A theoretical argument for complex-valued convolutional networks

Joan Bruna, Soumith Chintala, Yann LeCun,
Serkan Piantino, Arthur Szlam, and Mark Tygert

Facebook Artificial Intelligence Research

March 12, 2015

Abstract

A complex-valued convolutional network (convnet) implements the repeated application of the following composition of three operations, recursively applying the composition to an input vector of nonnegative real numbers: (1) convolution with several complex-valued vectors followed by (2) taking the absolute value of every entry of the resulting vectors followed by (3) local averaging. For processing real-valued random vectors, complex-valued convnets can be viewed as "data-driven multiscale windowed power spectra," "data-driven multiscale windowed absolute spectra," "data-driven multiwavelet absolute values," or (in their most general configuration) "data-driven nonlinear multiwavelet packets." Indeed, complex-valued convnets can calculate multiscale windowed spectra when the convnet filters are windowed complex-valued exponentials. Standard real-valued convnets, using rectified linear units (ReLUs), sigmoidal (for example, logistic or tanh) nonlinearities, max. pooling, etc., do not obviously exhibit the same exact correspondence with data-driven wavelets (whereas for complex-valued convnets, the correspondence is much more than just a vague analogy).

This note develops "data-driven multiscale windowed spectra" for certain stochastic processes that are common in the modeling of time series (such as audio) and natural images (including patterns and textures). We motivate the construction of such multiscale spectra in the form of "local averages of multiwavelet absolute values" or (in the most general configuration) "nonlinear multiwavelet packets" and connect these to certain "complex-valued convolutional networks." A textbook treatment of all concepts and terms used above and below is given by [12]. Further information is available in the original work of [7], [15], [5], [4], [19], [16], [9], [20], and [18], for example. The work of [8], [13], [17], [2], and [3] also develops complex-valued convolutional networks (convnets). Renormalization group theory and its connection to convnets is discussed by [14]; this connection is incredibly insightful, though we leave further discussion to the cited work. Our exposition relies on nothing but the basic signal processing treated by [12].

For simplicity, we first limit consideration to the special case of a doubly infinite sequence of nonnegative random variables X_k, where k ranges over the integers. This input data will be the result of convolving an unmeasured independent and identically distributed (i.i.d.) sequence Z_k, where k ranges over the integers, with an unknown sequence of real numbers f_k, where k ranges over the integers (this latter sequence is known as a "filter," whereas the i.i.d. sequence is known as "white noise"):

$$X_j = \sum_{k=-\infty}^{\infty} f_{j-k}\, Z_k \tag{1}$$

for any integer j. Such a sequence X_k, with k ranging over the integers, is a (strictly) "stationary stochastic process." The terminology "strictly stationary" refers to the fact that lagging or shifting the process preserves the probability distribution of the process: for any integer l, the shift $Y_k = X_{k-l}$, where k ranges over the integers, satisfies

$$Y_j = \sum_{k=-\infty}^{\infty} f_{j-k}\, Z'_k \tag{2}$$

for any integer j, where $Z'_k = Z_{k-l}$; the sequence Z'_k, with k ranging over the integers, is i.i.d. with the same distribution as Z_k, where k ranges over the integers.

The associated "absolute spectrum" is

$$\tilde{X}(\omega) = \lim_{n\to\infty} \mathbf{E} \left| \frac{1}{\sqrt{2n+1}} \sum_{k=-n}^{n} e^{-ik\omega} X_k \right| \tag{3}$$

for any real number ω (usually we consider not just any, but instead restrict consideration to a sequence running from 0 to about 2π). Please note that lagging or shifting the process changes neither the probability distribution of the process (since the process is stationary) nor the absolute spectrum: for any integer l, the shift $Y_k = X_{k-l}$ yields $\tilde{Y}(\omega) = \tilde{X}(\omega)$ for any real number ω, due to the absolute value in (3).

Similarly, the associated "power spectrum" is

$$\tilde{\tilde{X}}(\omega) = \lim_{n\to\infty} \mathbf{E} \left| \frac{1}{\sqrt{2n+1}} \sum_{k=-n}^{n} e^{-ik\omega} X_k \right|^2 \tag{4}$$

for any real number ω; there is an extra squaring under the expectation in (4) compared to (3). Again, lagging or shifting the process changes neither the probability distribution of the process nor the power spectrum: for any integer l, the shift $Y_k = X_{k-l}$ yields $\tilde{\tilde{Y}}(\omega) = \tilde{\tilde{X}}(\omega)$ for any real number ω, due to the absolute value in (4). The remainder of the present paper focuses on the absolute spectrum; most of the discussion applies to the power spectrum, too.

Remark 1. The absolute spectrum can be more robust than the power spectrum, in the same sense that the mean absolute deviation can be more robust than the variance or standard deviation. The power spectrum is more fundamental in a certain sense, yet the absolute spectrum may be preferable for applications to machine learning. We suspect that both work about the same. We focus on the absolute spectrum to simplify the exposition.

In practice, the input data is rarely strictly stationary, but usually only locally stationary, that is, (1) becomes

$$X_j = \sum_{k=-\infty}^{\infty} f^{(j)}_{j-k}\, Z_k \tag{5}$$

for any integer j, where $f^{(j)}_k$ changes much more slowly when changing j than when changing k. To accommodate such data, we introduce windowed spectra; for any even nonnegative-valued sequence g_k, with k ranging through the integers — this sequence could be samples of a Gaussian or any other window suitable for Gabor analysis (the data itself will determine g during training) — we consider

$$\tilde{X}_l(\omega) = \frac{1}{2n+1} \sum_{j=-n+l}^{n+l} \left| \frac{1}{\sqrt{2n+1}} \sum_{k=-\infty}^{\infty} e^{-ik\omega} g_{k-j} X_k \right| \tag{6}$$

for any integer l, with some positive integer n. The extra summation in (6) averages away noise and is a kind of approximation to the expected value in (3). Usually g_k is fairly close to 1 for $k = -n, -n+1, \ldots, n-1, n$, and g_k is fairly close to 0 for $|k| > n$, making a reasonably smooth transition between 0 and 1. The most important difference between (3) and (6) is the absence of a limit in the latter (hence the terminology, "localized" spectrum).

Due to the absolute value, (6) is equivalent to

$$\tilde{X}_l(\omega) = \frac{1}{2n+1} \sum_{j=-n+l}^{n+l} \left| \frac{1}{\sqrt{2n+1}} \sum_{k=-\infty}^{\infty} g_{j-k}(\omega)\, X_k \right| \tag{7}$$

for any even nonnegative-valued sequence g_k, with k ranging through the integers, where

$$g_k(\omega) = e^{ik\omega} g_k \tag{8}$$

for any integer k. Please note that the right-hand side of (7) is just a convolution followed by the absolute value followed by local averaging; this will facilitate fitting/learning/training using data — enabling a "data-driven" approach.

In most cases, the ideal choices of n and width of the window in (7), that is, the ideal number of indices for which g_k is substantially nonzero, are far from obvious. Often, in fact, multiple widths are relevant (say, wider for lower-frequency variations than for higher frequency). Not knowing the ideal a priori, we use multiple windows on multiple scales. An especially efficient multiscale implementation processes the results of the lowest-frequency channels recursively. For the lowest frequency, $\omega = 0$, and when X_k is nonnegative for every integer k (for example, the input X_k could be the $\tilde{X}_k$ arising from previous processing), (7) simplifies to

$$\tilde{X}_l(0) = \frac{1}{\sqrt{2n+1}} \sum_{k=-\infty}^{\infty} h_{l-k} X_k \tag{9}$$

for any integer l, where

$$h_l = \frac{1}{2n+1} \sum_{j=-n+l}^{n+l} g_j \tag{10}$$

for any integer l, and again g_j, with j ranging through the integers, is an even sequence of nonnegative real numbers. The result of (9) is simply a convolution with the input sequence, and further convolutions — say via recursive processing of the form in (7) — can undo this convolution and set the effective window however desired in later stages. The deconvolution and subsequent convolution with the windowed exponential of a later stage is numerically stable if the later window is wider than the preceding. In particular, recursively processing the zero-frequency channels in this way can implement a "wavelet transform" (if each recursive stage considers only two values for ω, one zero and one nonzero — see Figure 1) or a "multiwavelet transform" (if each recursive stage considers multiple values for ω, with one of the values being zero — see Figure 2). For multidimensional signals, multiwavelets detect local directionality beyond what wavelets provide. If we recursively process the higher-frequency channels, too, then we obtain a "nonlinear wavelet packet transform" or a "nonlinear multiwavelet packet transform" — a kind of nonlinear iterated filter bank — see Figure 3. Linearly recombining the different frequency channels can realize local rotation-invariance and other potentially desirable properties. The transforms just discussed are undecimated, but interleaving appropriate decimation or subsampling applied to the sequences yields the usual decimated transforms.

Remark 2. In practice, appropriate decimation or subsampling is important to avoid overfitting in the data-driven approach discussed below, by limiting the number of degrees of freedom appropriately. Even when the signal is not a strictly stationary stochastic process, the averaging in (7) — the leftmost summation — performs Coifman-Donoho "cycle spinning" to avoid artifacts that would otherwise arise due to windows' partitioning after subsampling. The averaging reduces the variance; wider local averaging would reduce the variance still further.

Remark 3. Sequences that are finite rather than doubly infinite provide only enough information for estimating a smoothed version of the spectrum. Alternatively, a finite amount of data provides information for estimating multiscale windowed spectra yielding time-frequency (or space-Fourier) resolution similar to the multiresolution analysis of wavelets.

Remark 4. SIFT, HOG, SURF, etc. of [10], [11], [6], [1], et al. are more analogous to multiwavelets than to multiwavelet packets more generally.

The "multiwavelet transform" constitutes a desirable baseline model. We can easily adapt to the data the choices of windows and indeed the whole recursive structure of the processing (whether restricting the recursion to the zero-frequency channels, or also allowing the recursive processing of higher-frequency channels). Viewing the convolutional filters that serve as windowed exponentials as parameters, the desirable baseline is just one member of a parametric family of models. This parametric family is known as a "complex-valued convolutional network." We can fit (i.e., learn or train) the parameters to the data via optimization procedures (such as stochastic gradient descent) in conjunction with "backpropagation" (backpropagation is the chain rule of Calculus applied to calculate gradients of our recursively composed operations). For "supervised learning," we optimize according to a specified objective, usually using the multiscale spectra as inputs to a scheme for classification or regression, as detailed by [9], for example.

Finally, while the above analysis concerns X_k, where k ranges over the integers, extending the above to analyze $X_{j,k}$, where j and k range over the integers, is straightforward — the latter could be a "locally homogeneous random field." Moreover, the infinite range of the integers is far from essential; implementations on computers obviously use only finite sequences. Furthermore, the above construction is appropriate for processing any locally stationary stochastic process, not just filtered white noise. For instance, the construction can enable a multiresolution analysis of "regularity" (or "smoothness") that easily distinguishes between low-pass filtered i.i.d. Gaussian noise and a sinusoid with a random phase offset (for example, $X_k = \sin(\pi(k + J)/1000)$ for any integer k, where J is an integer drawn uniformly at random from 1, 2, ..., 2000). More generally, the above construction should enable discriminating between many interesting classes of stochastic processes, commensurate with the ability of multiwavelet-based multiresolution analysis to measure "regularity," "intermittency," distributional characteristics (say, Gaussian versus Poisson), etc. In any case, every model in the parametric family constituting the complex-valued convnet calculates relevant features, windowed spectra of the form in (6) and (7). The absolute values in (6) and (7) are the key nonlinearity and are a reflection of the local stationarity — the local translation-invariance — of the process and its relevant features.

Acknowledgements

We would like to thank Keith Adams, Lubomir Bourdev, Rob Fergus, Armand Joulin, Manohar Paluri, Marc'Aurelio Ranzato, and Rachel Ward.

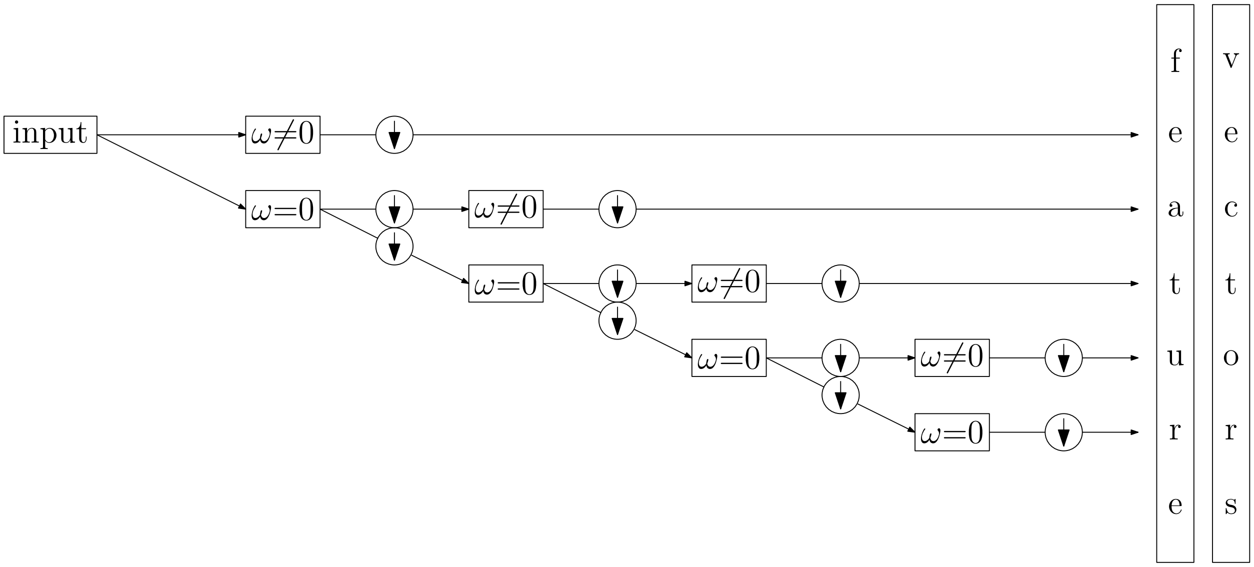

Figure 1: A flow chart for the "wavelet transform" of an input vector: each box "ω=0" corresponds to (7) with ω=0 or (9); each box "ω≠0" corresponds to (7) — convolution followed by taking the absolute value of every entry followed by local averaging; each circle "↓" corresponds to subsampling (say, retaining only every other entry)

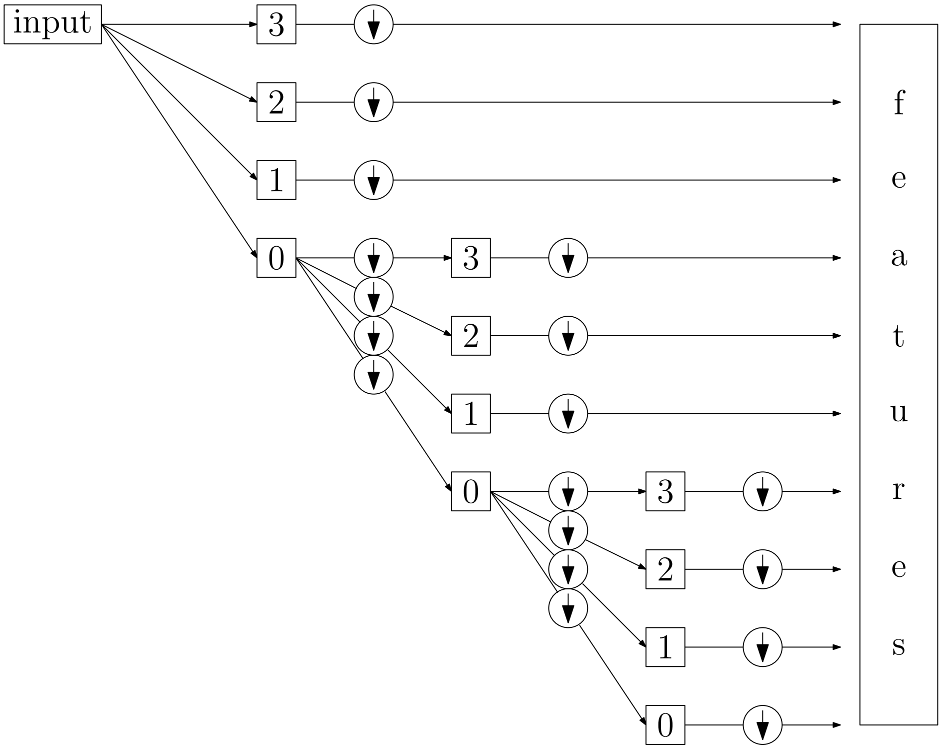

Figure 2: A flow chart for the "multiwavelet transform" of an input vector: each box "0" corresponds to (7) with ω=0 or (9); each box "1," "2," or "3" corresponds to (7) for different convolutional filters, but always with convolution followed by taking the absolute value of every entry followed by local averaging; each circle "↓" corresponds to subsampling (say, retaining only every fourth entry)

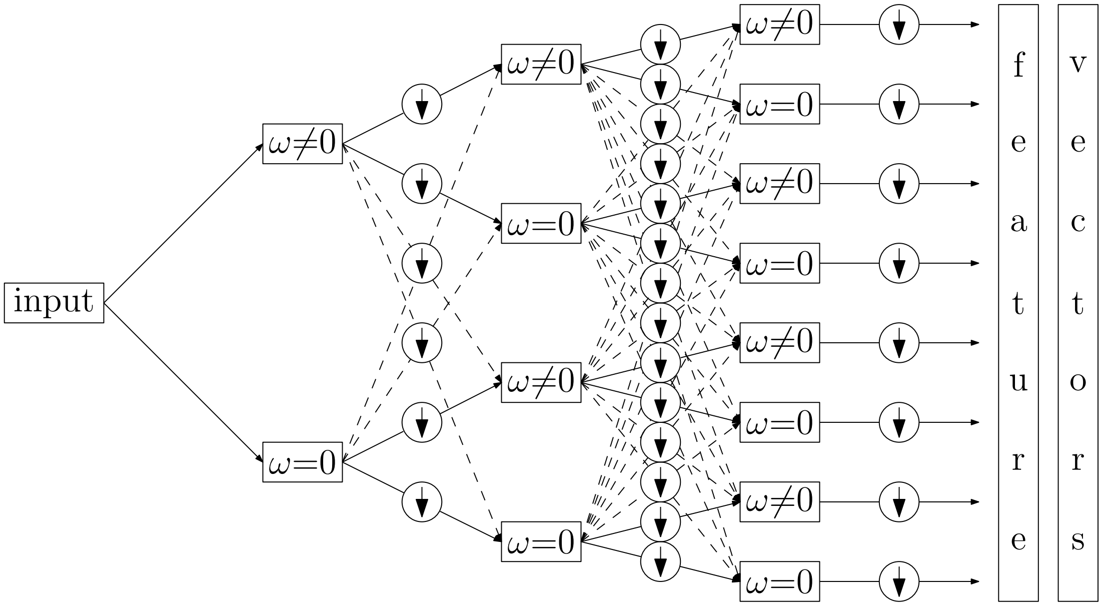

Figure 3: A flow chart for the "nonlinear wavelet packet transform" of an input vector: each box "ω=0" corresponds to (7) with ω=0 or (9); each box "$\omega\neq$0" corresponds to (7) — convolution followed by taking the absolute value of every entry followed by local averaging; each circle "↓" corresponds to subsampling (say, retaining only every other entry); the dashed arrows can involve downweighting the associated summands (and the convolutional filter can be different for every arrow); Figure 1 is essentially a special case of the present figure for which some of the convolutional filters simply deconvolve the preceding local averaging (omitting some of the subsampling)

References

[1] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, *Speeded-up robust features (SURF)*, Computer Vision Image Understanding, 110 (2008), pp. 346–359.

[2] J. Bruna and S. Mallat, *Invariant scattering convolutional networks*, IEEE Trans. Pattern Analysis Machine Intel., 35 (2013), pp. 1872–1886.

[3] J. Bruna, S. Mallat, E. Bacry, and J.-F. Muzy, *Intermittent process analysis with scattering moments*, Ann. Statist., 43 (2015), pp. 323–351.

[4] R. R. Coifman and D. Donoho, *Translation-invariant denoising*, in Wavelets and Statistics, A. Antoniadis and G. Oppenheim, eds., vol. 103 of Lecture Notes in Statistics, Springer, 1995, pp. 125–150.

[5] R. R. Coifman, Y. Meyer, S. Quake, and M. V. Wickerhauser, *Signal processing and compression with wavelet packets*, in Wavelets and Their Applications, J. S. Byrnes, J. L. Byrnes, K. A. Hargreaves, and K. Berry, eds., vol. 442 of NATO ASI Series C: Mathematical and Physical Sciences, Springer, 1994, pp. 363–379.

[6] N. Dalal and B. Triggs, *Histograms of oriented gradients for human detection*, in IEEE Computer Society Conf. Computer Vision and Pattern Recognition 2005, vol. 1, IEEE, June 2005, pp. 886–893.

[7] I. Daubechies, *Ten Lectures on Wavelets*, CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, 1992.

[8] R. Haensch and O. Hellwich, *Complex-valued convolutional neural networks for object detection in PolSAR data*, in 8th European Conf. EUSAR, IEEE, June 2010, pp. 1–4.

[9] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, *Gradient-based learning applied to document recognition*, Proc. IEEE, 86 (1998), pp. 2278–2324.

[10] D. G. Lowe, *Object recognition from local scale-invariant features*, in Proc. 7th IEEE Internat. Conf. Computer Vision, vol. 2, IEEE, September 1999, pp. 1150–1157.

[11] ——, *Distinctive image features from scale-invariant keypoints*, Internat. J. Computer Vision, 60 (2004), pp. 91–110.

[12] S. Mallat, *A Wavelet Tour of Signal Processing: the Sparse Way*, Academic Press, 3rd ed., 2008.

[13] ——, *Recursive interferometric representations*, in Proc. EUSIPCO Conf. 2010, EURASIP, August 2010, pp. 716–720.

[14] P. Mehta and D. J. Schwab, *An exact mapping between the variational renormalization group and deep learning*, Tech. Rep. 1410.3831, arXiv, October 2014.

[15] Y. Meyer, *Wavelets and Operators*, vol. 37 of Cambridge Studies in Advanced Mathematics, Cambridge University Press, 1993.

[16] Y. Meyer and R. R. Coifman, *Wavelets: Calderón-Zygmund and Multilinear Operators*, vol. 48 of Cambridge Studies in Advanced Mathematics, Cambridge University Press, 1997.

[17] T. Poggio, J. Mutch, J. Leibo, L. Rosasco, and A. Tacchetti, *The computational magic of the ventral stream: sketch of a theory (and why some deep architectures work)*, Tech. Rep. MIT-CSAIL-TR-2012-035, MIT CSAIL, Cambridge, MA, December 2012.

[18] L. R. Rabiner and R. W. Schafer, *Introduction to Digital Speech Processing*, vol. 1 of Foundations and Trends in Signal Processing, NOW, 2007.

[19] E. P. Simoncelli and W. T. Freeman, *The steerable pyramid: a flexible architecture for multi-scale derivative computation*, in Proc. Internat. Conf. Image Processing 1995, vol. 3, IEEE, October 1995, pp. 444–447.

[20] A. Srivastava, A. B. Lee, E. P. Simoncelli, and S. Zhu, *On advances in statistical modeling of natural images*, J. Math. Imaging Vision, 18 (2003), pp. 17–33.