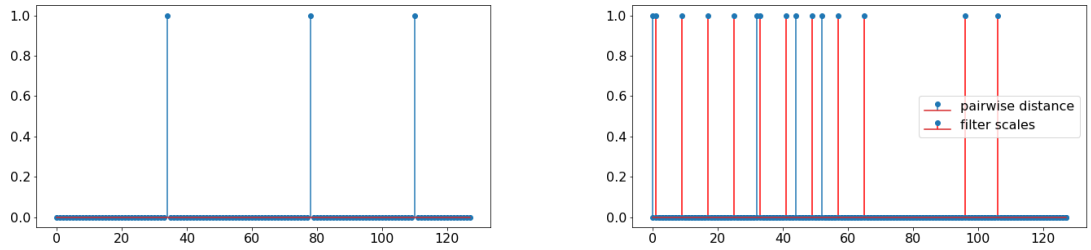


Figure 3.4 Choosing filter scales depending on the pairwise distance between signal spikes. **Left:** A sparse signal with $n = 128$, three spikes and $\min_{DD} = 8$. **Right:** The blue spikes show what are the pairwise distance. The orange spikes show what are the scales we choose.

Figure 3.4 gives one example of how we choose the scales. The left figure shows a signal x with three spikes, where the spikes are located at $u = 34, 78, 110$. This gives the pairwise distance set as $\mathcal{D}(x) = \{0, 32, 44, 52\}$ and $\min_{DD} = 8$. The blue spikes in right figure of Figure 3.4 show $\mathcal{D}$. With the logic described in the last paragraph, we sample the scales every 8 integers from 1 to 64, and sample three scales from 65 to 128, which is shown in the red spikes. As one can see, this strategy ensures there is at least one scale located between d_i and d_{i+1} for every i. In our experiment, we find two frequencies are enough to get the algorithm to converge. Therefore we randomly choose two frequencies when defining the filters. Our synthesis results are presented in the next section.

To solve the optimization problem in Equation 3.19, we initialize x' to be a random uniform noise and use gradient descent to update x' until the loss is reduced. However, in experiment, it is hard to match all target statistics together, therefore we propose a greedy method to randomly match each statistic. We first randomly permute the parameter set Λ. Then we choose the first $\{\xi_{i_1}, s_{i_1}\}$ to compute the target statistics and match it, i.e., use gradient descent to update x' until $\ell_1(x, x') = (f_{x,\xi_{i_1}}(s_{i_1}) - f_{x',\xi_{i_1}}(s_{i_1}))^2$ is reduced. Then we add in the next parameters $\{\xi_{i_2}, s_{i_2}\}$ and update x' to reduce the loss $\ell_1(x, x') = \sum_{j=1}^{2}(f_{x,\xi_{i_j}}(s_{i_j}) - f_{x',\xi_{i_j}}(s_{i_j}))^2$. We repeatedly add in the new parameters after the previous ones are matched until all parameters are added to promote convergence. After adding in all statistics and the process is converged, we reshuffle the parameter set Λ and repeat the

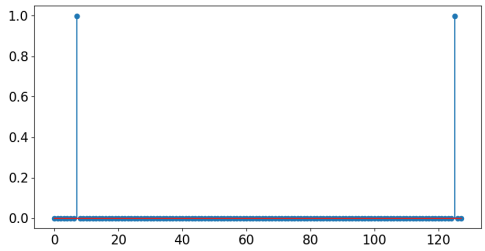
process. This process is repeated several times to avoid local minimum. The next section shows some of our numerical results.

3.5 Numerical results

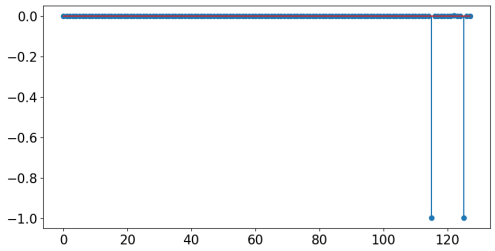
Figure 3.5 shows our synthesis results for sparse signals with different number of diracs and different $\min_{DD}$. Each row shows the result from one single test. The left column shows the original signals while the right column shows the corresponding synthesized signals. All signals are successfully synthesized up to translation, reflection or change of sign, proving the completeness of our selected measurements on sparse signals.

3.6 Conclusion

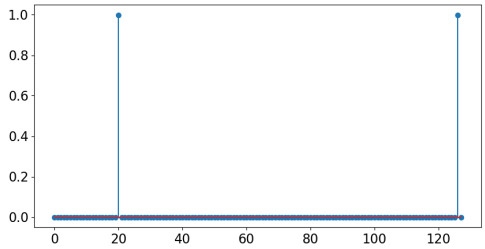
In this chapter, we proposed a two-layer hybrid scattering model that characterize the singularities of piecewise polynomials. We proved our second layer measurements can be used to capture the support and heights of sparse signals. We also designed an algorithm that use such measurements to recover sparse signals and successfully synthesized the original signals up to translation and reflection.



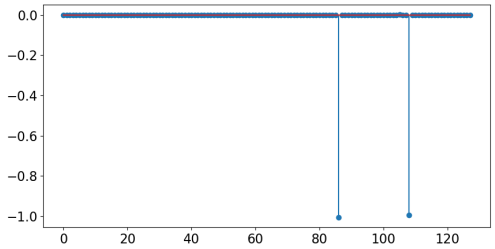
(a) Original $k = 2, \min_{DD} = 10$.



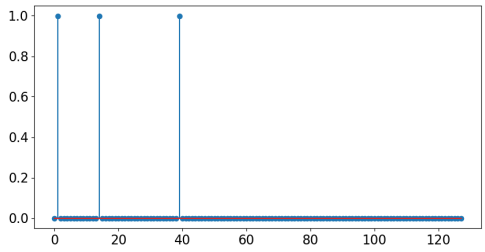
(b) Synthesized.



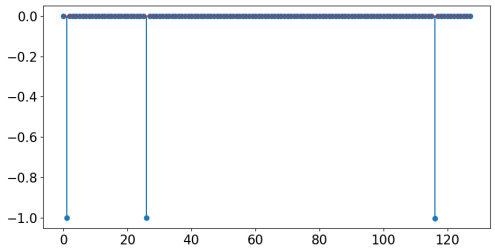
(c) Original $k = 2, \min_{DD} = 22$.



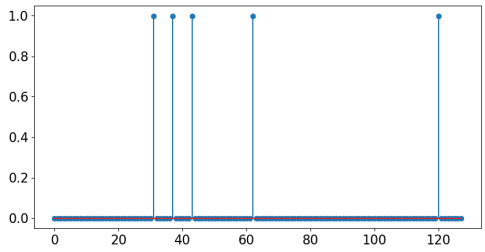
(d) Synthesized.



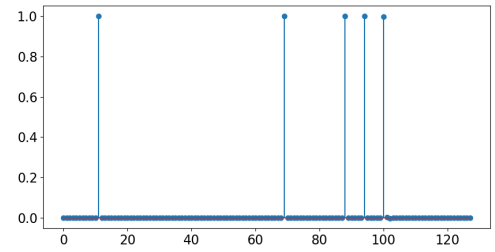
(e) Original $k = 3, \min_{DD} = 12$.



(f) Synthesized.



(g) Original $k = 5, \min_{DD} = 6$.



(h) Synthesized.

Figure 3.5 Sparse signals reconstructed up to a global reflection and translation. Each row shows the result from a single test. **Left column:** Original signals. **Right column:** Synthesized signals.

CHAPTER 4

SCATTERING STATISTICS OF GENERALIZED SPATIAL POISSON POINT PROCESSES

4.1 Introduction

In the last chapter we presented our work on analyzing the generalized scattering transform of deterministic sparse signals. In this chapter, we extend our work to stochastic processes. A lot of signal data in learning tasks can be modelled as a stochastic process, including texture discrimination [24], texture synthesis [33, 34], time series analysis (e.g., finance) [35], and wireless networks [36]. In many scenarios it is natural to model the signal data as the points of a (potentially complex) spatial point process. Furthermore, there are numerous other fields, including stochastic geometry [37], forestry [38], geoscience [39] and genetics [40], in which spatial point processes are used to model the underlying generating process of certain phenomena (e.g., earthquakes). CNNs have shown impressive results on these tasks. However, precise theoretical understanding of these results is lacking. Motivated by the existing empirical results, as well as the potential for numerous others in yet untapped research, we consider the capacity of CNNs to capture the statistical properties of spatial point processes.

Recall the definitions from Section 2.6. The scattering transform consists of an alternating cascade of linear wavelet transforms and complex modulus nonlinearities. In this chapter, we examine a generalized scattering transform that utilizes a broader class of filters, which are also the filters we used to analyze sparse signals in Chapter 3. But these filters still include wavelets as a special case. Our main focus is on scattering architectures constructed with filters that have small spatial support as is the case in most traditional CNNs.

Expected wavelet scattering moments for stochastic processes with stationary increments were introduced in [41], where it is shown that such moments capture important statistical

information of one-dimensional Poisson processes, fractional Brownian motion, α-stable Lévy processes, and a number of other stochastic processes. In this chapter, we extend the notion of scattering moments to our generalized architecture, and in the process of doing so, we recover many of the important small scale results in [41]. However, the main contributions contained here consist of new results for more general spatial point processes, including inhomogeneous Poisson point processes, which are not stationary and do not have stationary increments. The collection of expected scattering moments is a non-parametric model for these processes, which we prove captures important summary statistics of inhomogeneous, compound spatial Poisson point processes.

The remainder of this chapter is organized as follows. Expected scattering moments are introduced in Section 4.2. Sections 4.3 and 4.4 analyze the first-order and second-order scattering moments of inhomogeneous, compound spatial Poisson point processes. Section 4.5 compares the scattering moments of one-dimensional Poisson processes to two self-similar processes, fractional Brownian motions and the α-stable process. Section 4.6 presents stylized numerical examples to highlight certain aspects of the presented theory. A short conclusion is given in Section 4.8. All proofs are in the appendices, in addition to details on the numerical work.

4.2 Expected Scattering Moments for Random Signed Measures

Let $\psi \in \mathbf{L}^2(\mathbb{R})$ be a compactly supported mother wavelet with dilations $\psi_j(t) = 2^{-j}\psi(2^{-j}t)$, let and $X(t), t \in \mathbb{R}$, be a stochastic process with stationary increments defined on the real line. In [41], first-order wavelet scattering moments are defined as $SX(j) = \mathbb{E}[|\psi_j * X|]$, where the expectation does not depend on t since if $X(t)$ has stationary increments, then $X*\psi_j(t)$ is stationary so long as ψ_j is a wavelet. Much of the mathematical analysis of wavelet scattering moments relies on the fact that they can be rewritten as $SX(j) = \mathbb{E}[|\overline{\psi}_j * dX|]$, where $\overline{\psi}_j$ is the primitive of ψ_j, i.e., $d\overline{\psi}_j = \psi_j$. This reformulation motivates us to define scattering moments as the integration of a filter, which is not necessarily a wavelet, against

a random signed measure $Y(dt)$.

To that end, let $w \in \mathbf{L}^2(\mathbb{R}^d)$ be a continuous window function with support contained in the unit cube $[0,1]^d$. Denote by $w_s(t) = w\left(\frac{t}{s}\right)$ the dilation of w, supported on the cube $Q_s = [0,s]^d$, and set $g_\gamma(t)$ to be the Gabor-type filter with scale s and central frequency $\xi \in \mathbb{R}^d$,

$$g_\gamma(t) = w_s(t)e^{i\xi\cdot t}, \quad \gamma = (s,\xi), \;\; t \in \mathbb{R}^d\,. \tag{4.1}$$

Recall that this is the same type of filter we defined in Section 3.2 from Chapter 3.

For a random signed measure $Y(dt)$ we define the first-order $\mathbf{L}^p$ scattering moments, $1 \le p < \infty$, at location t as

$$S[\gamma,p]Y(t) \coloneqq \mathbb{E}[|g_\gamma * Y(t)|^p] \coloneqq \mathbb{E}\left[\left|\int_{\mathbb{R}^d} g_\gamma(t-u)\,Y(du)\right|^p\right]. \tag{4.2}$$

Note there is no assumption on the stationarity of $Y(du)$, which is why these scattering moments a priori depend on t. We define invariant (i.e., location independent) first-order scattering coefficients of Y by

$$SY(\gamma,p) = \lim_{R\to\infty} \frac{1}{(2R)^d} \int_{|t_i|\le R} \mathbb{E}[|g_\gamma * Y(t)|^p]\,dt\,, \tag{4.3}$$

if the limit on the right hand side exists.

We call Y a periodic measure if there exists $T > 0$ such that for any Borel set B, the family of sets $B + Te_i = \{b + Te_i : b \in B\}$ satisfies

$$Y(B) \overset{d}{=} Y(B + Te_i), \quad \forall\, 1 \le i \le d\,,$$

where $\{e_i\}_{i\le d}$ is the standard orthonormal basis for $\mathbb{R}^d$. In this case one can verify, by approximating g_γ with simple functions, that $(g_\gamma * Y)(t + Te_i) \overset{d}{=} (g_\gamma * Y)(t)$, and therefore

$$S[\gamma,p]Y(t + Te_i) = S[\gamma,p]Y(t), \quad \forall\, t \in \mathbb{R}^d\,.$$

Thus the limit in (4.3) exists, and

$$SY(\gamma,p) = \frac{1}{T^d} \int_{Q_T} \mathbb{E}[|g_\gamma * Y(t)|^p]\,dt\,. \tag{4.4}$$

Note that in the special case when the distribution of $Y(B)$ depends only on the Lebesgue measure of B, then $S[\gamma, p]Y(t)$ is independent of t and the above limit (4.3) exists with $SY(\gamma, p) = S[\gamma, p]Y(t)$ for any $t \in \mathbb{R}^d$.

First-order scattering moments compute summary statistics of the measure Y based upon its responses against the filters g_γ. Higher-order summary statistics can be obtained by computing first-order scattering moments for larger powers p, or by cascading lower-order modulus nonlinearities as in a CNN. This leads us to define second-order scattering moments by

$$S[\gamma, p, \gamma', p']Y(t) = \mathbb{E}\left[||g_\gamma * Y|^p * g_{\gamma'}(t)|^{p'}\right] .$$

First-order invariant scattering moments collapse additional information by aggregating the variations of the random measure Y, which removes information related to the intermittency of Y. Second-order invariant scattering moments augment first-order scattering moments by iterating on the cascade of linear filtering operations and nonlinear $|\cdot|^p$ operators, thus recovering some of this lost information. They are defined (assuming the limit on the right exists) by

$$SY(\gamma, p, \gamma', p') = \lim_{R\to\infty} \frac{1}{(2R)^d} \int_{|t_i|\leq R} \mathbb{E}\left[||g_\gamma * Y|^p * g_{\gamma'}(t)|^{p'}\right] \, dt \, .$$

The collection of (invariant) scattering moments is a set of non-parametric statistical measurements of the random measure Y. In the following sections, we analyze these moments for arbitrary frequencies ξ and small scales s, thus allowing the filters g_γ to serve as a model for the learned filters in CNNs. In particular, we will analyze the asymptotic behavior of the scattering moments as the scale parameter s decreases to zero.

4.3 First-Order Scattering Moments of Generalized Poisson Processes

We consider the case where $Y(dt)$ is an inhomogeneous, compound spatial Poisson point process. Such processes generalize ordinary Poisson point processes by incorporating variable charges (heights) at the points of the process and a non-uniform intensity for the locations

of the points. They thus provide a flexible family of point processes that can be used to model many different phenomena. In this section we consider first-order scattering moments of these generalized Poisson processes. In Sec. 4.3.1 we provide a review of such processes, and in Sec. 4.3.2 we show that first-order scattering moments capture a significant amount of statistical information related these processes, particularly when using very localized filters.

4.3.1 Inhomogeneous, Compound Spatial Poisson Point Processes

Let $\lambda(t)$ be a continuous function on $\mathbb{R}^d$ such that

$$0 < \lambda_{\min} := \inf_t \lambda(t) \leq \|\lambda\|_\infty < \infty \,, \tag{4.5}$$

and let $N(dt)$ be an inhomogeneous Poisson point process with intensity function $\lambda(t)$. That is,

$$N(dt) = \sum_{j=1}^{\infty} \delta_{t_j}(dt)$$

is a random measure, concentrated on a countable set of points $\{t_j\}_{j=1}^{\infty}$, such that for all Borel sets $B \subset \mathbb{R}^d$, the number of points of N in B, denoted $N(B)$, is a Poisson random variable with parameter

$$\Lambda(B) = \int_B \lambda(t)\, dt \,, \tag{4.6}$$

i.e.,

$$\mathbb{P}[N(B) = n] = e^{-\Lambda(B)} \frac{(\Lambda(B))^n}{n!} \,,$$

and $N(B)$ is independent of $N(B')$ for all sets B' that do not intersect B. Now let $(A_j)_{j=1}^{\infty}$ be a sequence of i.i.d. random variables independent of N, and let $Y(dt)$ be the random signed measure that gives charge A_j to each point t_j of N, i.e.,

$$Y(dt) = \sum_{j=1}^{\infty} A_j \delta_{t_j}(dt) \,. \tag{4.7}$$

We refer to $Y(dt)$ as an inhomogeneous, compound Poisson point process. For a Borel set $B \subset \mathbb{R}^d$, $Y(B)$ has a compound Poisson distribution and we will (in a slight abuse of

notation) write

$$Y(B) = \sum_{j=1}^{N(B)} A_j \, .$$

In many of our proofs, it will be convenient to consider the random measure $|Y|^p(dt)$ defined formally by

$$|Y|^p(dt) \coloneqq \sum_{j=1}^{\infty} |A_j|^p \delta_{t_j}(dt).$$

For a further overview of these processes, and closely related marked point processes, we refer the reader to Section 6.4 of [42].

4.3.2 First-order Scattering Asymptotics

Computing the convolution of g_γ with $Y(dt)$ gives

$$(g_\gamma * Y)(t) = \int_{\mathbb{R}^d} g_\gamma(t-u)\, Y(du) = \sum_{j=1}^{\infty} A_j g_\gamma(t - t_j) \, ,$$

which can be interpreted as a waveform g_γ emitting from each location t_j. Invariant scattering moments aggregate the random interference patterns in $|g_\gamma * Y|$. The results below show that the expectation of these interferences, for small scale waveforms g_γ, encode important statistical information related to the point process.

For notational convenience, we let

$$\Lambda_s(t) := \Lambda\left([t-s,t]^d\right) = \int_{[t-s,t]^d} \lambda(u)\, du$$

denote the expected number of points of N in the support of $g_\gamma(t - \cdot)$. If $\lambda(t)$ is a periodic function in each coordinate with period T, then $\Lambda_s(t) = \Lambda_s(t + Te_i)$ for $1 \leq i \leq d$ and therefore, the invariant scattering coefficients of Y may be defined as in (4.4).

Theorem 6. *Let $1 \leq p < \infty$ and suppose that $Y(dt)$ is an inhomogeneous, compound Poisson point process as defined above, where $(A_j)_{j=1}^{\infty}$ is an i.i.d. sequence of random variables, $\mathbb{E}[|A_1|^p] < \infty$ and $\lambda(t)$ is a continuous intensity function satisfying (4.5). Then for every*

$t \in \mathbb{R}^d$, *every* $\gamma = (s, \xi)$ *such that* $s^d \|\lambda\|_\infty < 1$, *and for every* $m \geq 1$.

$$S[\gamma, p]Y(t) = \sum_{k=1}^{m} e^{-\Lambda_s(t)} \frac{(\Lambda_s(t))^k}{k!} \mathbb{E}\left[\left|\sum_{j=1}^{k} A_j w(V_j) e^{is\xi \cdot V_j}\right|^p\right] + \epsilon(m, s, \xi, t)\,, \tag{4.8}$$

where the error term $\epsilon(m, s, \xi, t)$ *satisfies*

$$|\epsilon(m, s, \xi, t)| \leq C_{m,p} \frac{\|\lambda\|_\infty}{\lambda_{\min}} \|w\|_p^p \mathbb{E}[|A_1|^p] \|\lambda\|_\infty^{m+1} s^{d(m+1)} \tag{4.9}$$

and $V_1, V_2, \ldots$ *is an i.i.d. sequence of random variables, independent of the* A_j*, taking values in the unit cube* Q_1 *and with density*

$$p_V(v) = \frac{s^d}{\Lambda_s(t)} \lambda(t - vs)\,, \quad v \in Q_1\,.$$

The main idea of the proof of Theorem 6 is to condition on $N\left([t-s, t]^d\right)$, which is the number of points in the support of g_γ, and to use the fact that

$$\mathbb{P}\left[N\left([t-s, t]^d\right) > m\right] = O\left(\left(s^d \|\lambda\|_\infty\right)^{m+1}\right), \quad \forall\, s^d \|\lambda\|_\infty < 1\,.$$

Theorem 6 shows that even at small scales the scattering moments $S[\gamma, p]Y(t)$ depend upon higher-order information related to the distribution of the points, encapsulated by the term $(\Lambda_s(t))^k$, regardless of the scattering moment p. However, the influence of the higher-order terms diminishes rapidly as the scale of the filter shrinks, which is indicated by the bound (4.9) on the error function. Theorem 6 also shows that p^{th} scattering moments depend on the p^{th} moments of the charges, $(A_j)_{j=1}^\infty$. The next result uses Theorem 6 to examine the behavior of scattering moments for small filters in the asymptotic regime as the scale $s \to 0$.

Theorem 7. *Let* $1 \leq p < \infty$, *and suppose that* $Y(dt)$ *is an inhomogeneous, compound Poisson point process satisfying the same assumptions as in Theorem 6. Let* $\gamma_k = (s_k, \xi_k)$ *be a sequence of scale and frequency pairs such that* $\lim_{k\to\infty} s_k = 0$. *Then*

$$\lim_{k\to\infty} \frac{S[\gamma_k, p]Y(t)}{s_k^d} = \lambda(t) \mathbb{E}[|A_1|^p] \|w\|_p^p\,. \tag{4.10}$$

Furthermore, if $\lambda(t)$ *is periodic with period* T *along each coordinate, then*

$$\lim_{k\to\infty} \frac{SY(\gamma_k, p)}{s_k^d} = m_1(\lambda) \mathbb{E}[|A_1|^p] \|w\|_p^p, \quad \textit{where} \quad m_1(\lambda) = \frac{1}{T^d} \int_{Q_T} \lambda(t)\, dt\,. \tag{4.11}$$

Theorem 7 is proved via asymptotic analysis of the $m = 1$ case of Theorem 6. The key to the proof, which is similar to the technique used to prove Theorem 2.1 of [41], is that in a small cube $[t - s, t]^d$ there is at most one point of N with overwhelming probability. Therefore, when s is very small, with very high probability, $|g_\gamma * Y|^p(t) = (|g_\gamma|^p * |Y|^p)(t)$.

This theorem shows that for small scales the scattering moments $S[\gamma, p]Y(t)$ encode the intensity function $\lambda(t)$, up to factors depending upon the summary statistics of the charges $(A_j)_{j=1}^\infty$ and the window w. Recall that $\Lambda(B)$, defined in (4.6), determines the concentration of events within the set B. Thus even a one-layer location dependent scattering network yields considerable information regarding the underlying data generation, at least in the case of inhomogeneous Poisson processes. However, it is often the case, e.g., [43], that invariant statistics are utilized. In this case (4.11) shows that invariant scattering statistics mix the mean of $\lambda(t)$ and the p^{th} moment of the charge magnitudes. However, we can decouple these statistics as we now explain.

As a special case, Theorem 7 proves that for non-compound inhomogeneous Poisson processes (i.e., $A_j = 1$ for all $j \geq 1$), small scale scattering moments recover $\lambda(t)$ or $m_1(\lambda)$, depending on whether one computes invariant or time-dependent scattering moments. For compound processes, we can add an additional nonlinearity, namely the signum function sgn, which when applied to the Poisson point process in (4.7) yields,

$$\overline{Y}(dt) = \text{sgn}[Y(dt)] = \sum_{j=1}^{\infty} \delta_{t_j}(dt)\,.$$

Thus computing $S\overline{Y}(\gamma, p)$ and the ratio $SY(\gamma, p)/S\overline{Y}(\gamma, p)$ at small scales decouples the mean of $\lambda(t)$ from the p^{th} moment of $|A_1|$. We remark that the signum function is a simple perceptron and is closely related to the sigmoid nonlinearity, which is used in many neural networks. We further remark that the computation of $S\overline{Y}$ constitutes a small two-layer network, consisting of the nonlinear sgn function, the linear filtering by the collection of filters g_γ, the nonlinear p^{th} modulus $|\cdot|^p$, and the linear integration operator.

If $Y(dt)$ is a homogeneous Poisson process, then $\lambda(t)$ is constant, meaning that (4.10) and

(4.11) are equivalent. In the case of ordinary (non-compound) Poisson processes, Theorem 7 recovers the constant intensity. For periodic $\lambda(t)$ and invariant scattering moments, the effect of higher-order moments of $\lambda(t)$ can be partially isolated by considering higher-order expansions (e.g., $m > 1$) in (4.8). The next theorem considers second-order expansions and illustrates their dependence on the second moment of $\lambda(t)$.

Theorem 8. *Let $1 \leq p < \infty$, and suppose $Y(dt)$ is an inhomogeneous, compound Poisson point process satisfying the same assumptions as in Theorem 6. If $\lambda(t)$ is periodic with period T in each coordinate, and if $(\gamma_k)_{k\geq 1} = (s_k, \xi_k)_{k\geq 1}$, is a sequence of scale and frequency pairs such that $\lim_{k\to\infty} s_k = 0$ and $\lim_{k\to\infty} s_k\xi_k = L \in \mathbb{R}^d$, then*

$$\lim_{k\to\infty} \left(\frac{SY(\gamma_k, p)}{s_k^{2d}\mathbb{E}[|A_1|^p]\mathbb{E}[|V_k|^p]} - \frac{1}{T^d}\int_{Q_T} \frac{\Lambda_{s_k}(t)}{s_k^{2d}}\, dt \right)$$
$$= m_2(\lambda) \left(\frac{\mathbb{E}\left[\left|A_1 w(U_1) e^{iL\cdot U_1} + A_2 w(U_2) e^{iL\cdot U_2}\right|^p\right]}{2\|w\|_p^p \mathbb{E}[|A_1|^p]} \right), \tag{4.12}$$

where $m_2(\lambda) = T^{-d}\int_{Q_T} \lambda(t)^2\, dt$; U_1, U_2 are independent uniform random variables on Q_1; and $(V_k)_{k\geq 1}$ is a sequence of random variables independent of the A_j taking values in the unit cube Q_1 and with respective densities,

$$p_{V_k}(v) = \frac{s_k^d}{\Lambda_{s_k}(t)} \lambda(t - vs_k), \quad v \in Q_1 .$$

We first remark that the scale normalization on the left hand side of (4.12) is s^{-2d}, compared to a normalization of s^{-d} in Theorem 7. Thus even though (4.12) is written as a small scale limit, intuitively Theorem 8 is capturing information at moderately small scales that are larger than the scales considered in Theorem 7. This is further indicated by the term multiplied against $m_2(\lambda)$ on the right hand side of (4.12), which depends on two points of the process (as indicated by the presence of two charges A_1 and A_2).

Unlike Theorem 7, which gives a way to compute $m_1(\lambda)$, Theorem 8 does not allow one to compute $m_2(\lambda)$ since it would require knowledge of $\Lambda_{s_k}(t)$ in addition to the distribution from which the charges $(A_j)_{j=1}^{\infty}$ are drawn. However, Theorem 8 does show that at moderately small scales the invariant scattering coefficients depend non-trivially on the second

moment of $\lambda(t)$. This behavior at moderately small scales can be used to distinguish between, for example, an inhomogeneous Poisson point process with intensity function $\lambda(t)$ and a homogeneous Poisson point process with constant intensity $\lambda_0 = m_1(\lambda)$, whereas Theorem 7 indicates that at very small scales the two processes will have the same invariant scattering moments.

4.4 Second-Order Scattering Moments of Generalized Poisson Processes

We prove that second-order scattering moments, in the small scale regime, encode higher-order moment information about the charges $(A_j)_{j=1}^{\infty}$.

Theorem 9. *Let $1 \leq p, p' < \infty$ and $q = pp'$. Suppose that $Y(dt)$ is an inhomogeneous Poisson point process satisfying the same assumptions as in Theorem 6 as well as the additional assumption that $\mathbb{E}|A_1|^q < \infty$. Let $\gamma_k = (s_k, \xi_k)$ and $\gamma_k' = (s_k', \xi_k')$ be two sequences of scale and frequency pairs such that $s_k' = cs_k$ for some fixed constant $c > 0$ and $\lim_{k\to\infty} s_k \xi_k = L \in \mathbb{R}^d$. Then,*

$$\lim_{k\to\infty} \frac{S[\gamma_k, p, \gamma_k', p']Y(t)}{s_k^{d(p'+1)}} = K\lambda(t)\mathbb{E}[|A_1|^q]\,, \tag{4.13}$$

where

$$K \coloneqq \left\| g_{c,L/c} * |g_{1,0}|^p \right\|_{p'}^{p'}\,,$$

is a constant depending on p, p', c, L, and w. Furthermore, if $\lambda(t)$ is periodic with period T along each coordinate, then

$$\lim_{k\to\infty} \frac{SY(\gamma_k, p, \gamma_k', p')}{s_k^{d(p'+1)}} = Km_1(\lambda)\mathbb{E}[|A_1|^q]\,. \tag{4.14}$$

Note that the scaling factor $s^{-d(p'+1)}$ depends on p' but not p. Intuitively this corresponds to the behavior $\| |g_{\gamma_k}|^p * g_{\gamma_k'} \|_{p'}^{p'} \approx s_k^{d(p'+1)}$ as $s_k \to 0$. Theorem 9 proves that second-order scattering moments capture higher-order moments of the charges $(A_j)_{j=1}^{\infty}$ via two pairs of lower-order filtering and modulus operators. If $p, p' > 1$, then $q = pp'$ will be larger than

either p or p' and the result above will give us information about the higher order moment $\mathbb{E}|A_1|^q$.

It is also useful to consider the $p = 1$ case. Indeed, in Sec. 4.5 below it is shown that first-order invariant scattering moments can distinguish Poisson point processes from fractional Brownian motion and α-stable processes, if $p = 1$, but may fail to do so for larger values of p. However, Theorem 7 shows that first-order invariant scattering moments for $p = 1$ will not be able to distinguish between the various different types of Poisson point processes with a one-layer network at very small scales. Theorem 9 shows that a second-order calculation that augments the first-order calculation with $p = 1$ and $p' > 1$, will capture a higher-order moment of the charges $(A_j)_{j=1}^{\infty}$.

4.5 Poisson Point Processes Compared to Self Similar Processes

For one-dimensional processes (i.e., $d = 1$), we show that first-order invariant scattering moments can distinguish between inhomogeneous, compound Poisson point processes and certain self-similar processes. In particular, we show that if $X(t)$ is either an α-stable process or a fractional Brownian motion (fBM), then the corresponding first-order scattering moments will have different asymptotic behavior for infinitesimal scales than in the case of a Poisson point process. Similar results were initially reported in [41]; here we generalize those results to the non-wavelet filters g_γ defined in (4.1) and for general p^{th} scattering moments, and further clarify their usefulness in the context of the new results presented in Sec. 4.3 and Sec. 4.4. As in [41], the proof will be based on the scaling relationships of these processes and therefore will not be able to distinguish between α-stable processes and fBM[1]. The key will be proving a lemma that says if a stochastic process X has a scaling relation, then that scaling relation is inherited by integrals of deterministic functions against dX.

More precisely, for a stochastic process $X(t)$, $t \in \mathbb{R}$, we consider the convolution of the

[1]We note that [41] proves that second-order scattering moments defined with wavelet filters do distinguish between α-stable processes and fBM, but we do not pursue this direction in this project as we are concerned primarily with point processes.

filter g_γ with the noise dX defined by

$$g_\gamma * dX(t) = \int_{\mathbb{R}} g_\gamma(t-u)\, dX(u)\,,$$

and define (in a slight abuse of notation) the first-order scattering moments at time t by

$$S[\gamma, p]X(t) = \mathbb{E}[|g_\gamma * dX(t)|^p]\,. \tag{4.15}$$

In the case where $X(t)$ is a compound, inhomogeneous Poisson (counting) process, $Y = dX$ will be a compound Poisson random measure and the scattering moments defined in (4.15) will coincide with the first-order scattering moments defined in (4.2).

The following two theorems analyze the small scale first-order scattering moments when X is either an α-stable process, for $1 < \alpha \leq 2$, or fractional Brownian motion. Thus dX will be stable Lévy noise or fractional Gaussian noise, respectively. These results show that the asymptotic decay of the corresponding scattering moments is guaranteed to differ from Poisson point processes, in the case $p = 1$. We also note that both α-stable processes and fBM have stationary increments; therefore the scattering moments do not depend on time and

$$S[\gamma, p]X(t) = SX(\gamma, p) = \lim_{R\to\infty} \frac{1}{2R} \int_{|u|\leq R} \mathbb{E}[|g_\gamma * dX(u)|^p]\, du\,, \quad \forall\, t \in \mathbb{R}\,.$$

Theorem 10. *Let $1 \leq p < \infty$ and suppose $X(t)$ is a symmetric α-stable process for some $p < \alpha \leq 2$. Let $\gamma_k = (s_k, \xi_k)$ be a sequence of scale and frequency pairs such that $\lim_{k\to\infty} s_k = 0$ and $\lim_{k\to\infty} s_k\xi_k = L \in \mathbb{R}$. Then,*

$$\lim_{k\to\infty} \frac{SX(\gamma_k, p)}{s_k^{p/\alpha}} = \mathbb{E}\left[\left|\int_0^1 w(u)e^{iLu}\, dX(u)\right|^p\right]\,.$$

Theorem 11. *Let $1 \leq p < \infty$, suppose $X(t)$ is a fractional Brownian motion with Hurst parameter $0 < H < 1$. Assume that the window function w has bounded variation on $[0, 1]$, and let $\gamma_k = (s_k, \xi_k)$ be a sequence of scale and frequency pairs such that $\lim_{k\to\infty} s_k = 0$ and $\lim_{k\to\infty} s_k\xi_k = L \in \mathbb{R}$. Then,*

$$\lim_{k\to\infty} \frac{SX(\gamma_k, p)}{s_k^{pH}} = \mathbb{E}\left[\left|\int_0^1 w(u)e^{iLu}\, dX(u)\right|^p\right]\,.$$

The key to proving Theorem 10 and Theorem 11 is the lemma stated in Appendix 7.3, which shows that if $X(t)$ is a self-similar process, then, then stochastic integrals against dX satisfy an identity corresponding to the scaling relation of $X(t)$.

Together, these two theorems indicate that first-order invariant scattering moments distinguish inhomogeneous, compound Poisson processes from both α-stable processes and fractional Brownian motion except in the cases where $p = \alpha$ or $p = 1/H$. In particular, if X is a Brownian motion, then SX will distinguish X from a Poisson point process except in the case that $p = 2$. For this reason, it appears that $p = 1$ is the best choice of the parameter p for the purposes of distinguishing a Poisson point process from a self-similar process. In the case of a multi-layer network, it is advisable to set $p = 1$. Larger values of p' in the second layer can then allow us to determine the higher moments of the arrival heights $(A_j)_{j=1}^{\infty}$.

4.6 Numerical Illustrations

We carry out several experiments to numerically validate the previously stated results and to illustrate their capacity for distinguishing between different types of random processes. In all of the experirments below, we will hold the frequency ξ constant while we let the scale s decrease to zero.

4.6.1 Homogeneous, compound Poisson point processes with the same intensities

We generated three different types of homogeneous compound Poisson point processes, all with the same intensity $\lambda(t) \equiv \lambda_0 = 0.01$. The three point processes are Y_1 (ordinary), Y_2 (Gaussian), and Y_3 (Rademacher), where the charges are sampled according to $(A_{1,j})_{j=1}^{\infty} \equiv 1$, $(A_{2,j})_{j=1}^{\infty} \sim \mathcal{N}(0, \sqrt{\pi/2})$, and $(A_{3,j})_{j=1}^{\infty} \sim$ Rademacher distribution (i.e., ± 1 with equal probability). The charges of the three signals have the same first moment $\mathbb{E}[|A_{i,j}|] = 1$ and different second moment with $\mathbb{E}[|A_{1,j}|^2] = \mathbb{E}[|A_{3,j}|^2] = 1$ and $\mathbb{E}[|A_{2,j}|^2] = \frac{\pi}{2}$. Theorem 7 thus predicts that $p = 1$ invariant first-order scattering moments will not be able to

distinguish between the three processes, but $p = 2$ invariant first-order scattering moments will distinguish the Gaussian Poisson point process from the other two. Figure 4.1 illustrates this point by plotting the normalized invariant scattering moments for $p = 1$ and $p = 2$.

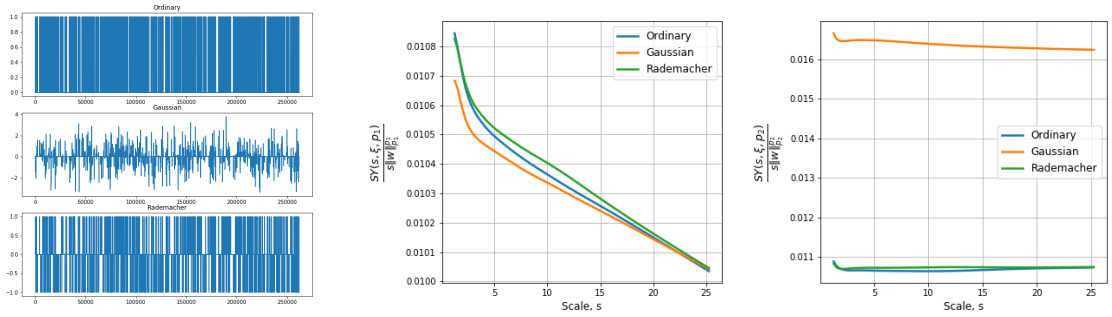


Figure 4.1 First-order invariant scattering moments for three types of homogeneous compound Poisson point processes with the same intensity λ_0. **Left:** *Top:* ordinary Poisson point process. *Middle:* Gaussian compound Poisson point process with normally distributed charges. *Bottom:* Rademacher compound Poisson point process with charges drawn from the Rademacher distribution. **Middle:** Normalized invariant scattering moments $SY(s,\xi,1)/s\|w\|_1$ (i.e., $p = 1$), which all converge to 0.01 as $s \to 0$ (up to numerical errors) since $\lambda_0 \mathbb{E}[|A_1|]$ is the same for all three point processes. **Right:** Normalized invariant scattering moments $SY(s,\xi,2)/s\|w\|_2^2$ (i.e., $p = 2$). In this case the ordinary Poisson point process and the Rademacher Poisson point process still converge to the same value as $s \to 0$ since $\mathbb{E}[|A_1|^2] = 1$ for both of them. However, the Gaussian Poisson point process converges to a different value since $\mathbb{E}[|A_1|^2] = \pi/2$ for this process.

4.6.2 Homogeneous, compound Poisson point processes with different intensities and charges

We consider two homogeneous, compound Poisson point processes with different intensities and different charge distributions, but which nevertheless have the same first-order invariant scattering moments with $p = 1$ due to the mixing of intensity and charge information in (4.11). The first compound Poisson point process has constant intensity $\lambda_1 = 0.01$ and charges $A_{1,j} \sim \mathcal{N}(0, 1)$, whereas the second has intensity $\lambda_2 = 0.01/\sqrt{2}$ and $A_{2,j} \sim \mathcal{N}(0, 2)$. In this way, $\lambda_1 \mathbb{E}[|A_{1,j}|] = \lambda_2 \mathbb{E}[|A_{2,j}|] = 0.01 \cdot \sqrt{2/\pi} \approx 0.008$, but $\lambda_1 \mathbb{E}[|A_{1,j}|^2] = 0.01$ and $\lambda_2 \mathbb{E}[|A_{2,j}|^2] = 0.01 \cdot \sqrt{2} \approx 0.014$. Figure 4.2 plots the normalized invariant scattering moments for $p = 1$ and $p = 2$.

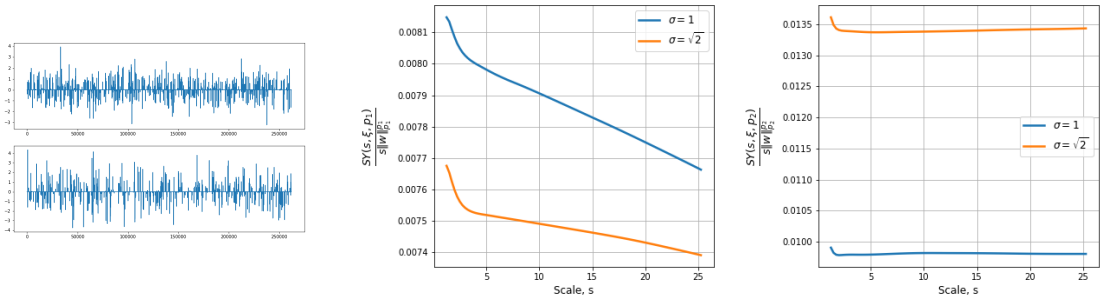


Figure 4.2 First-order invariant scattering moments for two homogeneous, Gaussian compound Poisson point processes with different intensity and variance. **Left:** *Top:* Homogeneous compound Poisson point process with intensity $\lambda_1 = 0.01$ and charges $A_{1,j} \sim \mathcal{N}(0,1)$. *Bottom:* Homogeneous compound Poisson point process with intensity $\lambda_2 = {}^{0.01}/\sqrt{2}$ and charges $A_{2,j} \sim \mathcal{N}(0,2)$. The two point processes are difficult to distinguish, visually. **Middle:** Normalized invariant scattering moments ${}^{SY(s,\xi,1)}/_{s\|w\|_1}$ (i.e., $p = 1$), which both converge to approximately 0.08 up to numerical error, thus indicating that these moments cannot distinguish the two processes. **Right:** Normalized invariant scattering moments ${}^{SY(s,\xi,2)}/_{s\|w\|_2^2}$ (i.e., $p = 2$). The two process are distinguished as $s \to 0$ since the values $\lambda_1 \mathbb{E}[|A_{1,j}|^2] = 0.01$ and $\lambda_2 \mathbb{E}[|A_{2,j}|^2] \approx 0.014$ differ by a significant margin.

4.6.3 Inhomogeneous, non-compound Poisson point processes

We also consider inhomogeneous Poisson point processes. We use the intensity function $\lambda(t) = 0.01(1 + 0.5\sin({}^{2\pi t}/_{N}))$ to generate inhomogeneous process. To estimate $S[\gamma, p]Y(t)$, we average the modulus of the scattering transform at time t over 1000 realizations. Figure 4.3 plots the scattering moments for inhomogeneous process at different time.

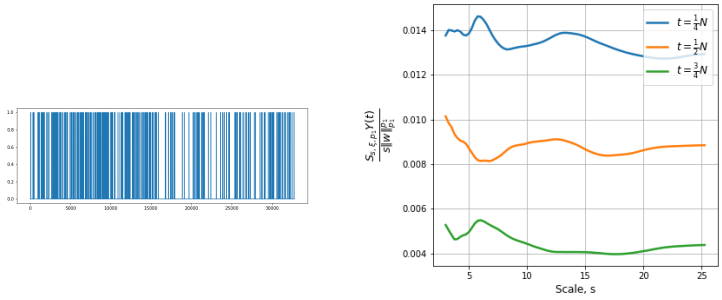


Figure 4.3 First-order invariant scattering moments for inhomogeneous non-compound Poisson point processes. **Left:** Inhomogeneous non-compound Poisson point process with intensity $\lambda(t) = 0.01(1 + 0.5\sin({}^{2\pi t}/_{N}))$. **Right:** Scattering moments ${}^{S[\gamma,p]Y(t)}/_{s\|w\|_p^p}$ for inhomogeneous non-compound Poisson point process at $t_1 = {}^{N}/_{4}$, $t_2 = {}^{N}/_{2}$, $t_3 = {}^{3N}/_{4}$. Note that $\lambda(t_1) = 0.015$, $\lambda(t_2) = 0.01$, $\lambda(t_3) = 0.005$. The plots show that for inhomogeneous process, scattering coefficients at time t converges to the intensity at that time.

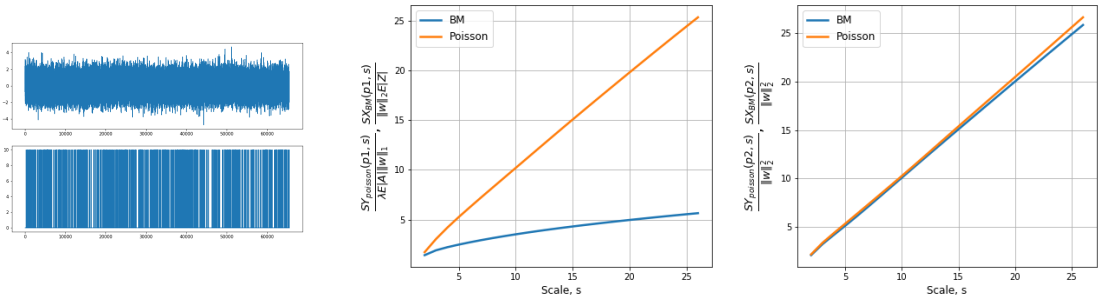


Figure 4.4 First-order invariant scattering moments for Brownian motion and Poisson point process. **Left:** *Top:* Brownian motion with Hurst parameter $H = 1/2$. *Bottom:* Ordinary Poisson point process. **Middle:** Normalized scattering moments for Brownian Motion $\left({}^{SX_{BM}(s,\xi,1)}/_{\|w\|_2 E|Z|}\right)$ and Poisson point process $\left({}^{SY_{poisson}(s,\xi,1)}/_{\lambda E|A|\|w\|_1}\right)$ at $p = 1$. This shows the convergence rate of normalized scattering is $\sqrt{s}$ for Brownian motion and s for Poisson process, indicating the 1st moment can distinguish Brownian motion and Poisson point process. **Right:** Normalized scattering moments for Brownian Motion $\left({}^{SX_{BM}(s,\xi,2)}/_{\|w\|_2^2}\right)$ and Poisson point process $\left({}^{SY_{poisson}(s,\xi,2)}/_{\|w\|_2^2}\right)$ at $p = 2$. Both normalized scattering moments have convergence rate s, so the 2nd moment scattering cannot distinguish the two processes.

4.6.4 Homogeneous, non-compound Poisson point process and self similar process

We consider Brownian motion with Hurst parameter $H = \frac{1}{2}$ and compare it with Poisson point process with intensity $\lambda = 0.01$ and charges $(A)_{j=1}^{\infty} \equiv 10$. Figure 4.4 shows that the 2nd moments cannot distinguish between Brownian motion and Poisson point process while the 1st moments can.

4.7 Scattering GAN

We are also interested in the capacity of scattering moments on Poisson point processes. Ideally, we want to generate new signals through the scattering measurements and show the realizations are from the given Poisson point process. We are less interested in synthesizing one specific realization of the process but more interested in generating new realizations through the distribution of scattering moments. Therefore, instead of minimizing the ℓ_2 loss between scattering coefficients, which we did in Chapter 3, we use a GAN model to study the high dimensional distribution of the scattering moments. Figure 4.5 shows the structure of our model, where we insert a scattering propagator S between the generator G and the

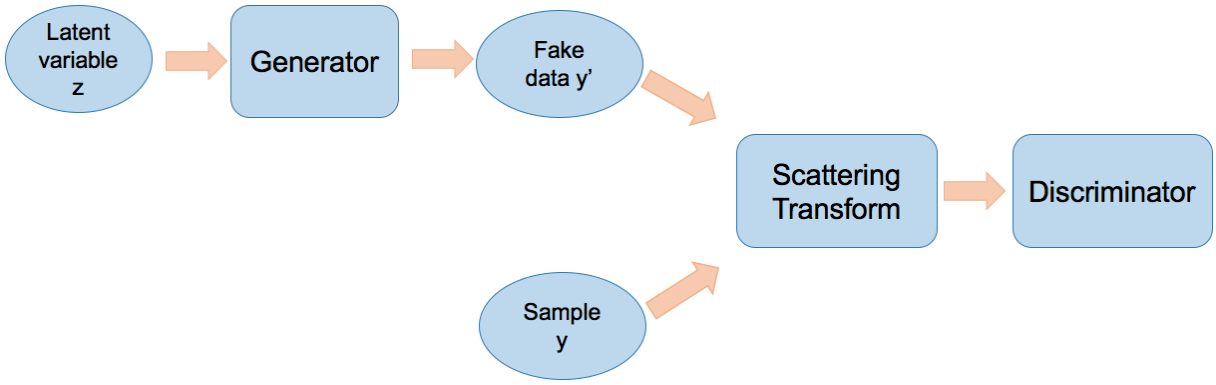


Figure 4.5 Scattering-GAN to study the capacity of scattering moments on Poisson point process. Similar to ordinary GAN, the generator takes in a random vector z and generates fake data y'. Also, the discriminator aims to distinguish fake representations from real. By inserting a scattering module between G and D, the discriminator tries to distinguish $S(y')$ from $S(y)$. When the model trains successfully, $S(y')$ has the same distribution as $S(y)$. By checking the similarity between y' and y, we learn the capacity of scattering moments.

discriminator D.

Given realizations $\{y_i\}_i$ from a Poisson point process Y, suppose their scattering moments $\{S(y_i)\}_i$ are samples from an unknown high dimensional distribution $\mathcal{P}_S$. In the scattering model, the discriminator tries to distinguish between the real scattering coefficients $S(y_i)$ and fake ones $S(y_i')$, while the generator is generating signals y_i' that have scattering coefficients $S(y_i')$ that match $S(y_i)$. Figure 4.6 shows the generated signals from the scattering GAN through a numerical experiment.

4.8 Conclusion

We have constructed Gabor-filter scattering transforms for random measures on $\mathbb{R}^d$, and stochastic processes on $\mathbb{R}$. Our construction is closely related to [41], but extends their work in several important ways. First, while our Gabor-type filters include dyadic wavelets as a special case, they also include many other families of filters. We also do not assume that the random measure Y is stationary, and consider compound, possibly inhomogeneous, Poisson random measures on $\mathbb{R}^d$, in addition to ordinary Poisson processes on $\mathbb{R}$. We do note however, that [41] provides a detailed analysis of self-similar processes and multifractal

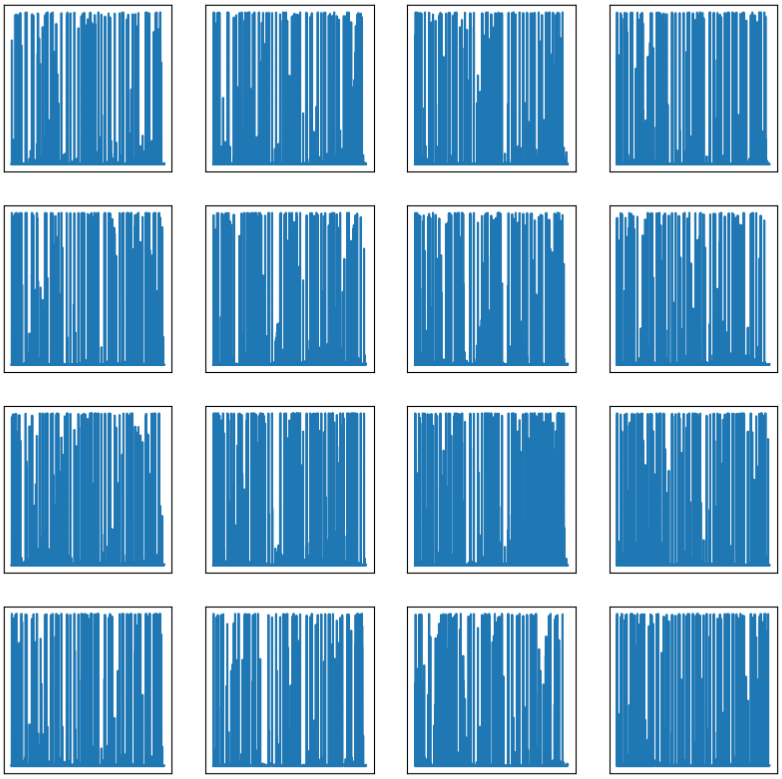


Figure 4.6 Generated signals through scattering GAN. We use realizations with length $n = 2^{12}$ from a homogeneous ordinary Poisson point process, i.e., $\lambda(t) \equiv \lambda_0$ and $A_i \equiv 1$, as training data. We use $\{2^{j/2}\}_{j=0}^{22}$ as scales for filters and apply a one-layer scattering operator to compute the scattering moments. Sigmoid is applied at the last layer in the generator. The generated signals are sparse, although not as sparse as training data. This is natural since Sigmoid forces $A_i' \in (0, 1)$, thus $\mathbb{E}[A_i'] < \mathbb{E}[A_i]$. According to theorem 6, $\lambda_0' > \lambda_0$, which we verified numerically.

random measures, whereas we have primarily focused on models of random sparse signals. We believe the results presented here open up several avenues of future research. Firstly, we have assumed throughout most of this chapter that the points of our random measures were distributed according to a possibly inhomogenous Poisson process. It would be interesting to discover if our measurements can distinguish these signals from other point processes. Secondly, it would be interesting to explore the use of these measurements for a variety of machine learning tasks such as synthesizing new signals. In the next chapter, we describe our work on using similar measurements for texture image synthesis.

CHAPTER 5

TEXTURE SYNTHESIS VIA PROJECTION ONTO MULTISCALE, MULTILAYER STATISTICS

5.1 Introduction

In the last chapter, we presented our work on generalized scattering transforms of stochastic processes. In practice, one class of signals that can be modeled by stochastic processes is texture images. This is motivated from the fact that texture images usually contain a type of random repetition of a potentially complex pattern. A natural task about texture images, which is related to the completeness of a model, is the texture synthesis problem. This task asks one to generate new, perceptually accurate, texture images given a limited (often single) realization of the texture class in question. Within the field of image generative models, the texture synthesis problem is appealing because it allows for types of statistical analysis that are not possible in general image generation. Recent works have proposed to use generative adversarial networks (GANs) [44] to perform texture synthesis and related tasks [45, 46, 47, 48]. GANs are also used to expand non-stationary texture images [49], proving the ability to capture large scale structures. Classically, texture synthesis models fall into two categories [50]: (i) non-parametric patch rearrangement methods that extract microscopic patterns from the reference image and randomly arrange these patterns in a new image; and (ii) parametric statistic-matching models that extract a set of empirical statistics from the reference texture, and generate a new image by selecting a random image with a similar statistical profile.

This chapter addresses the second type of model based on statistical matching. In [51, 52], the authors reproduce micro-textures by randomizing the phase of Fourier coefficients of the input texture. In some other works, the filtered responses of texture images are matched based on the maximum entropy principle [53, 54, 55, 56]. Such statistical models have two

challenges: (i) What is the set of statistics needed to characterize a large class of textures? and (ii) Given the statistical profile of a reference image, how does one generate a random image with the same statistical profile? Gatys and collaborators [3] had great success in addressing these two challenges by extracting the covariance statistics of the filter responses at various layers of the VGG-19 network [1], and then generating a new image with matching statistics via back-propagation and stochastic gradient descent, which is reviewed in Section 2.4. This work in turn inspired several subsequent methods, including [15, 57, 58, 59].

Despite the success of [3], though, the model is not perfect and many open questions remain for statistics-based texture synthesis models. Indeed, in a follow up paper [60], it is observed that high quality texture images can be synthesized using only a one-layer network with random, multiscale filters and rectified linear unit (ReLU) nonlinearity. The combination of the two papers [3, 60] raises questions with respect to the trade-off between network depth and the sizes of the receptive fields of the filters in the network. Additionally, putting the use of *random* filters aside, the use of a single layer of *multiscale* filters parallels classical work in the field that uses the statistics of multiscale wavelet coefficients to synthesize textures [61, 53, 2, 62]. Multi-scale CNN models are also designed to maintain high resolution in texture synthesis [63, 64]. Among these methods, the algorithm of Portilla and Simoncelli [2] is particularly notable for its use of statistics based on the modulus and phase of complex wavelet valued coefficients, in addition to its impressive performance which is often still bench-marked against today.

In this chapter we propose a multiscale, multilayer, nonlinear feature extractor for images based upon real-valued wavelet transforms, which in turn yields a set of statistics for use in texture synthesis. In addition to drawing inspiration from Portilla and Simoncelli [2] and Gatys *et al.* [3], the model presented in this chapter also draws upon ideas from the wavelet scattering transform [10], which itself has shown good results for the synthesis of gray-scale textures [43]. Nevertheless, our algorithm has several novel aspects that we use to investigate the texture synthesis problem, and which provide insight into image feature extraction via

convolutional networks. More specifically:

- We provide an analysis of the types of filters required to obtain good synthesis results when combined with the ReLU nonlinearity.
- We investigate the trade-off between network depth and the maximum scale of the wavelet filters.
- We propose a CNN architecture that uses the ReLU nonlinearity and is provably invertible at each layer, which in turn allows us to adapt the projection synthesis algorithm of [2] to our setting.
- We demonstrate our theoretical findings numerically through example synthesized images, and also compare our results to [2] and [3].

In Section 5.2 we present our statistical model in detail, while Section 5.3 describes our synthesis algorithm. Section 5.4 provides detailed numerical results, and Section 5.5 introduces implementation details. Section 5.6 contains a short conclusion.

5.2 Model

Set $\mathbb{T}^2 := [-T, T]^2$ and $\mathbb{R}_+ := [0, \infty)$, and let $x : \mathbb{T}^2 \to \mathbb{R}_+$ be a texture image, which we shall assume is in $\mathbf{L}^2(\mathbb{T}^2)$. A statistics-based matching algorithm for texture synthesis specifies a family of (nonlinear) functions $U_k : \mathbf{L}^2(\mathbb{T}^2) \to \mathbf{L}^2(\mathbb{T}^2)$ and extracts a family of empirical statistics Sx from x based on

$$Sx = (S_k x)_k \,, \quad S_k x := \frac{1}{(2T)^2} \int_{\mathbb{T}^2} U_k x(u)\, du \,.$$

A new texture $y \in \mathbf{L}^2(\mathbb{T}^2)$ is synthesized by drawing y from the set of images with similar statistical profiles:

$$y \sim \mathcal{I}_x := \{z \in \mathbf{L}^2(\mathbb{T}^2) : \|Sz - Sx\| \leq \varepsilon\} \,. \tag{5.1}$$

If $x \sim X$, where $(X(u))_{u \in \mathbb{T}^2}$ is a stochastic process, and if $U_k X$ is stationary and ergodic, then $S_k x \to \mathbb{E}[U_k X]$ as $T \to \infty$. Thus, we can think of $S_k x$ as approximating the statistics $\mathbb{E}[U_k X]$ of the unknown process that generated x.

The model (5.1) is appealing because the statistical profile Sx determines the texture class. It is thus paramount to determine a good set of functions $(U_k)_k$, and hence statistics $(S_k)_k$, the pursuit of which has ramifications in human and computer vision [65, 66, 67].

The method of Portilla and Simoncelli [2] defines the majority of their statistics by leveraging a complex valued wavelet transform and extracting statistics from the modulus and phase of the resulting wavelet coefficients. The first layer of our model also uses multiscale wavelet filters, but they are real valued and we replace the modulus and phase nonlinearities with the ReLU nonlinearity. In Sections 5.2.1 and 5.2.2 we explain how the proper selection of such wavelet filters, though, when combined with ReLU, can distinguish between certain types of patterns in the same way that modulus and phase can.

On the other hand, Gatys *et al.* [3] define their statistics using the Gram matrices of the filter responses at various layers in the VGG-19 network. The receptive field of the filters of the VGG-19 network are small, only 3×3 pixels, but the depth and pooling of the VGG-19 network allows such statistics to still capture complex multiscale patterns in texture images. Akin to the VGG network, in Section 5.2.3 we expand our set of functions U_k by computing a second wavelet transform and ReLU nonlinearity. Such a procedure is inspired by the wavelet scattering transform [10], but as we will describe differs from the scattering transform in several significant ways.

5.2.1 Wavelet filters

Let $\widehat{x}(\omega)$, for frequencies $\omega \in \Omega := \{\pi k/T : k \in \mathbb{Z}^2\} \subset \mathbb{R}^2$, denote the Fourier transform of x: $\widehat{x}(\omega) := \int_{\mathbb{T}^2} x(u) e^{-iu\cdot\omega}\, du$. A wavelet $\psi \in \mathbf{L}^2(\mathbb{T}^2)$ is an oscillating waveform that is localized in both space and frequency and has zero average. Inspired by previous work in wavelet based image processing, as well as recent analyses of the filters of the VGG network [68], we make use of three types of wavelets. Figure 5.1 shows the three wavelet families and we will define the wavelets in the following context.

The first two are directional wavelet filters. We select one even directional filter and one

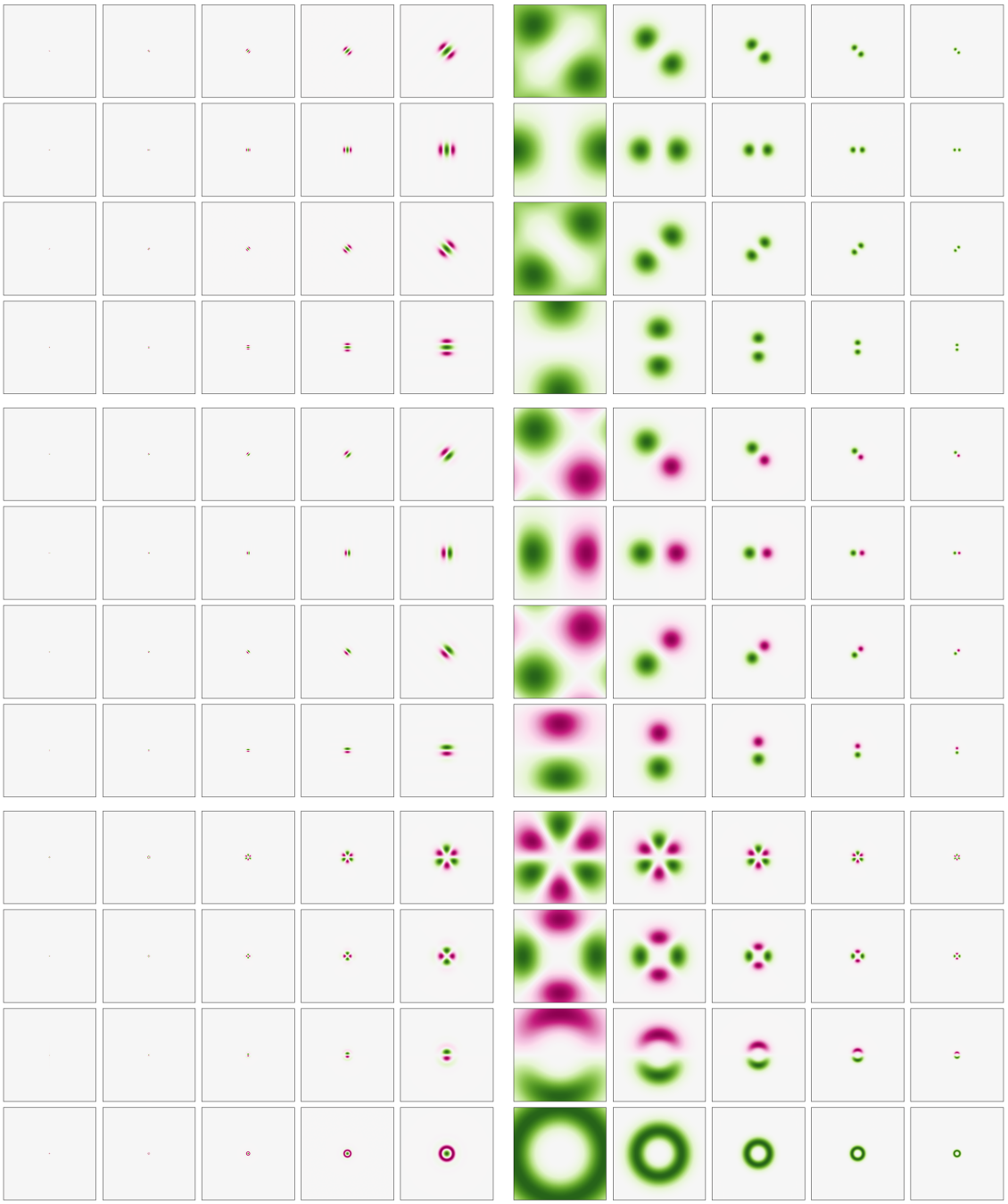


Figure 5.1 Wavelet families. **Upper**: Even directional wavelets in space and frequency (FFT). **Middle**: Odd directional wavelets in space and frequency (FFT). **Lower**: Omnidirectional wavelets in space and frequency (FFT). Each block shows the wavelet family with different scales and oscillations.

odd directional filter:

$$\psi^e(u) := g(u)\cos(\xi \cdot u)\,,$$
$$\psi^o(u) := g(u)\sin(\xi \cdot u)\,,$$

where g is an even window function and $\xi \in \mathbb{R}^2$ is the central frequency of the wavelets. These wavelets oscillate in the direction ξ and have localized Fourier transforms around ξ and $-\xi$. These wavelets are rotated to obtain waveforms oscillating in different directions:

$$\psi^\beta_\theta(u) := \psi^\beta(R_\theta^{-1}u)\,, \quad \beta \in \{e, o\}\,,$$

where $R_\theta \in \mathrm{SO}(2)$ is the 2×2 rotation matrix about the angle $\theta \in [0, \pi)$. We use M angles $\theta \in \Theta_M := \{m\pi/M : 0 \le m < M\}$.

The third type of wavelet is based on the polar coordinate representation $u = (r, \varphi) \in [0, \infty) \times [0, 2\pi)$, and oscillates along the angle parameter φ:

$$\psi^p_\ell(u) := a_\ell(u)\cos(\ell\varphi)\,,$$

where $\ell \in \mathbb{Z}$ is the frequency of oscillation along the angle φ. If $a_\ell(u) = \widetilde{a}_\ell(r)$, then the function a_ℓ determines the frequency of oscillation of the filter along the radial parameter. In this case, $|\widehat{\psi^p}(\omega)|$ has an essential support in the shape of an annulus and the filter is omnidirectional. We restrict $0 \le \ell < L$ and select $a_\ell(u) = \widetilde{a}_\ell(r)$ to be an oscillatory function that oscillates at a frequency approximately proportional to $L - 1 - \ell$, ensuring that the overall frequency support of ψ^p_ℓ is approximately fixed.

Directional filters such as ψ^e_θ and ψ^o_θ are common in image processing and various analyses of CNN filters, e.g., [68], have shown that commonly used CNNs learn directional filters. In Section 5.2.2 we will motivate the seemingly redundant choice of using both an even and odd directional wavelet filter. By examining the filters of the VGG-19 network, though, one also finds omnidirectional filters. In practice (see Section 5.4) we find that such filters improve the quality of synthesized textures in which the image patterns do not align with a small subset of directions.

All wavelets are dilated at dyadic scales to obtain a multiscale family of waveforms:

$$\psi^{\beta}_{j,\alpha}(u) := 2^{-2j}\psi^{\beta}_{\alpha}(2^{-j}u)\,, \quad j \in \mathbb{Z}\,, (\alpha,\beta) \in \{(\theta,e),(\theta,o),(\ell,p)\}\,.$$

In our experiment, for directional wavelets, we use $g = g_\sigma$ as a gaussian function with variance σ^2. For the three wavelets, they all have local support both in space and frequency. As j increase from 0 to $J-1$ (from left to right in each block in Figure 5.1), the wavelet has larger support in space and smaller support in frequency. For directional wavelets, the wavelet support varies in directions with rotations (from top to bottom in each block) to capture directional oscillations in images. For omnidirectional wavelets, the wavelet either oscillates radially or angularly or both. The total number of oscillations is fixed across the four wavelets.

Numerically, we may assume that $\widehat{x}(\omega)$ is supported on frequencies ω contained in $[-\pi,\pi]^2$. By design the collection of wavelet filters have collective frequency support in a ball, which we can assume is the frequency ball of radius π. In this case we complement the wavelet filters with two additional filters: (i) a non-negative low pass filter $\phi \in \mathbf{L}^2(\mathbb{T}^2)$ that has Fourier transform essentially supported around the origin (since wavelets have zero average); and (ii) a high pass filter $h \in \mathbf{L}^2(\mathbb{T}^2)$ that has Fourier transform essentially supported outside of the frequency ball $\{\omega \in \Omega : |\omega| \leq \pi\}$ (in other words, $\widehat{h}$ is supported in the "corners" of $[-\pi,\pi]^2$). The scales 2^j of the wavelet filters are restricted to $0 \leq j < J$, where $J \leq J_{\max} = O(\log_2 T)$, and we dilate ϕ to the scale 2^J via $\phi_J(u) := 2^{-2J}\phi(2^{-J}u)$.

The wavelet transform we use in this chapter computes the convolution of x with all the aforementioned filters:

$$W_J x := \{x * \phi_J\,,\, x * h\,,\, x * \psi^{\beta}_{j,\alpha} :$$
$$0 \leq j < J, (\alpha,\beta) \in \{(\theta,e),(\theta,o),(\ell,p)\},\ \ \theta \in \Theta_M,\ \ 0 \leq \ell < L\}\,.$$

A group of filters $\{f_k\}_k$ is said to form a frame for signals x such that $\mathrm{supp}(\widehat{x}) \subset [-\pi,\pi]^2$ if there exists two constants $0 < A \leq B < +\infty$ such that:

$$A \leq \sum_k |\widehat{f}_k(\omega)|^2 \leq B\,, \quad \forall\, \omega \in \Omega \cap [-\pi,\pi]^2\,.$$

We can define the dual filters as $\widehat{\widetilde{f}}_k(\omega) := \frac{\overline{\widehat{f}_k(\omega)}}{\sum_k |\widehat{f}_k(\omega)|^2}$. An image x can be reconstructed from its filtrations by the filters $\{f_k\}_k$ using the dual filters $\{\widetilde{f}_k\}_k$ and the formula:

$$x = \sum_k x * f_k * \widetilde{f}_k \,. \tag{5.2}$$

For appropriately chosen parameters, the collection of filters used to define the wavelet transform W_J forms a frame. As such, we can recover x from $W_J x$ using (5.2). This property will be important in Section 5.3 for developing an algorithm by which to synthesize a new texture.

5.2.2 First layer statistics

We extract directly from the image x the mean, variance, skewness, and kurtosis of the image intensities $(x(u))_{u\in\mathbb{T}^2}$, in addition to the min/max intensities. We then consider the low pass filtering $x * \phi_J$. Since $x(u) \geq 0$ and $\phi_J(u) \geq 0$, the mean of $x * \phi_J$ is proportional to the mean of x and does not need to be computed. We do add in the variance of the values $(x * \phi_J(u))_{u\in\mathbb{T}^2}$ to Sx. We also add in the variance of the high pass coefficients $(x * h(u))_{u\in\mathbb{T}^2}$. These statistics are also used by Portilla and Simoncelli, and one can find additional motivation for their usefulness in [2].

In order to simplify notation, let $\lambda = (j, \alpha, \beta) \in \Lambda$ be any admissible triplet for the wavelets $\psi^{\beta}_{j,\alpha}$ described in Section 5.2.1, and denote these wavelets by ψ_λ. Like the low pass coefficients and the high pass coefficients, we could compute only the variance of the values $(x * \psi_\lambda(u))_{u\in\mathbb{T}^2}$, but in doing so we would miss important correlations between patterns in x at different scales, orientations, and angular frequencies, as captured by our wavelets. In fact, results in [43] indicate that the using only the variance of wavelet coefficients does not result in good texture synthesis for certain types of textures. An alternative would be to compute the covariance between $x * \psi_\lambda$ and $x * \psi_{\lambda'}$, but the frequency localization of the wavelets means that such statistics will be nearly zero, and hence meaningless, for most pairs (λ, λ'). One possible solution is to apply a pointwise nonlinear function $\sigma : \mathbb{R} \to \mathbb{R}_+$ to the wavelet

coefficients, effectively pushing the high frequencies of $x * \psi_\lambda$ down to the low frequencies for each λ, which in turn generates non-trivial correlations between $x * \psi_\lambda$ and $x * \psi_{\lambda'}$. In [2], Portilla and Simoncelli decompose complex-valued wavelet coefficients into their modulus and phase (two nonlinear transforms), and compute covariance-type statistics of the wavelet modulus coefficients and of the phase coefficients. More recently, Zhang and Mallat [69] developed a wavelet phase harmonic nonlinear transform (also for complex wavelets) and used the resulting covariance statistics for texture synthesis of select gray-scale textures.

In this work we set $\sigma(t) := \max(0, t)$, which is the rectified linear unit (ReLU) nonlinearity. In order to obtain an invertible transform, we also multiply the wavelet coefficients by ± 1, thus yielding the nonlinear wavelet transform:

$$U_J^1 x := \{x * \phi_J \,,\, x * h \,,\, \sigma(\gamma \cdot x * \psi_\lambda) : \gamma = \pm 1, \;\; \lambda \in \Lambda\} \,.$$

Since $t = \sigma(t) - \sigma(-t)$, one can recover $W_J x$ from $U_J^1 x$ and hence one can recover x using (5.2). We compute the Gram matrix correlation statistics between all pairs of nonlinear coefficients in $U_J^1 x$,

$$C_x^1(\lambda, \gamma, \lambda', \gamma') := \frac{1}{(2T)^2} \int_{\mathbb{T}^2} \sigma(\gamma \cdot x * \psi_\lambda(u)) \sigma(\gamma' \cdot x * \psi_{\lambda'}(u)) \, du \tag{5.3}$$

with the exception of $\lambda = \lambda'$ and $\gamma = -\gamma'$ as $C_x^1(\lambda, \gamma, \lambda, -\gamma) = 0$.

Note that $|t| = \sigma(t) + \sigma(-t)$, and hence the statistics (5.3) subsume the covariance statistics between wavelet absolute value coefficients, which are similar to the wavelet modulus statistics computed in [2]. It was observed in [69] that the ReLU nonlinearity is related to phase information in complex valued wavelet coefficients. In [2], Portilla and Simoncelli motivate the inclusion of phase by considering two one-dimensional signals, a Dirac function and a step function. These two signals cannot be distinguished by the wavelet modulus coefficients alone. ReLU wavlet coefficients, on the other hand, can distinguish a Dirac function from a step function for either even or odd wavelets. However, the next theorem shows they have trouble distinguishing the relative intensity of these functions unless even and odd wavelets are used together.

Theorem 12. *Define* $y_1(t) := \delta(t)$ *and* $y_2(t) := \mathbf{1}_{[0,\infty)}(t)$. *Let* $\widetilde{\psi}^e \in \mathbf{L}^2(\mathbb{R})$ *be a one-dimensional even wavelet, and let* $\widetilde{\psi}^o \in \mathbf{L}^2(\mathbb{R})$ *be a one-dimensional odd wavelet. Define the* 2×2 *correlation matrices* $C^{\beta}_{y_k}$ *as:*

$$C^{\beta}_{y_k}(\gamma, \gamma') := \int_{\mathbb{R}} \sigma(\gamma \cdot y_k * \widetilde{\psi}^{\beta}(t)) \sigma(\gamma' \cdot y_k * \widetilde{\psi}^{\beta}(t))\, dt\,,$$

for $\beta \in \{e, o\}$ *and* $\gamma \in \{-1, +1\}$. *Then* $C^e_{y_1} \neq C^e_{-y_1}$ *and* $C^e_{y_2} = C^e_{-y_2}$, *while* $C^o_{y_1} = C^o_{-y_1}$ *and* $C^o_{y_2} \neq C^o_{-y_2}$.

Figure 5.2 shows examples of the wavelets and corresponding Fourier transforms.

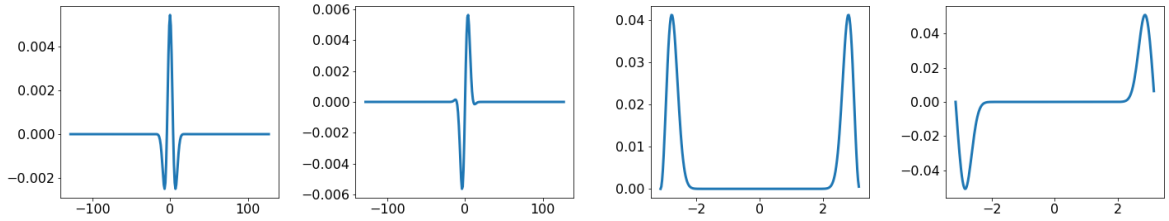


Figure 5.2 1D wavelets. From left to right: 1D even wavelet, 1D odd wavelet, FFT (real part) of even wavelet, FFT (imagery part) of odd wavelet.

In order to prove Theorem 12, we need the following lemma.

Lemma 3. *Let* $f(u)$ *be an odd function, we have*

$$\int_{\mathbb{R}} \sigma(\gamma \cdot f)\sigma(\gamma' \cdot f) = \int_{\mathbb{R}} \sigma(-\gamma \cdot f)\sigma(-\gamma' \cdot f)$$

for $\gamma, \gamma' \in \{-1, +1\}$.

Proof of Theorem 12. We first prove the theorem for the 1D Dirac function $y_1(t)$. First the wavelet transform of y_1 is:

$$y_1 * \psi^o(u) = \psi^o(u), \quad y_1 * \psi^e(u) = \psi^e(u)$$

Figure 5.3 shows the wavelet transforms for y_1 and $-y_1$. Since $\psi^o(u)$ is an odd function, with Lemma 3 we have $C^o_{y_1} = C^o_{-y_1}$. However since $\psi^e(u)$ is an even function, generally we have $C^e_{y_1}(+1, +1) \neq C^e_{-y_1}(+1, +1)$. Therefore $C^e_{y_1} \neq C^e_{-y_1}$. Figure 5.4 verifies this conclusion numerically.

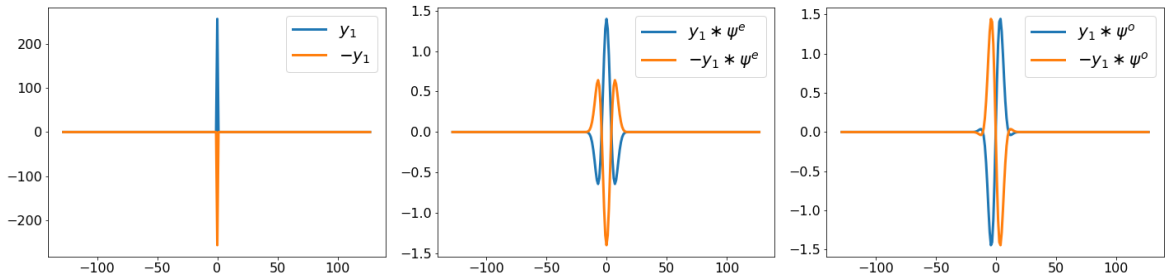


Figure 5.3 Dirac functions and wavelet coefficients. **Left**: Two Dirac functions y_1 and $-y_1$. **Middle**: Wavelet coefficients with the even wavelet. **Right**: Wavelet coefficients with the odd wavelet.

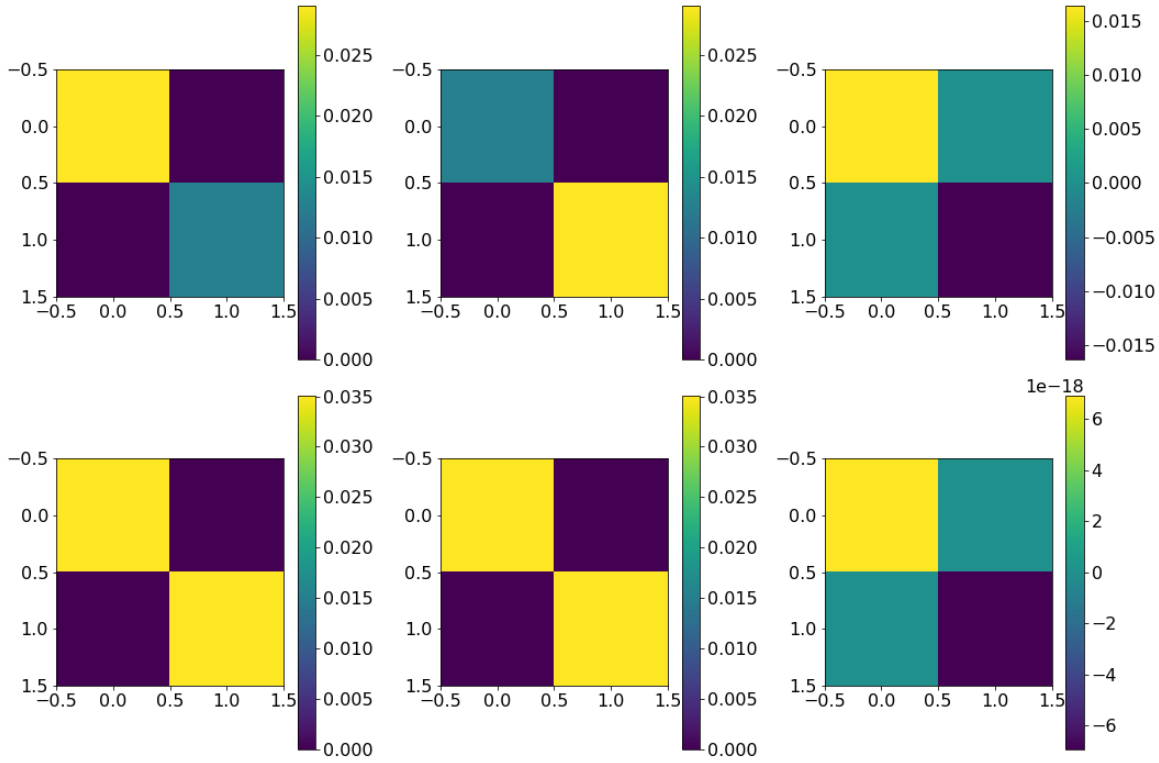


Figure 5.4 Covariance matrix for diracs. Upper row from left to right: $C^e_{y_1}$, $C^e_{-y_1}$, $C^e_{y_1} - C^e_{-y_1}$. Lower row from left to right: $C^o_{y_1}$, $C^o_{-y_1}$, $C^o_{y_1} - C^o_{-y_1}$. This numerically verified that even wavelet is able to distinguish the two dirac functions from the covariance statistics while odd wavelet cannot.

Now we prove the theorem for the jump function $y_2(t)$. The wavelet transforms satisfy:

$$y_2 * \psi^\beta(u) = \int_{\mathbb{R}} y_2(u-t)\psi^\beta(t)dt = \int_{-\infty}^{u} \psi^\beta(t)dt \tag{5.4}$$

for $\beta \in \{e, o\}$. Figure 5.5 illustrates the convolution of the even and odd wavelet with y_2

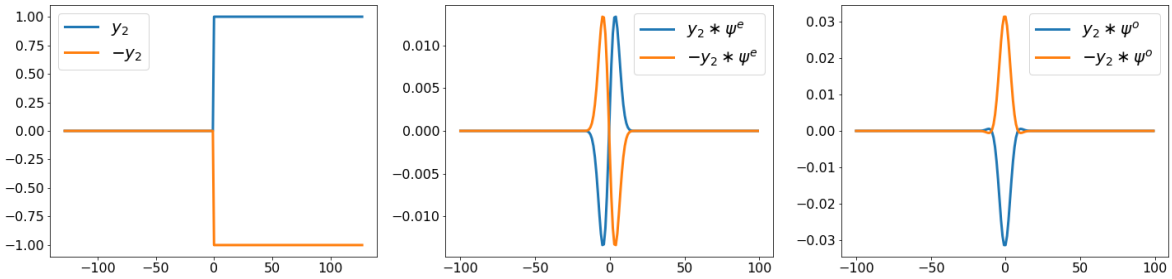


Figure 5.5 Jump functions and wavelet coefficients. **Left**: Two jump functions y_2 and $-y_2$. **Middle**: Wavelet coefficients with the even wavelet. **Right**: Wavelet coefficients with the odd wavelet.

and $-y_2$.

Remark 1. *Since ψ^e is an integrable even function, then $f^e(u) = \int_{-\infty}^{u} \psi^e(t)dt$ is an odd function.*

Remark 2. *Since ψ^o is an integrable odd function, then $f^o(u) = \int_{-\infty}^{u} \psi^o(t)dt$ is an even function.*

Remark 1 with Lemma 3 shows we have $C^e_{y_2} = C^e_{-y_2}$. Remark 2 means that generally we also have $C^o_{y_2} \neq C^o_{-y_2}$. Figure 5.6 gives the numerical verification.

□

Theorem 12 shows both even and odd wavelets are necessary in our model. For images x, the Dirac signal y_1 is similar to a dividing line that separates two regions of the same shade, which occurs in many types of texture images. This result shows that ReLU nonlinear wavelet coefficient correlations, when computed with an odd wavelet, cannot correctly determine the brightness of the dividing line relative to the regions it separates. Similarly, ReLU nonlinear wavelet coefficients, when computed with an even wavelet, cannot determine whether the color gradient across an edge is positive or negative. Numerical results illustrating these effects are given in Section 5.4.1.

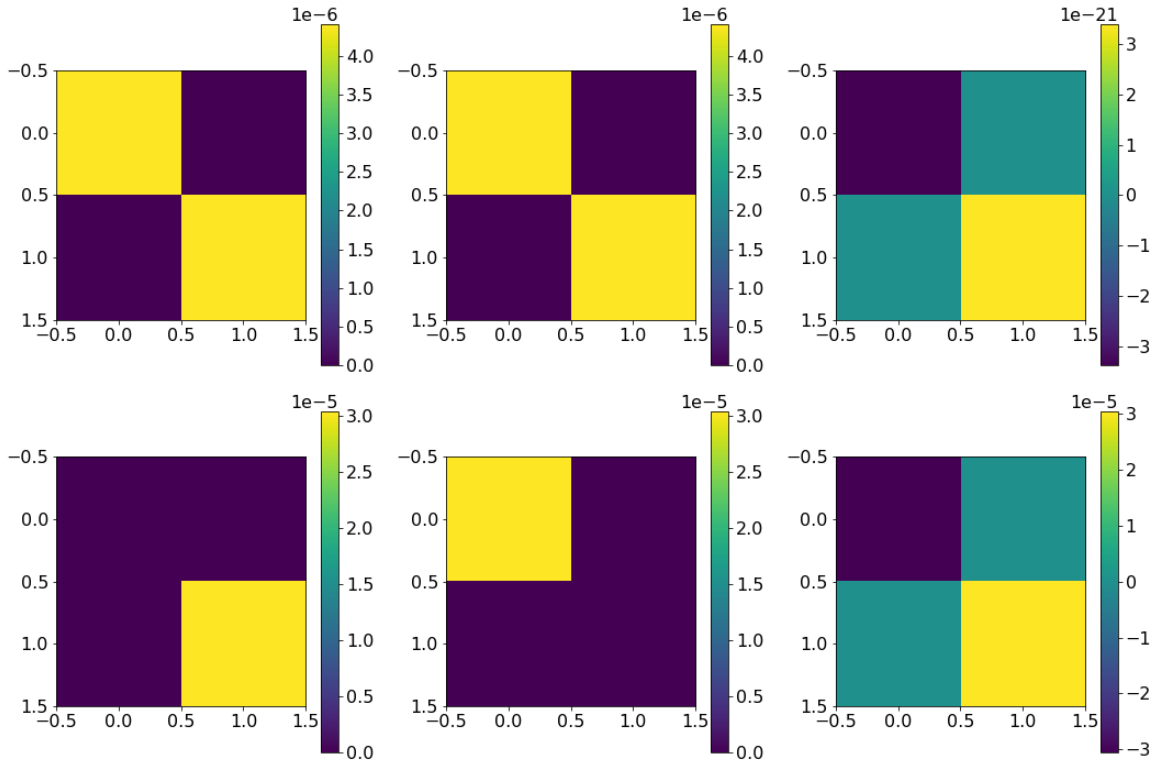


Figure 5.6 Upper row from left to right: $C^e_{y_2}$, $C^e_{-y_2}$, $C^e_{y_2} - C^e_{-y_2}$. Lower row from left to right: $C^o_{y_2}$, $C^o_{-y_2}$, $C^o_{y_2} - C^o_{-y_2}$. This numerically verified that odd wavelet is able to distinguish the two jump functions from the covariance statistics while the even wavelet cannot.

5.2.3 Second layer statistics

ReLU wavelet correlation statistics can be complemented by two-layer statistics that are derived from feature maps that combine image information across scales before iterating the operator U^1_J. In particular, we compute

$$U^2_J x := \left\{ U^1_J \left(\sum_{j=0}^{J-1} \sigma(\gamma \cdot x * \psi^\beta_{j,\alpha}) \right) : (\alpha, \beta) \in \{(\theta, e), (\theta, o), (\ell, p)\}, \right.$$
$$\left. \gamma = \pm 1, \;\; \theta \in \Theta_M, \;\; 0 \le \ell < L \right\}.$$

and recall

$$U^1_J x := \{ x * \phi_J \,,\, x * h \,,\, \sigma(\gamma \cdot x * \psi_\lambda) : \gamma = \pm 1, \;\; \lambda \in \Lambda \}.$$

We then compute the variance statistics of the low and high pass maps of $U^2_J x$, and the correlation statistics between all pairs of the nonlinear wavelet maps contained in $U^2_J x$, with

the same exceptions as in the first layer.

By iterating upon the map U_J^1, the map U_J^2 bears some similarity to the wavelet scattering transform [10]. However, there are several important differences. As already discussed, we utilize the ReLU nonlinearity and a family of real valued wavelets, as opposed to complex valued wavelets and the modulus nonlinearity. Furthermore, the map U_J^1 defined here is invertible, unlike the scattering propagation operator. Finally, before iterating the map U_J^1, we sum over the scale index j of the nonlinear maps $\sigma(\gamma \cdot x * \psi_{j,\alpha}^{\beta})$. This operation is akin to a 1×1 convolution operation in the VGG-19 CNN (and other CNNs), but unlike in the VGG network in which the filters being summed over have receptive fields with the same size, here we aggregate over nonlinear multiscale wavelet filtrations that allows our network to link together correlated patterns at multiple scales. This operation also has the effect of reducing the number of second layer maps, and hence statistics. Perhaps surprisingly, the operator U_J^2 is also invertible under appropriate conditions on the wavelets defined in Section 5.2.1.

Theorem 13. *If $\{\phi_J, h, \psi_{j,\theta}^o : 0 \leq j < J,\ \theta \in \Theta_M\}$ forms a frame and if $\widehat{g}$ is non-negative, radial, and a decreasing function of $|\omega|$, then $x \mapsto \{x * \phi_J\,,\, x * h\,,\, U_J^2 x\}$ is invertible.*

Before proving Theorem 13, we first prove the following lemma.

Lemma 4. *If $\{\phi_J, h, \psi_{j,\theta}^o : 0 \leq j < J,\ \theta \in \Theta_M\}$ forms a frame and if $\widehat{g}$ is non-negative, radial, and a decreasing function of $|\omega|$, then $\{\phi_J, h, \sum_{j=0}^{J-1} \psi_{j,\theta}^o :\ \theta \in \Theta_M\}$ also forms a frame.*

Proof of Lemma 4. With the definition of frame, we know there exist two constants $0 < A \leq B < \infty$ such that:

$$A \leq |\widehat{\phi}_J(\omega)|^2 + |\widehat{h}(\omega)|^2 + \sum_{j,\theta} |\widehat{\psi}_{j,\theta}^o(\omega)|^2 \leq B,\ \forall\, \omega \in \Omega \cap [-\pi, \pi]^2\,.$$

we need to prove there exist two constants $0 < A' \leq B' < \infty$ such that:

$$A' \leq |\widehat{\phi}_J(\omega)|^2 + |\widehat{h}(\omega)|^2 + \sum_{\theta} |\sum_{j} \widehat{\psi}_{j,\theta}^o(\omega)|^2 \leq B',\ \forall\, \omega \in \Omega \cap [-\pi, \pi]^2\,.$$

The upper bound always exists as long as we have a finite number of filters. Therefore we only prove the lower bound. The key point is to prove:

$$\sum_j |\widehat{\psi}^o_{j,\theta}(\omega)|^2 \leq |\sum_j \widehat{\psi}^o_{j,\theta}(\omega)|^2, \forall \theta \in \Theta_M \tag{5.5}$$

Without loss of generosity, we set $\theta = 0$ and omit this notation in the following proof. Recall the odd directional wavelet $\psi^o(u) = g(u)\sin(\xi \cdot u)$, which has Fourier transform:

$$\widehat{\psi}^o(\omega) = \frac{\widehat{g}_\sigma(\omega - \xi) - \widehat{g}_\sigma(\omega + \xi)}{2i} \tag{5.6}$$

Bringing equation (5.6) into equation (5.5), we need to prove:

$$\sum_j |\widehat{g}_{\sigma,j}(\omega - \xi) - \widehat{g}_{\sigma,j}(\omega + \xi)|^2 \leq |\sum_j \widehat{g}_{\sigma,j}(\omega - \xi) - \widehat{g}_{\sigma,j}(\omega + \xi)|^2 \tag{5.7}$$

If $\widehat{g}$ is non-negative, radial, and a decreasing function of $|\omega|$, one can prove:

- If $|\omega - \xi| < |\omega + \xi|$, then $\widehat{g}_\sigma(\omega - \xi) - \widehat{g}_\sigma(\omega + \xi) \geq 0$. For any such ω we also have $|2^{-j}\omega - \xi| < |2^{-j}\omega + \xi|$, and $\widehat{g}_\sigma(2^{-j}\omega - \xi) - \widehat{g}_\sigma(2^{-j}\omega + \xi) \geq 0$.
- If $|\omega - \xi| > |\omega + \xi|$, then $\widehat{g}_\sigma(\omega - \xi) - \widehat{g}_\sigma(\omega + \xi) \leq 0$. For any such ω we also have $|2^{-j}\omega - \xi| > |2^{-j}\omega + \xi|$, and $\widehat{g}_\sigma(2^{-j}\omega - \xi) - \widehat{g}_\sigma(2^{-j}\omega + \xi) \leq 0$.
- If $|\omega - \xi| = |\omega + \xi|$, then $\widehat{g}_\sigma(\omega - \xi) - \widehat{g}_\sigma(\omega + \xi) = 0$. For any such ω we also have $|2^{-j}\omega - \xi| = |2^{-j}\omega + \xi|$, and $\widehat{g}_\sigma(2^{-j}\omega - \xi) - \widehat{g}_\sigma(2^{-j}\omega + \xi) = 0$.

Then for all ω, $\widehat{g}_\sigma(\omega - \xi) - \widehat{g}_\sigma(\omega + \xi)$ and $\widehat{g}_{\sigma,j}(\omega - \xi) - \widehat{g}_{\sigma,j}(\omega + \xi)$ have the same sign for all j (either non-positive or non-negative). One can prove: $(\sum_i a_i)^2 \geq \sum_i a_i^2$ if all the a_i are non-negative or non-positive. Therefore, equation (5.7) is true, and $|\sum_j \widehat{\psi}^o_{j,\theta}(\omega)|^2 \geq \sum_j |\widehat{\psi}^o_{j,\theta}(\omega)|^2$ for all θ. The lemma is proved. □

With Lemma 4, now we prove Theorem 13.

Proof of Theorem 13. If $\{\phi_J, h, \psi^o_{j,\theta} : 0 \leq j < J,\ \theta \in \Theta_M\}$ forms a frame, then

$$\{\phi_J\,,\, h\,,\, \psi^\beta_{j,\alpha} : 0 \leq j < J, (\alpha, \beta) \in \{(\theta, e), (\theta, o), (\ell, p)\},\ \theta \in \Theta_M,\ 0 \leq \ell < L\}\,.$$

also forms a frame, i.e.,

$$x = x * \phi_J * \widetilde{\phi_J} + x * h * \widetilde{h} + \sum_{j=0}^{J-1} \sum_{\alpha,\beta} x * \psi_{j,\alpha}^{\beta} * \widetilde{\psi_{j,\alpha}^{\beta}}, \ \forall x \in \mathbf{L}^2(\mathbb{T}^2) \tag{5.8}$$

Therefore we can reconstruct $\sum_{j=0}^{J-1} \sigma(\gamma \cdot x * \psi_{j,\alpha}^{\beta})$ from $U_J^2 x$. With $t = \sigma(t) - \sigma(-t)$, we are able to get $\sum_{j=0}^{J-1} x * \psi_{j,\alpha}^{\beta}$, which can also be written as $x * \sum_{j=0}^{J-1} \psi_{j,\alpha}^{\beta}$. At this point, we have the following updated responses

$$\{x * \phi_J, x * h, x * \sum_{j=0}^{J-1} \psi_{j,\alpha}^{\beta}, (\alpha, \beta) \in \{(\theta, e), (\theta, o), (\ell, p)\}\}$$

With Lemma 4, $\{\phi_J, h, \sum_{j=0}^{J-1} \psi_{j,\theta}^{o}\}$ forms a frame, then $\{\phi_J, h, \sum_{j=0}^{J-1} \psi_{j,\alpha}^{\beta}\}$ also forms a frame. Therefore we can reconstruct the image x from the above responses:

$$x = x * \phi_J * \widetilde{\phi_J} + x * h * \widetilde{h} + \sum_{\alpha,\beta} x * \sum_{j=0}^{J-1} \psi_{j,\alpha}^{\beta} * \widetilde{\sum_{j=0}^{J-1} \psi_{j,\alpha}^{\beta}} \tag{5.9}$$

□

5.3 Synthesis algorithm

Since both operator U_J^1 and operator U_J^2 (combined with the low pass and high pass coefficients) are invertible, we can adapt the iterative projection algorithm of [2] in order to synthesize a new texture x^* with approximately the same statistical profile as x. We describe our version of the projection algorithm in this section. Algorithm 1 summarizes the following synthesis process.

Let us first collect the statistics described in Section 5.2:

- $S_J^0 x :=$ six pixel intensity statistics, given by the mean, variance, skewness, kurtosis, min, and max of $(x(u))_{u \in \mathbb{T}^2}$.
- $S_J^1 x := \{\mathrm{Var}(x * \phi_J), \mathrm{Var}(x * h), C_x^1\}$, which are the statistics derived from the first layer coefficients.

Algorithm 1 Projection algorithm for texture synthesis

Input reference image x ;
Output new texture image x^* ;
compute target statistics $S_J x = (S_J^0 x, S_J^1 x, S_J^2 x)$;
initialize: $x^* \leftarrow x^0$ (uniform noise) ;
$x^* \leftarrow \text{mod\_intensities}(x^*, S_J^0 x)$;
while error $> \varepsilon$ **do**
  compute $U_J^1 x^* = \{U_J^1 x^*_{\phi_J}, U_J^1 x^*_h, U_J^1 x^*_\psi\}$;
  $U_J^1 x^*_{\phi_J} \leftarrow \text{mod\_variance}(U_J^1 x^*_{\phi_J}, \text{Var}(x * \phi_J))$;
  $U_J^1 x^*_h \leftarrow \text{mod\_variance}(U_J^1 x^*_h, \text{Var}(x * h))$;
  $U_J^1 x^*_\psi \leftarrow \text{mod\_correlation}(U_J^1 x^*_\psi, C_x^1)$;
  compute $U_J^2 x^*$ from $U_J^1 x^*_\psi$;
  $U_J^2 x^*_{\phi_J} \leftarrow \text{mod\_variance}(U_J^2 x^*_{\phi_J}, S_J^2 x_{\phi_J})$;
  $U_J^2 x^*_h \leftarrow \text{mod\_variance}(U_J^2 x^*_h, S_J^2 x_h)$;
  $U_J^2 x^*_\psi \leftarrow \text{mod\_correlation}(U_J^2 x^*_\psi, S_J^2 x_\psi)$;
  reconstruct $U_J^1 x^*_\psi \leftarrow \text{reconstruct\_layer1}(U_J^2 x^*_{\phi_J}, U_J^2 x^*_h, U_J^2 x^*_\psi)$;
  reconstruct $x^* \leftarrow \text{reconstruct\_x}(U_J^1 x^*_{\phi_J}, U_J^1 x^*_h, U_J^1 x^*_\psi)$;
  $x^* \leftarrow \text{mod\_intensities}(x^*, S_J^0 x)$;
  update error $\|S_J x^* - S_J x\|$;
end

- $S_J^2 x := \{S_J^2 x_{\phi_J}, S_J^2 x_h, C_x^2\}$, which consists of the second layer low pass variances ($S_J^2 x_{\phi_J}$) and high pass variances ($S_J^2 x_h$), and the second layer correlation statistics between ReLU wavelet maps (C_x^2).

Now let $U_J^1 x_\psi$ denote the collection of nonlinear ReLU wavelet coefficient maps of $U_J^1 x$; let $U_J^2 x_{\phi_J}$ denote the collection of second layer low pass maps; let $U_J^2 x_h$ denote the collection of second layer high pass maps; and let $U_J^2 x_\psi$ denote the collection of second layer nonlinear ReLU wavelet coefficient maps. Given a reference image x, we start by computing its statistical profile $S_J x = (S_J^0 x, S_J^1 x, S_J^2 x)$. We then initialize our synthesized image with a random noise image x^0 where each $x^0(u)$ is an i.i.d. sample from the uniform distribution.

Let x^t denote the synthesized image after t iterations. The algorithm first updates x^t by directly modifying its intensities $(x^t(u))_{u \in \mathbb{T}^2}$ so that $S_J^0 x^t = S_J^0 x$. It then computes $U_J^1 x^t$ using the modified x^t. The low pass coefficients $x^t * \phi_J$ and the high pass coefficients $x^t * h$ are adjusted so that $\text{Var}(x^t * \phi_J) = \text{Var}(x * \phi_J)$ and $\text{Var}(x^t * h) = \text{Var}(x * h)$. All of these

steps are computed in the exact same fashion as [2]. Finally, the nonlinear coefficient maps $U_J^1 x_\psi^t$ are adjusted to match the target correlation matrix so that $C_{x^t}^1 = C_x^1$; this step is performed in a way that is similar to how [2] updates the wavelet modulus coefficients. If the algorithm is using only first layer statistics, it then inverts $U_J^1 x^t$ to obtain x^{t+1}, and the process repeats itself.

On the other hand, if the algorithm is using second layer statistics, it then decomposes the updated maps $U_J^1 x_\psi^t$ further by computing $U_J^2 x^t$. The algorithm updates the collection of second layer maps $U_J^2 x^t$ by matching $S_J^2 x^t$ to $S_J^2 x$ in a similar fashion as the first layer. At this point, the algorithm inverts the updated collection $\{x^t * \phi_J\,,\, x^t * h\,,\, U_J^2 x^t\}$ using Theorem 13 to obtain x^{t+1}.

5.4 Numerical Results

We implement several texture synthesis experiments with the goals of (i) numerically verifying theoretical assertions made in previous sections; (ii) understanding the effect of hyperparameter choices on the quality of synthesized textures; and (iii) comparing to other commonly used algorithms. Our texture images are taken from DTD database[1] and CG Texture database[2]. Every image is resized to 256×256 for consistency. In our experiments, the wavelet transform is implemented in the frequency field using the fast Fourier transform. Therefore, we also periodize certain images to avoid border effect [70]. For directional wavelets, we fix the total number of rotations as $M = 4$. For omnidirectional wavelets, we fix the total number of oscillations at $L = 4$. The maximum scale is $J_{\max} = 6$. In the following subsections, we numerically illustrate the advantages of using all three types of filters (Section 5.4.1); we examine the role of the maximum scale 2^J (Section 5.4.2); and we compare the one-layer synthesis to the two-layer synthesis, thus examining the role of network depth (Section 5.4.3). Finally, in Section 5.4.4 we also compare our results to Gatys *et al.* [3] and Portilla and Simoncelli [2].

[1] https://www.robots.ox.ac.uk/ vgg/data/dtd/
[2] https://www.textures.com/

5.4.1 Filter comparison

Theorem 12 proves both even and odd wavelets are important for texture synthesis. Figure 5.7 validates this theory numerically. We see, for example, that synthesis with odd wavelets is prone to blurring edges and even flipping the colors of enclosed regions that should have the same color (e.g., the last row of Figure 5.7). When using only even wavelets, images with banded colors, for example the third row of Figure 5.7, are blurred. Synthesized images using both even and odd wavelets are generally a clear improvement over their single wavelet-type counterparts.

Figure 5.8 shows the necessity for using omnidirectional wavelets. The advantages are clearest for rounded shapes or swirls. Such patterns have no clear direction. For example, in the swirly texture in the last row of Figure 5.8, the omnidirectional wavelets do a better job of reproducing the long swirls. The round-shaped pebbles and dots (rows one and three of Figure 5.8) have smoother edges and a cleaner background when adding the omnidirectional wavelets.

5.4.2 Comparison of maximum scale

Figure 5.9 shows synthesis results with different numbers of scales from the two layer model. With $J = 3$, the statistics are not able to capture macroscopic patterns. Therefore the edges of synthesized bricks (rows one and three) and frames (second to last row) are not straight or continuous. The reproduced house (second row) and dots (third last row) are more random compared to the original image. Larger scales can also capture longer swirls (last row). However, $J = 5$ and $J = 6$ achieve equivalent perceptual accuracy, proving there is no need to add in larger scales than $J = 5$. Indeed, the effective receptive field of the two layer synthesis with $J = 5$ is equivalent to the single layer receptive field of $J_{\max} = 6$. We also observe that using smaller scales gives more variety of the pattern arrangements, while large scale statistics have the potential to duplicate the reference image.

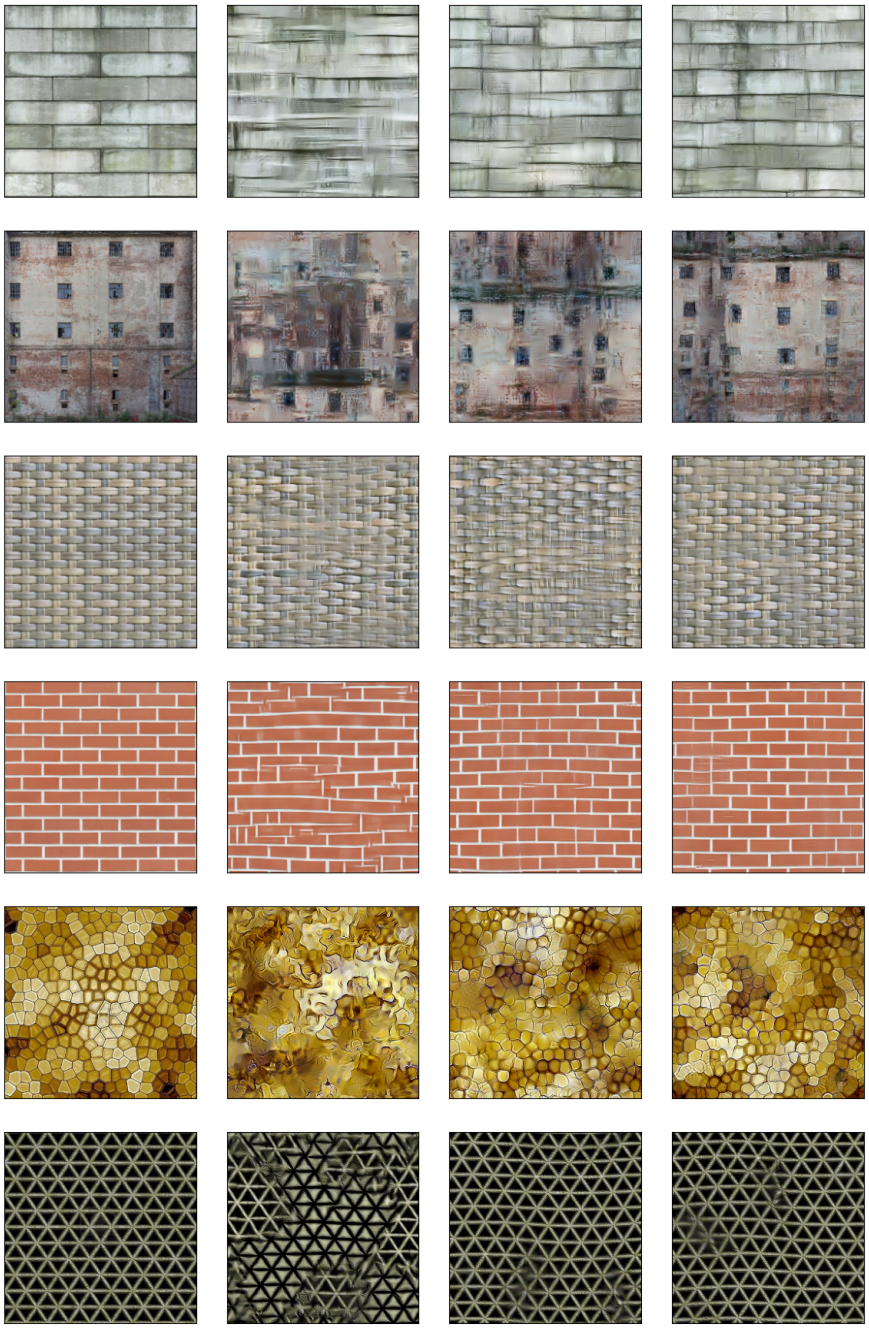


Figure 5.7 Synthesis results from one layer ($J = 6$) with different types of wavelet filters. **Left:** Original image. **Middle left:** First layer synthesis results with only odd wavelets. **Middle right:** First layer synthesis results with only even wavelets. **Right:** First layer synthesis results with both even and odd wavelets.

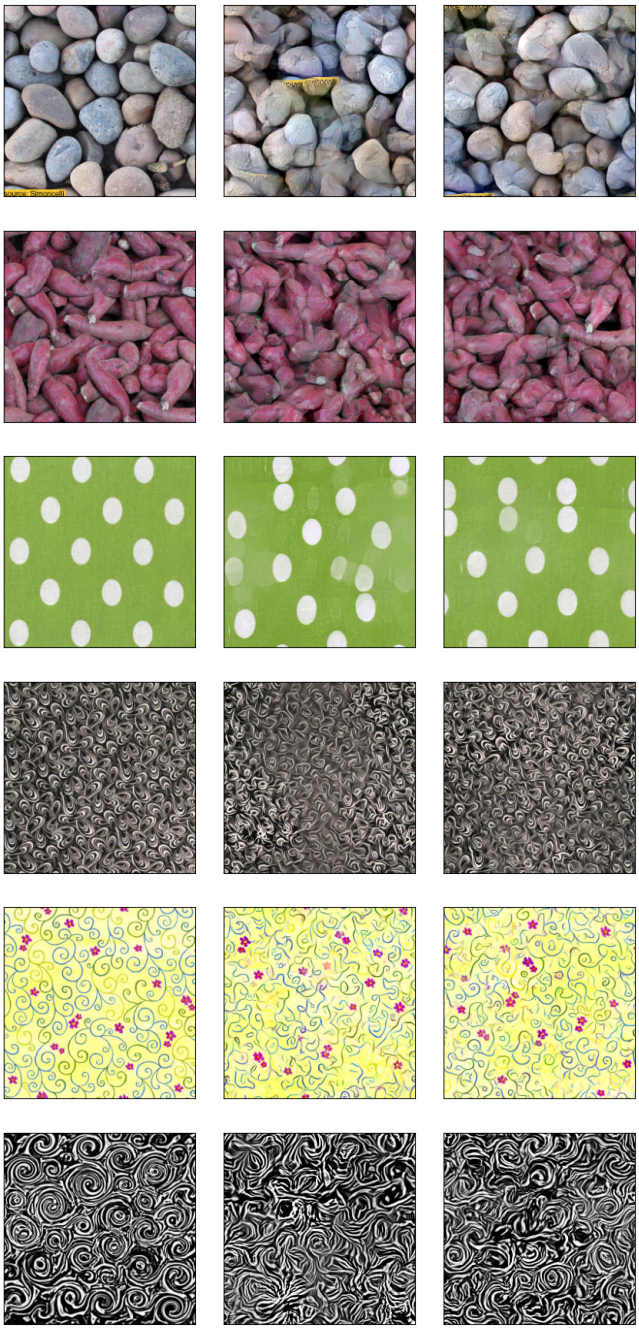


Figure 5.8 Synthesis results from two layers ($J = 5$) with/without omnidirectional wavelets. **Left:** Original image. **Middle:** 2nd layer synthesis with even and odd wavelets. **Right:** 2nd layer synthesis with even, odd and omnidirectional wavelets.

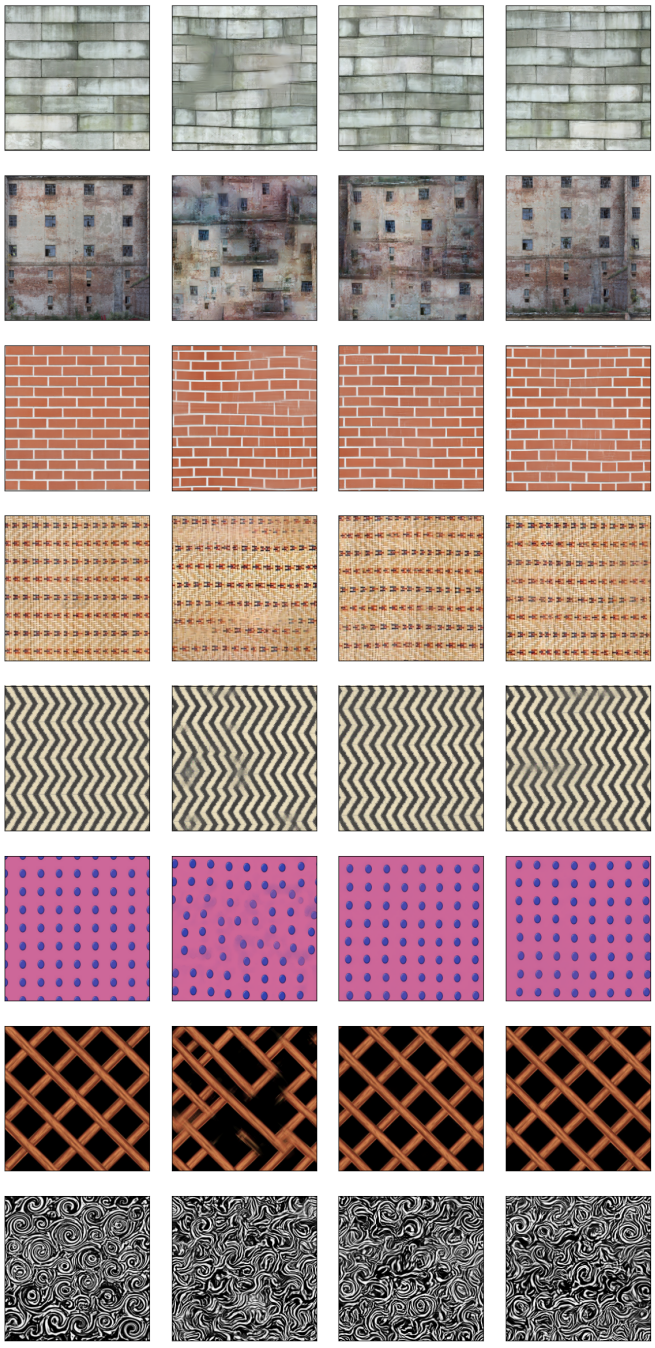


Figure 5.9 Synthesis results from two layers with different number of scales. **Left:** Original image. **Middle Left:** 2nd layer synthesis results with $J = 4$. **Middle Right:** 2nd layer synthesis results with $J = 5$. **Right:** 2nd layer synthesis results with $J = 6$.

5.4.3 Layers analysis

For many texture images, the one-layer model can synthesize images of high quality. However for images with more complicated structures, multiple layers can provide a boost in visual quality. As discussed in [10], deeper layers decompose high frequency information that is aggregated into large frequency bins with a single wavelet transform. Figure 5.10 shows images that achieved better quality with second layer statistics. For most images, the one-layer model captures general structures while the two-layer model refines the details, e.g., reproducing more accurate shapes, preserving long edges and swirls, fixing blurriness.

5.4.4 Methods comparison

We use even and odd directional wavelets, along with omnidirectional wavelets as our pre-selected filters in our final model. We also set $J = 5$ and use the two-layer model. Our results are compared with [2, 3] in Figure 5.11.

The textures synthesized by our model are generally equivalent to, or superior than, the images synthesized by Portilla and Simoncelli [2]. In fact, while not depicted in Figure 5.11, this result holds even if we restrict to one layer, indicating the combination of the ReLU and the selection of even, odd, and omnidirectional filters may provide a more complete statistical description of texture images.

With respect to Gatys *et al.* [3], the results are more nuanced. Our results are generally superior for textures with long, rigid edges even though their model is much deeper than our model. Additionally, textures with rigid patterns, but not necessarily long straight lines (e.g., left, last row; right, rows five, eight, nine) also have visually more appealing synthesis results via our method. These results can be attributed to the use of multiscale filters, although, even then, the results of Section 5.4.2 suggest that such an analysis might be too simplistic. For example, it is possible that a depth three wavelet network with $J = 4$ might also achieve similar performance to our current implementation with two layers and $J = 5$, which if so would raise questions with respect to other aspects of the VGG network.

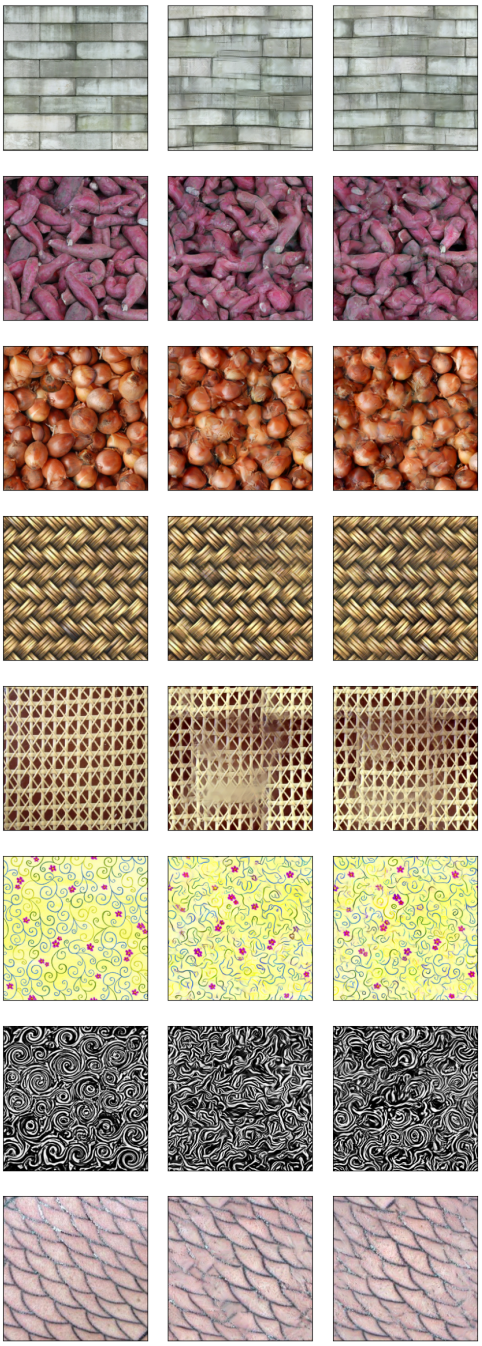


Figure 5.10 Synthesis results with one layer model and two layer model. **Left:** Original image. **Middle:** 1st layer synthesis results. **Right:** 2nd layer synthesis results.

Moving on we see that [3] obtains superior performance for images with long, non-rigid curves, such as the pebbles and onions (left, rows three and nine), fireworks (left, row eight) and swirling type images (right, rows seven and ten). Nevertheless, these results are in line with the observations in Section 5.4.3, which suggested that these types of textures require depth in order to capture their complex patterns.

Other images exhibit more subtle differences. For example, in the cracked earth image (left, row four), the synthesized image of [3] creates a bold effect on the cracks that is not present in the original. In the crossing image (left, row ten), the background illumination pattern is only correctly preserved by our method. Lastly, the multi-colored dots (right, last row) have a cleaner background with [3], but our method creates dots with colors that are not present in the original image, thus showing greater variability.

5.5 Implementation details

In this section, we describe more implementation details and analysis on numerics.

5.5.1 Reduction of second layer statistics

Recall at the second layer we compute:

$$U_J^2 x := \left\{ U_J^1 \left(\sum_{j=0}^{J-1} \sigma(\gamma \cdot x * \psi_{j,\alpha}^{\beta}) \right) : (\alpha, \beta) \in \{(\theta, e), (\theta, o), (\ell, p)\}, \;\; \theta \in \Theta_M, \;\; 0 \leq \ell < L \right\}.$$

that is:

$$\begin{aligned} U_J^2 x := \Bigg\{ & \sum_{j_1=0}^{J-1} \sigma(\gamma_1 \cdot x * \psi_{j_1,\alpha_1}^{\beta_1}) * \phi_J, \; \sum_{j_1=0}^{J-1} \sigma(\gamma_1 \cdot x * \psi_{j_1,\alpha_1}^{\beta_1}) * h, \\ & \sigma\left(\gamma_2 \cdot \left(\sum_{j_1=0}^{J-1} \sigma(\gamma_1 \cdot x * \psi_{j_1,\alpha_1}^{\beta_1}) \right) * \psi_{j_2,\alpha_2}^{\beta_2} \right) : \\ & (\alpha_1, \beta_1), (\alpha_2, \beta_2) \in \{(\theta, e), (\theta, o), (\ell, p)\}, \, \gamma_1, \gamma_2 \in \{+1, -1\} \Bigg\}. \end{aligned}$$

In particular for the third item, we apply another layer of the wavelet transform to the first layer responses. In numerical experiments for directional wavelets, we find $\sum_{j_1=0}^{J-1} \sigma(\gamma_1 \cdot x *$

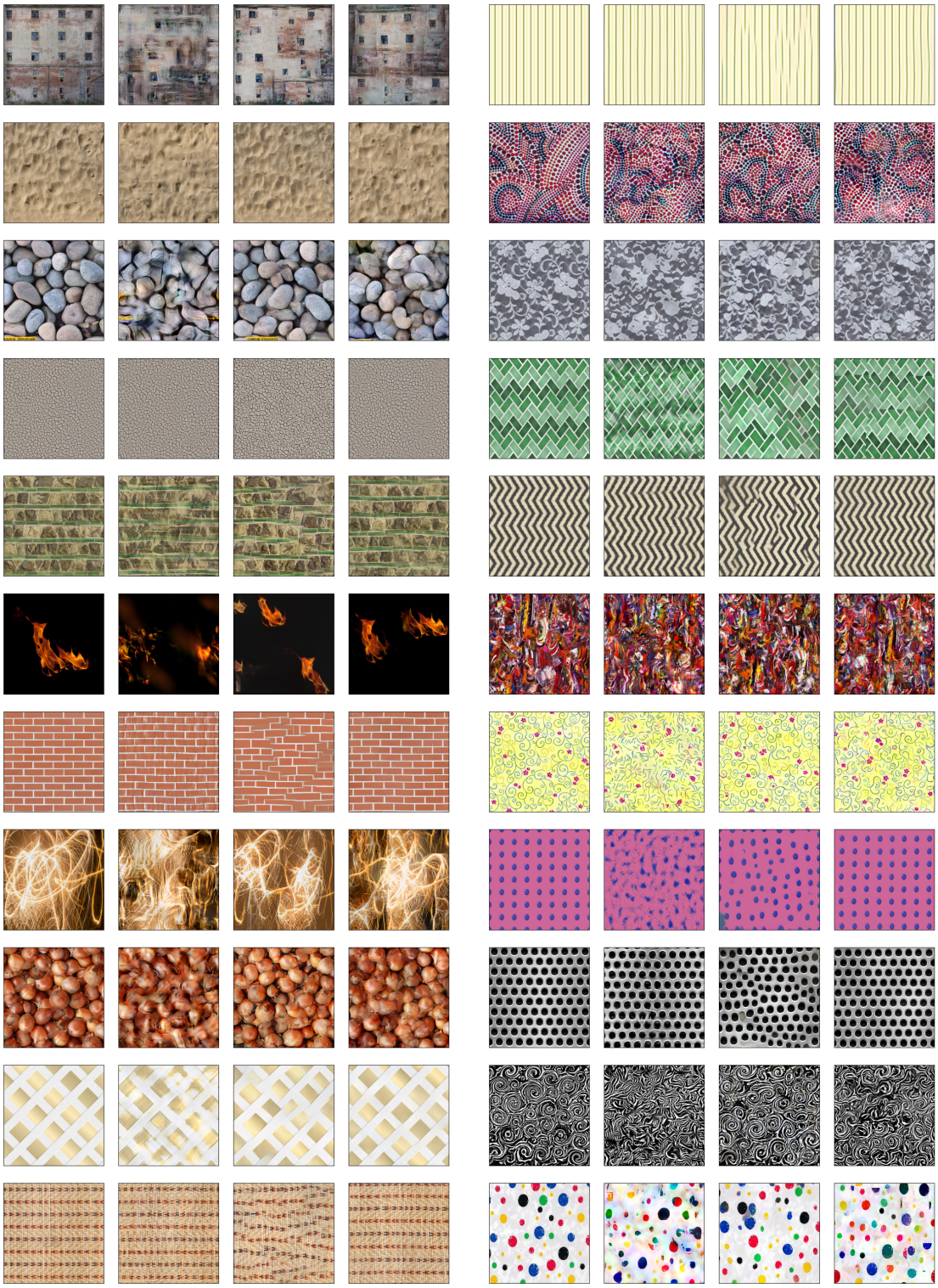


Figure 5.11 Synthesis results compared to other models. **Left:** Original images. **Middle Left:** Results from Portilla and Simoncelli [2]. **Middle Right:** Results from Gatys *et al.* [3]. **Right:** Results from our two layer model.

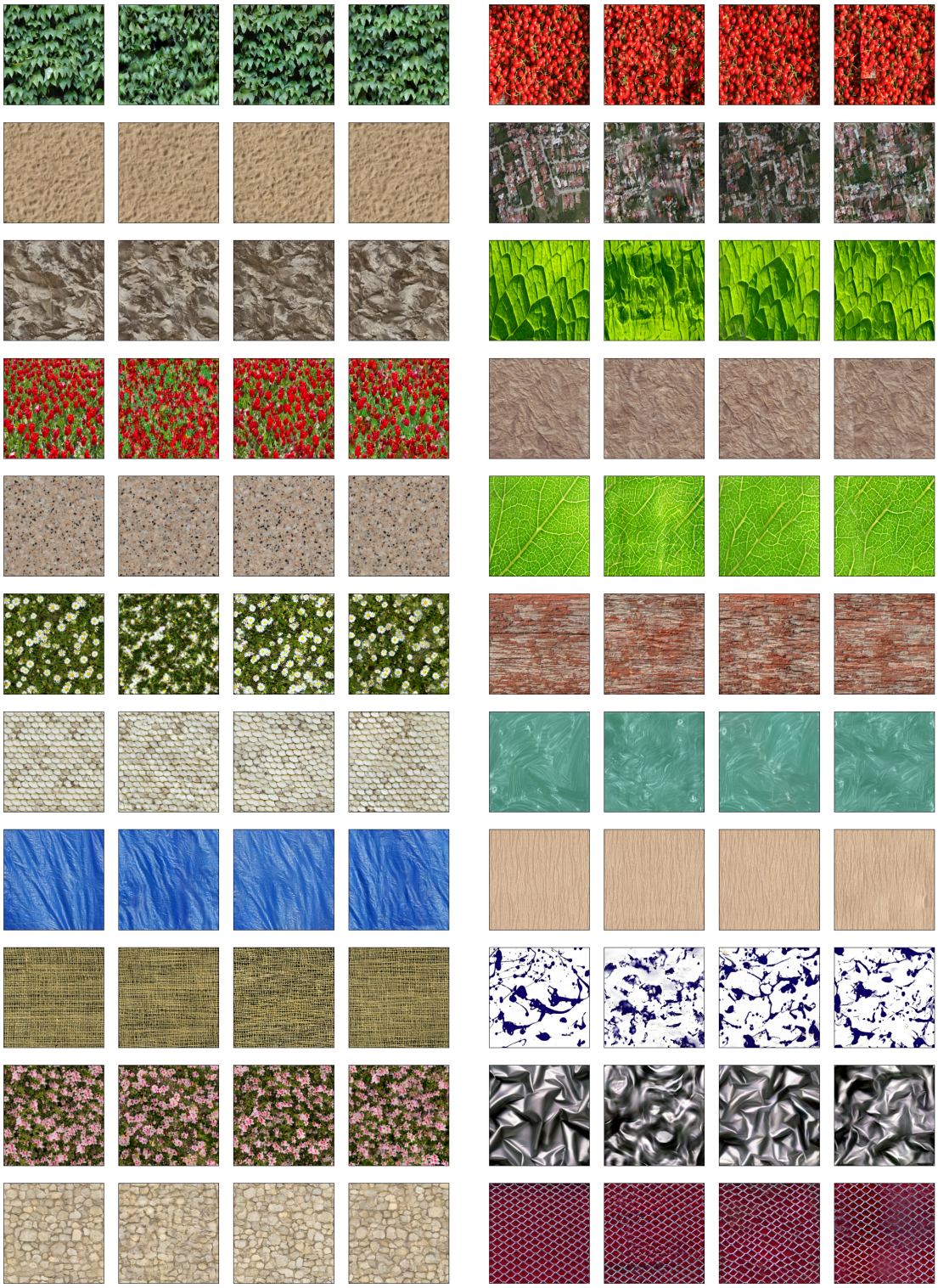


Figure 5.12 Synthesis results compared to other models. **Left:** Original images. **Middle Left:** Results from Portilla and Simoncelli [2]. **Middle Right:** Results from Gatys *et al.* [3]. **Right:** Results from our two layer model.

$\psi^{\beta_1}_{j_1,\theta_1}$) is essentially supported at the direction $\theta = \theta_1$. Therefore to reduce the number of total statistics and save computation, we set $\theta_2 = \theta_1$, i.e., the second layer of directional wavelets has the same direction as the first layer directional responses. We also add a residual wavelet to keep track of the residual frequencies and match the variance. We numerically verified with this restriction, there is little loss in the synthesized image quality.

5.5.2 Matching of second layer statistics

When we use the second layer statistics to synthesize texture images, we initialize the image with the first layer result. This is also equivalent to matching the first layer statistics until convergence, then matching both first layer and second layer statistics. In practice, we see the synthesized image has better quality than matching both first and second layer statistics from noise, i.e., matching both from the beginning.

Figure 5.13 compares the synthesized images using these two strategies. In particular, the middle right column shows synthesized images initialized from first layer result and the right column shows synthesized images initialized from noise. We notice matching only first layer statistics significantly reduces the first layer loss and the learned images are already well structured. Initialized from such images, second layer statistics refine the details. On the other hand, initializing from noise fails to reduce the first layer loss to a very small value. The synthesized images generally lose large structures and are not as good as results from the other strategy.

5.5.3 Analysis of number of iterations

In this section we discuss the relationship among the number of iterations, loss, and image quality. Figures 5.14, 5.15, 5.16 show the synthesis process of different textures. For all of these tests, we run second layer synthesis and run for 600 iterations. We plot the logarithm of the relative loss in these plots, i.e., $\log_{10}(\text{loss})$ where $\text{loss} = \frac{\|S^i_J x - S^i_J x^\star\|}{\|S^i_J x\|}$ for $i = 1, 2$, x is the reference image, and $x^\star$ is the synthesized image.

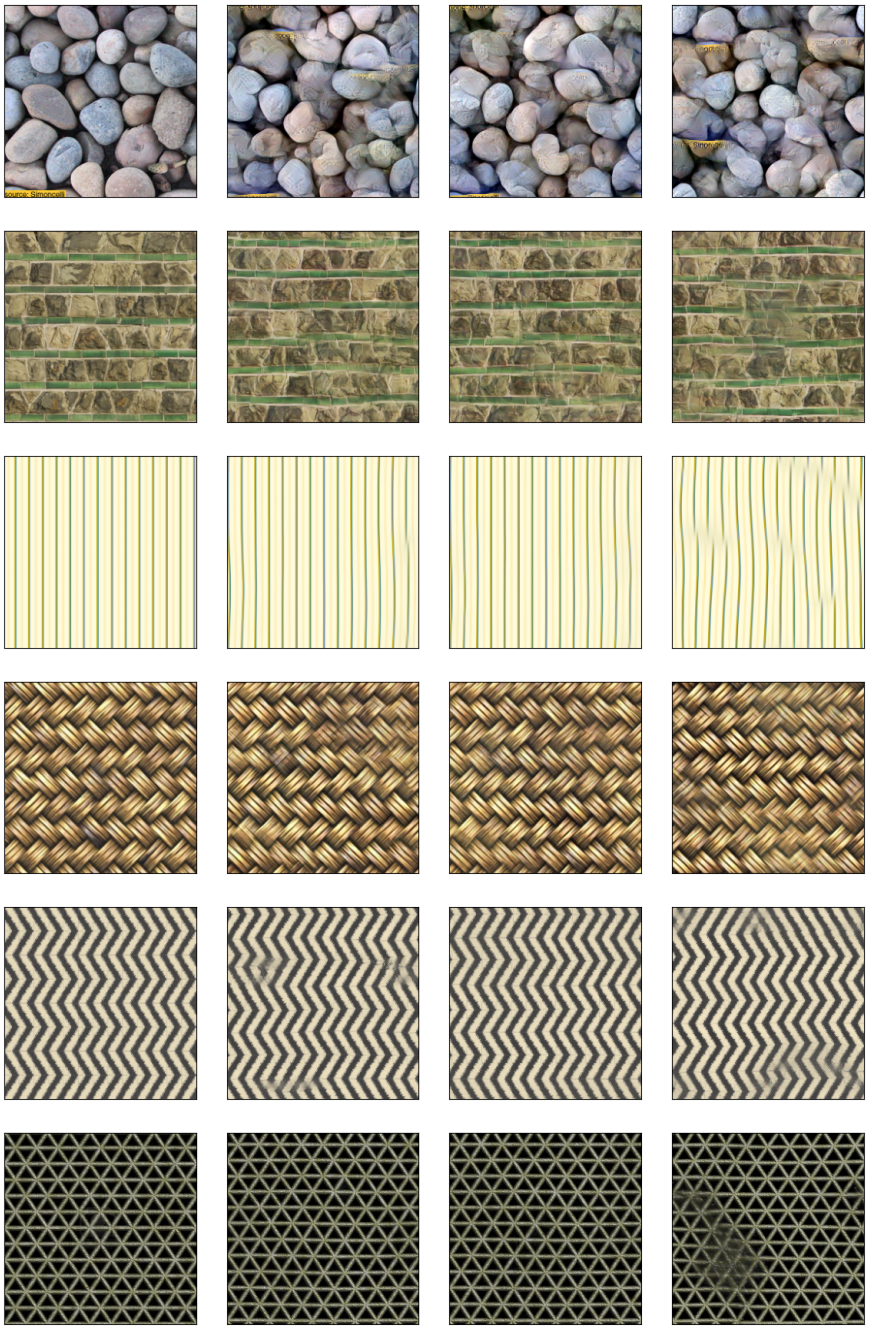


Figure 5.13 **Left:** Original images. **Middle left:** Images synthesized from first layer. **Middle right:** Images synthesized from second layer, initialized from first layer result. **Right:** Images synthesized from second layer, initialized from uniform noise.

Figure 5.14 shows the synthesis process of micro-textures plus flowers. We start from noise to match the second layer to simplify the analysis. After iteration 0, the images already look much better than noise, although the colors are not fully matched. At iteration 40, the second layer loss is reduced to approximately $10^{-3.5}$ and the synthesized images are of high quality. As we continue the algorithm, at iteration 100 and 590, the images smoothly shift from one sample to another sample which are from the same texture class and the loss barely changed. In general, for such micro-textures, the algorithm converges fast and the loss is reduced long before we stop.

Figure 5.15 shows the synthesis process for textures that have more structure. Still we start from noise. At iteration 0, the colors are mismatched, similar to Figure 5.14. At iteration 40 and 100, the colors are matched and general structures are learned, but the holes and frames are not aligned perfectly. Also the relative loss is not reduced. Finally at iteration 590, the second layer loss is reduced to around 10^{-4} and the synthesized images are of good quality. The algorithm on these types of images generally takes longer to converge than for micro-textures. As the number of iterations increases, the image quality is continuously improved.

Figure 5.16 shows the synthesis process of a texture that has intricate patterns. Here we start from the first layer result to show how the second layer statistics improve the image quality. At iteration 0, which is essentially the synthesized image from the first layer experiment, there are barely swirls but only curves. As the algorithm goes on, we notice the first layer loss actually increases while the second layer loss is decreasing. The increasing first layer loss generally does not affect image quality so long as the second layer loss decreases. By the last iteration, the synthesized image contains longer and smoother swirls.

5.6 Conclusion

We presented a unique texture synthesis algorithm that melds aspects of Portilla and Simoncelli [2], Gatys *et al.* [3], and Mallat [10], while also incorporating new ideas on filter design,

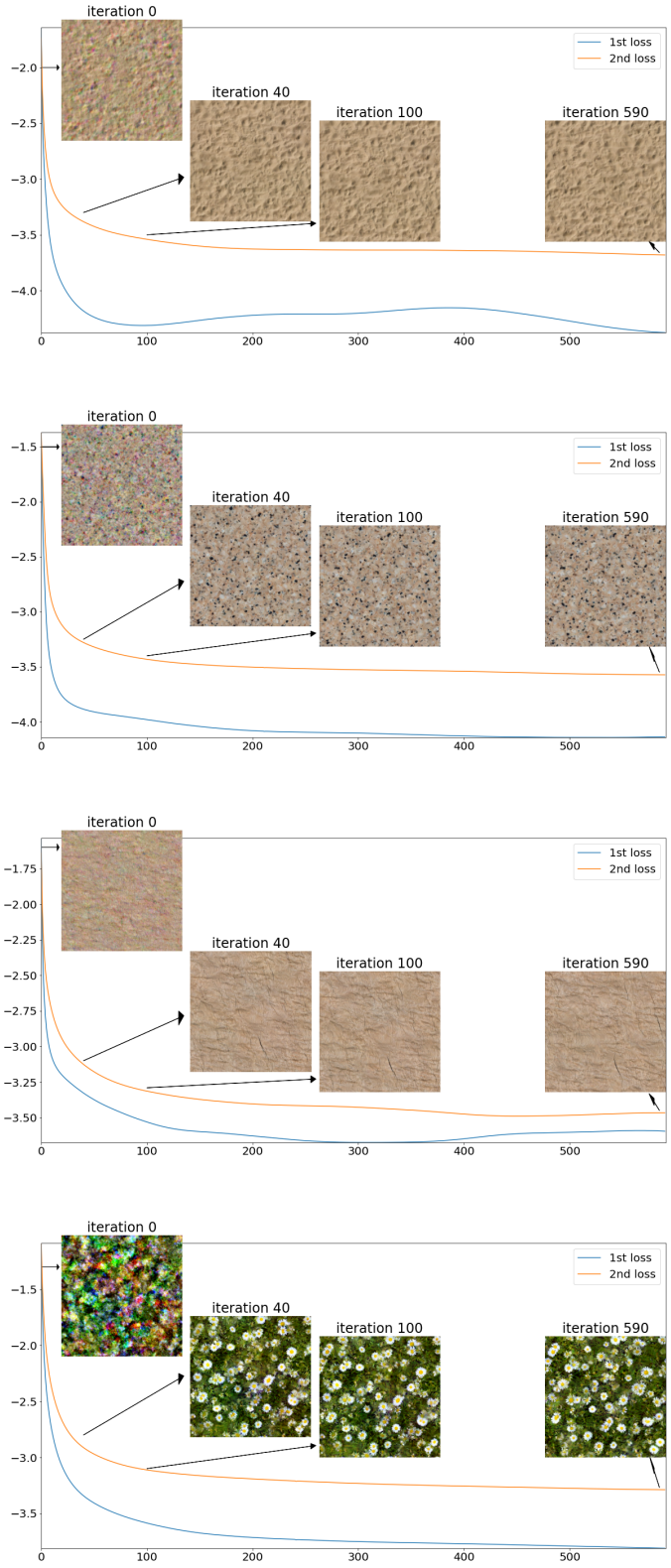


Figure 5.14 The synthesis process for different micro-textures plus flowers.

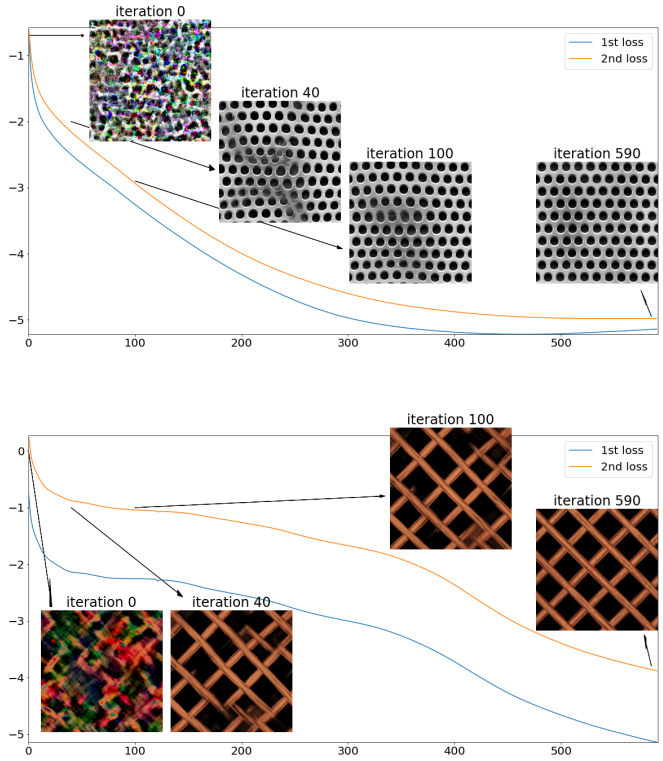


Figure 5.15 The synthesis process for different macro-textures with rigid patterns.

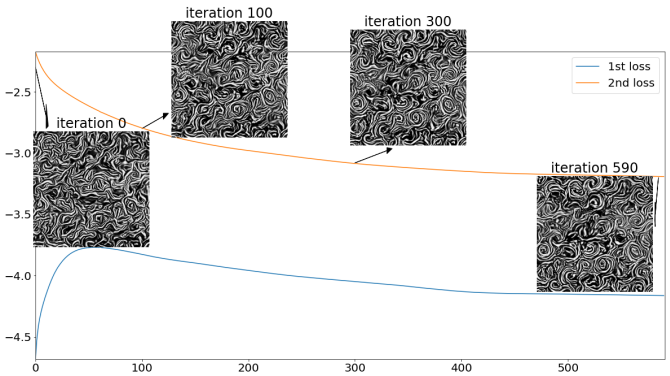


Figure 5.16 The synthesis process when initializing from the first layer synthesis, for a texture with complex patterns.

multi-layer structure, and the invertibility of CNNs. Our numerical analysis provides insight into the workings of statistics-based texture synthesis algorithms. Synthesized textures are competitive with the state-of-the-art and in some cases superior to [3], thus providing a potential alternative. Nevertheless, issues such as the trade-off between network depth and filter scale are not fully resolved, and invite future research endeavors, which we will explore in the next chapter.

CHAPTER 6

MULTILAYER MODEL ANALYSIS

6.1 Introduction

In this chapter, we continue our analysis of the statistical texture synthesis model proposed in Chapter 5. Intuitively, large filters and deep layers increase the size of receptive fields in the network and preserve long range dependence of the image, but they do so in different ways. From the numerical tests, we noticed the model with large filters keeps long straight lines and the multilayer model preserves long curves. On the other hand, the one layer wavelet model does not capture intricate curves and the multilayer VGG model struggles to reproduce rigid patterns. There has also been a discussion on the trade-off between model depth and the size of convolution filters in [15, 3].

In this chapter, we develop mathematical analysis on filter size and model depth. We investigate certain types of texture classes and discuss the necessity of large filters. We also discuss the relationship between model depth and the higher order derivatives of a signal. Specifically we continue the analysis of Chapter 5 in the following ways:

- We proposed a model to represent the ReLU architecture.
- We provide new perspective on understanding the even and odd wavelets.
- We prove if the largest filter is not large enough to meet certain criteria, the gram matrix fails to distinguish different textures.
- We prove with small filters, deep models intricately partition the high order derivatives of the given signal.

6.2 Multilayer multiscale model

6.2.1 Filters

Let $\phi : \mathbb{R}^2 \to \mathbb{R}$ be an low pass filter with the following conditions:

- $\phi(u) \geq 0, \forall u \in \mathbb{R}^2$.
- ϕ is isotropic, i.e., $\phi(u) = \phi(R_\theta u), \forall \theta$. $R_\theta : \mathbb{R}^2 \to \mathbb{R}^2$ is the rotation operator (rotate by θ).
- ϕ is compactly supported in $[-a, a]^2 \in \mathbb{R}^2$.

Define the wavelet family $\{\psi_{k,n}\}_{k,n}$ such that

$$\psi_{k,n}(u) = D^{(n)}_{\vec{v}_k}\phi(u) \tag{6.1}$$

is the n-th derivative of ϕ along the direction $\vec{v}_k$ where $\vec{v}_k = R_\theta(\vec{v}_0)$, $\vec{v}_0 = (1, 0)$, $\theta = \frac{k\pi}{K}$. K is the total number of rotations. The filters can also be dilated:

$$\psi_{j,k,n}(u) = 2^{-2j}\psi_{k,n}(2^{-j}u),\ 0 \leq j < J$$

Since

$$\begin{aligned} D^{(n)}_{\vec{v}_k}\phi_j(u) &= D^{(n)}_{\vec{v}_k}(2^{-2j}\phi(2^{-j}u)) \\ &= 2^{-2j} \cdot 2^{-nj} D^{(n)}_{\vec{v}_k}(\phi(2^{-j}u)) \\ &= 2^{-nj}\psi_{j,k,n}(u), \end{aligned}$$

we also have $\psi_{j,k,n}(u) = 2^{nj} D^{(n)}_{\vec{v}_k}\phi_j(u)$.

Remark 3. *When $n = 1$, $\psi_{k,n}$ is an odd directional wavelet with direction $\theta = \frac{k\pi}{K}$; when $n = 2$, $\psi_{k,n}$ is an even directional wavelet with direction $\theta = \frac{k\pi}{K}$.*

The derivative wavelets defined in Equation 6.1 are not exactly the same directional wavelets used in Chatper 5. The definition is different and the derivative wavelets has compact support. However since the derivative wavelets at $n = 1, 2$ look similar to the

directional wavelets, they can be thought of as a generalized version of directional wavelets with more flexibility by adjusting n. Therefore in the following context, we are going to analyze the filters from Chapter 5 through the derivative wavelets defined in Equation 6.1.

6.2.2 Wavelet transform

Let $x : \mathbb{R}^2 \to \mathbb{R}$ be a 2D signal. The wavelet transform can be rewritten as

$$\begin{aligned} x * \psi_{j,k,n}(u) &= 2^{nj} x * D^{(n)}_{\vec{v}_k} \phi_j(u) \\ &= 2^{nj} D^{(n)}_{\vec{v}_k} (x * \phi_j)(u) \end{aligned}$$

The above equation shows the wavelet transform of a signal with a derivative wavelet equals to the derivative of the signal convolved with the low pass up to a constant 2^{nj}, which depends on the dimension n and scale parameter j.

Lemma 5. *If x is isotropic, i.e., $x(u) = x(R_\theta u), \forall u$, then $x * \psi_{j,k_1,n} = x * \psi_{j,k_2,n}, \forall k_1, k_2$.*

Proof. If x is isotropic, then

$$\begin{aligned} x * \psi_{j,k_1,n}(u) &= 2^{nj} D^{(n)}_{\vec{v}_{k_1}} (x * \phi_j)(u) \\ (x, \phi_j \text{ are isotropic}) &= 2^{nj} D^{(n)}_{\vec{v}_{k_2}} (x * \phi_j)(u) \\ &= x * \psi_{j,k_2,n}(u) \end{aligned}$$

□

6.2.3 Gram matrix

Recall in Chapter 5, we use the gram matrix between ReLU wavelet response to represent a texture image for synthesis and achieved good performance. Here we generalize the definition

for derivative filters. Define gram matrix as

$$\begin{aligned}
&Gx(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) \\
&\quad := \langle \mathrm{ReLU}(\epsilon_1 \cdot x * \psi_{j_1,k_1,n_1}), \mathrm{ReLU}(\epsilon_2 \cdot x * \psi_{j_2,k_2,n_2}) \rangle \\
&\quad = \int_{\mathbb{R}^2} \mathrm{ReLU}(\epsilon_1 \cdot x * \psi_{j_1,k_1,n_1}(u)) \cdot \mathrm{ReLU}(\epsilon_2 \cdot x * \psi_{j_2,k_2,n_2}(u)) du \\
&\quad = 2^{n_1 j_1 + n_2 j_2} \int_{\mathbb{R}^2} \mathrm{ReLU}\left(\epsilon_1 \cdot D^{(n_1)}_{\vec{v}_{k_1}}(x * \phi_{j_1})(u)\right) \cdot \mathrm{ReLU}\left(\epsilon_2 \cdot D^{(n_2)}_{\vec{v}_{k_2}}(x * \phi_{j_2})(u)\right) du
\end{aligned} \tag{6.2}$$

where $\epsilon_1, \epsilon_2 \in \{+1, -1\}$. In order to better analyze the correlation, we are going to decompose the integral into the correlation and the integral support. Let

$$\begin{aligned}
Ex(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) := \Big\{ u \in \mathbb{R}^2 : &\epsilon_1 \cdot D^{(n_1)}_{\vec{v}_{k_1}}(x * \phi_{j_1})(u) \geq 0, \vec{v}_{k_1} = R_{\theta_1}\vec{v}_0, \theta_1 = \frac{k_1\pi}{K}, \\
&\epsilon_2 \cdot D^{(n_2)}_{\vec{v}_{k_2}}(x * \phi_{j_2})(u) \geq 0, \vec{v}_{k_2} = R_{\theta_2}\vec{v}_0, \theta_2 = \frac{k_2\pi}{K} \Big\}
\end{aligned} \tag{6.3}$$

denote the ReLU support that the integral is integrated on. We can rewrite the gram matrix as

$$\begin{aligned}
&Gx(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) = \\
&\quad \epsilon_1\epsilon_2 2^{n_1 j_1 + n_2 j_2} \int_{Ex(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2)} D^{(n_1)}_{\vec{v}_{k_1}}(x * \phi_{j_1})(u) \cdot D^{(n_2)}_{\vec{v}_{k_2}}(x * \phi_{j_2})(u) du
\end{aligned} \tag{6.4}$$

The following special cases include the statistical representations we computed at the first layer for texture images in Chapter 5.

- Consider $n_1 = n_2 = 0$, there is no derivatives or directional derivatives. We can assume $k_1 = k_2 = 0$. If $x \geq 0$, e.g., natural images, then $x * \phi_j \geq 0$. Then

$$\begin{aligned}
Gx(j_1,j_2,0,0,0,0,+1,+1) &= \int_{\mathbb{R}^2} x * \phi_{j_1} \cdot x * \phi_{j_2} dt \\
&= \frac{1}{(2\pi)^2} \int_{\mathbb{C}^2} \widehat{x * \phi_{j_1}}(\omega) \cdot \widehat{x * \phi_{j_2}}^*(\omega) d\omega \\
&= \frac{1}{(2\pi)^2} \int_{\mathbb{C}^2} \widehat{x}(\omega) \cdot \widehat{\phi_{j_1}}(\omega) \cdot \widehat{x}^*(\omega) \cdot \widehat{\phi_{j_2}}^*(\omega) d\omega \\
(\textit{suppose } j_1 \geq j_2) &\approx \frac{1}{(2\pi)^2} \int_{\mathbb{C}^2} |\widehat{x}(\omega)|^2 \cdot |\widehat{\phi_{j_1}}(\omega)|^2 d\omega \\
&= \|x * \phi_{j_1}\|_2^2
\end{aligned}$$

Therefore $Gx(j_1, j_2, 0, 0, 0, 0, +1, +1)$ captures the ℓ_2 norm of the low pass responses.

- $Gx(j_1, j_2, k_1, k_2, 1, 1, \epsilon_1, \epsilon_2)$ captures the odd wavelet gram matrix between different angles and scales.
- $Gx(j_1, j_2, k_1, k_2, 2, 2, \epsilon_1, \epsilon_2)$ captures the even wavelet gram matrix between different angles and scales.
- $Gx(j_1, j_2, k_1, k_2, 1, 2, \epsilon_1, \epsilon_2)$ captures the gram matrix between even and odd wavelets cross angles and scales.

6.3 Texture with multiple straight lines

Recall in Figure 5.9 (fourth row) and Figure 5.11 (right, first row) from Chapter 5, our model with large scale filters preserve long straight lines while both the model with only small scale filters and the model from Gatys fail to do so. In this section, we consider a special class of signals with parallel straight lines. We prove that multiscale, especially large scale filters, are essential for recovering textures with long lines, which we define in the following.

Let $x(u_1, u_2) = \sum_i \alpha_i \mathbb{1}_{\delta_i}(u_1)$ where $\delta_i \in \Delta \subset \mathbb{R}$. Then we have the wavelet transform as

$$\begin{aligned} x * \psi_{j,k,n}(u) &= 2^{nj} D^{(n)}_{\vec{v}_k}(x * \phi_j)(u) \\ &= 2^{nj} D^{(n)}_{\vec{v}_k}(\sum_{\delta_i \in \Delta} \alpha_i \cdot \phi_j^{1d}(u_1 - \delta_i)) \end{aligned}$$

where $u = (u_1, u_2)$ and ϕ_j^{1d} is the one dimensional version of ϕ_j. Let $\vec{v}_0 = (1, 0), \vec{v}_{-1} = (0, 1)$. Then

$$\begin{aligned} D^{(n)}_{\vec{v}_0}(x * \phi_j)(u) &= \sum_{\delta_i} \alpha_i \frac{d^n}{du_1^n} \phi_j^{1d}(u_1 - \delta_i) \\ D^{(n)}_{\vec{v}_{-1}}(x * \phi_j)(u) &= 0 \end{aligned}$$

Therefore for any $\vec{v}_k = R_{\theta_k}(\vec{v}_0) = (\cos\theta_k, \sin\theta_k)$,

$$\begin{aligned} D^{(n)}_{\vec{v}_k}(x * \phi_j)(u) &= \cos^n\theta_k \cdot D^{(n)}_{\vec{v}_0}(x * \phi_j)(u) \\ &= \cos^n\theta_k \cdot \sum_{\delta_i} \alpha_i \frac{d^n}{du_1^n}\phi_j^{1d}(u_1 - \delta_i) \end{aligned}$$

And also,

$$x * \psi_{j,k,n}(u) = 2^{nj} \cdot \cos^n\theta_k \cdot \sum_{\delta_i} \alpha_i \frac{d^n}{du_1^n}\phi_j^{1d}(u_1 - \delta_i) \tag{6.5}$$

6.3.1 Gram matrix

The gram matrix for this type of signal is

$$\begin{aligned} &Gx(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) \\ &\quad := \langle(\mathrm{ReLU}(\epsilon_1 \cdot x * \psi_{j_1,k_1,n_1}), \mathrm{ReLU}(\epsilon_2 \cdot x * \psi_{j_2,k_2,n_2})\rangle \\ &\quad = \int_{Ex(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2)} \epsilon_1\epsilon_2 2^{n_1 j_1 + n_2 j_2} \cdot \cos^{n_1}\theta_{k_1} \cdot \cos^{n_2}\theta_{k_2} \sum_{\delta_i} \alpha_i \frac{d^{n_1}}{du_1^{n_1}}\phi_{j_1}^{1d}(u_1 - \delta_i) \\ &\quad \cdot \sum_{\delta_i} \alpha_i \frac{d^{n_2}}{du_1^{n_2}}\phi_{j_2}^{1d}(u_1 - \delta_i)\, du_1 \\ &\quad = \epsilon_1\epsilon_2 2^{n_1 j_1 + n_2 j_2} \cdot \cos^{n_1}\theta_{k_1} \cdot \cos^{n_2}\theta_{k_2} \int_{Ex(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2)} \\ &\quad \sum_{\delta_{i_1}} \sum_{\delta_{i_2}} \alpha_{i_1}\alpha_{i_2} \frac{d^{n_1}}{du_1^{n_1}}\phi_{j_1}^{1d}(u_1 - \delta_{i_1}) \frac{d^{n_2}}{du_1^{n_2}}\phi_{j_2}^{1d}(u_1 - \delta_{i_2})\, du_1 \end{aligned} \tag{6.6}$$

Note

$$\begin{aligned} Ex(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) = \Bigg\{ u \in \mathbb{R}^2 : \epsilon_1 \cdot \cos^{n_1}\theta_{k_1} \sum_{\delta_i} \alpha_i \frac{d^{n_1}}{du_1^{n_1}}\phi_{j_1}^{1d}(u_1 - \delta_i) > 0, \\ \epsilon_2 \cdot \cos^{n_2}\theta_{k_2} \sum_{\delta_i} \alpha_i \frac{d^{n_2}}{du_1^{n_2}}\phi_{j_2}^{1d}(u_1 - \delta_i) > 0 \Bigg\} \end{aligned} \tag{6.7}$$

Equation 6.7 shows the integral support only depends on u_1. Moreover, it is contained in the union of the band around $u_1 = \delta_i$. Our next section explores more on when J is small and the bands don't overlap among different δ_i.

Besides the following remark shows, when the scales (j_1, j_2), derivative orders (n_1, n_2) and signs (ϵ_1, ϵ_2) are fixed and only vary the filters' directions (k_1, k_2), the gram matrix between the two ReLU responses is proportional to a value that depends on k_1, k_2.

Remark 4. $\frac{Gx(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2)}{Gx(j_1,j_2,k_1',k_2',n_1,n_2,\epsilon_1,\epsilon_2)} = \frac{\cos^{n_1}\theta_{k_1}\cdot\cos^{n_2}\theta_{k_2}}{\cos^{n_1}\theta_{k_1'}\cdot\cos^{n_2}\theta_{k_2'}}$

Moreover, the following remark summarizes, when the angle $\theta_{k_1} = m\pi + \frac{\pi}{2}$ or $\theta_{k_2} = m\pi + \frac{\pi}{2}$, the gram matrix is zero.

Remark 5. *If* $\cos^{n_1}\theta_{k_1} = 0$ *or* $\cos^{n_2}\theta_{k_2} = 0$*, then*

$$Gx(j_1, j_2, k_1, k_2, n_1, n_2, \epsilon_1, \epsilon_2) = 0,\ \forall j_1, j_2, n_1, n_2, \epsilon_1, \epsilon_2.$$

6.3.2 Small J

Now let us assume J is small. Recall ϕ is supported in $[-a, a]$, then ϕ_j has support $[-2^j a, 2^j a]$ and ϕ_{J-1} has support $[-2^{J-1}a, 2^{J-1}a]$. Let d_{min} demote the smallest pairwise distance between the diracs in x. The following theorem proves when J is small, the gram matrix cannot distinguish one line texture from another under certain conditions.

Theorem 14. *Let* $x_1(u) = \sum_{\delta_{i_1}\in\Delta_1} \alpha_{i_1}\mathbb{1}_{\delta_{i_1}}(u_1)$ *and* $x_2(u) = \sum_{\delta_{i_2}\in\Delta_2} \beta_{i_2}\mathbb{1}_{\delta_{i_2}}(u_1)$*. Let* d^1_{min} *and* d^2_{min} *denote the smallest pairwise distance between diracs of* x_1 *and* x_2*, respectively. Assume* ϕ *is supported in* $[-a, a]$*. If* $2^J a < d^i_{min}$ *for* $i \in \{1, 2\}$ *and* $\sum_{\delta_{i_1}\in\Delta_1} \alpha^2_{i_1} = \sum_{\delta_{i_2}\in\Delta_2} \beta^2_{i_2}$*, then* $Gx_1 = Gx_2$.

Note the above theorem is related to Theorem 3 in Chapter 3. The 2D signal in this section can be decomposed into a sparse signal along one direction and a constant along another direction. Theorem 14 provides a necessary condition to distinguish two signals of such type while Theorem 3 provides a sufficient condition to identify a 1D sparse signal.

Proof of Theorem 17. Consider the signal x_1. When $2^J a < d^1_{min}$, $\phi^{1d}_{j_1}(u_1 - \delta_i)$ does not overlap with $\phi^{1d}_{j_2}(u_1 - \delta_{i'})$ for any $\delta_i, \delta_{i'} \in \Delta_1$, $i \neq i'$, $0 \leq j_1, j_2 < J$. Therefore the nonzero set from

(6.7) can be decomposed into

$$Ex_1(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) = \bigcup_{\delta_i\in\Delta_1}\Big\{u\in\mathbb{R}^2 : \epsilon_1\cdot\cos^{n_1}\theta_{k_1}\alpha_i\frac{d^{n_1}}{du_1^{n_1}}\phi_{j_1}^{1d}(u_1-\delta_i)>0, \\ \epsilon_2\cdot\cos^{n_2}\theta_{k_2}\alpha_i\frac{d^{n_2}}{du_1^{n_2}}\phi_{j_2}^{1d}(u_1-\delta_i)>0\Big\} \\ :== \bigcup_{\delta_i\in\Delta_1} Ex_1^i(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) \tag{6.8}$$

Note

$$Ex_1^i(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) = Ex_1^{i'}(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) - \delta_{i'} + \delta_i \tag{6.9}$$

i.e., if

$$t' \in Ex_1^{i'}(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2),$$

then

$$t = t' - \delta_{i'} + \delta_i \in Ex_1^i(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2).$$

Also the argument inside the integral from Equation 6.6 is nonzero only along the diagonal:

$$\sum_{\delta_{i_1}}\sum_{\delta_{i_2}}\alpha_{i_1}\alpha_{i_2}\frac{d^{n_1}}{dt_1^{n_1}}\phi_{j_1}^{1d}(t_1-\delta_{i_1})\frac{d^{n_2}}{dt_1^{n_2}}\phi_{j_2}^{1d}(t_1-\delta_{i_2}) = \sum_{\delta_{i_1}}\alpha_{i_1}^2\frac{d^{n_1}}{dt_1^{n_1}}\phi_{j_1}^{1d}(t_1-\delta_{i_1})\frac{d^{n_2}}{dt_1^{n_2}}\phi_{j_2}^{1d}(t_1-\delta_{i_1}) \tag{6.10}$$

Inserting (6.8) and (6.10) into (6.6) we get

$$\begin{aligned} &Gx_1(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) \\ &\quad= \epsilon_1\epsilon_2 2^{n_1j_1+n_2j_2}\cdot\cos^{n_1}\theta_{k_1}\cdot\cos^{n_2}\theta_{k_2} \\ &\quad\sum_{\delta_i\in\Delta_1}\int_{Ex_1^i(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2)}\alpha_{i_1}^2\frac{d^{n_1}}{du_1^{n_1}}\phi_{j_1}^{1d}(u_1-\delta_{i_1})\frac{d^{n_2}}{du_1^{n_2}}\phi_{j_2}^{1d}(u_1-\delta_{i_1})du_1 \\ &\quad= \epsilon_1\epsilon_2 2^{n_1j_1+n_2j_2}\cdot\cos^{n_1}\theta_{k_1}\cdot\cos^{n_2}\theta_{k_2} \\ &\quad\sum_{\delta_i\in\Delta_1}\alpha_{i_1}^2\int_{Ex_1^i(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2)}\frac{d^{n_1}}{du_1^{n_1}}\phi_{j_1}^{1d}(u_1-\delta_{i_1})\frac{d^{n_2}}{du_1^{n_2}}\phi_{j_2}^{1d}(u_1-\delta_{i_1})du_1 \\ \text{(with (6.9))} &= \epsilon_1\epsilon_2 2^{n_1j_1+n_2j_2}\cdot\cos^{n_1}\theta_{k_1}\cdot\cos^{n_2}\theta_{k_2}\left(\sum_{\delta_i\in\Delta_1}\alpha_{i_1}^2\right) \\ &\quad\int_{Ex_1^0(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2)}\frac{d^{n_1}}{du_1^{n_1}}\phi_{j_1}^{1d}(u_1-\delta_0)\frac{d^{n_2}}{du_1^{n_2}}\phi_{j_2}^{1d}(u_1-\delta_0)du_1 \end{aligned} \tag{6.11}$$

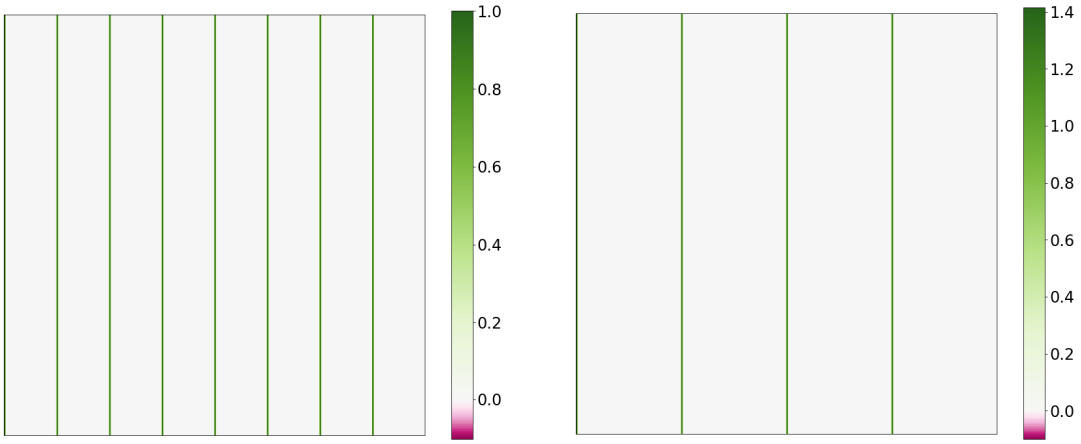


Figure 6.1 Two line texture images that have the same summation of height squared. **Left:** $x_1(u) = \sum_{i=1}^{8} \mathbb{1}_{32i}(u_1),\ u \in [\mathbb{Z} \bigcap [0, 256)]^2$. **Right:** $x_2(u) = \sum_{i=1}^{4} \sqrt{2} \cdot \mathbb{1}_{64i}(u_1),\ u \in [\mathbb{Z} \bigcap [0, 256)]^2$.

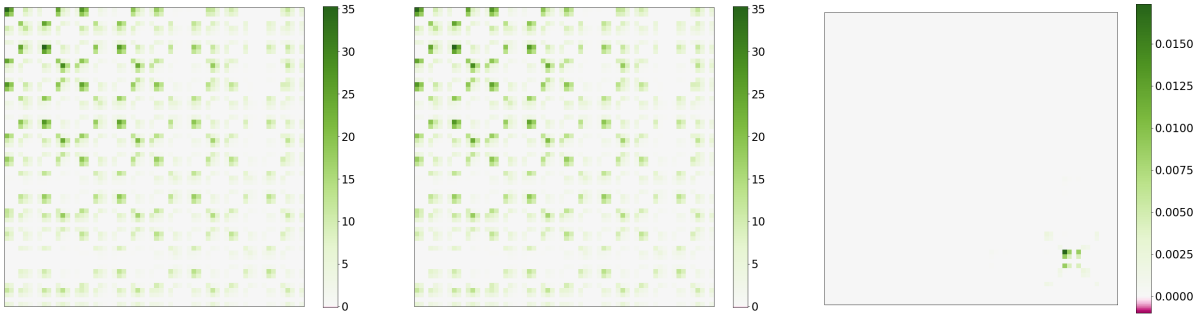


Figure 6.2 The gram matrices and the difference for the two line textures in Figure 6.1. **Left:** The gram matrix Gx_1 between ReLU responses for texture x_1. **Middle:** The gram matrix Gx_2 between ReLU responses for texture x_2. **Right:** The difference between the two gram matrices $|Gx_1 - Gx_2|$.

One can get similar results for x_2 with $\sum_{\delta_i \in \Delta_1} \alpha_{i_1}^2$ substituted by $\sum_{\delta_i \in \Delta_2} \beta_{i_1}^2$. Then the conclusion is proved. □

Figure 6.1 and 6.2 verify Theorem 14 numerically. The two line textures $x_1(u) = \sum_{i=1}^{8} \mathbb{1}_{32i}(u_1)$ and $x_2(u) = \sum_{i=1}^{4} \sqrt{2} \cdot \mathbb{1}_{64i}(u_1),\ u \in [\mathbb{Z} \bigcap [0, 256)]^2$ both have summation of height squared equal to 8. For the selected wavelet family, we choose $J = 2$ so that $2^J a < d_{min}$ where $d_{min} = 32$ in this example. Figure 6.2 shows there is little difference between the two gram matrices except for the numerical errors, thus verifying our conclusion numerically.

6.3.3 Deep layers

One may argue that the gram matrix fails in distinguishing the two signals due to the fact that we only have one layer of convolution. In this section, we prove that even with multiple layers, if the filters are tiny, the gram matrix still fails to distinguish different line textures.

Recall the first layer wavelet coefficients for the line texture:

$$x * \psi_{j,k,n}(u) = 2^{nj} \cdot \cos^n \theta_k \cdot \sum_{\delta_i} \alpha_i \frac{d^n}{du_1^n} \phi_j^{1d}(u_1 - \delta_i) \tag{6.12}$$

Remind $\sigma(x) = (ReLU)(x)$. Let $\frac{d^n}{du_1^n}\phi_{j,\delta_i}^{1d}(u) = \frac{d^n}{du_1^n}\phi_j^{1d}(u - \delta_i)$ and $\odot$ represent the element-wise multiplication. The second layer response can be developed in a similar way:

$$\begin{aligned}
&\sigma(x * \psi_{j_1,k_1,n_1}) * \psi_{j_2,k_2,n_2}(u) \\
=&\sigma(x * \psi_{j_1,k_1,n_1}) * 2^{n_2 j_2} \cdot D_{\vec{v}_{k_2}}^{(n_2)} \phi_{j_2}(u) \\
=&2^{n_2 j_2} D_{\vec{v}_{k_2}}^{(n_2)} [\sigma(x * \psi_{j_1,k_1,n_1})] * \phi_{j_2}(u) \\
=&2^{n_2 j_2} \cdot \cos^{n_2} \theta_{k_2} \cdot D_{\vec{v}_0}^{(n_2)} [\sigma(x * \psi_{j_1,k_1,n_1})] * \phi_{j_2}(u) \\
=&2^{n_2 j_2} \cdot \cos^{n_2} \theta_{k_2} \cdot \left(\sigma'(x * \psi_{j_1,k_1,n_1}) \odot \frac{d^{n_2}}{du^{n_2}} [x * \psi_{j_1,k_1,n_1}] \right) * \phi_{j_2}(u) \\
=&2^{n_2 j_2} \cdot \cos^{n_2} \theta_{k_2} \cdot \left(\sigma'(2^{n_1 j_1} \cdot \cos^{n_1} \theta_{k_1} \cdot \sum_{\delta_i} \alpha_i \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1,\delta_i}^{1d}) \odot \right. \\
&\left. \frac{d^{n_2}}{du^{n_2}} [2^{n_1 j_1} \cdot \cos^{n_1} \theta_{k_1} \cdot \sum_{\delta_i} \alpha_i \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1,\delta_i}^{1d}] \right) * \phi_{j_2}(u) \\
=&2^{n_1 j_1 + n_2 j_2} \cdot \cos^{n_1} \theta_{k_1} \cdot \cos^{n_2} \theta_{k_2} \cdot \left(\sigma'(\cos^{n_1} \theta_{k_1} \cdot \sum_{\delta_i} \alpha_i \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1,\delta_i}^{1d}) \odot \sum_{\delta_i} \alpha_i \frac{d^{n_1+n_2}}{du_1^{n_1+n_2}} \phi_{j_1,\delta_i}^{1d} \right) * \phi_{j_2}(u)
\end{aligned}$$

When J is small such that $\frac{d^n}{du_1^n}\phi_{j,\delta_i}^{1d}(u)$ do not overlap at different i, the last line of the above equation equals to

$$2^{n_1 j_1 + n_2 j_2} \cdot \cos^{n_1} \theta_{k_1} \cdot \cos^{n_2} \theta_{k_2} \cdot \sum_{\delta_i} \alpha_i \left(\sigma'(\cos^{n_1} \theta_{k_1} \cdot \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1,\delta_i}^{1d}) \odot \frac{d^{n_1+n_2}}{du_1^{n_1+n_2}} \phi_{j_1,\delta_i}^{1d} \right) * \phi_{j_2}(u).$$

The term inside the big parenthesis is supported in $[\delta_i - 2^{j_1}a, \delta_i + 2^{j_1}a]$. With the convolution with ϕ_{j_2}, the support is $[\delta_i - 2^{j_1+j_2}a, \delta_i + 2^{j_1+j_2}a]$. This term is equivariant to translations on δ_i. Therefore when we apply the ReLU and gram matrix at the second layer, we get the

same conclusion as the first layer result. Define

$$G_2x(\lambda,\lambda') = \int \sigma(\epsilon_2 \cdot \sigma(\epsilon_1 \cdot x * \psi_{j_1,k_1,n_1}) * \psi_{j_2,k_2,n_2})(u) \cdot \sigma(\epsilon_2' \cdot \sigma(\epsilon_1' \cdot x * \psi'_{j_1,k_1,n_1}) * \psi'_{j_2,k_2,n_2})(u)du$$

where $\lambda = (j_1, k_1, n_1, \epsilon_1, j_2, k_2, n_2, \epsilon_2)$.

Theorem 15. *Let* $x_1(u) = \sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1} \mathbb{1}_{\delta_{i_1}}(u_1)$ *and* $x_2(u) = \sum_{\delta_{i_2} \in \Delta_2} \beta_{i_2} \mathbb{1}_{\delta_{i_2}}(u_1)$. *Let* d^1_{min} *and* d^2_{min} *denote the smallest pairwise distance between diracs of* x_1 *and* x_2*, respectively. Assume* ϕ *is supported in* $[-a, a]$. *If* $2^{2J}a < d^i_{min}$ *for* $i \in \{1, 2\}$ *and* $\sum_{\delta_{i_1} \in \Delta_1} \alpha^2_{i_1} = \sum_{\delta_{i_2} \in \Delta_2} \beta^2_{i_2}$, *then* $G_2x_1 = G_2x_2$.

More generally, define G_mx to be the m-layer gram matrix. We have

Theorem 16. *Let* $x_1(u) = \sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1} \mathbb{1}_{\delta_{i_1}}(u_1)$ *and* $x_2(u) = \sum_{\delta_{i_2} \in \Delta_2} \beta_{i_2} \mathbb{1}_{\delta_{i_2}}(u_1)$. *Let* d^1_{min} *and* d^2_{min} *denote the smallest pairwise distance between diracs of* x_1 *and* x_2*, respectively. Assume* ϕ *is supported in* $[-a, a]$. *If* $2^{mJ}a < d^i_{min}$ *for* $i \in \{1, 2\}$ *and* $\sum_{\delta_{i_1} \in \Delta_1} \alpha^2_{i_1} = \sum_{\delta_{i_2} \in \Delta_2} \beta^2_{i_2}$, *then* $G_mx_1 = G_mx_2$.

Proof. Let $S_mx[\lambda_1, \ldots, \lambda_m](u) = \sigma(\ldots \sigma(x_1 * \psi_{\lambda_1}) \cdots * \psi_{\lambda_m})(u)$ denote the m-th layer response of x. Let $f_m[\delta_i, \lambda_1, \ldots, \lambda_m](u) = \sigma(\ldots \sigma(\mathbb{1}_{\delta_i} * \psi_{\lambda_1}) \cdots * \psi_{\lambda_m})$ be the multilayer response at the m-th layer of the line function. Note $\mathbb{1}_{\delta_i}(u) = \mathbb{1}_{\delta_i}(u_1)$. Then the m-th layer response of line signal x_1 is:

$$S_mx_1[\lambda_1, \ldots, \lambda_m](u) = \sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1} f_m[\delta_{i_1}, \lambda_1, \ldots, \lambda_m](u)$$

Since ϕ is supported in $[-a, a]^2$, for any $0 \leq j < J$, ϕ_j is supported in $[-2^{J-1}a, 2^{J-1}a]^2$. Therefore $f_m[\delta_{i_1}, \lambda_1, \ldots, \lambda_m]$ is supported in the band: $\{u \in \mathbb{R}^2 : t_1 \in [-2^{m(J-1)}a, 2^{m(J-1)}a\}$. The gram matrix for the line function is :

$$G_m\mathbb{1}_{\delta_{i_1}}[\lambda_1, \ldots, \lambda_m, \lambda'_1, \ldots, \lambda'_m] = \int f_m[\delta_{i_1}, \lambda_1, \ldots, \lambda_m](u) \cdot f_m[\delta_{i_1}, \lambda'_1, \ldots, \lambda'_m](u)du$$

Note $G_m\mathbb{1}_{\delta_{i_1}}[\lambda_1, \ldots, \lambda_m, \lambda'_1, \ldots, \lambda'_m]$ remains a constant when changing δ_i. Since $2^{mJ}a < d^i_{min}$ for $i = 1, 2$, $f_m[\delta_{i_1}, \lambda_1, \ldots, \lambda_m]$ does not overlap with $f_m[\delta_{i'_1}, \lambda_1, \ldots, \lambda_m]$ for any $\delta_{i_1} \neq \delta_{i'_1}$.

With all of these facts,

$$\begin{aligned} G_m x_1[\lambda_1, \ldots, \lambda_m, \lambda'_1, \ldots, \lambda'_m] &= \int \sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1} f_m[\delta_{i_1}, \lambda_1, \ldots, \lambda_m](u) \cdot \\ &\qquad \sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1} f_m[\delta_{i_1}, \lambda'_1, \ldots, \lambda'_m](u) \\ &= \sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1}^2 \int f_m[\delta_{i_1}, \lambda_1, \ldots, \lambda_m](u) \cdot f_m[\delta_{i_1}, \lambda'_1, \ldots, \lambda'_m](u) \\ &= (\sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1}^2) G_m \mathbb{1}_\delta[\lambda_1, \ldots, \lambda_m, \lambda'_1, \ldots, \lambda'_m] \end{aligned}$$

We get similar result for x_2. Therefore, $G_m x_1 = G_m x_2$. □

Theorem 16 shows interesting relationship between model depth m, largest filter scale J and the minimum distance d_{min} between the lines. It also matches the observation of Figure 5.9 (fourth row, sixth row) from Chapter 5. According to Figure 5.9, when $J = 4$ the model is not able to identify the relative positions of the long lines while $J = 5$ and $J = 6$ fix the issue and are able to reproduce the lines. Motivated by Theorem 16, to avoid the inability to distinguish such textures, one can either increase filter size J or increase model depth m to break the condition $2^{mJ} a < d^i_{min}$. Note the VGG model is deep but is still not able to reproduce long lines (Figure 5.11, left, fifth row; right, first row). We infer this is resulted from the pooling function and invite further research.

6.4 Frame-like texture

In this section we analyze another type of textures. Figure 6.3 shows one sample from this class.

Let $x(u_1, u_2) = \sum_{\delta_i \in \Delta_1} \alpha_i \mathbb{1}_{\delta_i}(u_1) + \sum_{\xi_i \in \Delta_2} \beta_i \mathbb{1}_{\xi_i}(u_2)$. The wavelet transform for this signal is:

$$\begin{aligned} x * \psi_{j,k,n}(u) &= 2^{nj} D^{(n)}_{\vec{v}_k}(x * \phi_j(u)) \\ &= 2^{nj} \left[\cos^n \theta_k \sum_{\delta_i \in \Delta_1} \alpha_i \frac{d^n}{du_1^n} \phi_j^{1d}(u_1 - \delta_i) + \sin^n \theta_k \sum_{\xi_i \in \Delta_2} \beta_i \frac{d^n}{du_2^n} \phi_j^{1d}(u_2 - \xi_i) \right] \end{aligned}$$

Insert it into the gram matrix at Equation 6.2 we get:

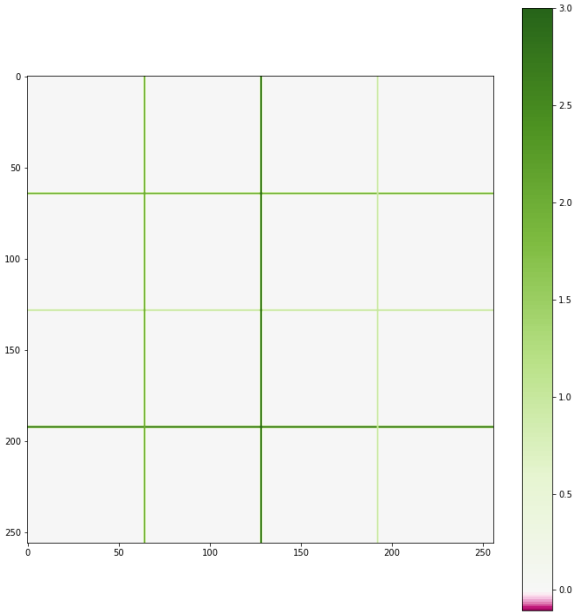


Figure 6.3 Frame texture.

$$
\begin{aligned}
&Gx(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) \\
&\quad = \epsilon_1\epsilon_2 2^{n_1 j_1 + n_2 j_2} \cdot \Bigg[\\
&\quad \cos^{n_1}\theta_{k_1} \cdot \cos^{n_2}\theta_{k_2} \int_{Ex} \sum_{\delta_{i_1}} \alpha_{i_1} \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_{i_1}) \cdot \sum_{\delta_{i_2}} \alpha_{i_2} \frac{d^{n_2}}{du_1^{n_2}} \phi_{j_2}^{1d}(t_1 - \delta_{i_2})\, du_1 du_2 \\
&+ \cos^{n_1}\theta_{k_1} \cdot \sin^{n_2}\theta_{k_2} \int_{Ex} \sum_{\delta_{i_1}} \alpha_{i_1} \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_{i_1}) \cdot \sum_{\xi_{i_2}} \beta_{i_2} \frac{d^{n_2}}{du_2^{n_2}} \phi_{j_2}^{1d}(u_2 - \xi_{i_2})\, du_1 du_2 \\
&+ \sin^{n_1}\theta_{k_1} \cdot \cos^{n_2}\theta_{k_2} \int_{Ex} \sum_{\xi_{i_1}} \beta_{i_1} \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_2 - \xi_{i_1}) \cdot \sum_{\delta_{i_2}} \alpha_{i_2} \frac{d^{n_2}}{du_2^{n_2}} \phi_{j_2}^{1d}(u_1 - \delta_{i_2})\, du_1 du_2 \\
&+ \sin^{n_1}\theta_{k_1} \cdot \sin^{n_2}\theta_{k_2} \int_{Ex} \sum_{\xi_{i_1}} \beta_{i_1} \frac{d^{n_1}}{du_2^{n_1}} \phi_{j_1}^{1d}(u_2 - \xi_{i_1}) \cdot \sum_{\xi_{i_2}} \beta_{i_2} \frac{d^{n_2}}{du_2^{n_2}} \phi_{j_2}^{1d}(u_2 - \xi_{i_2})\, du_1 du_2 \Bigg]
\end{aligned}
\tag{6.13}
$$

where Ex denotes

$$\begin{aligned} Ex(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) = \Big\{ & u \in \mathbb{R}^2 : \\ & \epsilon_1 \Big[\cos^{n_1}\theta_{k_1} \sum_{\delta_i \in \Delta_1} \alpha_i \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_i) + \sin^{n_1}\theta_{k_1} \sum_{\xi_i \in \Delta_2} \beta_i \frac{d^{n_1}}{du_2^{n_1}} \phi_{j_1}^{1d}(u_2 - \xi_i) \Big] > 0, \\ & \epsilon_2 \Big[\cos^{n_2}\theta_{k_2} \sum_{\delta_i \in \Delta_1} \alpha_i \frac{d^{n_2}}{du_1^{n_2}} \phi_{j_2}^{1d}(u_1 - \delta_i) + \sin^{n_2}\theta_{k_2} \sum_{\xi_i \in \Delta_2} \beta_i \frac{d^{n_2}}{du_2^{n_2}} \phi_{j_2}^{1d}(u_2 - \xi_i) \Big] > 0 \Big\} \end{aligned} \tag{6.14}$$

If $\theta_k \in \{0, \frac{\pi}{2}\}$, $\cos\theta_k = 0$ or $\sin\theta_k = 0$. Therefore if $\theta_{k_1}, \theta_{k_2} \in \{0, \frac{\pi}{2}\}$, the four items in Equation 6.13 only have one that is nonzero:

- If $\theta_{k_1} = \theta_{k_2} = 0$, only the first item is nonzero. Ex can be simplified as:

$$\begin{aligned} Ex(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) = \Big\{ u \in \mathbb{R}^2 : & \epsilon_1 \cos^{n_1}\theta_{k_1} \sum_{\delta_i \in \Delta_1} \alpha_i \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_i) > 0, \\ & \epsilon_2 \cos^{n_2}\theta_{k_2} \sum_{\delta_i \in \Delta_1} \alpha_i \frac{d^{n_2}}{du_1^{n_2}} \phi_{j_2}^{1d}(u_1 - \delta_i) > 0 \Big\} \end{aligned}$$

  Then with the same analysis as Theorem 14, the gram matrix can be simplified as:

$$\begin{aligned} Gx(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) = & \epsilon_1\epsilon_2 2^{n_1 j_1 + n_2 j_2} \cdot \cos^{n_1}\theta_{k_1} \cdot \cos^{n_2}\theta_{k_2} \cdot (\sum_{\delta_i} \alpha_i^2) \cdot \\ & \int_{Ex^0_{u_1}} \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_0) \cdot \frac{d^{n_2}}{du_1^{n_2}} \phi_{j_2}^{1d}(u_1 - \delta_0)\, du_1 \end{aligned} \tag{6.15}$$

  where $Ex^0_{u_1} = \{u \in \mathbb{R}^2 : \epsilon_1 \cos^{n_1}\theta_{k_1} \alpha_0 \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_0) > 0,\ \epsilon_2 \cos^{n_2}\theta_{k_2} \alpha_0 \frac{d^{n_2}}{du_1^{n_2}} \phi_{j_2}^{1d}(u_1 - \delta_0) > 0$

- Similarly if $\theta_{k_1} = \theta_{k_2} = \frac{\pi}{2}$, the gram matrix is simplified as:

$$\begin{aligned} Gx(j_1,j_2,k_1,k_2,n_1,n_2,\epsilon_1,\epsilon_2) = & \epsilon_1\epsilon_2 2^{n_1 j_1 + n_2 j_2} \cdot \sin^{n_1}\theta_{k_1} \cdot \sin^{n_2}\theta_{k_2} \cdot (\sum_{\xi_i} \beta_i^2) \cdot \\ & \int_{Ex^0_{u_2}} \frac{d^{n_1}}{du_2^{n_1}} \phi_{j_1}^{1d}(u_2 - \xi_0) \cdot \frac{d^{n_2}}{du_2^{n_2}} \phi_{j_2}^{1d}(u_2 - \xi_0)\, du_2 \end{aligned} \tag{6.16}$$

- If $\theta_{k_1} = 0, \theta_{k_2} = \frac{\pi}{2}$, Ex can be simplified as:

$$Ex(j_1, j_2, k_1, k_2, n_1, n_2, \epsilon_1, \epsilon_2) = \Big\{ u \in \mathbb{R}^2 : \epsilon_1 \cos^{n_1} \theta_{k_1} \sum_{\delta_i \in \Delta_1} \alpha_i \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_i) > 0, \\ \epsilon_2 \sin^{n_2} \theta_{k_2} \sum_{\xi_i \in \Delta_2} \alpha_i \frac{d^{n_2}}{du_2^{n_2}} \phi_{j_2}^{1d}(u_2 - \delta_i) > 0 \Big\}$$

$$:= Ex_{u_1} \otimes Ex_{u_2}$$

The gram matrix is simplified as:

$$\begin{aligned} & Gx(j_1, j_2, k_1, k_2, n_1, n_2, \epsilon_1, \epsilon_2) \\ =& \epsilon_1 \epsilon_2 2^{n_1 j_1 + n_2 j_2} \cdot \cos^{n_1} \theta_{k_1} \cdot \sin^{n_2} \theta_{k_2} \\ & \int_{Ex} \sum_{\delta_{i_1}} \alpha_{i_1} \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_{i_1}) \cdot \sum_{\delta_{i_2}} \beta_{i_2} \frac{d^{n_2}}{du_2^{n_2}} \phi_{j_2}^{1d}(u_2 - \xi_{i_2}) \, du_1 du_2 \\ =& \epsilon_1 \epsilon_2 2^{n_1 j_1 + n_2 j_2} \cdot \cos^{n_1} \theta_{k_1} \cdot \sin^{n_2} \theta_{k_2} \\ & \int_{Ex_{u_1}} \sum_{\delta_{i_1}} \alpha_{i_1} \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_{i_1}) \, du_1 \cdot \int_{Ex_{u_2}} \sum_{\delta_{i_2}} \beta_{i_2} \frac{d^{n_2}}{du_2^{n_2}} \phi_{j_2}^{1d}(u_2 - \xi_{i_2}) \, du_2 \\ =& \epsilon_1 \epsilon_2 2^{n_1 j_1 + n_2 j_2} \cdot \cos^{n_1} \theta_{k_1} \cdot \sin^{n_2} \theta_{k_2} \cdot \\ & (\sum_{\delta_{i_1}} \alpha_{i_1}) \cdot \int_{Ex_{t_1}^0} \frac{d^{n_1}}{du_1^{n_1}} \phi_{j_1}^{1d}(u_1 - \delta_0) \, du_1 \cdot (\sum_{\delta_{i_2}} \beta_{i_2}) \cdot \int_{Ex_{u_2}^0} \frac{d^{n_2}}{du_2^{n_2}} \phi_{j_2}^{1d}(u_2 - \xi_0) \, du_2 \end{aligned} \tag{6.17}$$

- If $\theta_{k_1} = 0, \theta_{k_2} = \frac{\pi}{2}$, the gram matrix is simplified as:

$$\begin{aligned} & Gx(j_1, j_2, k_1, k_2, n_1, n_2, \epsilon_1, \epsilon_2) \\ =& \epsilon_1 \epsilon_2 2^{n_1 j_1 + n_2 j_2} \cdot \sin^{n_1} \theta_{k_1} \cdot \cos^{n_2} \theta_{k_2} \cdot \\ & (\sum_{\xi_{i_1}} \beta_{i_1}) \cdot \int_{Ex_{u_2}^0} \frac{d^{n_1}}{du_2^{n_1}} \phi_{j_1}^{1d}(u_2 - \xi_0) \, du_2 \cdot (\sum_{\delta_{i_2}} \alpha_{i_2}) \cdot \int_{Ex_{u_1}^0} \frac{d^{n_2}}{du_1^{n_2}} \phi_{j_2}^{1d}(u_1 - \xi_0) \, du_1 \end{aligned} \tag{6.18}$$

With the analysis above we have the following theorem:

Theorem 17. *Let* $x(u) = \sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1} \mathbb{1}_{\delta_{i_1}}(u_1) + \sum_{\delta_{i_2} \in \Delta_2} \beta_{i_2} \mathbb{1}_{\delta_{i_2}}(u_2)$ *and* $x'(u) = \sum_{\delta_{i_1} \in \Delta_1'} \alpha'_{i_1} \mathbb{1}_{\delta_{i_1}}(u_1) + \sum_{\delta_{i_2} \in \Delta_2'} \beta'_{i_2} \mathbb{1}_{\delta_{i_2}}(u_2)$. *Let* d_{min} *and* d'_{min} *denote the smallest pairwise distance between parallel lines of* x *and* x'*, respectively, either along* u_1 *or* u_2*. Assume* ϕ *is supported in* $[-a, a]^2$ *and* $\theta_k \in \{0, \frac{\pi}{2}\}$*. If* $2^J a < d_{min}$ *and* $2^J a < d'_{min}$*, then* $Gx = Gx'$ *with the following conditions:*

- $\sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1}^2 = \sum_{\delta_{i_1} \in \Delta_1'} \alpha_{i_1}'^2$
- $\sum_{\xi_{i_2} \in \Delta_2} \beta_{i_2}^2 = \sum_{\xi_{i_2} \in \Delta_2'} \beta_{i_2}'^2$
- $\sum_{\delta_{i_1} \in \Delta_1} \alpha_{i_1} \cdot \sum_{\xi_{i_2} \in \Delta_2} \beta_{i_2} = \sum_{\delta_{i_1} \in \Delta_1'} \alpha_{i_1}' \cdot \sum_{\xi_{i_2} \in \Delta_2'} \beta_{i_2}'$

Theorem 17 explains the observation of Figure 5.9 (seventh row) from Chapter 5. When $J = 4$, the gram representations are not able to reproduce the cross lines in the frame texture while larger J resolves the issue.

6.5 Multilayer model

In this section, we discuss the properties of multilayer ReLU model. In the following context we use $\sigma(f) := \text{ReLU}(f)$.

Let $x : \mathbb{R} \to \mathbb{R}$ and $\phi : \mathbb{R} \to \mathbb{R}$ to be a one dimensional low pass filter with $\phi(u) \geq 0$. Define the wavelets

$$\psi_n(u) = \phi^{(n)}(u),\ u \in \mathbb{R}$$

By rescaling we get a wavelet family $\{\psi_{j,n}\}_{j,n}$:

$$\psi_{j,n}(u) = \phi_j^{(n)}(u)$$

Note $\psi_{j,n}(u) = 2^{nj} \frac{d^n}{du^n} \phi_j(u)$. Then for the one layer wavelet coefficients we have

$$x * \psi_{j,n} = x * 2^{nj} \frac{d^n \phi_j}{du^n} = 2^{nj} \cdot (\frac{d^n}{du^n} x * \phi_j) = 2^{nj} \cdot x^{(n)} * \phi_j$$

If j is small, i.e., ϕ_j is a tiny filter which is usually the case in CNNs, the convolution with ϕ_j can be regarded as a local smoothing operator. The above wavelet coefficients captures the n-th order of derivative of x.

6.5.1 Multilayer ReLU model

Now we consider deeper layers of ReLU responses. The following lemma is an essential property needed about the n-th order derivatives of ReLU function. Note $\frac{d}{du}[\sigma(f(u))] =$

$\sigma'(f(u)) \cdot f'(u)$ and $\sigma'(c \cdot f) = \sigma'(f)$ if $c > 0$. Since $\sigma^{(n)}(f) = 0, \forall n > 1$, we have the following lemma:

Lemma 6. $\frac{d^n}{du^n}\sigma(f(u)) = \sigma'(f(u)) \cdot f^{(n)}(u)$.

Proof. We use induction to prove the statement. First we know $\frac{d}{du}[\sigma(f(u))] = \sigma'(f(u)) \cdot f'(u)$ which aligns with the statement at $n = 1$. Suppose for $n - 1$, we have

$$\frac{d^{n-1}}{du^{n-1}}\sigma(f(u)) = \sigma'(f(u)) \cdot f^{(n-1)}(u)$$

Consider n,

$$\begin{aligned}
\frac{d^n}{du^n}\sigma(f(u)) &= \frac{d}{du}\left(\frac{d^{n-1}}{du^{n-1}}\sigma(f(u))\right) \\
&= \frac{d}{du}\left(\sigma'(f(u)) \cdot f^{(n-1)}(u)\right) \\
&= \sigma''(f(u)) \cdot f^{(n-1)}(u) + \sigma'(f(u)) \cdot f^{(n)}(u) \\
(\textit{since } \sigma''(f(u)) = 0) &= \sigma'(f(u)) \cdot f^{(n)}(u)
\end{aligned}$$

We prove the conclusion. □

Let us first look at the second layer model

$$
\begin{aligned}
\sigma(\epsilon_1 \cdot x * \psi_{j_1,n_1}) * \psi_{j_2,n_2} &= \sigma(\epsilon_1 \cdot x * \psi_{j_1,n_1}) * 2^{n_2 j_2} \cdot \frac{d^{n_2}}{du^{n_2}} \phi_{j_2} \\
&= 2^{n_2 j_2} \cdot \frac{d^{n_2}}{du^{n_2}} \left[\sigma(\epsilon_1 \cdot x * \psi_{j_1,n_1}) * \phi_{j_2} \right] \\
&= 2^{n_2 j_2} \cdot \frac{d^{n_2}}{du^{n_2}} \left[\sigma(\epsilon_1 \cdot x * \psi_{j_1,n_1}) \right] * \phi_{j_2} \\
&= \epsilon_1 2^{n_2 j_2} \cdot \sigma'(\epsilon_1 \cdot x * \psi_{j_1,n_1}) \odot \frac{d^{n_2}}{du^{n_2}} \left[x * \psi_{j_1,n_1} \right] * \phi_{j_2} \\
&= \epsilon_1 2^{n_2 j_2} \cdot \sigma'(\epsilon_1 \cdot x * \psi_{j_1,n_1}) \odot \frac{d^{n_2}}{du^{n_2}} \left[x * 2^{n_1 j_1} \frac{d^{n_1}}{du^{n_1}} \phi_{j_1} \right] * \phi_{j_2} \\
&= \epsilon_1 2^{n_1 j_1 + n_2 j_2} \cdot \sigma'(\epsilon_1 \cdot x * \psi_{j_1,n_1}) \odot \frac{d^{n_1+n_2}}{du^{n_1+n_2}} \left[x * \phi_{j_1} \right] * \phi_{j_2} \\
&= \epsilon_1 2^{n_1 j_1 + n_2 j_2} \cdot \sigma'(\epsilon_1 \cdot x * \psi_{j_1,n_1}) \odot \left[x^{(n_1+n_2)} * \phi_{j_1} \right] * \phi_{j_2} \\
&= \epsilon_1 2^{n_1 j_1 + n_2 j_2} \cdot \sigma'(\epsilon_1 \cdot 2^{n_1 j_1} \cdot x^{(n_1)} * \phi_{j_1}) \odot \left[x^{(n_1+n_2)} * \phi_{j_1} \right] * \phi_{j_2} \\
&= \epsilon_1 2^{n_1 j_1 + n_2 j_2} \cdot \sigma'(\epsilon_1 \cdot x^{(n_1)} * \phi_{j_1}) \odot \left[x^{(n_1+n_2)} * \phi_{j_1} \right] * \phi_{j_2}
\end{aligned}
\tag{6.19}
$$

With the definition of ReLU, we know $\sigma'(\epsilon_1 \cdot x^{(n_1)} * \phi_{j_1}) \neq 0$ only when $\epsilon_1 \cdot x^{(n_1)} * \phi_{j_1} > 0$. If ϕ_{j_1} is tiny enough, the condition is approximately $\epsilon_1 \cdot x^{(n_1)} > 0$. Therefore it can be concluded that $\sigma(\epsilon_1 \cdot x * \psi_{j_1,n_1}) * \psi_{j_2,n_2}$ captures the $(n_1 + n_2)$-th order of derivative of x at where $\epsilon_1 \cdot x^{(n_1)} > 0$. With similar logic, we can prove the following theorem for deeper layers of ReLU models.

Theorem 18. *Let f_k be the k-layer ReLU response, i.e., $f_k = \sigma(\dots \sigma(\epsilon_2 \cdot \sigma(\epsilon_1 x * \psi_{j_1,n_1}) * \psi_{j_2,n_2}) \dots) * \psi_{j_k,n_k}$. Then*

$$
f_k = (\prod_{i=1}^{k-1} \epsilon_i) \cdot 2^{\sum_{i=1}^{k} n_i j_i} \left\{ \sigma'_{k-1} \odot \left[\dots \sigma'_1 \odot (x^{(\sum_{i=1}^{k} n_i)} * \phi_{j_1}) \cdots * \phi_{j_{k-1}} \right] \right\} * \phi_{j_k} \tag{6.20}
$$

where $\sigma'_k = \sigma'(\epsilon_k \cdot f_k)$.

Proof. We use induction to prove the theorem. First for $k = 2$, we prove the conclusion in

(6.19). Suppose the statement holds for $k-1$, let us consider the case for k:

$$\begin{aligned} f_k &= \sigma(\epsilon_{k-1} f_{k-1}) * \psi_{n_k, j_k} \\ &= 2^{n_k j_k} \frac{d^{n_k}}{du^{n_k}} \left[\sigma(\epsilon_{k-1} f_{k-1}) \right] * \phi_{j_k} \\ &= \epsilon_{k-1} \cdot 2^{n_k j_k} \left[\sigma'(\epsilon_{k-1} f_{k-1}) \odot \frac{d^{n_k}}{du^{n_k}} f_{k-1} \right] * \phi_{j_k} \end{aligned} \tag{6.21}$$

Since

$$\begin{aligned} \frac{d^{n_k}}{du^{n_k}} f_{k-1} &= \frac{d^{n_k}}{du^{n_k}} \sigma(\epsilon_{k-2} f_{k-2}) * \psi_{j_{k-1}, n_{k-1}} \\ &= \sigma(\epsilon_{k-2} f_{k-2}) * \frac{d^{n_k}}{du^{n_k}} \psi_{j_{k-1}, n_{k-1}} \\ &= \sigma(\epsilon_{k-2} f_{k-2}) * \frac{d^{n_k}}{du^{n_k}} (2^{n_{k-1} j_{k-1}} \frac{d^{n_{k-1}}}{du^{n_{k-1}}} \phi_{j_{k-1}}) \\ &= \sigma(\epsilon_{k-2} f_{k-2}) * (2^{n_{k-1} j_{k-1}} \frac{d^{n_{k-1}+n_k}}{du^{n_{k-1}+n_k}} \phi_{j_{k-1}}) \\ &= \frac{1}{2^{j_{k-1} n_k}} \sigma(\epsilon_{k-2} f_{k-2}) * \psi_{j_{k-1}, n_{k-1}+n_k} \end{aligned}$$

Let $f^*_{k-1} = \sigma(\epsilon_{k-2} f_{k-2}) * \psi_{j_{k-1}, n_{k-1}+n_k}$, the only difference between f^*_{k-1} and f_{k-1} is that in f^*_{k-1} the last filter substitute n_{k-1} with $n_{k-1} + n_k$. Insert this change into (6.20), we get

$$f^*_{k-1} = (\prod_{i=1}^{k-2} \epsilon_i) \cdot 2^{n_k j_{k-1} + \sum_{i=1}^{k-1} n_i j_i} \left\{ \sigma'_{k-2} \odot \left[\dots \sigma'_1 \odot (x^{(\sum_{i=1}^{k} n_i)} * \phi_{j_1}) \cdots * \phi_{j_{k-2}} \right] \right\} * \phi_{j_{k-1}}$$

Therefore

$$\frac{d^{n_k}}{du^{n_k}} f_{k-1} = (\prod_{i=1}^{k-2} \epsilon_i) \cdot 2^{\sum_{i=1}^{k-1} n_i j_i} \left\{ \sigma'_{k-2} \odot \left[\dots \sigma'_1 \odot (x^{(\sum_{i=1}^{k} n_i)} * \phi_{j_1}) \cdots * \phi_{j_{k-2}} \right] \right\} * \phi_{j_{k-1}}$$

Insert this into Equation 6.21 we prove the theorem. □

The above theorem concludes interesting results about very deep neural networks. In most CNNs, the convolution filters are of small size, which aligns with the case when j_k are all small. In that case, the convolutions with ϕ_j can be regarded as tiny local smoothing operators that we can ignore for now. The theorem concludes that under this condition, the k-layer ReLU model captures $(\sum_{i=1}^{k} n_i)$-th order derivatives of x at where $(\prod_{i=1}^{l} \epsilon_i) \cdot x^{(\sum_{i=1}^{l} n_i)} > 0, \forall 1 \leq l < k$. High order derivatives are preserved based on fine partitions of low order derivatives. Specifically when the filters are tiny, n usually equals to 0 or 1. If n is exactly 1,

the theorem proves that the n-th layer deep model preserves the n-th order derivatives of x based on partitions of i-th order derivatives for $0 < i < n$. If n is either 0 or 1, the theorem implies similar conclusions. If n is exactly 0, i.e., all the filters are small low pass filters, the deep model is less related to derivatives of x. In general, deep models smooth out either the signal or the high order derivatives of the signal. Also deep layers increase the receptive fields, thus incorporating information from more neighbor pixels and preserving long range dependence.

CHAPTER 7

RANDOM FIELDS

A random field refers to a random function over an arbitrary domain, e.g., $\mathbb{R}^d$. The function value at each point in the domain can be think of as a random variable. And the function values at different points usually have some correlation. On the other hand, texture images from the same class usually have similar repeated patterns but generally different, which can be reflected from the correlation and randomness of a random field. Therefore random fields can be used to model texture images. In this chapter, we show more of our work on scattering transform on random fields. We extend the work in [43] to d dimensional space. We also discuss the relationship between scattering moments and power spectrum of a stochastic process.

7.1 Scattering moments of self-similar processes

Before we introduce the scattering moments of random fields, we first show and prove the following properties of random fields that are stationary or has stationary increments. Let $\psi(u) : u \in \mathbb{R}^d$ be a wavelet, i.e., $\int_{\mathbb{R}^d} \psi(u)du = 0$.

Lemma 7. *If $\{X(u)\}_{u\in\mathbb{R}^d}$ has stationary increments, $\{X * \psi(u)\}_{u\in\mathbb{R}^d}$ is stationary.*

Proof. for all u and v:

$$\begin{aligned}
X * \psi(u) &= \int X(u-u')\psi(u')du' \\
\text{(since } \psi \text{ has zero average)} \quad &= \int X(u-u')\psi(u')du' - X(u)\int \psi(u')du' \\
&= \int (X(u-u') - X(u))\psi(u')du' \\
&\overset{d}{=} \int (X(v-u') - X(v))\psi(u')du' \\
&= \int X(v-u')\psi(u')du' \\
&= X * \psi(v)
\end{aligned}$$

Therefore, $X * \psi(u) \stackrel{d}{=} X * \psi(v)\, \forall u, v$, and thus $\{X * \psi(u)\}_{u \in \mathbb{R}^d}$ is stationary. □

Remark 6. *Similarly if $\{X(u)\}_{u \in \mathbb{R}^d}$ is stationary, $\{X * \psi(u)\}_{u \in \mathbb{R}^d}$ is stationary.*

Let σ be a modulus operator or a ReLU operator, i.e., $\sigma(x) = |x|$ or $\sigma(x) = \text{ReLU}(x)$. Let $\{\psi_{j,\theta}\}_{j,\theta}$ be a family of wavelets where

$$\psi_{j,\theta}(u) = 2^{-dj}\psi(2^{-j}R_{-\theta}u),\ \forall u \in \mathbb{R}^d.$$

The following theorem explores the first order scattering moments of self similar processes.

Theorem 19. *Suppose $\{X(u)\}_{u \in \mathbb{R}^d}$ is a self similar process of order H and has stationary increments. The first order m-th scattering moments of $\{X(u)\}_{u \in \mathbb{R}^d}$ satisfy:*

$$\frac{\mathbb{E}[\sigma^m(X * \psi_{j_1,\theta_1})]}{\mathbb{E}[\sigma^m(X * \psi_{\theta_1})]} = 2^{mHj_1} \tag{7.1}$$

Moreover, if $\{X(u)\}_{u \in \mathbb{R}^d}$ is isotropic:

$$\frac{\mathbb{E}[\sigma^m(X * \psi_{j_1,\theta_1})]}{\mathbb{E}[\sigma^m(X * \psi)]} = 2^{mHj_1} \tag{7.2}$$

Proof.

$$\begin{aligned}
X * \psi_{j_1,\theta_1}(u) &= \int_{\mathbb{R}^d} X(u') \cdot \psi_{j_1,\theta_1}(u - u')du' \\
&= \int_{\mathbb{R}^d} X(u') \cdot 2^{-nj_1}\psi_{\theta_1}(2^{-j_1}(u - u'))du' \\
(\text{let } v' = 2^{-j_1}u') \ &= \int_{\mathbb{R}^d} X(2^{j_1}v') \cdot 2^{-nj_1}\psi_{\theta_1}(2^{-j_1}u - v'))2^{nj_1}dv' \\
(\text{since } X \text{ is self similar}) \ &\stackrel{d}{=} \int_{\mathbb{R}^d} 2^{Hj_1}X(v') \cdot \psi_{\theta_1}(2^{-j_1}u - v'))dv' \\
&= 2^{Hj_1}X * \psi_{\theta_1}(2^{-j_1}u)
\end{aligned} \tag{7.3}$$

Since X is stationary, we have $X * \psi_{\theta_1}(u)$ is stationary. Therefore $\mathbb{E}[\sigma^m(X * \psi_{\theta_1}(2^{-j_1}u))] = \mathbb{E}[\sigma^m(X * \psi_{\theta_1}(u))]$ and the first result develops naturally.

Moreover, if $\{X(u)\}_{u\in\mathbb{R}^d}$ is isotropic:

$$\begin{aligned} X * \psi_{\theta_1}(u) &= \int_{\mathbb{R}^d} X(u') \cdot \psi_{\theta_1}(u-u')du' \\ &= \int_{\mathbb{R}^d} X(u') \cdot \psi(R_{\theta_1}^{-1}(u-u'))du' \\ (\text{let } v' = R_{\theta_1}^{-1}u') \ &= \int_{\mathbb{R}^d} X(R_{\theta_1}v') \cdot \psi(R_{\theta_1}^{-1}u - v')dv' \\ (\text{since } X \text{ is isotropic}) \ &\stackrel{d}{=} \int_{\mathbb{R}^d} X(v') \cdot \psi(R_{\theta_1}^{-1}u - v')dv' \\ &= X * \psi(R_{\theta_1}^{-1}u) \end{aligned}$$

Since $X * \psi$ is stationary, $\mathbb{E}[\sigma^m(X * \psi(R_{\theta_1}^{-1}u))] = \mathbb{E}[\sigma^m(X * \psi(u))]$, we verified the second result. □

Theorem 19 implies that for a self similar process of order H with stationary increments, if we think of $\mathbb{E}[\sigma^m(X * \psi_{\theta_1})]$ as a normalization term, the first order scattering moments is proportional to a value determined by the scale j of the wavelet, the moments m and H. Moreover if the process is isotropic, the normalization term is substitute by $\mathbb{E}[\sigma^m(X*\psi)]$ and the scattering moments are invariant to the rotation parameter θ. It provides a necessary condition for identifying a self similar process and also captures the order parameter H of the process through the first order scattering moments. In general, this theorem indicates the pattern of scattering moments of a self similar process. The following theorem explores more about the second order scattering moments of such processes. Since it involves rotation difference, we focus on $\mathbb{R}^2$ for simplification in the following theorem.

Theorem 20. *Suppose $(X(u))_{u\in\mathbb{R}^2}$ is a self similar process of order H and has stationary increments. The second order scattering moments satisfy:*

$$\frac{\mathbb{E}[\sigma(\sigma(X * \psi_{j_1,\theta_1}) * \psi_{j_2,\theta_2})]}{\mathbb{E}[\sigma(X * \psi_{j_1,\theta_1})]} = \frac{\mathbb{E}[\sigma(\sigma(X * \psi_{\theta_1}) * \psi_{j_2-j_1,\theta_2})]}{\mathbb{E}[\sigma(X * \psi_{\theta_1})]} \tag{7.4}$$

Moreover, if $(X(u))_{u\in\mathbb{R}^2}$ is isotropic:

$$\frac{\mathbb{E}[\sigma(\sigma(X * \psi_{j_1,\theta_1}) * \psi_{j_2,\theta_2})]}{\mathbb{E}[(X * \psi_{j_1,\theta_1})]} = \frac{\mathbb{E}[\sigma(\sigma(X * \psi) * \psi_{j_2-j_1,\theta_2-\theta_1})]}{\mathbb{E}[\sigma(X * \psi)]} \tag{7.5}$$

Proof. From Theorem 19 we have $X * \psi_{j_1,\theta_1}(u) \overset{d}{=} 2^{Hj_1} X * \psi_{\theta_1}(2^{-j_1}u)$. Therefore:

$$\begin{aligned}
\sigma(X * \psi_{j_1,\theta_1}) * \psi_{j_2,\theta_2}(u) &= \int_{\mathbb{R}^2} \sigma(X * \psi_{j_1,\theta_1}(u')) \cdot \psi_{j_2,\theta_2}(u-u')du' \\
&\overset{d}{=} \int_{\mathbb{R}^2} \sigma(2^{Hj_1} X * \psi_{\theta_1}(2^{-j_1}u')) \cdot \psi_{j_2,\theta_2}(u-u')du' \\
(\text{let } v' = 2^{-j_1}u') \ &= \int_{\mathbb{R}^2} \sigma(2^{Hj_1} X * \psi_{\theta_1}(v')) \cdot \psi_{j_2,\theta_2}(u-2^{j_1}v')2^{nj_1}dv' \\
&= \int_{\mathbb{R}^2} \sigma(2^{Hj_1} X * \psi_{\theta_1}(v')) \cdot \psi_{j_2-j_1,\theta_2}(2^{-j_1}u-v')dv' \\
&= 2^{Hj_1}\sigma(X * \psi_{\theta_1}) * \psi_{j_2-j_1,\theta_2}(2^{-j_1}u)
\end{aligned}$$

Since $\sigma(X*\psi_1)$ is stationary, from Remark 6 we have $\sigma(X*\psi_1)*\psi_2$ is stationary. Therefore:

$$\begin{aligned}
\frac{\mathbb{E}[\sigma(\sigma(X * \psi_{j_1,\theta_1}) * \psi_{j_2,\theta_2})]}{\mathbb{E}[\sigma(X * \psi_{j_1,\theta_1})]} &= \frac{2^{Hj_1}\mathbb{E}[\sigma(\sigma(X * \psi_{\theta_1}) * \psi_{j_2-j_1,\theta_2})]}{2^{Hj_1}\mathbb{E}[\sigma(X * \psi_{\theta_1})]} \\
&= \frac{\mathbb{E}[\sigma(\sigma(X * \psi_{\theta_1}) * \psi_{j_2-j_1,\theta_2})]}{\mathbb{E}[\sigma(X * \psi_{\theta_1})]}
\end{aligned}$$

The first result if verified. If $(X(u))_{u\in\mathbb{R}^2}$ is isotropic:

$$\begin{aligned}
\sigma(X * \psi_{\theta_1}) * \psi_{j_2-j_1,\theta_2}(u) &= \int_{\mathbb{R}^2} \sigma(X * \psi_{\theta_1}(u')) \cdot \psi_{j_2-j_1,\theta_2}(u-u')du' \\
&\overset{d}{=} \int_{\mathbb{R}^2} \sigma(X * \psi(R_{\theta_1}^{-1}u')) \cdot \psi_{j_2-j_1,\theta_2}(u-u')du' \\
(\text{let } v' = R_{\theta_1}^{-1}u') \ &= \int_{\mathbb{R}^2} \sigma(X * \psi(v')) \cdot \psi_{j_2-j_1,\theta_2}(u-R_{\theta_1}^{-1}v')dv' \\
&= \int_{\mathbb{R}^2} \sigma(X * \psi(v')) \cdot \psi_{j_2-j_1,\theta_2-\theta_1}(R_{\theta_1}u-v')dv' \\
&= \sigma(X * \psi) * \psi_{j_2-j_1,\theta_2-\theta_1}(R_{\theta_1}u)
\end{aligned}$$

Therefore:

$$\frac{\mathbb{E}[\sigma(\sigma(X * \psi_{\theta_1}) * \psi_{j_2-j_1,\theta_2})]}{\mathbb{E}[\sigma(X * \psi_{\theta_1})]} = \frac{\mathbb{E}[\sigma(\sigma(X * \psi) * \psi_{j_2-j_1,\theta_2-\theta_1})]}{\mathbb{E}[\sigma(X * \psi)]}$$

We verified the second result. □

Theorem 20 implies that, for a self similar process with order H, the second order scattering moments $\mathbb{E}[\sigma(\sigma(X * \psi_{j_1,\theta_1}) * \psi_{j_2,\theta_2})]$ normalized by the first scattering moments $\mathbb{E}[\sigma(X * \psi_{j_1,\theta_1})]$, is determined by the scale difference $j_2 - j_1$ where j_1, j_2 are the scales of the two wavelets at the two layers. Moreover, if the process is isotropic, the second order

scattering moments is also determined by the rotation difference $\theta_2 - \theta_1$, where θ_1, θ_2 are the rotations of the two wavelets at the two layers. This theorem provides another necessary condition for identifying a self similar process. It describes the pattern of the second order moments and the parameter H is also captured in such statistics.

7.2 Power spectrum

Definition 1. *Suppose $\{X(u)\}_{u\in\mathbb{R}^d}$ is a stationary process with zero mean. The covariance of X is defined as $R_X(\tau) = \mathbb{E}X(0)\overline{X(\tau)}$. Then the power spectrum of $\{X(u)\}_{u\in\mathbb{R}^d}$ is defined as:*

$$\widehat{R_X}(\omega) = \mathcal{F}(R_X)(\omega) = \int_{\mathbb{R}^d} R_X(\tau)e^{-i\tau\omega}d\tau$$

Lemma 8. *Suppose $\{X(u)\}_{u\in\mathbb{R}^d}$ is a stationary process and ψ is a wavelet. Let $Y(u) = X * \psi(u)$. It can be proved that Y is also stationary from Remark 1. Then*

$$\widehat{R_Y}(\omega) = \widehat{R_X}(\omega)|\widehat{\psi}(-\omega)|^2 \tag{7.6}$$

Proof.

$$
\begin{aligned}
\widehat{R_Y}(\omega) &= \int R_Y(\tau) e^{-i\tau\omega} d\tau \\
&= \int \mathbb{E}[Y(0)\overline{Y(\tau)}] e^{-i\tau\omega} d\tau \\
&= \int \mathbb{E}[X * \psi(0)\overline{X * \psi(\tau)}] e^{-i\tau\omega} d\tau \\
&= \int \mathbb{E}[\int X(u)\cdot\psi(-u)du \cdot \int \overline{X(v)\cdot\psi(\tau - v)}dv] e^{-i\tau\omega} d\tau \\
\text{(by Fubini's Thm.)} \quad &= \mathbb{E}\int\int\int X(u)\cdot\psi(-u)\cdot\overline{X(v)}\cdot\overline{\psi(\tau-v)}\cdot e^{-i\tau\omega} du dv d\tau \\
&= \mathbb{E}\int\int X(u)\cdot\psi(-u)\cdot\overline{X(v)}\cdot[\int \overline{\psi(\tau-v)}\cdot e^{-i\tau\omega} d\tau] dv du \\
\text{(let } \tau' = \tau - v, d\tau' = d\tau) \quad &= \mathbb{E}\int\int X(u)\cdot\psi(-u)\cdot\overline{X(v)}\cdot[\int \overline{\psi(\tau')}\cdot e^{-i(\tau'+v)\omega} d\tau'] dv du \\
&= \mathbb{E}\int\int X(u)\cdot\psi(-u)\cdot\overline{X(v)}\cdot\overline{\hat{\psi}(-\omega)}\cdot e^{-iv\omega} dv du \\
\text{(let } v = u + v', dv = dv') \quad &= \overline{\hat{\psi}(-\omega)}\mathbb{E}\int\int X(u)\cdot\psi(-u)\cdot\overline{X(u+v')}\cdot e^{-i(u+v')\omega} dv' du \\
\text{(by Fubini's Thm.)} \quad &= \overline{\hat{\psi}(-\omega)}\int\int \psi(-u)\mathbb{E}[X(u)\cdot\overline{X(u+v')}]\cdot e^{-i(u+v')\omega} dv' du \\
&= \overline{\hat{\psi}(-\omega)}\int\int \psi(-u)\cdot R_X(v')\cdot e^{-i(u+v')\omega} dv' du \\
&= \overline{\hat{\psi}(-\omega)}\hat{\psi}(-\omega)\widehat{R_X}(\omega) \\
&= |\hat{\psi}(-\omega)|^2\widehat{R_X}(\omega)
\end{aligned}
$$

□

Lemma 9. *Let $\{X(u)\}_{u\in\mathbb{R}^d}$ be a stationary process and let $\{\psi_j\}_j$ be a group of wavelets that are complex analytic. Suppose $\sum_j |\widehat{\psi_j}|^2(\omega) + \sum_j |\widehat{\psi_j}|^2(-\omega) = 2, \forall\omega$. Then*

$$\sum_j \mathbb{E}|X * \psi_j|^2 = VarX$$

Proof. Let $Y_j = X * \psi_j$. From Lemma 8,

$$\widehat{R_{Y_j}}(\omega) = \widehat{R_X}(\omega)|\widehat{\psi_j}(-\omega)|^2$$

Therefore,

$$\begin{aligned}\sum_j \widehat{R_{Y_j}}(\omega) &= \sum_j \widehat{R_X}(\omega)|\widehat{\psi_j}(-\omega)|^2 \\ &= \widehat{R_X}(\omega)\sum_j |\widehat{\psi_j}(-\omega)|^2 \\ &= \begin{cases} 2\widehat{R_X}(\omega), \omega < 0 \\ 0, \omega > 0 \end{cases}\end{aligned}$$

Taking the inverse Fourier transform at $\tau = 0$ of both sides, we get

$$\sum_j R_{Y_j}(0) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \widehat{R_Y}(\omega)d\omega = \frac{2}{2\pi}\int_{-\infty}^{0} \widehat{R_X}(\omega)d\omega = R_X(0)$$

Since $R_X(0) = \mathrm{Var}X$

$$\sum_j \mathrm{Var}(Y_j) = \mathrm{Var}X$$

Since ψ_j is a wavelet, $\mathbb{E}Y_j = \mathbb{E}X * \psi_j = 0$. We have the result:

$$\sum_j \mathbb{E}|X * \psi_j|^2 = \sum_j \mathbb{E}|Y_j|^2 = \sum_j \mathrm{Var}(Y_j) = \mathrm{Var}X$$

□

Remark 7. *Note if $\sigma(x) = ReLU(x)$, then $|X * \psi_j|^2 = \sigma^2(X * \psi_j) + \sigma^2(-X * \psi_j)$ when X and ψ_j are both real valued. Then Lemma 9 can be extended to the ReLU operator:*

$$\sum_j \mathbb{E}[\sigma^2(X * \psi_j)] + \sum_j \mathbb{E}[\sigma^2(-X * \psi_j)] = VarX$$

7.3 Power spectrum and scattering equivalence

For now let us work on $\mathbb{R}$. Define $\psi_\lambda(t) = \sqrt{\lambda}\psi(\lambda t)$ for any $\lambda \in (0, \infty)$, and assume ψ is admissible, meaning that

$$C_\psi := \int_0^\infty \frac{|\widehat{\psi}(\omega)|^2}{\omega}\, d\omega < \infty\,.$$

Let $(X(t))_{t\in\mathbb{R}}$ be a stationary process. Define the first order quadratic scattering moments of X as:

$$SX(\lambda) := \mathbb{E}|X * \psi_\lambda|^2\,, \quad \forall\, \lambda \in (0, \infty)\,.$$

Theorem 21. *Let X, Y be two real-valued stationary processes.*

$$\widehat{R}_X(\omega) = \widehat{R}_Y(\omega) \quad a.e.\ \omega \in \mathbb{R} \quad \Longleftrightarrow \quad SX(\lambda) = SY(\lambda) \quad \forall\, \lambda \in (0, \infty)\,.$$

To prove the theorem, we need Lemma 2.1 from [71], which we restate in the following.

Lemma 10. *Let $f \in \mathbf{L}^2(\mathbb{R})$ be continuous and assume $f(\omega) = f(-\omega)$. $\widehat{\psi}$ has compact support and satisfy Condition 1. Then*

$$\int f(\omega)|\widehat{\psi}_\lambda(\omega)|^2 d\omega = 0\, \forall \lambda > 0 \Longrightarrow f = 0\ a.e.$$

Condition 1. *Define*

$$|\widehat{\psi}^+_\lambda(\omega)|^2 = (|\widehat{\psi}_\lambda(\omega)|^2 + |\widehat{\psi}_\lambda(-\omega)|^2) \cdot \mathbb{1}(\omega \geq 0).$$

If for any finite sequence $\{\omega_i\}_{i=1}^n$ of distinct positive frequencies, the collection $\{|\widehat{\psi}^+_\lambda(\omega_i)|^2\}_{i=1}^n$ are linearly independent functions of λ, we say the wavelet ψ satisfies the linear independence condition.

Proof of Theorem 21. $\Longrightarrow$:

If ψ is a wavelet, then from Lemma 8, we have:

$$\widehat{R}_X(\omega) = \widehat{R}_Y(\omega) \quad \text{a.e. } \omega \in \mathbb{R} \Longrightarrow \widehat{R}_{X*\psi_\lambda}(\omega) = \widehat{R}_{Y*\psi_\lambda}(\omega) \quad \text{a.e. } \omega \in \mathbb{R}, \forall \lambda$$

Then by taking inverse Fourier tansform of the power spectrum we have:

$$R_{X*\psi_\lambda}(\tau) = R_{Y*\psi_\lambda}(\tau) \quad \forall \tau \in \mathbb{R}, \lambda$$

Let $\tau = 0$, we have:

$$\mathbb{E}|X * \psi_\lambda|^2 = \mathbb{E}|Y * \psi_\lambda|^2$$

which proves this direction.

$\Longleftarrow$:

$$\begin{aligned}
\mathbb{E}|X * \psi_\lambda(t)|^2 &= \mathbb{E}\int X(u)\cdot\psi_\lambda(t-u)du\overline{\int X(v)\cdot\psi_\lambda(t-v)dv} \\
&= \mathbb{E}\int\int X(u)\cdot\overline{X}(v)\cdot\psi_\lambda(t-u)\cdot\overline{\psi_\lambda}(t-v)dudv \\
(\text{Let } v=u+\tau) &= \mathbb{E}\int\int X(u)\cdot\overline{X}(u+\tau)\cdot\psi_\lambda(t-u)\cdot\overline{\psi_\lambda}(t-u-\tau)dud\tau \\
&= \int\int \mathbb{E}X(u)\overline{X}(u+\tau)\cdot\psi_\lambda(t-u)\cdot\overline{\psi_\lambda}(t-u-\tau)dud\tau \\
&= \int\int R_X(\tau)\cdot\psi_\lambda(t-u)\cdot\overline{\psi_\lambda}(t-u-\tau)dud\tau \\
&= \int\psi_\lambda(t-u)\cdot\int R_X(\tau)\cdot\overline{\psi_\lambda}(t-u-\tau)d\tau du \\
&= \int\psi_\lambda(t-u)\cdot R_X * \overline{\psi_\lambda}(t-u)du \\
(\text{Let } t=0 \text{ and } v=-u) &= \int\psi_\lambda(v)\cdot R_X * \overline{\psi_\lambda}(v)dv \\
(\text{By Plancherel Thm.}) &= \int\widehat{R_X * \overline{\psi_\lambda}}(\omega)\cdot\widehat{\overline{\psi_\lambda}}(\omega)d\omega \\
&= \int\widehat{R_X}(\omega)\cdot\widehat{\overline{\psi_\lambda}}(\omega)\cdot\widehat{\psi_\lambda}(-\omega)d\omega \\
&= \int\widehat{R_X}(\omega)\cdot\overline{\widehat{\psi_\lambda}}(-\omega)\cdot\widehat{\psi_\lambda}(-\omega)d\omega \\
&= \int\widehat{R_X}(\omega)\cdot|\widehat{\psi_\lambda}(-\omega)|^2d\omega \\
&= \int\widehat{R_X}(-\omega)\cdot|\widehat{\psi_\lambda}(\omega)|^2d\omega \\
(\widehat{R_X} \text{ is even}) &= \int\widehat{R_X}(\omega)\cdot|\widehat{\psi_\lambda}(\omega)|^2d\omega
\end{aligned}$$

If $SX(\lambda) = SY(\lambda)$, we have:

$$\int(\widehat{R_X} - \widehat{R_Y})(\omega)\cdot|\widehat{\psi_\lambda}(\omega)|^2d\omega = 0$$

According to Lemma 11 stated in the following, we have $\widehat{R_X}(\omega) = \widehat{R_X}(-\omega)$. For stationary, real-valued X, Lemma 8 implies $\mathbb{E}|X * \psi_\lambda(t)|^2 = \int\widehat{R_X}(\omega)\cdot|\widehat{\psi_\lambda}(\omega)|^2d\omega$ for all ψ. Suppose $\widehat{\psi_\lambda}$ is an indicator function supported in interval $[a,b]$, we know the integration of $\widehat{R_X}$ in interval $[a,b]$ is positive. By taking arbitrary a and b, we can conclude that $\widehat{R_X}$ is positive almost everywhere. With Lemma 10, we prove the result. $\square$

Lemma 11. *If X is stationary and real-valued, we have R_X is real-valued and $R_X(\tau) = R_X(-\tau)$. Then we also have $\widehat{R_X}$ is real-valued and $\widehat{R_X}(\omega) = \widehat{R_X}(-\omega)$*

Proof. Since

$$R_X(\tau) = \mathbb{E}X(0)X(\tau) = \mathbb{E}X(-\tau)X(0) = R_X(-\tau)$$

$R_X(\tau)$ is real-valued and even.

For $\widehat{R_X}$,

$$\widehat{R_X}(\omega) = \int R_X(\tau)e^{-i\tau\omega}d\tau = \int R_X(\tau)\cos(\tau\omega) - i \cdot R_X(\tau)\sin(\tau\omega)d\tau$$

Since R_X is even and sin() is odd, we know the imagery part is 0. Therefore, R_X is real-valued.

$$\widehat{R_X}(-\omega) = \int R_X(\tau)e^{i\tau\omega}d\tau = \int R_X(-\tau)e^{i\tau\omega}d\tau = \int R_X(v)e^{-iv\omega}dv = \widehat{R_X}(\omega)$$

we proved the result. □

Theorem 21 demonstrates the power spectral is equivalent to the scattering moments of a real-valued stationary process. In particular, two processes with the same power spectrum is equivalent to, the two processes has the same first order second scattering moments for any scattering parameter λ. Since a stationary process is determined by its power spectrum, this theorem provides a necessary and sufficient condition for identifying such processes.

APPENDIX

PROOF FOR CHAPTER 3

The Proof of the Lemmas

The Proof of Lemma 1. By linearity, it suffices to show that p is not the zero function unless $\alpha_1 = \ldots, \alpha_N = 0$. We will consider the case where $|\gamma_N| > |\gamma_k|$ for all $k = 1, 2, \ldots, N-1$. The other cases, where $|\gamma_1| > |\gamma_k|$ for all $2 \leq k \leq N$ or where $|\gamma_1| = |\gamma_N| > |\gamma_k|$ for all $2 \leq k \leq N-1$ are similar. The n-th derivative of p is given by

$$p^{(n)}(x) = \sum_{k=1}^{N} \alpha_k \gamma_k^n \mathrm{e}^{\mathrm{i}\gamma_k x}.$$

Therefore,

$$\lim_{n\to\infty} \frac{p^{(n)}(0)}{\gamma_N^n} = \alpha_N,$$

so in particular, there exists n such that $p^{(n)}(0) \neq 0$, and therefore p is not uniformly zero. □

The proof of Lemma 2. In the case where $p = 2m$ is even, then by (3.13) and (3.14), $|p_i|^{2m} \in \mathcal{E}(md_i)$ for each i. Therefore, since $d_1 > d_2, d_3, d_4$, it follows from (3.15) that

$$|p_1|^{2m} + |p_2|^{2m} - |p_3|^{2m} - |p_4|^{2m}$$

is an element of $\mathcal{E}(md_1)$ and therefore vanishes on a set of measure zero. Now consider the case where $p = 2m+1$ is odd. Squaring both sides of (3.16) implies

$$p_5 = 2(|p_1 p_2|^{2m+1} - |p_3 p_4|^{2m+1}),$$

where $p_5 \coloneqq |p_1|^{4m+2} + |p_2|^{4m+2} - |p_3|^{4m+2} - |p_4|^{4m+2}$. Thus, squaring both sides again gives

$$p_6 \coloneqq 8|p_1 p_2 p_3 p_4|^{2m+1}$$

where $p_6 = p_5^2 - 4|p_1 p_2|^{4m+2} - 4|p_3 p_4|^{4m+2}$. Therefore, squaring both sides one final time implies that

$$p_6^2 - 64|p_1 p_2 p_3 p_4|^{4m+2} = 0.$$

However, since $d_1 > d_2, d_3, d_4$, repeatedly applying (3.13), (3.14), and (3.15), we see that $(p_6^2 - 64|p_1p_2p_3p_4|^2) \in \mathcal{E}(4md_1)$ and therefore vanishes on a set of measure zero. $\square$

In addition to the lemmas from Chapter 3, we also need the following lemmas to prove Theorem 5 and 4.

Lemma 12. *Let $m \geq 1$ be an odd integer, and let $a, b, c, d, C \in \mathbb{R}$, $a, b, c, d \neq 0$. Let $p(\theta) = a + be^{i\theta}$, and $q(\theta) = c + de^{i\theta}$. If there are more than $4m$ distinct $\theta \in [0, 2\pi]$ such that*

$$|p(\theta)|^m - |q(\theta)|^m = C,$$

then $ab = cd$ and $a^2 + b^2 = c^2 + d^2$.

Lemma 13. *Let $m \geq 1$ be an integer (not necessarily odd) and let $a, b, c, d, C \in \mathbb{R}, \gamma > 0, \kappa \neq 0, 1$. Then the set of θ such that*

$$\left|a + be^{i\theta} + ce^{i(\gamma+1)\theta}\right|^m - \left|\kappa a + \frac{1}{\kappa}be^{i\theta} + \kappa ce^{i(\gamma+1)\theta}\right|^m = C \tag{7}$$

has measure zero.

The Proof of Lemma 12. If θ is a solution to

$$|p(\theta)|^p - |q(\theta)|^p = C,$$

then

$$|p(\theta)|^{2p} - |q(\theta)|^{2p} - C^2 = 2|q(\theta)|^p C.$$

Therefore, $f(\theta) = 0$, where $f : \mathbb{R} \to \mathbb{R}$ is the function defined by

$$f(\theta) \coloneqq \left(|p(\theta)|^{2p} - |q(\theta)|^{2p} - C^2\right)^2 - 4|q(\theta)|^{2p}C^2.$$

Since

$$|p(\theta)|^2 = a^2 + b^2 + 2ab\cos(\theta) \quad \text{and} \quad |q(\theta)|^2 = c^2 + d^2 + cd\cos(\theta), \tag{8}$$

$f(\theta)$ is a trigonometric polynomial of degree at most $2p$ which by assumption has more than $4p$ zeros in $[0, 2\pi]$. This implies that $f(\theta)$ is uniformly zero. By (8)

$$f(\theta) = \left((a^2 + b^2 + 2ab\cos(\theta))^p - (c^2 + d^2 + 2cd\cos(\theta))^p - C^2\right)^2 - 4C^2(c^2 + d^2 + 2cd\cos(\theta))^p$$

and so setting the $\cos^{2p}(\theta)$ coefficient equal to zero implies

$$0 = (2^p a^p b^p - 2^p c^p d^p)^2$$

which implies $ab = cd$ since p is odd. If $p \geq 3$, then $2(p-1) > p$. Therefore, using the binomial formula and setting the $\cos^{2(p-1)}(\theta)$ coefficient of $f(\theta)$ equal to zero implies

$$\left(\binom{p}{p-1}(a^2 + b^2)(2ab)^{p-1} - \binom{p}{p-1}(c^2 + d^2)(2cd)^{p-1}\right)^2 = 0,$$

but since $ab = cd$ this implies that $a^2 + b^2 = c^2 + d^2$. On the other hand, if $p = 1$, using the fact that $ab = cd$ we see that

$$f(\theta) = \left(a^2 + b^2 - (c^2 + d^2) - C\right)^2 = 4C^2(c^2 + d^2 + 2cd\cos(\theta)$$

Therefore, $f(\theta)$ can only be uniformly equal to zero if $C = 0$ and $a^2 + b^2 = c^2 + d^2$. □

The Proof of Lemma 13. Let

$$p(\theta) = a + b\mathrm{e}^{\mathrm{i}\theta} + c\mathrm{e}^{\mathrm{i}(\gamma+1)\theta} \quad \text{and} \quad q(\theta) = \kappa a + \frac{1}{\kappa}b\mathrm{e}^{\mathrm{i}\theta} + \kappa c\mathrm{e}^{\mathrm{i}(\gamma+1)\theta}.$$

Then by (7) we see

$$|p(\theta)|^p = |q(\theta)|^p + C,$$

Squaring both sides yields,

$$|p(\theta)|^{2p} - |q(\theta)|^{2p} - C^2 = 2|q(\theta)|^p C,$$

and therefore if θ satisfies (7) it is a solution to $f(\theta) = 0$, where

$$f(\theta) \coloneqq \left(|p(\theta)|^{2p} - |q(\theta)|^{2p} - C^2\right)^2 - 4|q(\theta)|^{2p}C^2 = 0.$$

$f(\theta)$ is an element of the class $\mathcal{E}$ of generalized exponential polynomials introduced earlier. Therefore, it will follow that f vanishes on a set of measure zero as soon as we show that f is not uniformly zero. We will verify that that the lead cofficient of f is nonzero unless $c = \pm 1$. Using the trigonometric identities $\sin^2(x) + \cos^2(x) = 1$ and $\cos(x - y) = \cos(x)\cos(y) + \sin(x)\sin(y)$ we see that

$$|p(\theta)|^2 = a^2 + b^2 + c^2 + 2ab\cos(\theta) + 2bc\cos(\gamma\theta) + 2ac\cos((\gamma+1)\theta)$$

and likewise

$$|q(\theta)|^2 = \kappa^2 a^2 + \frac{1}{\kappa^2}b^2 + \kappa^2 c^2 + 2ab\cos(\theta) + 2bc\cos(\gamma\theta) + 2\kappa^2 ac\cos((\gamma+1)\theta).$$

Therefore, the lead coefficient of $f(\theta)$ vanishes if and only if $\kappa^2 = 1$. □

The Proof of Theorem 4

Proof. Choose $\xi_1, \xi_2, \ldots, \xi_L$ i.i.d. from any probability distribution which is absolutely continuous with respect to the Lebesgue measure. Since x is collision free, with probability one, each of the $\xi_\ell \Delta_{i,i+1}(x)$ are distinct modulo 2π, i.e.

$$\xi_\ell \Delta_{i,i+1}(x) \not\equiv \xi_{\ell'} \Delta_{i',i'+1}(x) \mod 2\pi \tag{9}$$

for all $1 \le i, i' \le k-1$ and $1 \le \ell, \ell' \le L$, except when $(i, \ell) = (i', \ell')$.

Assume these $\xi_\ell \Delta_{i,i+1}$ are distinct, and let $y(t)$ be any signal defined as in 3.17 such that $\mathcal{D}(y) = \mathcal{D}(x) =: \mathcal{D}$, and $\partial_s^2 f_{x,\xi_\ell}(d) = \partial_s^2 f_{y,\xi_\ell}(d)$ for all $d \in \mathcal{D}$ and for all $1 \le \ell \le L-1$, and $\sum_{i=1}^k |b_i|^p = \sum_{i=1}^k |a_i|^p$. Note that $y(t)$ depends on $\xi_1, \ldots, \xi_{L-1}$, but is independent of ξ_L. By CITE TURNPIKE and the assumption that $x(t)$ and $y(t)$ are collision free, the fact that $\mathcal{D}(x) = \mathcal{D}(y)$ implies that the support sets of x and y are equivalent up to translation and reflection, so we may assume without loss of generality that $\Delta_{i,j}(x) = \Delta_{i,j}(y) =: \Delta_{i,j}$ for all $1 \le i \le j \le k$. We will show that $\vec{b}$ must be given by

$$b_i = \begin{cases} \frac{1}{c}a_i & \text{if } i \text{ is odd} \\ ca_i & \text{if } i \text{ is even} \end{cases},$$

where $c = \pm 1$ or

$$|c|^p = \frac{\sum_{i=1}^{\lfloor \frac{k+1}{2} \rfloor} |a_{2i-1}|^p}{\sum_{i=1}^{\lfloor \frac{k}{2} \rfloor} |a_{2i}|^p}. \tag{10}$$

Next, we will show that, with probability one, if c satisfies 10, but $c \neq \pm 1$, that $\partial_s^2 f_{x,\xi_L}(\Delta_{1,3}) \neq \partial_s^2 f_{y,\xi_L}(\Delta_{1,3})$. Since $y(t)$ $\big($ and therefore $\vec{b}\big)$ was chosen to depend on $\xi_1, \ldots, \xi_{L-1}$, but not ξ_L, these two facts together will imply that, with probability one, if $y(t)$ is any signal such that $\mathcal{D}(y) = \mathcal{D}(x) =: \mathcal{D}$, and $\partial_s^2 f_{x,\xi_\ell}(d) = \partial_s^2 f_{y,xi_\ell}(d)$ for all $d \in \mathcal{D}$ and all $1 \leq \ell \leq L$, then $\vec{b} = \pm \vec{a}$ and therefore y is equivalent to $\pm x$ up to reflection and translation.

(3.6) implies that for all $1 \leq \ell \leq L-1$ and all $1 \leq i \leq k-1$ we have

$$\begin{aligned} |a_i + a_{i+1} e^{i\xi_\ell \Delta_{i,i+1}}|^p - |a_{i+1}|^p - |a_i|^p &= \partial_s^2 f_{x,\xi_\ell}(\Delta_{i,i+1}) \\ &= \partial_s^2 f_{y,\xi_\ell}(\Delta_{i,i+1}) \\ &= |b_i + b_{i+1} e^{i\xi_\ell \Delta_{i,i+1}}|^p - |b_{i+1}|^p - |b_i|^p. \end{aligned}$$

Therefore, for all $1 \leq i \leq k-1$, $\xi_1 \Delta_{i,i+1}, \ldots, \xi_{L-1} \Delta_{i,i+1}$ are all $L-1$ solutions, which are distinct modulo 2π, to

$$|a_i + a_{i+1} e^{i\theta}|^p - |b_i + b_{i+1}^{i\theta}|^p = |b_i|^p + |b_{i+1}|^p - |a_i|^p - |a_{i+1}|^p.$$

Since $L - 1 \geq 4p$, Lemma 12 implies that

$$a_i a_{i+1} = b_i b_{i+1} \tag{11}$$

and

$$a_i^2 + a_{i+1}^2 = b_i^2 + b_{i+1}^2 \tag{12}$$

for all $1 \leq i \leq k-1$. It follows from (11) that

$$b_i = \begin{cases} \frac{1}{c} a_i & \text{if } i \text{ is odd} \\ c a_i & \text{if } i \text{ is even,} \end{cases} \tag{13}$$

where $c \coloneqq \frac{a_1}{b_1}$. Combining (13) with the assumption that

$$\sum_{i=1}^{k} |a_i|^p = \sum_{i=1}^{k} |b_i|^p$$

implies that either $c = \pm 1$ or that c satisfies (10). Now, we will show that, with probability one, if c satisfies (10), but $c \neq \pm 1$, then $\partial_s^2 f_{\xi_L}[x](\Delta_{1,3}) \neq \partial_s^2 f_{\xi_L}[y](\Delta_{1,3})$.

By 3.7, if $\partial_s^2 f_{\xi_L}[x](\Delta_{1,3}) = \partial_s^2 f_{\xi_L}[y](\Delta_{1,3})$, then

$$\begin{aligned}&|a_1 + a_2 e^{i\xi_L \Delta_{1,2}} + a_3 e^{i\xi_L \Delta_{1,3}}|^p + |a_2|^p - |a_2 e^{i\xi_L \Delta_{1,2}} + a_3 e^{i\xi_L \Delta_{1,3}}|^p - |a_1 + a_2 e^{i\xi_L \Delta_{1,2}}|^p \\ =&|b_1 + b_2 e^{i\xi_L \Delta_{1,2}} + b_3 e^{i\xi_L \Delta_{1,3}}|^p + |b_2|^p - |b_2 e^{i\xi_L \Delta_{1,2}} + b_3 e^{i\xi_L \Delta_{1,3}}|^p - |b_1 + b_2 e^{i\xi_L \Delta_{1,2}}|^p. \quad (14)\end{aligned}$$

But combining (11) and (12) implies that for all i either $(a_i, a_{i+1}) = \pm(b_i, b_{i+1})$ or $(a_i, a_{i+1}) = \pm(b_{i+1}, b_i)$. In either case, we have that

$$|a_1 + a_2 e^{i\xi_L \Delta_{1,2}}| = |b_1 + b_2 e^{i\xi_L \Delta_{1,2}}| \quad \text{and} \quad |a_2 e^{i\xi_L \Delta_{1,2}} + a_3 e^{i\xi_L \Delta_{1,3}}| = |b_2 e^{i\xi_L \Delta_{1,2}} + b_3 e^{i\xi_L \Delta_{1,3}}|.$$

Combining this with (14) gives

$$|a_1 + a_2 e^{i\xi_L \Delta_{1,2}} + a_3 e^{i\xi_L \Delta_{1,3}}|^p + |a_2|^p = |b_1 + b_2 e^{i\xi_L \Delta_{1,2}} + b_3 e^{i\xi_L \Delta_{1,3}}|^p + |b_2|^p. \quad (15)$$

However, by Lemma 13 the set of $\xi_L \in \mathbb{R}$ such that (15) holds has measure zero, unless $c = \pm 1$. Since we assumed that the distribution of ξ_L was absolutely continuous with respect to the Lebesgue measure, this completes the proof. □

The Proof of Theorem 5. The proof is quite similar to the proof of Theorem 4. Choose $\xi_1, \ldots, \xi_L$ i.i.d from any probability distribution which is absolutely continuous with respect to the Lebesgue measure, and again note that with probability one each of the $\xi_\ell \Delta_{i,i+1}$ are distinct modulo 2π. Let $y(t)$ be any signal defined as in (3.18) such that $\mathcal{D}(y) = \mathcal{D}(x) =: \mathcal{D}$, and $\partial_s^2 f_{\xi_\ell}[x](d) = \partial_s^2 f_{\xi_\ell}[y](d)$ for all $d \in \mathcal{D}$ and for all $1 \leq \ell \leq L-1$, and $\sum_{i=1}^k |b_i|^p = \sum_{i=1}^k |a_i|^p$. As before, we may assume that $\Delta_{i,j}(x) = \Delta_{i,j}(y) =: \Delta_{i,j}$ for all $1 \leq i \leq j \leq k$.

Since $\partial_s^2 f_{\xi_\ell}[x](\Delta_{i,i+1}) = \partial_s^2 f_{\xi_\ell}[x](\Delta_{i,i+1})$ if follows from (3.6) that for all $1 \leq \ell \leq L$, $1 \leq i \leq k-1$,

$$|a_i + a_{i+1} e^{i\xi_\ell \Delta_{i,i+1}}|^{2m} - |a_i|^{2m} - |a_{i+1}|^{2m} = |b_i + b_{i+1} e^{i\xi_\ell \Delta_{i,i+1}}|^{2m} - |b_i|^{2m} - |b_{i+1}|^{2m}.$$

Therefore, for all $1 \leq i \leq k-1$, $\xi_1 \Delta_{i,i+1}, \ldots, \xi_{L-1} \Delta_{i,i+1}$ are $L-1$ solutions to

$$h(\theta) \coloneqq |a_i + a_{i+1} e^{i\theta}|^{2m} - |b_i + b_{i+1} e^{i\theta}|^{2m} + |b_i|^{2m} + |b_{i+1}|^{2m} - |a_i|^{2m} - |a_{i+1}|^{2m} = 0$$

which are distint modulo 2π. Using the facts that

$$|a_i + a_{i+1}e^{i\theta}|^2 = a_i^2 + a_{i+1}^2 + 2a_i a_{i+1}\cos(\theta) \quad \text{and} \quad |b_i + b_{i+1}e^{i\theta}|^2 = b_i^2 + b_{i+1}^2 + 2b_i b_{i+1}\cos(\theta)$$

we see that

$$h(\theta) = (a_i^2 + a_{i+1}^2 + 2a_i a_{i+1}\cos(\theta))^m - (b_i^2 + b_{i+1}^2 + 2b_i b_{i+1}\cos(\theta))^m + b_i^{2m} + b_{i+1}^{2m} - a_i^{2m} - a_{i+1}^{2m}$$

is a trigonometic polynomial of degree at most m with at least $L-1$ distinct roots. Since $L - 1 \geq 2m + 1 > 2m$, this implies that h must be uniformly zero. In particular, setting the lead coefficient equal to zero implies

$$(a_i a_{i+1})^m = (b_i b_{i+1})^m$$

for all $1 \leq i \leq k-1$. Using the binomial theorem and setting the $\cos^{m-1}(\theta)$ coefficient equal to zero gives

$$(a_i^2 + a_{i+1}^2)^{m-1} a_i a_{i+1} = (b_i^2 + b_{i+1}^2)^{m-1} b_i b_{i+1}.$$

Combining the last two equations gives

$$a_i^2 + a_{i+1}^2 = b_i^2 + b_{i+1}^2 \quad \text{and} \quad a_i a_{i+1} = b_i b_{i+1}.$$

As in the proof of Theorem 4, these two facts imply that

$$b_i = \begin{cases} \frac{1}{c} a_i & \text{if } i \text{ is odd} \\ c a_i & \text{if } i \text{ is even} \end{cases}$$

for $c = \frac{a_1}{b_1}$, and the assumption that

$$\sum_{i=1}^{k} |a_i|^p = \sum_{i=1}^{k} |b_i|^p$$

implies that either $c = \pm 1$ or

$$|c|^p = \pm \frac{\sum_{i=1}^{\lfloor \frac{k+1}{2} \rfloor} |a_{2i-1}|^p}{\sum_{i=1}^{\lfloor \frac{k}{2} \rfloor} |a_{2i}|^p}. \tag{16}$$

As in the proof of Theorem 4 it follows from Lemma 13 if c satisfies (16), but $c \neq \pm 1$, then $\partial_s^2 f_{\xi_L}[x](\Delta_{1,3}) \neq \partial_s^2 f_{\xi_L}[y](\Delta_{1,3})$ with probability one. Therefore, the proof is complete. $\square$

Proof of Theorem 6

To prove Theorem 6 we will need the following lemma.

Lemma 14. *Let Z be a Poisson random variable with parameter λ. Then for all $\alpha \in \mathbb{R}$, $m \in \mathbb{N}$,*

$$\mathbb{E}\left[Z^\alpha \mathbb{1}_{\{Z>m\}}\right] = \sum_{k=m+1}^{\infty} e^{-\lambda}\frac{\lambda^k}{k!}k^\alpha \leq C_{m,\alpha}\lambda^{m+1}\,, \quad \forall\, 0<\lambda<1\,.$$

Proof. For $0<\lambda<1$ and $k\in\mathbb{N}$, $e^{-\lambda}\lambda^k \leq 1$. Therefore,

$$\begin{aligned}
\mathbb{E}\left[Z^\alpha \mathbb{1}_{\{Z>m\}}\right] &= \sum_{k=m+1}^{\infty} e^{-\lambda}\frac{\lambda^k}{k!}k^\alpha \\
&= \lambda^{m+1}\sum_{k=0}^{\infty} e^{-\lambda}\frac{\lambda^k}{(k+m+1)!}(k+m+1)^\alpha \\
&\leq \lambda^{m+1}\sum_{k=0}^{\infty}\frac{(k+m+1)^\alpha}{(k+m+1)!} \\
&= C_{\alpha,m}\lambda^{m+1}\,.
\end{aligned}$$

□

Proof. **[Theorem 6]** Recalling the definitions of $Y(dt)$ and $S[\gamma,p]Y(t)$, and setting $N_s(t) = N\left([t-s,t]^d\right)$, we see

$$\begin{aligned}
S[\gamma,p]Y(t) &= \mathbb{E}\left[\left|\int_{[s-t,t]^d} w\left(\frac{t-u}{s}\right)e^{i\xi\cdot(t-u)}\,Y(du)\right|^p\right] \\
&= \mathbb{E}\left[\left|\sum_{j=1}^{N_s(t)} A_j w\left(\frac{t-t_j}{s}\right)e^{i\xi\cdot(t-t_j)}\right|^p\right]\,,
\end{aligned}$$

where $t_1, t_2, \ldots t_{N_s(t)}$ are the points $N(t)$ in $[t-s,t]^d$. Conditioned on the event that $N_s(t)=k$, the locations of the k points on $[t-s,t]^d$ are distributed as i.i.d. random variables $Z_1,\ldots,Z_k$ taking values in $[t-s,t]^d$ with density

$$p_Z(z) = \frac{\lambda(z)}{\Lambda_s(t)}\,, \quad z\in[t-s,t]^d\,.$$

Therefore, the random variables

$$V_i := \frac{t - Z_i}{s}$$

take values in the unit cube $Q_1 = [0,1]^d$ and have density

$$p_V(v) = \frac{s^d}{\Lambda_s(t)} \lambda(t - vs)\,, \quad v \in Q_1\,.$$

Note that in the special case that N is homogeneous, i.e. $\lambda(t) \equiv \lambda_0$ is constant, the V_i are uniform random variables on Q_1.

Our proof will be based off of conditioning on $N_s(t)$. For $N_s(t) = k \geq 1$,

$$\begin{aligned}\mathbb{E}\left[\left|\sum_{j=1}^{N_s(t)} A_j w\left(\frac{t-t_j}{s}\right) e^{i\xi\cdot(t-t_j)}\right|^p : N_s(t) = k\right] &= \mathbb{E}\left[\left|\sum_{j=1}^{k} A_j w(V_j) e^{is\xi\cdot V_j}\right|^p\right] \\ &\leq \frac{\|\lambda\|_\infty}{\lambda_{\min}} k^p \mathbb{E}[|A_1|^p] \|w\|_p^p\,, \qquad (17)\end{aligned}$$

where (17) follows from (i) the independence of the random variables A_j and V_j; (ii) the fact that for any sequence of i.i.d. random variables $Z_1, Z_2, \ldots$,

$$\mathbb{E}\left[\left|\sum_{n=1}^{k} Z_n\right|^p\right] \leq k^{p-1}\mathbb{E}\left[\sum_{n=1}^{k} |Z_n|^p\right] = k^p \mathbb{E}[|Z_1|^p]\,;$$

and (iii) the fact that

$$\mathbb{E}[|w(V_i)|^p] = \int_{Q_1} |w(v)|^p p_V(v)\, dv \leq \frac{\|\lambda\|_\infty}{\lambda_{\min}} \|w\|_p^p\,.$$

Therefore, since $\mathbb{P}[N_s(t) = k] = e^{-\Lambda_s(t)} \cdot {(\Lambda_s(t))^k}/{k!}$,

$$\begin{aligned}&\mathbb{E}\left[\left|\sum_{j=1}^{N_s(t)} A_j w\left(\frac{t-t_j}{s}\right) e^{i\xi\cdot(t-t_j)}\right|^p\right] = \\ &= \sum_{k=0}^{\infty} e^{-\Lambda_s(t)} \frac{(\Lambda_s(t))^k}{k!} \mathbb{E}\left[\left|\sum_{j=1}^{N_s(t)} A_j w\left(\frac{t-t_j}{s}\right) e^{i\xi\cdot(t-t_j)}\right|^p : N_s(t) = k\right] \\ &= \sum_{k=1}^{\infty} e^{-\Lambda_s(t)} \frac{(\Lambda_s(t))^k}{k!} \mathbb{E}\left[\left|\sum_{j=1}^{k} A_j w(V_j) e^{is\xi\cdot V_j}\right|^p\right] \\ &= \sum_{k=1}^{m} e^{-\Lambda_s(t)} \frac{(\Lambda_s(t))^k}{k!} \mathbb{E}\left[\left|\sum_{j=1}^{k} A_j w(V_j) e^{is\xi\cdot V_j}\right|^p\right] + \epsilon(m, s, \xi, t)\,,\end{aligned}$$

where

$$\epsilon(m,s,t,\xi) := \sum_{k=m+1}^{\infty} e^{-\Lambda_s(t)} \frac{(\Lambda_s(t))^k}{k!} \mathbb{E}\left[\left|\sum_{j=1}^{k} A_j w(V_j) e^{is\xi\cdot V_j}\right|^p\right].$$

By (17) and Lemma 14, if s is small enough so that $\Lambda_s(t) \le s^d \|\lambda\|_\infty < 1$, then:

$$\begin{aligned}
\epsilon(m,s,\xi,t) &= \sum_{k=m+1}^{\infty} e^{-\Lambda_s(t)} \frac{(\Lambda_s(t))^k}{k!} \mathbb{E}\left[\left|\sum_{j=1}^{k} A_j w(V_j) e^{is\xi\cdot V_j}\right|^p\right] \\
&\le \frac{\|\lambda\|_\infty}{\lambda_{\min}} \mathbb{E}[|A_1|^p] \|w\|_p^p \sum_{k=m+1}^{\infty} e^{-\Lambda_s(t)} \frac{(\Lambda_s(t))^k}{k!} k^p \\
&\le C_{m,p} \frac{\|\lambda\|_\infty}{\lambda_{\min}} \mathbb{E}[|A_1|^p] \|w\|_p^p (\Lambda_s(t))^{m+1} \\
&\le C_{m,p} \frac{\|\lambda\|_\infty}{\lambda_{\min}} \mathbb{E}[|A_1|^p] \|w\|_p^p \|\lambda\|_\infty^{m+1} s^{d(m+1)}.
\end{aligned}$$

□

Proof of Theorem 7

Proof. **[Theorem 7]** Let (s_k, ξ_k) be a sequence of scale and frequency pairs such that $\lim_{k\to\infty} s_k = 0$. Applying Theorem 6 with $m = 1$, we obtain:

$$\begin{aligned}
\frac{S_{\gamma_k,p} Y(t)}{s_k^d} &= e^{-\Lambda_{s_k}(t)} \frac{\Lambda_{s_k}(t)}{s_k^d} \mathbb{E}\left[\left|A_1 w(V_{1,k}) e^{is\xi\cdot V_{1,k}}\right|^p\right] + \frac{\epsilon(1,s_k,\xi_k,t)}{s_k^d} \\
&= e^{-\Lambda_{s_k}(t)} \frac{\Lambda_{s_k}(t)}{s_k^d} \mathbb{E}[|A_1|^p] \mathbb{E}[|w(V_{1,k})|^p] + \frac{\epsilon(1,s_k,\xi_k,t)}{s_k^d},
\end{aligned}$$

where we write $V_{1,k} = V_1$ to emphasize the fact that the density of $V_{1,k}$ is:

$$p_{V_k}(v) = \frac{s_k^d}{\Lambda_{s_k}(t)} \lambda(t - v s_k).$$

Using the error bound (4.9), we see that:

$$\lim_{k\to\infty} \frac{\epsilon(1,s_k,\xi_k,t)}{s_k^d} = 0.$$

Furthermore, since $0 \le \Lambda_{s_k}(t) \le s_k^d \|\lambda\|_\infty$, we observe that:

$$\lim_{k\to\infty} e^{-\Lambda_{s_k}(t)} = 1,$$

and by the continuity of $\lambda(t)$,

$$\lim_{k\to\infty} \frac{\Lambda_{s_k}(t)}{s_k^d} = \lim_{k\to\infty} \frac{1}{s_k^d} \int_{[s_k-t,t]^d} \lambda(u)\, du = \lambda(t)\,. \tag{18}$$

Finally, by the continuity of $\lambda(t)$, we see that

$$p_{V_k}(v) \le \frac{\|\lambda\|_\infty}{\lambda_{\min}} \quad \text{and} \quad \lim_{k\to\infty} p_{V_k}(v) = 1\,, \quad \forall\, v \in Q_1\,. \tag{19}$$

Therefore, by the bounded convergence theorem,

$$\lim_{k\to\infty} \mathbb{E}[|w(V_1)|^p] = \lim_{k\to\infty} \int_{Q_1} |w(v)|^p p_{V_k}(v)\, dv = \int_{Q_1} |w(v)|^p \lim_{k\to\infty} p_{V_k}(v)\, dv = \|w\|_p^p\,.$$

That completes the proof of (4.10).

To prove (4.11), we assume that $\lambda(t)$ is periodic with period T along each coordinate and again use Theorem 6 with $m = 1$ to observe,

$$\frac{SY(s_k,\xi_k,p)}{s_k^d} = \mathbb{E}[|A_1|^p] \frac{1}{T^d} \int_{Q_T} e^{-\Lambda_{s_k}(t)} \frac{\Lambda_{s_k}(t)}{s_k^d} \int_{Q_1} |w(v)|^p p_{V_k}(v)\, dv\, dt + \frac{1}{T^d} \int_{Q_1} \frac{\epsilon(1,s_k,\xi_k,t)}{s_k^d}\, dt\,.$$

By (4.9), the second integral converges to zero as $k \to \infty$. Therefore,

$$\lim_{k\to\infty} \frac{SY(s_k,\xi_k,p)}{s_k^d} = \mathbb{E}[|A_1|^p] \|w\|_p^p \frac{1}{T^d} \int_{Q_T} \lambda(t)\, dt\,,$$

by the continuity of $\lambda(t)$ and the bounded convergence theorem. □

Proof of Theorem 8

Proof. **[Theorem 8]** We apply Theorem 6 with $m = 2$ and obtain:

$$\begin{aligned} S_{\gamma_k,p} Y(t) &= e^{-\Lambda_{s_k}(t)} \Lambda_{s_k}(t) \mathbb{E}[|A_1|^p] \mathbb{E}[|w(V_{1,k})|^p] \\ &\quad + e^{-\Lambda_{s_k}(t)} \frac{(\Lambda_{s_k}(t))^2}{2} \mathbb{E}\left[\left|A_1 w(V_{1,k}) e^{i s_k \xi_k \cdot V_{1,k}} + A_2 w(V_{2,k}) e^{i s_k \xi_k \cdot V_{2,k}}\right|^p\right] + \epsilon(2,s_k,\xi_k,t)\,, \end{aligned} \tag{20}$$

where $V_{i,k}$, $i = 1, 2$, are random variables taking values on the unit cube $Q_1 = [0,1]^d$ with densities,

$$p_{V_k}(v) = \frac{s_k^d}{\Lambda_{s_k}(t)} \lambda(t - v s_k)\,.$$

Dividing both sides in (20) by $s_k^{2d}\|w\|_p^p\mathbb{E}[|A_1|^p]$ and subtracting $\frac{\Lambda_{s_k}(t)}{s_k^{2d}}\frac{\mathbb{E}[|w(V_{1,k})|^p]}{\|w\|_p^p}$ yields:

$$\begin{aligned}\frac{S_{\gamma_k,p}Y(t)}{s_k^{2d}\|w\|_p^p\mathbb{E}[|A_1|^p]} - \frac{\Lambda_{s_k}(t)}{s_k^{2d}}\frac{\mathbb{E}[|w(V_{1,k})|^p]}{\|w\|_p^p} &= \frac{e^{-\Lambda_{s_k}(t)}\Lambda_{s_k}(t)-\Lambda_{s_k}(t)}{s_k^{2d}}\frac{\mathbb{E}[|w(V_{1,k})|^p]}{\|w\|_p^p} \\ &+ e^{-\Lambda_{s_k}(t)}\frac{(\Lambda_{s_k}(t))^2}{s_k^{2d}}\frac{\mathbb{E}\left[\left|A_1w(V_{1,k})e^{is_k\xi_k\cdot V_{1,k}}+A_2w(V_{2,k})e^{is_k\xi_k\cdot V_{2,k}}\right|^p\right]}{2\|w\|_p^p\mathbb{E}[|A_1|^p]} \\ &+ \frac{\epsilon(2,s_k,\xi_k,t)}{s_k^{2d}\|w\|_p^p\mathbb{E}[|A_1|^p]}\,.\end{aligned} \tag{21}$$

Using the error bound (4.9),

$$\lim_{k\to\infty}\frac{\epsilon(2,s_k,\xi_k,t)}{s_k^{2d}\|w\|_p^p\mathbb{E}[|A_1|^p]} = 0\,, \tag{22}$$

at a rate independent of t. Recalling (19) from the proof of Theorem 7, we use the fact that $\lim_{k\to\infty} p_{V_k} \equiv 1$ and the bounded convergence theorem to conclude,

$$\lim_{k\to\infty}\mathbb{E}\left[\left|A_1w(V_{1,k})e^{is_k\xi_k\cdot V_{1,k}}+A_2w(V_{2,k})e^{is_k\xi_k\cdot V_{2,k}}\right|^p\right] = \mathbb{E}\left[\left|A_1w(U_1)e^{iL\cdot U_1}+A_2w(U_2)e^{iL\cdot U_2}\right|^p\right]\,, \tag{23}$$

where U_i, $i=1,2$, are uniform random variables on the unit cube and $L = \lim_{k\to\infty} s_k\xi_k$. Similarly,

$$\lim_{k\to\infty}\frac{\mathbb{E}[|w(V_{1,k})|^p]}{\|w\|_p^p} = 1\,. \tag{24}$$

Lastly, recalling that $s_k \to 0$ as $k\to\infty$ and using (18) from the proof of Theorem 7, we see

$$\begin{aligned}\lim_{k\to\infty}\frac{e^{-\Lambda_{s_k}(t)}\Lambda_{s_k}(t)-\Lambda_{s_k}(t)}{s_k^{2d}} &= \lim_{k\to\infty}\left(\frac{\Lambda_{s_k}(t)}{s_k^d}\right)\lim_{k\to\infty}\left(\frac{e^{-\Lambda_{s_k}(t)}-1}{s_k^d}\right) \\ &= \lambda(t)\lim_{k\to\infty}\left(\frac{e^{-\Lambda_{s_k}(t)}-1}{s_k^d}\right) \\ &= -\lambda(t)^2\,.\end{aligned} \tag{25}$$

Now we integrate both sides of (21) over Q_T and divide by T^d. Taking the limit as $k\to\infty$, on the left hand side we get:

$$\begin{aligned}&\lim_{k\to\infty}\frac{1}{T^d}\int_{Q_T}\left(\frac{S_{\gamma_k,p}Y(t)}{s_k^{2d}\|w\|_p^p\mathbb{E}[|A_1|^p]} - \frac{\Lambda_{s_k}(t)}{s_k^{2d}}\frac{\mathbb{E}[|w(V_{1,k})|^p]}{\|w\|_p^p}\right)dt \\ &\quad= \lim_{k\to\infty}\left(\frac{SY(s_k,\xi_k,p)}{s_k^{2d}\|w\|_p^p\mathbb{E}[|A_1|^p]} - \frac{\mathbb{E}[|w(V_{1,k})|^p]}{\|w\|_p^p}\frac{1}{T^d}\int_{Q_T}\frac{\Lambda_{s_k}(t)}{s_k^{2d}}\,dt\right) \\ &\quad= \lim_{k\to\infty}\left(\frac{SY(s_k,\xi_k,p)}{s_k^{2d}\mathbb{E}[|w(V_{1,k})|^p]\mathbb{E}[|A_1|^p]} - \frac{1}{T^d}\int_{Q_T}\frac{\Lambda_{s_k}(t)}{s_k^{2d}}\,dt\right)\,,\end{aligned}$$

where we used the definition of the invariant scattering moments and (24). On the right hand side of (21), we use (24), (25) and the dominated convergence theorem to see that the first term is:

$$\lim_{k\to\infty}\frac{1}{T^d}\int_{Q_T}\frac{e^{-\Lambda_{s_k}(t)}\Lambda_{s_k}(t)-\Lambda_{s_k}(t)}{s_k^{2d}}\frac{\mathbb{E}[|w(V_{1,k})|^p]}{\|w\|_p^p}\,dt=\lim_{k\to\infty}\frac{1}{T^d}\int_{Q_T}\frac{e^{-\Lambda_{s_k}(t)}\Lambda_{s_k}(t)-\Lambda_{s_k}(t)}{s_k^{2d}}\,dt$$
$$=-\frac{1}{T^d}\int_{Q_T}\lambda(t)^2\,dt\,.$$

Using (18), (23), and the bounded convergence theorem, the second term of (21) is:

$$\lim_{k\to\infty}\frac{1}{T^d}\int_{Q_T}e^{-\Lambda_{s_k}(t)}\frac{(\Lambda_{s_k}(t))^2}{s_k^{2d}}\frac{\mathbb{E}\left[\left|A_1w(V_{1,k})e^{is_k\xi_k\cdot V_{1,k}}+A_2w(V_{2,k})e^{is_k\xi_k\cdot V_{2,k}}\right|^p\right]}{2\|w\|_p^p\mathbb{E}[|A_1|^p]}\,dt$$
$$=\frac{\mathbb{E}[|A_1w(U_1)e^{iL\cdot U_1}+A_2w(U_2)e^{iL\cdot U_2}|^p]}{2\|w\|_p^p\mathbb{E}[|A_1|^p]}\left(\frac{1}{T^d}\int_{Q_T}\lambda(t)^2\,dt\right).$$

Finally, the third term of (21) goes to zero using the bounded convergence theorem and (22). Putting together the left and right hand sides of (21) with these calculations finishes the proof. □

Proof of Theorem 9

Proof. **[Theorem 9]** As in the proof of Theorem 6, let $N_s(t)=N\left([t-s,t]^d\right)$ denote the number of points in the cube $[t-s,t]^d$. Then since the support of w is contained in $[0,1]^d$,

$$\left(g_{\gamma_k}*Y\right)(t)=\int_{[t-s_k,t]^d}w\left(\frac{t-u}{s_k}\right)e^{i\xi_k\cdot(t-u)}\,Y(du)=\sum_{j=1}^{N_{s_k}(t)}A_jw\left(\frac{t-t_j}{s_k}\right)e^{i\xi_k\cdot(t-t_j)}\,,$$

where $t_1,t_2,\ldots,t_{N_{s_k}(t)}$ are the points of N in $[t-s_k,t]^d$. Therefore, in the event that $N_{s_k}(t)=1$,

$$\left|\left(g_{\gamma_k}*Y\right)(t)\right|^p=\left(|g_{\gamma_k}|^p*|Y|^p\right)(t)\,,$$

and so, partitioning the space of possible outcomes based on $N_{s_k}(t)$, we obtain:

$$\begin{aligned}
|\left(g_{\gamma_k} * Y\right)(t)|^p &= |\left(g_{\gamma_k} * Y\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)=1\}} + \left(g_{\gamma_k} * Y\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)>1\}}|^p \\
&= |\left(g_{\gamma_k} * Y\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)=1\}}|^p + |\left(g_{\gamma_k} * Y\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)>1\}}|^p \\
&= \left(|g_{\gamma_k}|^p * |Y|^p\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)=1\}} + |\left(g_{\gamma_k} * Y\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)>1\}}|^p \\
&= \left(|g_{\gamma_k}|^p * |Y|^p\right)(t) + |\left(g_{\gamma_k} * Y\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)>1\}}|^p - \left(|g_{\gamma_k}|^p * |Y|^p\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)>1\}} \\
&= \left(|g_{\gamma_k}|^p * |Y|^p\right)(t) + e_k(t)\,,
\end{aligned}$$

where

$$e_k(t) := |\left(g_{\gamma_k} * Y\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)>1\}}|^p - \left(|g_{\gamma_k}|^p * |Y|^p\right)(t) \cdot \mathbb{1}_{\{N_{s_k}(t)>1\}}$$

Using the above, we can write the second order convolution term as:

$$\left(g_{\gamma_k'} * |g_{\gamma_k} * Y|\right)(t) = \left(g_{\gamma_k'} * |g_{\gamma_k}|^p * |Y|^p\right)(t) + \left(g_{\gamma_k'} * e_k\right)(t)\,.$$

The following lemma implies that $\left(g_{\gamma_k'} * e_k\right)(t)$ decays rapidly in $\mathbf{L}^{p'}$ at a rate independent of t.

Lemma 15. *There exists $\delta > 0$, independent of t, such that if $s_k < \delta$,*

$$\mathbb{E}\left[\left|\left(g_{\gamma_k'} * e_k\right)(t)\right|^p\right] \leq C(p, p', w, c, L)\frac{\|\lambda\|_\infty}{\lambda_{\min}}\|\lambda\|_\infty^2 s_k^{d(p'+2)}\,.$$

Once we have proved Lemma 15, equation (4.13) will follow once we show,

$$\lim_{k\to\infty} \frac{\mathbb{E}\left[\left|\left(g_{\gamma_k'} * |g_{\gamma_k}|^p * |Y|^p\right)(t)\right|^{p'}\right]}{s_k^{d(p'+1)}} = K(p, p', w, c, L)\lambda(t)\mathbb{E}[|A_1|^q]\,. \tag{26}$$

Let us prove (26) first and postpone the proof of Lemma 15. We will use the fact that the support of $g_{\gamma_k'} * |g_{\gamma_k}|^p$ is contained in $[0, s_k + s_k']^d$. Let $\tilde{s}_k := s_k + s_k'$, $N_k(t) := N_{\tilde{s}_k}(t)$, $\Lambda_k(t) := \Lambda_{\tilde{s}_k}(t)$, and let $t_1, t_2, \ldots, t_{N_k(t)}$ be the points of N in the cube $[t - \tilde{s}_k, t]^d$. We have that $\mathbb{P}[N_k(t) = n] = e^{-\Lambda_k(t)}\frac{(\Lambda_k(t))^n}{n!}$, and conditioned on the event that $N_k(t) = n$, the locations of the points $t_1, \ldots, t_n$ are distributed as i.i.d. random variables $Z_1(t), \ldots, Z_n(t)$

taking values in $[t-\tilde{s}_k, t]^d$ with density $p_{Z(t)}(z) = \frac{\lambda(z)}{\Lambda_k(t)}$. Therefore the i.i.d. random variables $\widetilde{V}_1(t), \ldots, \widetilde{V}_n(t)$ defined by $\widetilde{V}_i(t) := t - Z_i(t)$ take values in $[0, \tilde{s}_k]^d$ and have density

$$p_{\widetilde{V}(t)}(v) = \frac{\lambda(t-v)}{\Lambda_k(t)}, \quad v \in [0, \tilde{s}_k]^d .$$

Now, we condition on $N_k(t)$ to see that

$$\begin{aligned}
\mathbb{E}\left[\left|\left(g_{\gamma_k'} * |g_{\gamma_k}|^p * |Y|^p\right)(t)\right|^{p'}\right] =& \mathbb{E}\left[\left|\sum_{j=1}^{N_k(t)} |A_j|^p \left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(t-t_j)\right|^{p'}\right] \\
=& \sum_{n=1}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^n}{n!} \cdot \qquad (27) \\
& \mathbb{E}\left[\left|\sum_{j=1}^{N_k(t)} |A_j|^p \left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(t-t_j)\right|^{p'} : N_k(t) = n\right] \\
=& \sum_{n=1}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^n}{n!} \mathbb{E}\left[\left|\sum_{j=1}^{n} |A_j|^p \left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_j(t))\right|^{p'}\right] \\
=& e^{-\Lambda_k(t)} \Lambda_k(t) \mathbb{E}[|A_1|^q] \mathbb{E}\left[\left|\left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_1(t))\right|^{p'}\right] \qquad (28) \\
& + \sum_{n=2}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^n}{n!} \mathbb{E}\left[\left|\sum_{j=1}^{n} |A_j|^p \left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_j(t))\right|^{p'}\right] . \qquad (29)
\end{aligned}$$

The following lemma will be used to estimate the scaling of the term in (28).

Lemma 16. *For all $t \in \mathbb{R}^d$,*

$$\lim_{k\to\infty} \frac{\tilde{s}_k^d}{s_k^{d(p'+1)}} \mathbb{E}\left[\left|\left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_1(t))\right|^{p'}\right] = \|g_{c,L/c} * |g_{1,0}|^p\|_{p'}^{p'} . \qquad (30)$$

Furthermore, there exists $\delta > 0$, independent of t, such that if $s_k < \delta$ then

$$\frac{\tilde{s}_k^d}{s_k^{d(p'+1)}} \mathbb{E}\left[\left|\left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_1(t))\right|^{p'}\right] \leq 2\frac{\|\lambda\|_\infty}{\lambda_{\min}} C(p, p', w, c, L) . \qquad (31)$$

Proof. Making a change of variables in both u and v, and recalling the assumption that $s'_k = cs_k$, we observe that

$$\begin{aligned}
&\frac{\tilde{s}_k^d}{s_k^{d(p'+1)}}\mathbb{E}\left[\left|\left(g_{\gamma'_k} * |g_{\gamma_k}|^p\right)(\widetilde{V}_1(t))\right|^{p'}\right] \\
&\qquad = \frac{\tilde{s}_k^d}{s_k^{d(p'+1)}}\int_{\mathbb{R}^d} p_{\widetilde{V}(t)}(v)\left|\int_{\mathbb{R}^d} w\left(\frac{v-u}{s'_k}\right)e^{i\xi'_k\cdot(v-u)}\left|w\left(\frac{u}{s_k}\right)\right|^p du\right|^{p'} dv \\
&\qquad = \tilde{s}_k^d\int_{\mathbb{R}^d} p_{\widetilde{V}(t)}(s_k v)\left|\int_{\mathbb{R}^d} w\left(\frac{s_k(v-u)}{s'_k}\right)e^{is_k\xi'_k\cdot(v-u)}|w(u)|^p\, du\right|^{p'} dv \\
&\qquad = \int_{\mathbb{R}^d}\frac{\tilde{s}_k^d\lambda(t-s_k v)}{\Lambda_k(t)}\left|\int_{\mathbb{R}^d} w\left(\frac{u-v}{c}\right)e^{is'_k\xi'_k\cdot(u-v)/c}|w(u)|^p\, du\right|^{p'} dv\,. \qquad (32)
\end{aligned}$$

The continuity of $\lambda(t)$ implies that

$$\lim_{k\to\infty}\frac{\tilde{s}_k^d\lambda(t-s_k v)}{\Lambda_k(t)} = 1\,, \quad \forall\, v\in[0,1+c]^d\,.$$

Furthermore, the assumption $0<\lambda_{\min}\le\|\lambda\|_\infty<\infty$ implies

$$\frac{\tilde{s}_k^d\lambda(t-s_k v)}{\Lambda_k(t)} \le \frac{\|\lambda\|_\infty}{\lambda_{\min}}\,, \quad \forall\, k\ge 1\,. \qquad (33)$$

Therefore, (30) follows from the dominated convergence theorem and by the observation that the inner integral of (32) is zero unless $v\in[0,1+c]^d$. Equation (31) follows from inserting (33) into (32) and sending k to infinity. □

Since

$$\lim_{k\to\infty}\frac{\Lambda_k(t)}{\tilde{s}_k^d} = \lambda(t)\,,$$

the independence of $\widetilde{V}_1(t)$ and A_1, the continuity of $\lambda(t)$, and Lemma 16 imply that taking $k\to\infty$ in (28) yields:

$$\begin{aligned}
&\lim_{k\to\infty}\left(\frac{e^{-\Lambda_k(t)}\Lambda_k(t)\mathbb{E}[|A_1|^q]\mathbb{E}\left[|g_{\gamma'_k} * |g_{\gamma_k}|^p(\widetilde{V}_1(t))|^{p'}\right]}{s_k^{d(p'+1)}}\right) \\
&\qquad = \lim_{k\to\infty}\left(e^{-\Lambda_k(t)}\frac{\Lambda_k(t)}{\tilde{s}_k^d}\mathbb{E}[|A_1|^q]\frac{\tilde{s}_k^d}{s_k^{d(p'+1)}}\mathbb{E}\left[|g_{\gamma'_k} * |g_{\gamma_k}|^p(\widetilde{V}_1(t))|^{p'}\right]\right) \\
&\qquad = K(p,p',c,w,L)\lambda(t)\mathbb{E}[|A_1|^q]\,.
\end{aligned}$$

The following lemma shows that (29) is $O\left(s_k^{d(p'+2)}\right)$ (and converges at a rate independent of t), and therefore completes the proof of (4.13) subject to proving Lemma 15.

Lemma 17. *For all $\alpha \in \mathbb{R}$ there exists $\delta > 0$, independent of t, such that if $s_k < \delta$, then*

$$\sum_{n=2}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^n}{n!} n^{\alpha} \mathbb{E}\left[\left|\sum_{j=1}^{n} |A_j|^p \left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_j(t))\right|^{p'}\right] \leq C(p,p',w,c,\alpha,L) \frac{\|\lambda\|_\infty}{\lambda_{\min}} \|\lambda\|_\infty^2 \mathbb{E}[|A_1|^q] s_k^{d(p'+2)}.$$

Proof. For any sequence of i.i.d. random variables, $Z_1, Z_2, \ldots$, it holds that

$$\mathbb{E}\left[\left|\sum_{n=1}^{k} Z_n\right|^p\right] \leq k^{p-1} \mathbb{E}\left[\sum_{n=1}^{k} |Z_n|^p\right] = k^p \mathbb{E}\left[|Z_1|^p\right].$$

Therefore, by Lemma 14, Lemma 16, and the fact that the $\widetilde{V}_j(t)$ and A_i are i.i.d. and independent of each other, we see that if $s_k < \delta$, where δ is as in (31),

$$\begin{aligned}
&\sum_{n=2}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^n}{n!} n^{\alpha} \mathbb{E}\left[\left|\sum_{j=1}^{n} |A_i|^p \left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_j(t))\right|^{p'}\right] \\
\leq &\sum_{n=2}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^n}{n!} n^{\alpha} n^{p'} \mathbb{E}\left[|A_1|^q \left|\left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_1(t))\right|^{p'}\right] \\
= &\sum_{n=2}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^n}{n!} n^{p'+\alpha} \mathbb{E}[|A_1|^q] \mathbb{E}\left[\left|\left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_1(t))\right|^{p'}\right] \\
= &\mathbb{E}[|A_1|^q] \mathbb{E}\left[\left|\left(g_{\gamma_k'} * |g_{\gamma_k}|^p\right)(\widetilde{V}_1(t))\right|^{p'}\right] \sum_{n=2}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^n}{n!} n^{p'+\alpha} \\
\leq &C(p,p',w,c,L) \frac{\|\lambda\|_\infty}{\lambda_{\min}} \mathbb{E}[|A_1|^q] \frac{s_k^{d(p'+1)}}{\tilde{s}_k^d} \sum_{n=2}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^n}{n!} n^{p'+\alpha} \\
\leq &C(p,p',w,c,L,\alpha) \frac{\|\lambda\|_\infty}{\lambda_{\min}} \mathbb{E}[|A_1|^q] \frac{s_k^{d(p'+1)}}{\tilde{s}_k^d} (\Lambda_k(t))^2 \\
\leq &C(p,p',w,c,L,\alpha) \frac{\|\lambda\|_\infty}{\lambda_{\min}} \|\lambda\|_\infty^2 \mathbb{E}[|A_1|^q] s_k^{d(p'+2)},
\end{aligned}$$

where the last inequality uses the fact that $\Lambda_k(t) \leq \tilde{s}_k^d \|\lambda\|_\infty = (1+c)^d s_k^d \|\lambda\|_\infty$. □

We will now complete the proof of the theorem by proving Lemma 15.

Proof. **[Lemma 15]** Since

$$e_k(t) = |\left(g_{\gamma_k} * Y\right)(t)\mathbb{1}_{\{N_{s_k}(t)>1\}}|^p - \left(|g_{\gamma_k}|^p * |Y|^p\right)(t)\mathbb{1}_{\{N_{s_k}(t)>1\}},$$

we see that

$$\left|g_{\gamma_k'} * e_k(t)\right| \leq \left|g_{\gamma_k'} * \left(\left|\left(g_{\gamma_k} * Y\right)\mathbb{1}_{\{N_{s_k}(\cdot)>1\}}\right|^p\right)(t)\right| + \left|g_{\gamma_k'} * \left(\left(|g_{\gamma_k}|^p * |Y|^p\right)\mathbb{1}_{\{N_{s_k}(\cdot)>1\}}\right)(t)\right|.$$

First turning our attention to the second term, we note that

$$\begin{aligned}
&\left|g_{\gamma_k'} * \left(\left(|g_{\gamma_k}|^p * |Y|^p\right)\mathbb{1}_{\{N_{s_k}(\cdot)>1\}}\right)(t)\right| \\
&\quad= \left|\int_{[t-s_k',t]^d} w\left(\frac{t-u}{s_k'}\right) e^{i\xi_k'\cdot(t-u)}\left(|g_{\gamma_k}|^p * |Y|^p\right)(u)\mathbb{1}_{\{N_{s_k}(u)>1\}}\,du\right| \\
&\quad\leq \mathbb{1}_{\{N_k(t)>1\}}\int_{[t-s_k',t]^d} w\left(\frac{t-u}{s_k'}\right)\left(|g_{\gamma_k}|^p * |Y|^p\right)(u)\,du \\
&\quad= \mathbb{1}_{\{N_k(t)>1\}}\left(g_{s_k',0} * |g_{\gamma_k}|^p * |Y|^p\right)(t). \qquad (34)
\end{aligned}$$

since $N_{s_k}(u) \leq N_{s_k+s_k'}(t) = N_{\tilde{s}_k}(t) = N_k(t)$ for all $u \in [t-s_k', t]^d$. Therefore, conditioning on $N_k(t)$, if $s_k < \delta$,

$$\begin{aligned}
&\mathbb{E}\left[\left|g_{\gamma_k'} * \left(\left(|g_{\gamma_k}|^p * |Y|^p\right)\mathbb{1}_{\{N_{s_k}(\cdot)>1\}}\right)(t)\right|^{p'}\right] \\
&\quad\leq \mathbb{E}\left[\left|\mathbb{1}_{\{N_k(t)>1\}}\left(g_{s_k',0} * |g_{\gamma_k}|^p * |Y|^p\right)(t)\right|^{p'}\right] \\
&\quad= \sum_{n=2}^{\infty} e^{-\Lambda_k(t)}\frac{(\Lambda_k(t))^n}{n!}\mathbb{E}\left[\left|\sum_{j=1}^{n}|A_j|^p\left(g_{s_k',0} * |g_{\gamma_k}|^p\right)(\widetilde{V}_j(t))\right|^{p'}\right] \\
&\quad\leq C(p,p',w,c,L)\frac{\|\lambda\|_\infty}{\lambda_{\min}}\|\lambda\|_\infty^2\mathbb{E}[|A_1|^q]s_k^{d(p'+2)}
\end{aligned}$$

by Lemma 17. Now, turning our attention to the first term, note that

$$|(g_{\gamma_k} * Y)(t)|^p\,\mathbb{1}_{\{N_{s_k}(t)>1\}} \leq N_{s_k}(t)^{p-1}\left(|g_{\gamma_k}|^p * |Y|^p\right)(t)\mathbb{1}_{\{N_{s_k}(t)>1\}}.$$

Therefore, by the same logic as in (34)

$$\begin{aligned}
&\left|g_{\gamma_k'} * \left(\left|\left(g_{\gamma_k} * Y\right) \mathbb{1}_{\{N_{s_k}(\cdot)>1\}}\right|^p\right)(t)\right| \\
&\quad \leq \int_{[t-s_k',t]^d} w\left(\frac{t-u}{s_k'}\right) N_{s_k}(u)^{p-1} \left(|g_{\gamma_k}|^p * |Y|^p\right)(u) \mathbb{1}_{\{N_{s_k}(u)>1\}}\, du \\
&\quad \leq \mathbb{1}_{\{N_k(t)>1\}} N_k(t)^{p-1} \int_{[t-s_k',t]^d} w\left(\frac{t-u}{s_k'}\right) \left(|g_{\gamma_k}|^p * |Y|^p\right)(u)\, du \\
&\quad \leq \mathbb{1}_{\{N_k(t)>1\}} N_k(t)^{p-1} \left(g_{s_k',0} * \left(|g_{\gamma_k}|^p * |Y|^p\right)\right)(t).
\end{aligned}$$

So again conditioning on $N_k(t)$, and applying Lemma 17, we see that if $s_k < \delta$

$$\begin{aligned}
&\mathbb{E}\left[\left|g_{\gamma_k'} * \left(\left|\left(g_{\gamma_k} * Y\right) \mathbb{1}_{\{N_{s_k}(\cdot)>1\}}\right|^p\right)(t)\right|^{p'}\right] \\
&\quad \leq \sum_{n=2}^{\infty} e^{-\Lambda_k(t)} \frac{(\Lambda_k(t))^k}{n!} n^{p-1} \mathbb{E}\left[\left|\sum_{j=1}^{n} |A_j|^p \left|\left(g_{s_k',0} * |g_{\gamma_k}|\right)\left(\widetilde{V}_j(t)\right)\right|^p\right|^{p'}\right] \\
&\quad \leq C(p,p',w,c,L) \frac{\|\lambda\|_\infty}{\lambda_{\min}} \|\lambda\|_\infty^2 \mathbb{E}[|A_1|^q] s_k^{d(p'+2)}.
\end{aligned}$$

□

This completes the proof of (4.13). Line (4.14) follows from integrating with respect to t, observing that the error bounds in Lemmas 15 and 16 are independent of t, and applying the bounded convergence theorem.

□

Proofs of Results from Section 4.5

In order to prove Theorems 10 and 11, we will need the following lemma which shows that the scaling relationship of a self-similar process $X(t)$induces a similar relationship on stochastic integrals against $dX(t)$.

Lemma 18. *If X is a stochastic process that satisfies the scaling relation*

$$X(st) \stackrel{d}{=} s^\beta X(t) \tag{35}$$

for some $\beta > 0$, then for any measurable function $f : \mathbb{R} \to \mathbb{R}$,

$$\int_0^s f(u)\, dX(u) \stackrel{d}{=} s^\beta \int_0^1 f(su)\, dX(u)\,.$$

Proof. Let $X = (X(t))_{t\in\mathbb{R}}$ be a stochastic process satisfying (35), and let $\mathcal{P}_n = \{0 = t_0^n < t_1^n < \ldots < t_{K_n}^n = 1\}$ be a sequence of partitions of $[0,1]$ such that

$$\lim_{n\to\infty} \max_k \left\{|t_k^n - t_{k-1}^n|\right\} = 0.$$

Then, by the scaling relation (35),

$$\begin{aligned}\int_0^s f(u)\, dX(u) &= \lim_{n\to\infty} \sum_{k=0}^{K_n-1} f(st_k^n)\left(X(st_{k+1}^n) - X(st_k^n)\right) \\ &\stackrel{d}{=} s^\beta \lim_{n\to\infty} \sum_{k=0}^{K_n-1} f(st_k^n)\left(X(t_{k+1}^n) - X(t_k^n)\right) = s^\beta \int_0^1 f(su)\, dX(u)\,.\end{aligned}$$

□

We will now use Lemma 18 to prove Theorems 10 and 11.

Proof. **[Theorem 10]** Let $X = (X(t))_{t\in\mathbb{R}}$ be the α-stable process, $p < \alpha \leq 2$. Since X has stationary increments, its scattering coefficients do not depend on t and it suffices to analyze

$$\mathbb{E}\left[|(g_{\gamma_k} * dX)(0)|^p\right] = \mathbb{E}\left[\left|\int_{-s_k}^0 g_{\gamma_k}(u)\, dX(u)\right|^p\right] = \mathbb{E}\left[\left|\int_0^{s_k} g_{\gamma_k}(u)\, dX(u)\right|^p\right],$$

where the second equality uses the fact the distribution of X does not change if it is run in reverse, i.e.

$$(X(t))_{t\in\mathbb{R}} \stackrel{d}{=} (X(-t))_{t\in\mathbb{R}}$$

It is well known that $X(t)$ satisfies (35) for $\beta = 1/\alpha$. Therefore, by Lemma 18

$$\mathbb{E}\left[|(g_{\gamma_k} * dX)(0)|^p\right] = \mathbb{E}\left[\left|\int_0^{s_k} w\left(\frac{u}{s_k}\right) e^{i\xi_k u}\, dX(u)\right|^p\right] = s_k^{p/\alpha}\mathbb{E}\left[\left|\int_0^1 w(u) e^{i\xi_k s_k u}\, dX(u)\right|^p\right].$$

So,

$$\frac{\mathbb{E}\left[|(g_{\gamma_k} * dX)(0)|^p\right]}{s_k^{p/\alpha}} = \mathbb{E}\left[\left|\int_0^1 w(u) e^{i\xi_k s_k u}\, dX(u)\right|^p\right].$$

The proof will be complete as soon as we show that

$$\lim_{k\to\infty}\left(\mathbb{E}\left[\left|\int_0^1 w(u)e^{i\xi_k s_k u}\,dX(u)\right|^p\right]\right)^{1/p}=\left(\mathbb{E}\left[\left|\int_0^1 w(u)e^{iLu}\,dX(u)\right|^p\right]\right)^{1/p}.$$

By the triangle inequality,

$$\begin{aligned}&\left|\left(\mathbb{E}\left[\left|\int_0^1 w(u)e^{i\xi_k s_k u}\,dX(u)\right|^p\right]\right)^{1/p}-\left(\mathbb{E}\left[\left|\int_0^1 w(u)e^{iLu}\,dX(u)\right|^p\right]\right)^{1/p}\right|\\&\quad\le\left(\mathbb{E}\left[\left|\int_0^1 w(u)\left(e^{i\xi_k s_k u}-e^{iLu}\right)\,dX(u)\right|^p\right]\right)^{1/p}.\end{aligned}$$

Since $1\le p<\alpha$, we may choose p' strictly greater than 1 such that $p\le p'<\alpha$, and note that by Jensen's inequality

$$\left(\mathbb{E}\left[\left|\int_0^1 w(u)\left(e^{i\xi_k s_k u}-e^{iLu}\right)\,dX(u)\right|^p\right]\right)^{1/p}\le\left(\mathbb{E}\left[\left|\int_0^1 w(u)\left(e^{i\xi_k s_k u}-e^{iLu}\right)\,dX(u)\right|^{p'}\right]\right)^{1/p'},$$

and since $X(t)$ is a p'-integrable martingale, the boundedness of martingale transforms [72] (see also [73]) implies

$$\begin{aligned}&\left(\mathbb{E}\left[\left|\int_0^1 w(u)\left(e^{i\xi_k s_k u}-e^{iLu}\right)\,dX(u)\right|^{p'}\right]\right)^{1/p'}\\&\quad\le C_{p'}\sup_{0\le u\le 1}\left|w(u)\left(e^{i\xi_k s_k u}-e^{iLu}\right)\right|\mathbb{E}\left[|X_1|^{p'}\right]\le C_{p'}|s_k\xi_k-L|\|w\|_\infty\mathbb{E}\left[|X_1|^{p'}\right],\end{aligned}$$

which converges to zero by the continuity of w on $[0,1]$ and the assumption that $s_k\xi_k$ converges to L. □

Proof. **[Theorem 11]** Similarly to the proof of Theorem 10, it suffices to show that if a $(X(t))_{t\in\mathbb{R}}$ is fractional Brownian motion with Hurst parameter H, then

$$\lim_{k\to\infty}\left(\mathbb{E}\left[\left|\int_0^1 w(u)\left(e^{i\xi_k s_k u}-e^{iLu}\right)\,dX(u)\right|^p\right]\right)^{1/p}=0.$$

However, fractional Brownian motion is not a semi-martingale so we cannot apply Burkholder's theorem as we did in the proof of Theorem 10. Instead, we use the following result first established in [74] (see also [75], p. 48) which states that if $x(u)$ is any (deterministic) function

with bounded variation, and $y(u)$ is any function which is α-Hölder continuous, $0 < \alpha < 1$, then

$$\int_0^1 x(u)\, dy(u)$$

is well-defined as the limit of Riemann sums and

$$\left| \int_0^1 x(u)\, dy(u) - x(0)\,(y(1) - y(0)) \right| \leq C_\alpha \|x\|_{BV} \|y\|_\alpha \,,$$

where $\|\cdot\|_{BV}$ and $\|\cdot\|_\alpha$ are the bounded variation and α-Hölder seminorms respectively. For all k, the function $h_k(u) \coloneqq w(u)\left(e^{i\xi_k s_k u} - e^{iLu}\right) \coloneqq w(u) f_k(u)$ satisfies, $h_k(0) = 0$ and

$$\|h_k\|_{BV} \leq \|w\|_\infty \|f_k\|_{BV} + \|w\|_{BV} \|f_k\|_\infty \,.$$

One can check that the fact that $s_k \xi_k$ converges to L implies that f_k converges to zero in both $\mathbf{L}^\infty$ and in the bounded variation seminorm, and that therefore that $\|h_k\|_{BV}$ converges to zero.

It is well-known that fractional Brownian motion with Hurst parameter H admits a continuous modification which is α-Hölder continuous for any $\alpha < H$. Therefore,

$$\mathbb{E}\left[\left| \int_0^1 w(u) \left(e^{i\xi_k s_k u} - e^{iLu}\right)\, dX(u) \right|^p \right] \leq C_\alpha^p \|h_k\|_{BV}^p \mathbb{E}\left[\|X\|_\alpha^p \right] \,.$$

Lastly, one can use the Garsia-Rodemich-Rumsey inequality [76], to show that

$$\mathbb{E}[\|X\|_\alpha^p] < \infty \,.$$

for all $1 < p < \infty$. For details we refer the reader to the survey article [77]. Therefore,

$$\lim_{k \to 0} \mathbb{E}\left[\left| \int_0^1 w(u) \left(e^{i\xi_k s_k u} - e^{iLu}\right)\, dX(u) \right|^p \right] = 0$$

as desired.

□

Remark 8. *The assumption that w has bounded-variation was used to justify that the stochastic integral against fractional Brownian motion was well defined as the limit of Riemann sums because of its Hölder continuity and the above mentioned result of [74]. This allowed us to avoid the technical complexities of defining such an integral using either the Malliavin calculus or the Wick product.*

Details of Numerical Experiments

Definition of Filters

For all the numerical experiments, we take the window function w to be the smooth bump function

$$w(t) = \begin{cases} \exp\left(-\frac{1}{4t-4t^2}\right), & t \in (0,1) \\ 0, & \text{otherwise.} \end{cases}$$

Therefore for $\gamma = (s, \xi)$, our filters are given by

$$g_\gamma(t) = e^{i\xi t} w(t) = \begin{cases} e^{i\xi t} e^{-s^2/(4ts-4t^2)}, & t \in (0,s) \\ 0, & \text{otherwise} \end{cases}.$$

Frequencies

In all of our experiments, we hold the frequency, ξ, which we sample uniformly at random from $(0, 2\pi)$, constant while allowing the scale to decrease to zero.

Simulation of Poisson point process

We use the standard method to generate a realization of a Poisson point process. For Poisson point process with intensity λ, the time interval between two neighbor jumps follows exponential distribution:

$$\Delta_j \coloneqq t_j - t_{j-1} \sim \text{Exp}(\lambda)\,.$$

Therefore, taking the inverse cumulative distribution function, we sample the time interval between two neighbor jumps through:

$$\Delta_j = -\frac{\log U_j}{\lambda},$$

where U_j are i.i.d. uniform random variables on $[0, 1]$, and assign the charge A_j to the jump at location t_j.

For inhomogeneous Poisson process with intensity funciton $\lambda(t)$, we simulate the time interval based on the proposition from [78]. First define the cumulated intensity:

$$\Lambda(t) = \int_0^t \lambda(s)ds\,,$$

then generate the location of jumps t_j by the following algorithm:

Algorithm 2 Algorithm for simulating inhomogeneous Poisson point process

initialize $V = 0$, $t = 0$
while $t < N$ **do**
  generate $U \sim \mathcal{U}([0, 1])$
  $V \leftarrow V - \log U$
  $t = \inf\{v : \Lambda(v) < V\}$
  deliver t

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[2] Javier Portilla and Eero Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *International Journal of Computer Vision*, 40(1):49–71, 10 2000.

[3] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in neural information processing systems*, pages 262–270, 2015.

[4] Stephane Mallat. A wavelet tour of signal processing, third edition: The sparse way. *Academic Press, 3rd edition*, 2008.

[5] B. S. Manjunath and W. Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(8):837–842, August 1996.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[10] Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, 2012.

[11] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive Growing of GANs for Improved Quality, Stability, and Variation. *ArXiv e-prints*, October 2017.

[12] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative Adversarial Text to Image Synthesis. *ArXiv e-prints*, May 2016.

[13] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei Efros. Context encoders: Feature learning by inpainting. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[14] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.

[15] Leon A. Gatys, Alexander S. Ecker, and M. Bethge. A neural algorithm of artistic style. *ArXiv*, abs/1508.06576, 2015.

[16] Joakim Andén and Stéphane Mallat. Multiscale scattering for audio classification. In *Proceedings of the ISMIR 2011 conference*, pages 657–662, 2011.

[17] Joakim Andén and Stéphane Mallat. Deep scattering spectrum. *IEEE Transactions on Signal Processing*, 62(16):4114–4128, August 2014.

[18] G. Wolf, S. Mallat, and S.A. Shamma. Audio source separation with time-frequency velocities. In *2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, Reims, France, 2014.

[19] Guy Wolf, Stephane Mallat, and Shihab A. Shamma. Rigid motion model for audio source separation. *IEEE Transactions on Signal Processing*, 64(7):1822–1831, 2015.

[20] Joakim Andén, Vincent Lostanlen, and Stéphane Mallat. Classification with joint time-frequency scattering. arXiv:1807.08869, 2018.

[21] Joan Bruna and Stéphane Mallat. Classification with scattering operators. In *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1561–1566, 2011.

[22] Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886, August 2013.

[23] Laurent Sifre and Stéphane Mallat. Combined scattering for rotation invariant texture analysis. In *Proceedings of the ESANN 2012 conference*, 2012.

[24] Laurent Sifre and Stéphane Mallat. Rotation, scaling and deformation invariant scattering for texture discrimination. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013.

[25] Laurent Sifre and Stéphane Mallat. Rigid-motion scattering for texture classification. arXiv:1403.1687, 2014.

[26] Edouard Oyallon and Stéphane Mallat. Deep roto-translation scattering for object classification. In *Proceedings in IEEE CVPR 2015 conference*, 2015. arXiv:1412.8659.

[27] Matthew Hirn, Stéphane Mallat, and Nicolas Poilvert. Wavelet scattering regression of quantum chemical energies. *Multiscale Modeling and Simulation*, 15(2):827–863, 2017. arXiv:1605.04654.

[28] Michael Eickenberg, Georgios Exarchakis, Matthew Hirn, and Stéphane Mallat. Solid harmonic wavelet scattering: Predicting quantum molecular energy from invariant descriptors of 3D electronic densities. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pages 6540–6549, 2017.

[29] Michael Eickenberg, Georgios Exarchakis, Matthew Hirn, Stéphane Mallat, and Louis Thiry. Solid harmonic wavelet scattering for predictions of molecule properties. *Journal of Chemical Physics*, 148:241732, 2018.

[30] Xavier Brumwell, Paul Sinz, Kwang Jin Kim, Yue Qi, and Matthew Hirn. Steerable wavelet scattering for 3D atomic systems with application to Li-Si energy prediction. In *NeurIPS Workshop on Machine Learning for Molecules and Materials*, 2018.

[31] Stéphane Mallat and Iréne Waldspurger. Phase retrieval for the cauchy wavelet transform. *Journal of Fourier Analysis and Applications*, 21(6):1251–1309, 2015.

[32] Juri Ranieri, Amina Chebira, Yue M. Lu, and Martin Vetterli. Phase retrieval for sparse signals: Uniqueness conditions. *CoRR*, abs/1308.3058, 2013.

[33] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems 28*, pages 262–270, 2015.

[34] Joseph Antognini, Matt Hoffman, and Ron J. Weiss. Synthesizing diverse, high-quality audio textures. arXiv:1806.08002, 2018.

[35] Mikolaj Binkowski, Gautier Marti, and Philippe Donnat. Autoregressive convolutional neural networks for asynchronous time series. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 580–589, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

[36] Antoine Brochard, Bartłomiej Błaszczyszyn, Stéphane Mallat, and Sixin Zhang. Statistical learning of geometric characteristics of wireless networks. arXiv:1812.08265, 2018.

[37] Martin Haenggi, Jeffrey G. Andrews, François Baccelli, Olivier Dousse, and Massimo Franceschetti. Stochastic geometry and random graphs for the analysis and design of wireless networks. *IEEE Journal on Selected Areas in Communications*, 27(7):1029–1046, 2009.

[38] Astrid Genet, Pavel Grabarnik, Olga Sekretenko, and David Pothier. Incorporating the mechanisms underlying inter-tree competition into a random point process model to improve spatial tree pattern analysis in forestry. *Ecological Modelling*, 288:143–154, 09 2014.

[39] Frederic Paik Schoenberg. A note on the consistent estimation of spatial-temporal point process parameters. *Statistica Sinica*, 2016.

[40] V. Fromion, E. Leoncini, and P. Robert. Stochastic gene expression in cells: A point process approach. *SIAM Journal on Applied Mathematics*, 73(1):195–211, 2013.

[41] Joan Bruna, Stéphane Mallat, Emmanuel Bacry, and Jean-Francois Muzy. Intermittent process analysis with scattering moments. *Annals of Statistics*, 43(1):323 – 351, 2015.

[42] D. J. Daley and D. Vere-Jones. *An introduction to the theory of point processes. Vol. I.* Probability and its Applications (New York). Springer-Verlag, New York, second edition, 2003. Elementary theory and methods.

[43] Joan Bruna and Stéphane Mallat. Multiscale sparse microcanonical models. *Mathematical Statistics and Learning*, 1(3/4):257–315, 01 2018.

[44] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pages 2672–2680, 2014.

[45] Urs Bergmann, Nikolay Jetchev, and Roland Vollgraf. Learning texture manifolds with the periodic spatial GAN. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 469–477, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

[46] Nikolay Jetchev, Urs Bergmann, and Roland Vollgraf. Texture synthesis with spatial generative adversarial networks. *ArXiv*, abs/1611.08207, 2016.

[47] Nikolay Jetchev, Urs Bergmann, and C. Seward. Ganosaic: Mosaic creation with generative texture manifolds. *ArXiv*, abs/1712.00269, 2017.

[48] Wenqi Xian, Patsorn Sangkloy, Jingwan Lu, Chen Fang, F. Yu, and James Hays. Texturegan: Controlling deep image synthesis with texture patches. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8456–8465, 2018.

[49] Yang Zhou, Zhen Zhu, Xiang Bai, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Non-stationary texture synthesis by adversarial expansion. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 37(4), 2018.

[50] Adib Akl, Charles Yaacoub, Marc Donias, Jean-Pierre Da Costa, and Christian Germain. A survey of exemplar-based texture synthesis methods. *Computer Vision and Image Understanding*, 172:12–24, 2018.

[51] Bruno Galerne, Yann Gousseau, and Jean-Michel Morel. Micro-Texture Synthesis by Phase Randomization. *Image Processing On Line*, 1:213–237, 2011.

[52] B. Galerne, Y. Gousseau, and J. Morel. Random phase textures: Theory and synthesis. *IEEE Transactions on Image Processing*, 20:257–267, 2011.

[53] Song Chun Zhu, Yingnian Wu, and David Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):107–126, 1998.

[54] Y. Lu, Song-Chun Zhu, and Y. Wu. Learning frame models using cnn filters for knowledge visualization. *ArXiv*, abs/1509.08379, 2015.

[55] Valentin De Bortoli, Agnès Desolneux, Bruno Galerne, and Arthur Leclaire. Macrocanonical models for texture synthesis. In Jan Lellmann, Martin Burger, and Jan Modersitzki, editors, *Scale Space and Variational Methods in Computer Vision*, pages 13–24, Cham, 2019. Springer International Publishing.

[56] Valentin De Bortoli, A. Desolneux, Alain Durmus, B. Galerne, and A. Leclaire. Maximum entropy methods for texture synthesis: theory and practice. *SIAM J. Math. Data Sci.*, 3:52–82, 2021.

[57] D. Ulyanov, V. Lebedev, A. Vedaldi, and V. Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. In *ICML*, 2016.

[58] J. Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[59] Valentin De Bortoli, A. Desolneux, B. Galerne, and A. Leclaire. Macrocanonical models for texture synthesis. In *SSVM*, 2019.

[60] Ivan Ustyuzhaninov, Wieland Brendel, Leon Gatys, and Matthias Bethge. What does it take to generate natural textures? In *Proceedings of the International Conference on Learning Representations*, 2017.

[61] David J. Heeger and James R. Bergen. Pyramid-based texture analysis/synthesis. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 229–238, 1995.

[62] Thibaud Briand, Jonathan Vacher, Bruno Galerne, and Julien Rabin. The Heeger-Bergen pyramid-based texture synthesis algorithm. *Image Processing On Line*, 4:279–299, 2014.

[63] Nicolas Gonthier, Yann Gousseau, and Saïd Ladjal. High resolution neural texture synthesis with long range constraints. *CoRR*, abs/2008.01808, 2020.

[64] Xavier Snelgrove. High-resolution multi-scale neural texture synthesis. In *SIGGRAPH ASIA 2017 Technical Briefs*, SA '17, New York, NY, USA, 2017. ACM.

[65] Benjamin Balas, Lisa Nakano, and Ruth Rosenholtz. A summary statistic representation in peripheral vision explains visual crowding. *Journal of vision*, 9:13.1–18, 11 2009.

[66] Jeremy Freeman, Corey M Ziemba, David J Heeger, Eero P Simoncelli, and J Anthony Movshon. A functional and perceptual signature of the second visual area in primates. *Nature Neuroscience*, pages 974–981, 07 2013.

[67] Gouki Okazawa, Satohiro Tajima, and Hidehiko Komatsu. Image statistics underlying natural texture selectivity of neurons in macaque v4. *Proceedings of the National Academy of Sciences*, 112(4):E351–E360, 2015.

[68] R. Brüel Gabrielsson and G. Carlsson. Exposition and interpretation of the topology of neural networks. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 1069–1076, 2019.

[69] Sixin Zhang and S. Mallat. Maximum entropy models from phase harmonic covariances. *ArXiv*, abs/1911.10017, 2019.

[70] Lionel Moisan. Periodic plus Smooth Image Decomposition. working paper or preprint, May 2009.

[71] Matthew Hirn and Anna Little. Wavelet invariants for statistically robust multi-reference alignment. arXiv:1909.11062, 2019.

[72] Donald L. Burkholder. Sharp inequalities for martingales and stochastic integrals. In *Colloque Paul Lévy sur les processus stochastiques*, number 157-158 in Astérisque, pages 75–94. Société mathématique de France, 1988.

[73] Rodrigo Bañuelos and Gang Wang. Sharp inequalities for martingales with applications to the beurling-ahlfors and riesz transforms. *Duke Math. J.*, 80(3):575–600, 12 1995.

[74] L. C. Young. An inequality of the Hölder type, connected with Stieltjes integration. *Acta Math.*, 67:251–282, 1936.

[75] Peter K Friz and Martin Hairer. *A course on rough paths: with an introduction to regularity structures*. Springer, 2014.

[76] A. M. Garsia, E. Rodemich, and H. Rumsey Jr. A real variable lemma and the continuity of paths of some Gaussian processes. *Indiana University Mathematics Journal*, 20(6):565–578, 1970.

[77] G. Shevchenko. Fractional Brownian motion in a nutshell. In *International Journal of Modern Physics Conference Series*, volume 36 of *International Journal of Modern Physics Conference Series*, page 1560002, January 2015.

[78] E. (Erhan) Cinlar. *Introduction to stochastic processes*. Prentice-Hall, Englewood Cliffs, N.J., 1975.