Compression of Higher Dimensional Functions Containing Smooth Discontinuities

Venkat Chandrasekaran, Michael B. Wakin, Dror Baron, Richard G. Baraniuk
Department of Electrical and Computer Engineering, Rice University

***Abstract*— Discontinuities in data often represent the key information of interest. Efficient representations for such discontinuities are important for many signal processing applications, including compression, but standard Fourier and wavelet representations fail to efficiently capture the structure of the discontinuities. These issues have been most notable in image processing, where progress has been made on modeling and representing one-dimensional edge discontinuities along C^2 curves. Little work, however, has been done on efficient representations for higher dimensional functions or on handling higher orders of smoothness in discontinuities. In this paper, we consider the class of N-dimensional Horizon functions containing a C^K smooth singularity in $N-1$ dimensions, which serves as a manifold boundary between two constant regions; we first derive the optimal rate-distortion function for this class. We then introduce the *surflet* representation for approximation and compression of Horizon-class functions. Surflets enable a multiscale, piecewise polynomial approximation of the discontinuity. We propose a compression algorithm using surflets that achieves the optimal asymptotic rate-distortion performance for this function class. Equally important, the algorithm can be implemented using knowledge of only the N-dimensional function, without explicitly estimating the $(N-1)$-dimensional discontinuity.**

I. Introduction

A. Motivation

Discontinuities are prevalent in real-world data. Discontinuities often represent a boundary separating two regions and thus provide vital information. Edges in images illustrate this well; they usually separate two smooth regions and thus convey fundamental information about the underlying geometrical structure of the image. Therefore, representing discontinuities sparsely is an important goal for approximation and compression algorithms.

This work was supported by NSF grant CCR-9973188, ONR grant N00014-02-1-0353, AFOSR grant F49620-01-1-0378, and the Texas Instruments Leadership University Program.

Email: {venkatc, wakin, drorb, richb}@rice.edu. Web: dsp.rice.edu

Most discontinuities occur at a lower dimension than that of the data and moreover are themselves continuous. For instance, in images, the data is two-dimensional, while the edges essentially lie along one-dimensional curves. Wavelets model smooth regions in images well, but fail to represent edges sparsely and capture the coherent nature of these edges. Romberg *et al.* [1] have used wedgelets [2] to represent edges effectively and have suggested a framework using wedgelets to jointly encode all the wavelet coefficients corresponding to a discontinuity. Candès and Donoho [3] have proposed *curvelets* as an alternative sparse representation for discontinuities. However, a major disadvantage with these methods is that they are intended for discontinuities that belong to C^2 (the space of smooth functions having two continuous derivatives), and hence do not take advantage of higher degrees of smoothness of the discontinuities. Additionally, most of the analysis for these methods has not been extended beyond two dimensions.

Indeed, little work has been done on efficient representations for higher dimensional functions with discontinuities along smooth manifolds. There are a variety of situations for which such representations would be useful. Consider, for example, sparse video representation. Simple real-life motion of an object captured on video can be modeled as a discontinuity separating smooth regions in $N = 3$ dimensional space-time, with the discontinuity varying (moving) with time. Other examples include three-dimensional computer graphics ($N = 3$) and three-dimensional video ($N = 4$).

B. Contributions

In this paper, we consider the problem of representing and compressing elements of the function class $\mathcal{F}$, the space of N-dimensional *Horizon* functions [2] containing a C^K smooth $(N-1)$-dimensional singularity that separates two constant regions (see Fig. 1 for examples in 2-D and 3-D). Using the results of Kolmogorov [4] and Clements [5], we prove that the rate-distortion function

$D(R) \lesssim \left(\frac{1}{R}\right)^{\frac{K}{N-1}}$ is an optimal bound for this class.[1] Unfortunately, these papers do not suggest any constructive coding scheme. Cohen *et al.* [6] describe a coding scheme that, given explicit knowledge of the $(N-1)$-dimensional discontinuity, can be used to achieve the above rate-distortion performance; in practice, however, such explicit knowledge is unavailable.

This paper introduces a new representation for functions in the class $\mathcal{F}$. We represent Horizon-class functions using a collection of elements drawn from a dictionary of piecewise smooth polynomials at various scales. Each of these polynomials is called a *surflet*; the term "surflet" is derived from "surface"-let, because each of these polynomials approximates the discontinuity surface over a small region of the Horizon-class function.

In addition, we propose a tree-structured compression algorithm for surflets and establish that this algorithm achieves the optimal rate-distortion performance $D(R) \lesssim \left(\frac{1}{R}\right)^{\frac{K}{N-1}}$ for the class $\mathcal{F}$. Our method incorporates the following major features:

- Our algorithm operates directly on the N-dimensional function, without explicit knowledge of the $(N-1)$-dimensional discontinuity.
- We quantize and encode higher-order polynomial coefficients with lesser precision, without a substantial increase in distortion.
- Combining the notion of multiresolution with predictive coding provides significant gains in terms of rate-distortion performance.

By reducing the number of allowable polynomial elements, our quantization scheme leads us to an interesting insight. Conventional wisdom in the wavelets community maintains that higher-order polynomials are not practical for representing boundaries that are smoother than C^2, due to an assumed exponential explosion in the number of parameters and thus the size of the representation dictionary. A fascinating aspect of our solution is that the quantization scheme reduces the size of the surflet dictionary tremendously, making the approximation of smooth boundaries tractable.

In Sec. II, we introduce the problem, define our function class, and state the specific goal of our compression algorithm. We introduce surflets in Sec. III. In Sec. IV, we describe our compression algorithm in detail. Sec. V summarizes our contributions and insights. Finally, we direct the interested reader to a supplemental paper [7] that contains additional details and proofs for all theorems (omitted here for brevity).

[1] We focus here on *asymptotic* performance. We use the notation $f(R) \lesssim g(R)$ if there exists a constant C, possibly large but not dependent on R, such that $f(R) \leq Cg(R)$.

II. Definitions and Problem Setup

In this paper, we consider functions of N variables that contain a smooth discontinuity that is a function of $N-1$ variables. We denote vectors using boldface characters. Let $\mathbf{x} \in [0,1]^N$, and let x_i denote its i'th element. We denote the first $N-1$ elements of $\mathbf{x}$ by $\mathbf{y}$, i.e., $\mathbf{y} = [x_1, x_2, \cdots, x_{N-1}] \in [0,1]^{N-1}$.

A. Smoothness model for discontinuities

We first define the notion of smoothness for modeling discontinuities. A function of $N-1$ variables has smoothness of order $K > 0$, where $K = r + \alpha$, r is an integer, and $0 < \alpha \leq 1$, if the following criteria are met [4, 5]:

- all iterated partial derivatives with respect to the $N-1$ directions up to order r exist and are continuous;
- all such partial derivatives of order r satisfy a Lipschitz condition of order α (also known as a Hölder condition).[2]

We denote the space of such functions by $\mathcal{C}^K$. Observe that when K is an integer, $\mathcal{C}^K$ includes as a subset the traditional space C^K (where the function has $K = r+1$ continuous partial derivatives).

B. Multidimensional Horizon-class functions

Let b be a function of $N-1$ variables such that

$$b : [0,1]^{N-1} \to [0,1].$$

We define the function f of N variables such that

$$f : [0,1]^N \to \{0,1\}$$

according to the following:

$$f(\mathbf{x}) = \begin{cases} 1, & x_N \geq b(\mathbf{y}) \\ 0, & x_N < b(\mathbf{y}). \end{cases}$$

The function f is known as a Horizon-class function [2], where the function b defines a manifold horizon boundary between values 0 and 1.

In this paper, we consider the case where the horizon b belongs to $\mathcal{C}^K$, and we let $\mathcal{F}$ denote the class of all Horizon-class functions f containing such a discontinuity. As shown in Fig. 1, when $N = 2$ such a function can be interpreted as an image containing a smooth discontinuity that separates a 0-valued region below from

[2] A function $g \in \text{Lip}(\alpha)$ if $|g(\mathbf{y}+\mathbf{h}) - g(\mathbf{y})| \leq C|\mathbf{h}|^{\alpha}$ for all $\mathbf{y}, \mathbf{h}$.

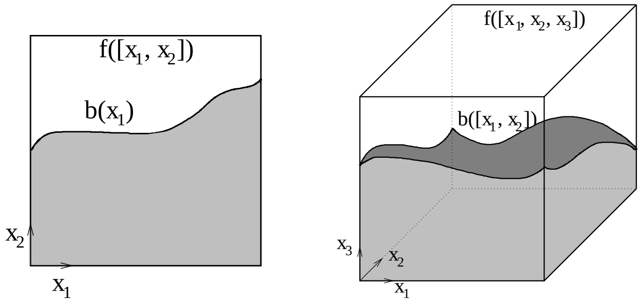

Fig. 1. *Example Horizon-class functions for $N = 2$ and $N = 3$.*

a 1-valued region above. For $N = 3$, f represents a cube with a two-dimensional smooth surface cutting across the cube, dividing it into two regions — 0-valued below the surface and 1-valued above it.

C. Problem formulation

Our goal is to encode an arbitrary function f in the Horizon class $\mathcal{F}$. We use the squared-L_2 metric to measure distortion between f and $\widehat{f}_R$, the approximation provided by the compression algorithm using R bits

$$D_2(f, \widehat{f}_R) = \int_{\mathbf{x}\in[0,1]^N} (f - \widehat{f}_R)^2.$$

Our performance measure is the asymptotic rate-distortion behavior.

We emphasize that our algorithm approximates f in N dimensions. The approximation $\widehat{f}_R$, however, can be viewed as a type of Horizon-class signal — our algorithm implicitly provides a piecewise polynomial approximation $\widehat{b}_R$ to the smooth discontinuity b. That is,

$$\widehat{f}_R(\mathbf{x}) = \begin{cases} 1, & x_N \geq \widehat{b}_R(\mathbf{y}) \\ 0, & x_N < \widehat{b}_R(\mathbf{y}) \end{cases} \tag{1}$$

for some piecewise polynomial $\widehat{b}_R$. From the definition of N-dimensional Horizon-class functions, it follows that

$$\begin{aligned} D_2(f, \widehat{f}_R) &= \int_{\mathbf{x}\in[0,1]^N} (f - \widehat{f}_R)^2 \\ &= \int_{\mathbf{y}\in[0,1]^{N-1}} |b - \widehat{b}_R| \\ &= D_1(b, \widehat{b}_R). \end{aligned} \tag{2}$$

Hence, optimizing for squared-L_2 distortion between f and $\widehat{f}_R$ is equivalent to optimizing for L_1 distortion between b and $\widehat{b}_R$.

The work of Clements [5] (extending Kolmogorov and Tihomirov [4]) regarding metric entropy establishes that no coder for functions $b \in \mathcal{C}^K$ can outperform the rate-distortion function

$$D_1(b, \widehat{b}_R) \lesssim \left(\frac{1}{R}\right)^{\frac{K}{N-1}}. \tag{3}$$

We have extended this result to the N-dimensional class $\mathcal{F}$.

Theorem 1 [7]: The optimal asymptotic rate-distortion performance for the class $\mathcal{F}$ of Horizon signals is given by

$$D_2(f, \widehat{f}_R) \lesssim \left(\frac{1}{R}\right)^{\frac{K}{N-1}}. \tag{4}$$

D. Compression strategies

We assume that a coder is provided explicitly with the function f. As can be seen from the above formulation, all of the critical information about the function f is contained in the discontinuity b. One would expect any efficient coder to exploit such a fact. Methods through which this is achieved may vary.

One can imagine a coder that *explicitly* encodes b and then constructs a Horizon-class approximation $\widehat{f}$. Knowledge of b could be provided from an external "oracle" [8], or b could conceivably be estimated from the provided data f. Wavelets provide an efficient method for compressing the smooth function b. Cohen *et al.* [6] describe a tree-structured wavelet coder that can be used to compress b with optimal rate-distortion performance (3). From (2) and (4), it follows that this wavelet coder is optimal for coding instances of f. In practice, however, a coder is not provided with explicit information of b, and a method for estimating b from f may be difficult to implement. Estimates for b may also be quite sensitive to noise in the data.

In this paper, we propose a compression algorithm that operates directly on the N-dimensional data f. The algorithm assembles an approximation $\widehat{f}_R$ that is Horizon-class (that is, it can be assembled using an estimate $\widehat{b}_R$), but it does not require explicit knowledge of b. We prove that this algorithm achieves the optimal rate-distortion performance (4). Although we omit the discussion in this paper, our algorithm can also be easily extended to similar function spaces containing smooth discontinuities. Our spatially localized approach, for example, allows for changes in the variable along which the discontinuity varies (assumed throughout this paper to be x_N).

III. The Surflet Dictionary

In this section, we define a discrete dictionary of N-dimensional atoms, called *surflets*, that can be used to construct approximations to the Horizon-class function f. Each surflet consists of a dyadic hypercube containing a Horizon-class function, with a discontinuity defined by a smooth polynomial. Sec. IV describes compression using surflet approximations.

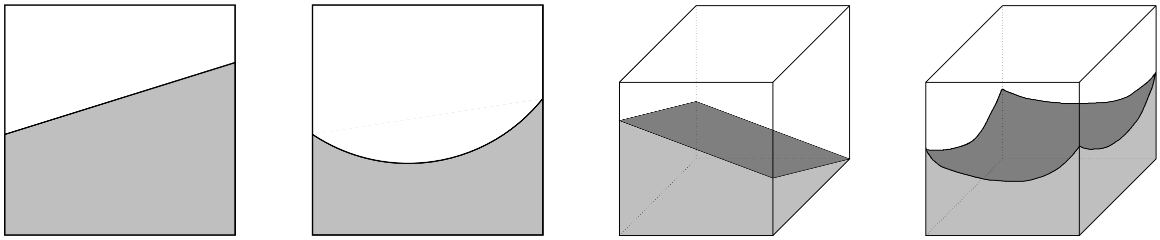

Fig. 2. *Example surflets, designed for (a)* $N=2$*,* $K\in(1,2]$*; (b)* $N=2$*,* $K\in(2,3]$*; (c)* $N=3$*,* $K\in(1,2]$*; (d)* $N=3$*,* $K\in(2,3]$.

A. Motivation — Taylor's theorem

The surflet atoms are motivated by the following property. If b is a function of $N-1$ variables in $\mathcal{C}^K$, then Taylor's theorem states that

$$\begin{aligned} b(\mathbf{y}+\mathbf{h}) &= b(\mathbf{y}) + \frac{1}{1!}\sum_{i_1=1}^{N-1} b_{y_{i_1}}(\mathbf{y})h_{i_1} \\ &+ \frac{1}{2!}\sum_{i_1,i_2=1}^{N-1} b_{y_{i_1}y_{i_2}}(\mathbf{y})h_{i_1}h_{i_2} + \cdots \\ &+ \frac{1}{r!}\sum_{i_1,\ldots,i_r=1}^{N-1} b_{y_{i_1}\cdots y_{i_r}}(\mathbf{y})h_{i_1}\cdots h_{i_r} \\ &+ O(\|\mathbf{h}\|^K), \end{aligned} \quad (5)$$

where $b_{y_1\cdots y_\ell}$ refers to the iterated partial derivatives of b with respect to $y_1,\ldots,y_\ell$ in that order. Note that there are $(N-1)^\ell$ ℓ'th order derivative terms.

Thus, over a small domain, the function b is well approximated using an r'th order polynomial (where the polynomial coefficients correspond to the partial derivatives of b evaluated at $\mathbf{y}$). Clearly, then, one method for approximating b on a larger domain would be to assemble a *piecewise polynomial* approximation, where each polynomial is derived from the local Taylor approximation of b. Consequently, these piecewise polynomials can be used to assemble a Horizon-class approximation of the function f. Surflets provide the N-dimensional framework for constructing such approximations and can be implemented without explicit knowledge of b or its derivatives.

B. Definition

A *dyadic hypercube* $X_j \subseteq [0,1]^N$ at scale $j\in\mathbb{N}$ is a domain that satisfies

$$X_j = [\beta_1 2^{-j},(\beta_1+1)2^{-j})\times\cdots\times[\beta_N 2^{-j},(\beta_N+1)2^{-j})$$

with $\beta_1,\beta_2,\ldots,\beta_N \in \{0,1,\ldots,2^j-1\}$. We explicitly denote the $(N-1)$-dimensional hypercube *subdomain* of X_j as

$$Y_j = [\beta_1 2^{-j},(\beta_1+1)2^{-j})\times\cdots\times[\beta_{N-1} 2^{-j},(\beta_{N-1}+1)2^{-j}). \quad (6)$$

The *surflet* $s(X_j;p;\cdot)$ is a Horizon-class function over the dyadic hypercube X_j defined through the polynomial p. For $\mathbf{x}\in X_j$ with corresponding $\mathbf{y} = [x_1,x_2,\cdots,x_{N-1}]$, we have

$$s(X_j;p;\mathbf{x}) = \begin{cases} 1, & x_N \ge p(\mathbf{y}) \\ 0, & \text{otherwise}, \end{cases}$$

where the polynomial $p(\mathbf{y})$ is defined as

$$\begin{aligned} p(\mathbf{y}) &= p_0 + \sum_{i_1=1}^{N-1} p_{1,i_1}\, y_{i_1} + \sum_{i_1,i_2=1}^{N-1} p_{2,i_1,i_2}\, y_{i_1}y_{i_2} + \cdots \\ &+ \sum_{i_1,\ldots,i_r=1}^{N-1} p_{r,i_1,i_2,\ldots,i_r}\, y_{i_1}y_{i_2}\cdots y_{i_r}. \end{aligned}$$

We call the polynomial coefficients $\{p_{\ell,i_1,\ldots,i_\ell}\}_{\ell=0}^r$ the *surflet coefficients*.[3] We note here that, in some cases, a surflet may be identically 0 or 1 over the entire domain X_j. Fig. 2 illustrates a collection of surflets with $N=2$ and $N=3$. We sometimes denote a generic surflet as $s(X_j)$, indicating only its region of support.

A surflet $s(X_j)$ approximates the function f over the dyadic hypercube X_j. One can cover the entire domain $[0,1]^N$ with a collection of dyadic hypercubes (possibly at different scales) and use surflets to approximate f over each of these smaller domains. For $N=3$, these surflets together look like piecewise smooth "surfaces" approximating the function f. Fig. 3 shows approximations for $N=2$ and $N=3$ obtained by combining localized surflets.

C. Discretization

We obtain a discrete surflet dictionary by quantizing the set of allowable surflet polynomial coefficients. For $\ell\in\{0,1,\ldots,r\}$, the surflet coefficient $p_{\ell,i_1,\ldots,i_\ell}$ at scale $j\in\mathbb{N}$ is restricted to values $\{n\cdot\Delta_{\ell,j}\}_{n\in\mathbb{Z}}$, where the stepsize satisfies

$$\Delta_{\ell,j} = 2^{-(K-\ell)j}. \quad (7)$$

[3] Because the ordering of terms $y_{i_1}y_{i_2}\cdots y_{i_\ell}$ in a monomial is not relevant, only $\binom{\ell+N-2}{\ell}$ monomial coefficients (not $(N-1)^\ell$) need to be encoded for order ℓ. We preserve the slightly redundant notation for ease of comparison with (5).

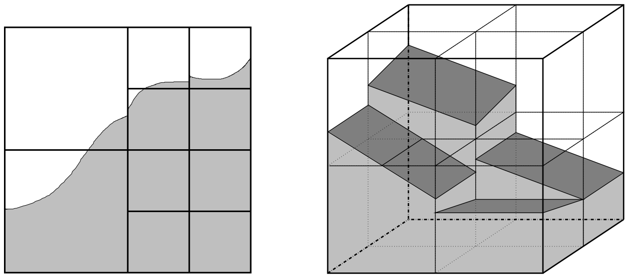

Fig. 3. *Example surflet tilings, (a) piecewise cubic with $N = 2$ and (b) piecewise linear with $N = 3$.*

The necessary range for n may depend on the function b. However, all derivatives are locally bounded, and so the relevant discrete surflet dictionary is actually finite for any realization of f.

These quantization stepsizes are carefully chosen to ensure the proper fidelity of surflet approximations without requiring excess bitrate. The key idea is that *higher-order terms can be quantized with lesser precision*, without increasing the residual error term in the Taylor approximation (5). In fact, Kolmogorov and Tihomirov [4] implicitly used this concept to establish the metric entropy for the class $\mathcal{C}^K$.

IV. Compression using Surflets

A. Overview

Using surflets, we propose a tree-based multiresolution approach to approximate and encode f. The approximation is arranged on a 2^N-tree, where each node in the tree at scale j represents a hypercube of sidelength 2^{-j}. Every node is either a leaf node (hypercube), or has 2^N children nodes (children hypercubes that perfectly tile the volume of the parent hypercube). Each node in the tree is labeled with a surflet. Leaf nodes provide the actual approximation to the function f, while interior nodes are useful for predicting and encoding their descendants. This framework allows for *adaptive* approximation of f — many small surflets can be used at fine scales for more complicated regions, while few large surflets will suffice to encode simple regions of f (such as those containing all 0 or 1).

Sec. IV-B discusses techniques for determining the proper surflet at each node. Sec. IV-C presents a method for pruning the tree depth according to the function f. Sec. IV-D describes the performance of a simple surflet encoder acting only on the leaf nodes. Sec. IV-E presents a more advanced surflet coder, using a top-down predictive technique to exploit the correlation among surflet coefficients.

B. Surflet Selection

Consider a node at scale j that corresponds to a dyadic hypercube X_j, and let Y_j be the $(N-1)$-dimensional subdomain of X_j as defined in (6).

In a situation where the coder is provided with explicit information about the discontinuity b and its derivatives, determination of the surflet at this node can proceed as implied in Sec. III. Specifically, the coder can construct the Taylor expansion of b around any point $\mathbf{y} \in Y_j$ and quantize the polynomial coefficients according to (7). To be precise, we choose

$$\mathbf{y} = [\beta_1 2^{-j}, \beta_2 2^{-j}, \ldots, \beta_{N-1} 2^{-j}]$$

and call this a *characteristic point*. We refer to the resulting surflet as the *quantized Taylor surflet*.[4] From (5), it follows that the squared-L_2 error of the quantized Taylor surflet approximation of f obeys

$$D_2(f, s(X_j)) = \int_{X_j} (f - s(X_j))^2 = O\left(2^{-j(K+N-1)}\right). \quad (8)$$

As discussed in Sec. II-D, our coder is not provided with explicit information of b. It is therefore important to define a technique that can obtain a surflet estimate directly from the data f. We assume that there exists a technique to compute the squared-L_2 error $D_2(f, s(X_j))$ between a given surflet $s(X_j)$ and the function f on the dyadic block. In such a case, we can search the finite surflet dictionary for the minimizer of this error. We refer to the resulting surflet as the *L_2-best surflet*. This surflet will necessarily obey (8) as well. Sections IV-D and IV-E discuss the coding implications of using each type of surflet.

C. Organization of Surflet Trees

Given a method for assigning a surflet to each tree node, it is also necessary to determine the proper dyadic segmentation for the tree approximation. This can be accomplished using the CART (or Viterbi) algorithm in a process known as *tree-pruning* [1, 2]. Tree-pruning proceeds from the bottom up, determining whether to prune the tree beneath each node (leaving it as a leaf node). Various criteria exist for making such a decision. In particular, the rate-distortion optimal segmentation can be obtained by minimizing the Lagrangian rate-distortion cost $D + \lambda R$ for a penalty term λ.

[4] For the purposes of this paper, all surflets used in the approximation that share the same characteristic point (e.g., along each column in Fig. 3) are required to be of the same scale and are assigned the same surflet parameters. This condition ensures that $\widehat{f}_R$ is Horizon-class but can be relaxed, depending on the application.

D. Leaf Encoding

An initial approach toward a surflet coder would encode a tree segmentation map denoting the location of leaf nodes, along with the quantized surflet coefficients at each leaf node.

Theorem 2 [7]: Using either the quantized Taylor surflets or the L_2-best surflets, a surflet leaf-encoder achieves asymptotic performance $D_2(f, \widehat{f}_R) \lesssim \left(\frac{\log R}{R}\right)^{\frac{K}{N-1}}$.

Comparing with (4), this simple coder is *near-optimal* in terms of rate-distortion performance.

E. Top-down Predictive Encoding

Achieving the optimal rate-distortion performance (4) requires a slightly more sophisticated coder that can exploit the correlation among nearby surflets. In this section, we briefly describe a top-down surflet coder that predicts surflet parameters from previously encoded values (see [7] for additional details).

The top-down predictive coder encodes an entire tree segmentation starting with the root node, and proceeding from the top down. Given a quantized surflet $s(X_j)$ at an interior node at scale j, we can encode its children surflets (scale $j+1$) according to the following procedure.

- **Parent-child prediction:** Let Y_j be the subdomain of X_j, and let $Y_{j+1} \subset Y_j$ be the single subdomain at scale $j+1$ that shares the same characteristic point with Y_j. Thus, for each surflet $s(X_{j+1})$ with subdomain on Y_{j+1}, every coefficient of $s(X_{j+1})$ is also a surflet coefficient of (the previously encoded) $s(X_j)$, but more precision must be provided to achieve (7). The coder provides the necessary bits.
- **Child-neighbor prediction:** We now use surflets encoded at scale $j+1$ (from Step 1) to predict the surflet coefficients for each of the remaining hypercube children of X_j. We omit the precise details but note that this prediction operates according to (5), with $||\mathbf{h}|| \sim 2^{-(j+1)}$ [7].

We have proved that the number of bits required to encode each surflet using the above procedure is independent of the scale j. Although the motivation for the above approach comes from the structure among Taylor series coefficients, the same prediction scheme will indeed work for L_2-best surflets.

Theorem 3 [7]: The top-down predictive coder using either quantized Taylor surflets or L_2-best surflets achieves the optimal rate-distortion performance $D_2(f, \widehat{f}_R) \lesssim \left(\frac{1}{R}\right)^{\frac{K}{N-1}}$.

Although only the leaf nodes provide the ultimate approximation to the function, the additional information encoded at interior nodes provides the *key* to efficiently encoding the leaf nodes. In addition, unlike the surflet leaf-encoder, this top-down approach yields a *progressive* bitstream — the early bits encode a low-resolution (coarse scale) approximation that is then refined using subsequent bits.

V. Conclusions

Our surflet-based compression framework provides a sparse representation of multidimensional functions with smooth discontinuities. We have presented a tractable method based on piecewise smooth polynomials to approximate and encode such functions. The insights that we gained, namely, quantizing higher-order terms with lesser precision and predictive coding to decrease bitrate, can be used to solve more sophisticated signal representation problems. In addition, our method requires knowledge only of the higher dimensional function and not the smooth discontinuity. Future work will focus on extending the surflet dictionary to *surfprints* (similar to the wedgeprints of [9]), which can be combined with wavelets to approximate higher dimensional functions that are *smooth* away from smooth discontinuities.

References

[1] J. K. Romberg, M. B. Wakin, and R. G. Baraniuk, "Multiscale Geometric Image Processing," in *SPIE Vis. Comm. and Image Proc.*, Lugano, Switzerland, July 2003.

[2] D. L. Donoho, "Wedgelets: Nearly-minimax estimation of edges," *Annals of Stat.*, vol. 27, pp. 859–897, 1999.

[3] E. J. Candès and D. L. Donoho, "Curvelets — A suprisingly effective nonadaptive representation for objects with edges," in *Curve and Surface Fitting*, A. Cohen, C. Rabut, and L. L. Schumaker, Eds. Vanderbilt University Press, 1999.

[4] A. N. Kolmogorov and V. M. Tihomirov, "ϵ-entropy and ϵ-capacity of sets in functional spaces," *Amer. Math. Soc. Transl. (Ser. 2)*, vol. 17, pp. 277–364, 1961.

[5] G. F. Clements, "Entropies of several sets of real valued functions," *Pacific J. Math.*, vol. 13, pp. 1085–1095, 1963.

[6] A. Cohen, W. Dahmen, I. Daubechies, and R. DeVore, "Tree approximation and optimal encoding," *J. App. Comp. Harmonic Analysis*, vol. 11, pp. 192–226, 2001.

[7] V. Chandrasekaran, M. B. Wakin, D. Baron, and R. Baraniuk, "Compressing piecewise smooth multidimensional functions using surflets: Rate-distortion analysis," Tech. Rep., ECE Dept., Rice University, 2004, Available at http://dsp.rice.edu/publications.

[8] M. N. Do, P. L. Dragotti, R. Shukla, and M. Vetterli, "On the compression of two-dimensional piecewise smooth functions," in *IEEE Int. Conf. on Image Proc. — ICIP '01*, Oct. 2001.

[9] J. K. Romberg, M. B. Wakin, and R. G. Baraniuk, "Approximation and compression of piecewise smooth images using a wavelet/wedgelet geometric model," in *IEEE Int. Conf. on Image Proc. — ICIP '03*, 2003.