Perceptual significance of kernel methods for natural image processing

Vikas Ramachandra and Truong Q. Nguyen

Abstract: We explore the unifying connection between kernel regression, Volterra series expansion and multiscale signal decomposition using recent results on function estimation for system identification. We show that using any of these techniques for (non-linear) image processing tasks is (approximately) equivalent. Further, we use the relation between wavelets and independent components of natural images. Kernel methods can be shown to be implicit Volterra series expansions, which are well approximated by wavelets. Wavelets are, in turn, well represented by independent components of natural images. Thus, it can be seen that kernel methods are also near optimal in terms of higher order statistical modeling and approximation of (natural) images. This explains the reason for good results often (perceptually) observed with the use of kernel methods for many image processing problems.

(1) Introduction: It is known that natural image statistics go well beyond second order moments. It was shown by [1] that the independent components of natural images are oriented edges at different scales, which are well captured by multiresolution decompositions, as shown by [2]. Recently, there has been a fair amount of work on the use of multiresolution decomposition of images and modifications at subbands for various problems including superresolution [3], denoising [1], deconvolution [2], image quality assessment [3] etc. There has also been a parallel stream of works which use kernel methods like kernel regression for image processing tasks including superresolution [4], image denoising [5], deconvolution [6], image edge enhancement [7] and so on. Most of these tasks can be cast as estimation problems. It is well known that the choice of kernel helps to regularize the solution of the estimation problem. Although, these techniques give good results, it is not immediately clear (for the image processing tasks mentioned) how much the regularization provided by the kernel is useful. Although [8] shows that there is considerable improvement in image estimation problems when higher order moments are considered, it is not directly motivated by natural image statistics, since the emphasis in [8] and [9] is more on using kernel methods for faster implicit higher order expansions of signals.

We proceed by reviewing the relation between kernel methods and Volterra expansions [9]. Volterra expansions are then shown to be approximated well by wavelet bases [10]. Thus, kernel methods can be well approximated by wavelet expansions. Further, it has been observed that the independent components of natural images are wavelet-like in nature [2],[11]. We briefly talk about extensions using sparse overcomplete basis representations and their relation to nonlinear ICA. Since kernel methods are well approximated by wavelets which are also related directly to natural image components, we argue that this is why kernel methods perform well perceptually.

(2) Volterra series and kernel regression: There is an (implicit) Volterra series for polynomial kernel regression. Let the input $x(t)$ and output $y(t)$, both functions of time, be related by a mapping $y(t) = H * x(t)$. Traditionally, $y(t)$ and $x(t)$ are related by a convolution as, $y(t) = H_1 x(t) = \int h^{(1)}(\tau)x(t-\tau)d\tau$ where H_1 gives the transfer function of the linear impulse response. For nonlinear systems, we can extend the linear response to form this Volterra series operator $y(t) = Vx(t) = H_1x(t) + H_2x(t) + \cdots$ where $H_nx(t) = \int h^{(n)}(\tau_1, \cdots, \tau_n)x(t-\tau_1)\cdots x(t-\tau_n)d\tau_1 \cdots d\tau_n$ is the nth order Volterra operator, and where $h^{(n)}(\tau_1, \tau_2, \cdots, \tau_n)$ are the Volterra kernels, and where $H_0x(t) = constant$. The Volterra series is equivalent to a Taylor series with memory, relating the output to present and past inputs. This also captures the multiplicative interaction between input terms. [15] has details of Volterra series convergence. A related class of mutually uncorrelated operators which are easier to estimate are the Weiner series operators. The Wiener operators G_n are linear combinations of Volterra operators up to order n. They can be obtained from the original Volterra series by Gram-Schmidt like orthogonalization.The discrete version of the Volterra operator is the function: $H_n(x) = \sum_{i_1=1}^{m} \cdots \sum_{i_n=1}^{m} h^{(n)}_{i_1 \cdots i_n} x_{i_1} \cdots x_{i_n}$ If the input data is m dimensional, the Volterra kernels are given by $h^{(n)}_{i_1 \cdots i_n}$. Discrete systems with compact domain can be approximated by finite, discrete Volterra series [9].

Volterra Series as a linear operator in RKHS: Material in this section is condensed from [9]. Given observations $(x_1, y_1), \cdots, (x_N, y_N)$, linear regression tries to estimate y as a function of x,$y = f(x) = \sum_{j=1}^{M} \gamma_j \varphi_j(x)$ using $\gamma_j \in R$ and a dictionary of M functions $\varphi_j : R^m \to R$. In the case of pth-order Volterra or Wiener series, this dictionary consists of all monomials of x up to order p. Instead of directly using the monomials as basis functions, it is possible to specify the dictionary in terms of a kernel function k via $\varphi_j(x) = k(x, z_j)$, using a set of points $z_1, \cdots, z_M$ from R^m. We consider positive definite kernels, i.e. functions k with the property that the Gram matrix $K_{ij} = k(x_i, x_j)$ is positive definite for all choices of the $x_1, \cdots, x_N$ from the input domain. Such kernels can be seen as a dot product in a linear space F, i.e., there exists a map ϕ such that $k(x, x') = \phi(x)^\top \phi(x')$. F

can be identified with a space of functions, $f(x) = \sum_{j=1}^{M} \gamma_j k(x, z_j)$ This space is a reproducing kernel Hilbert space (RKHS). Using the representer theorem, we can show that the optimal solution $f(x)$ can be expressed in terms of the training points x_js only, i.e. $f(x) = \sum_{j=1}^{M} \gamma_j k(x, x_j)$, where $\gamma_j \in R$. The optimal γ_js in the mean squared sense can be shown to be $\gamma = K^{-1}y$, where the K is the Gram matrix of the training input points x_j, and y is a stack of the training output points y_j. For a test point x, $y = f(x) = \gamma^\top k(x) = y^\top K^{-1} k(x)$ where $k(x) = [k(x, x_1), k(x, x_2), \cdots, k(x, x_N)]$. The Volterra operators can be expressed using the kernel framework. The nth-order Volterra operator is a weighted sum of all nth-order monomials of the input vector x. For $n = 0, 1, 2, \cdots$, we define the map ϕ_n as $\phi_n(x) = [\mathsf{x}_1^n, \mathsf{x}_1^{n-1}\mathsf{x}_2, \cdots, \mathsf{x}_1\mathsf{x}_2^{n-1}, \mathsf{x}_2^n, \cdots, \mathsf{x}_m^n]$and $\phi_0(x) = 1$, such that ϕ_n maps the input $x \in R^m$ into a vector $\phi_n(x) \in F_n = R^{m^n}$ containing all m^n ordered monomials of degree n evaluated at x. Using ϕ_n, we write the nth-order Volterra operator as a scalar product in F_n, $H_n(x) = \eta_n^\top \phi_n(x)$ with the coefficients stacked into the vector $\eta_n = [h_{1,1,\cdots,1}^n, h_{1,2,\cdots,1}^n, \cdots,]^\top \in F_n$. The functions ϕ_n form an RKHS characterized by the polynomial kernels, $\phi_n(x_1)^\top \phi_n(x_2) = (x_1 x_2)^n = k_n(x_1, x_2)$ The estimation problem can be solved directly if one applies the same idea to the entire pth-order Volterra series. The entire pth-order Volterra series is also a scalar product in F^p:$\sum_{p}^{n=0} H_n(x) = (\eta^p)^\top \phi^{(p)}(x)$ Also, the inner products are: $\phi^p(x_1)^\top \phi^p(x_1) = \sum_{p}^{n=0} \alpha_n^2 (x_1^\top x_2)^n = k^p(x_1, x_2)$. The output via a Volterra series for a given test point is: $y = f(x) = \sum_{p}^{n=0} H_n(x) = y^\top K_p^{-1} k^p(x)$ Thus, there is an equivalence between polynomial kernel regression and Volterra system estimation.

(3) Wavelet expansion of Volterra operators: The discussion in this section is condensed from [10]. A multiwavelet basis for $L^2(R)$, the vector space of square-integrable functions, is composed of the scaled translates and dilates of multiple wavelet functions $[\psi^1, \cdots, \psi^r]$. These multiwavelets are generated from r scaling functions $[\rho^1, \cdots, \rho^r]$. Because multiwavelets employ multiple scaling functions and wavelets, there is more freedom to design these functions to satisfy a greater range of properties, which enables better representation of many more system responses. By construction, the multiwavelets $[\psi^1, \cdots, \psi^r]$ satisfy the equations: $\rho^s(x) = \sqrt{2} \sum_{p,t} a_p^{s,t} \rho^t(2x - p), s = 1, \cdots, r; \psi^s(x) = \sqrt{2} \sum_{p,t} b_p^{s,t} \rho^t(2x - p), s = 1, \cdots, r$. Multiwavelets and scaling functions are formed recursively as linear combinations of scaling functions with half the support. The scaled translates and dilates of the scaling functions and multiwavelets are: $\rho_{j,k}^s(x) = 2^{j/2} \rho^s(2^j x - k), \psi_{j,k}^s(x) = 2^{j/2} \psi^s(2^j x - k)$ $j \in Z$ gives the resolution level and $k \in Z$ is the translation index. A given approximation space V_j can be decomposed as: $V_j = V_{j-1} \bigoplus W_{j-1}$, where W_{j-1} is the corresponding wavelet space. Recursively applying this relation, we get, $V_j = V_{j_0} \bigoplus W_{j_0} \bigoplus W_{j_0+1} \bigoplus \cdots \bigoplus W_{j-2} \bigoplus W_{j-1}$. In a multiwavelet multiresolution analysis, a function $f \in$

$L^2(R)$ is first approximated in terms of a fine-resolution space V_j as: $f_j(x) = \sum_r^{s=1} \sum_{k \in Z} a_{j,k}^s \rho_{j,k}^s(x)$ where $a_{j,k}^s$ are scaling function expansion coefficients and f_j denotes an approximation of function f at resolution level j. This can be written equivalently as $f_j(x) = \sum_r^{s=1} \sum_{k \in Z} a_{j,k}^s \rho_{j0,k}^s(x) + \sum_r^{s=1} \sum_{l=j_0}^{j-1} \sum_{k \in Z} b_{l,k}^s \psi_{l,k}^s(x)$ This representation is in terms of the scaling functions on the coarsest resolution level j_0 and the multiwavelets on levels j_0 through $j-1$. The multiscale expansion is often a very sparse, i.e. a function can often be accurately represented using a few multiwavelet coefficients. Higher-dimensional multiwavelets are obtained as the tensor products of the one-dimensional scaling functions and multiwavelets. The family of piecewise-polynomial multiwavelets is particularly well suited for the representation of Volterra kernels because the functions can easily be adapted to the domains of support of the kernels.

Multiwavelet representation of kernels and operators: Here, we explain how one would represent first order Volterra operators using multiwavelets. Extension of this to higher order operators, although straightforward, requires cumbersome notation and we avoid it here. The reader can find the derivations for higher order operators in [10].

First order operator approximation: We recall the first order operator is convolution. We start by approximating the input $x(t)$ with a zero order hold. If the input and output are sampled at a rate of 2^j Hz and there are a total of N data points, a zero-order hold approximation of the input can be written as: $x_j(t) = \sum_{k=1}^{N-1} x_{j,k} \chi_{j,k}(t)$ where the characteristic function $\chi_{j,k}$ is defined as: $\chi_{j,k}(t) = 2^{j/2} \chi(2^j t - k)$ and χ is the characteristic function of the interval $[0, 1]$. The zero-order hold is simply a piecewise constant approximation of the input. The coefficients $u_{j,k}$ are equivalent to scaled samples of the input. Discretizing the output at a sampling rate of 2^j Hz, equation (2) yields N equations for the discrete first-order outputs: $y_{1,j}(t_n) = \int_0^{t_n} h_1(\tau) x(t - \tau) d\tau, n = 1, \cdots, N$ where $t_n = 2^{-j} n$. The translated zero-order hold approximation of the input can be written as: $x_j(t_n - \tau) = \sum_{k=0}^{n-1} x_{j,n-k-1} \tau_{j,k}$ Next, the first-order kernel h_1 is represented in terms of the piecewise-polynomial multiwavelets. The first-order kernel can be approximated in terms of the (boundary-adapted) scaling functions on level j1 as: $h_{1,j_1}(\tau) = \sum_p \sum_r^{s=1} a_{j1,p}^s \rho_{j1,p}^s(\tau)$ Substituting equations (18) and (19) into equation (17), the expression for the output becomes, $y_{1,j}(t_n) = \sum_{k=0}^{n_1-1} \sum_p \sum_{s=1}^r x_{j,n-k-1} a_{j_1,p} \int_0^{T_1} \rho_{j1,p}^s(\tau) \chi_{j,k}(\tau) d\tau$, where$n_1 = nforn < 2^j T_1 and 2^j T_1 forn \geq 2^j T_1$ This can be written as: $\underline{y}_{1,j} = M_1 \underline{a}_1$ where $y_{1,j}$ is a vector of discrete first-order outputs and a_1 is a vector of scaling function coefficients that represent the first-order kernel. Similarly, for second and higher order operators, the kernel is expressed as a linear combination of a tensor product of scaling functions. To summarize, we see that a multiresolution approximation is possible for a Volterra series expansion. The output is expressed as a linear combination of the multiscale basis functions, the weights of which are dependent on the interaction between the input signal values.

(4) Multiresolution analysis and image statistics: Marginal distributions of oriented bandpass filter responses of natural images are highly kurtotic with sharp peaks at zero and much longer tails than Gaussian density, and have a number of important implications to sensory neural coding of natural visual scene [1]. A multiscale decomposition can capture many higher order image statistics as was shown in [2], where it was discussed that wavelet-like filters emerge on applying ICA (Independent component analysis) on natural images, which captures the higher order interactions between image pixels. Most basis vectors are localized in both space and frequency. It is clear that the basis functions representing lower frequencies are spatially more spread, but are also more localized in the frequency domain than those representing high frequencies. This is essentially the main property of wavelets.

Over-complete bases and nonlinear ICA: By selecting multiresolution bases from an overcomplete dictionary, we induce sparsity on the coefficients, (basis pursuit) which leads to a non-linear relation between the chosen bases and the input. In [14], learning sparse overcomplete representations was shown to be a nonlinear extension of ICA. This can explain the nonlinear response of the human visual cortical cells [13]. Thus, sparse extensions of ICA and overcomplete wavelet/ basis expansions are closely interrelated.

(5) Conclusion: In this paper, we discussed the equivalence of kernel regression and multiresolution analysis for image processing tasks which rely on natural image statistics. The relation was drawn by relating polynomial kernel methods to (implicit) Volterra series expansion, finding multiwavelet approximations for Volterra operators and using the relation between multiresolution decomposition and the independent components of natural images.

REFERENCES

[1] A.J. Bell and T.J. Sejnowski, "The "independent components" of natural scenes are edge filters," *Vision Research,* Dec. 1997.

[2] J. Hurri et. al, "Wavelets and natural image statistics," *Proc. Scandinavian Conf. on Image Analysis,* 1997.

[3] V. Ramachandra and T. Nguyen, "Superresolution with side information," Technical report, Videprocessing Lab, UCSD.

[1] J. Portilla et al., "Image Denoising using Gaussian Scale Mixtures in the Wavelet Domain," CS Technical Report TR2002-831, NYU.

[2] J. Fan and J. Koo, "Wavelet deconvolution," *IEEE Info. Theory,*2002.

[3] Z. Wang et.al, "Quality-aware images," *IEEE Image Proc,* 2006.

[4] K.S. Ni et. al, "Single Image Superresolution Based on Support Vector Regression," *IEEE ICASSP,* 2006.

[5] B. Scholkopf et al., "Input Space Versus Feature Space in Kernel-Based Methods," *IEEE Trans. on Neural Networks,* Sep 1999.

[6] D. Li et. al, "Blind Image Deconvolution Using Support Vector Regression," *IEEE ICASSP,* Mar 2005.

[7] M. Nitzberg et.al, "Nonlinear image filtering with edge and corner enhancement," *IEEE Trans. on Pattern Anal. Machine Intell.,* 1992.

[8] M. Franz and B. Scholkopf, "Implicit Wiener series for higher-order image analysis," *NIPS,* 2004.

[9] M. Franz and B. Scholkopf, "A unifying view of wiener & volterra theory & polynomial kernel regression," *Neural Computation.* 2006.

[10] R. Prazenica1 and A. Kurdila, ”Multiwavelet Constructions and Volterra Kernel Identification,” *Nonlinear Dynamics,* 2006.

[11] J. Portilla and E.P. Simoncelli, ”A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients.” *IJCV,* 2000.

[12] S. Chen et. al, ”Atomic decomposition by basis pursuit,” *SIAM Journal on Scientific Computing,* 1998.

[13] B. Olshausen and D. Field, ”Sparse coding with an overcomplete basis set: A strategy employed by V1?,” *Vision Research,* Vol. 37.

[14] M.S. Lewicki and T.J. Sejnowski, ”Learning Overcomplete Representations.” *Neural Computation,* 2000.

[15] S. Boyd et. al, ”Analytical Foundations of Volterra Series,” *IMA Journal of Maths. Control and Information,* Vol. 1, No. 3.

Authors’ affiliations: Vikas Ramachandra* and Truong Nguyen. The authors are with the University of California, San Diego, 9500 Gilman Drive, San Diego CA 92093. Email: vikas@ucsd.edu. Phone: 858 534 5669. * Corresponding author.