Discrete denoising of heterogeneous two-dimensional data

Taesup Moon
Yahoo! Labs
701 First Ave, Sunnyvale, CA, 94089
Email: tsmoon@ymail.com

Tsachy Weissman
Department of Electrical Engineering
Stanford University, Stanford, CA
Email: tsachy@stanford.edu

Jae-Young Kim
Department of Electrical Engineering
Stanford University, Stanford, CA
Email: jykim1@stanford.edu

Abstract—We consider discrete denoising of two-dimensional data with characteristics that may be varying abruptly between regions. Using a quadtree decomposition technique and space-filling curves, we extend the recently developed S-DUDE (Shifting Discrete Universal DEnoiser), which was tailored to one-dimensional data, to the two-dimensional case. Our scheme competes with a genie that has access, in addition to the noisy data, also to the underlying noiseless data, and can employ m different two-dimensional sliding window denoisers along m distinct regions obtained by a quadtree decomposition with m leaves, in a way that minimizes the overall loss. We show that, regardless of what the underlying noiseless data may be, the two-dimensional S-DUDE performs essentially as well as this genie, provided that the number of distinct regions satisfies $m = o(n)$, where n is the total size of the data. The resulting algorithm complexity is still linear in both n and m, as in the one-dimensional case. Our experimental results show that the two-dimensional S-DUDE can be effective when the characteristics of the underlying clean image vary across different regions in the data.

I. Introduction

Discrete denoising is the problem of reconstructing the components of a finite-alphabet sequence based on the observation of its Discrete Memoryless Channel (DMC)[1]-corrupted version. Universal discrete denoising, in which no statistical or other properties are known a priori about the underlying clean data and the goal is to attain optimum performance, was first considered and solved in [1]. The main result in [1] asserts that, regardless of what the underlying individual sequence may be, the Discrete Universal Denoiser (DUDE) attains the performance of the *best* sliding window denoiser that would be chosen by a genie who accesses, in addition to the noisy sequence, the underlying clean data. Recently, in [2], a generalization has been carried out for the case in which the characteristics of the underlying sequence change over time. The new scheme, called Shifting Discrete Universal Denoiser (S-DUDE), was shown to achieve the performance of the *best combination* of sliding window denoisers, allowing at most m shifts (i.e., switches from one sliding window denoiser to another) along the sequence, provided that m grows sub-linearly in the data size n, regardless of what the underlying noiseless sequence may be. It was also shown in [2]

that the scheme can be implemented efficiently via dynamic programming, with linear complexity both in n and m.

One of the domains in which DUDE found its application is image denoising. It was shown in the experimental results of [1] and [3] that DUDE achieves or often outperforms the best of several of the state-of-the-art image denoisers, many of which are sliding window schemes, for small-alphabet images. It is natural then to attempt to extend S-DUDE for images, namely, two-dimensional data, as well. The motivation is clear; images tend to have locally distinct characteristics, and allowing the sliding window denoisers to shift from one region to another may significantly improve the denoising performance compared to applying one fixed sliding window denoiser throughout all the data. However, whereas the extension of the DUDE to two-dimensional data was straightforward (cf. [1, Section VIII-C] and [3]), that of the S-DUDE is highly non-trivial, since it requires segmentation of the data, based on its noisy observation, into homogeneous regions in a way that minimizes the overall loss. Such segmentation is significantly more involved and often intractable, in contrast to the one-dimensional case of the S-DUDE, which only required to divide the data into distinct intervals with associated denoisers.

Due to above difficulty of general segmentation of data, we instead adopt a restricted class of segmentation schemes, the quadtree decomposition, to build a reference class of shifting two-dimensional sliding window denoisers. Then, we employ the space-filling Peano-Hilbert curve [4][5] to scan the data so that applying the original one-dimensional S-DUDE on the scanned data can achieve the best performance among the schemes in the reference class, regardless of the underlying clean data. The quadtree decomposition has been popular in image compression [6] and pattern recognition [8], and recently in [9], it has also been applied to denoising continuous-valued signals by viewing denoising as a low-rate lossy compression problem. The Peano-Hilbert curves have been used, among other applications, in universal compression of two-dimensional data in both the individual sequence setting [4] and the probabilistic setting [10]. A more general problem of scanning and predicting multi-dimensional data was considered in [11], [12]. The combination of the quadtree decomposition and the Peano-Hilbert scanning for image denoising is the main contribution of this paper.

Our resulting denoising scheme, 2-D S-DUDE, still en-

---

[1]The DMC is assumed known throughout this paper. This assumption is benign in applications where the DMC is easily learnable from the data.

joys the performance guarantee that parallels that of [2] for two-dimensional data. The use of the Peano-Hilbert scan is essential to obtain a scheme of which complexity remains linear in both the data size and the number of distinct segments m, while an effort of directly finding the best quadtree decomposition may have resulted in a scheme with complexity exponential in m. We show the effectiveness of our scheme by experimental results that demonstrate 2-D S-DUDE outperforming 2-D DUDE, particularly for images of space-varying heterogeneous characteristics.

The rest of the paper is organized as follows: Section II collects notation and preliminary results necessary for the paper. Our algorithm and the main theoretical guarantee are given in Section III, and the experimental results are presented in Section IV. Concluding remarks with a discussion of future work are given in Section V.

II. NOTATION, PRELIMINARIES, AND MOTIVATION

A. Notation

We follow the notation of [2]. Let $\mathcal{X}, \mathcal{Z}, \hat{\mathcal{X}}$ denote, respectively, the alphabet of the clean, noisy, and reconstructed sources, which are assumed to be finite. As in [1], [2], the noisy sequence is a DMC-corrupted version of the clean one, where the channel matrix is denoted by $\mathbf{\Pi} = \{\Pi(x, z)\}_{x \in \mathcal{X}, z \in \mathcal{Z}}$, and $\Pi(x, z)$ stands for the probability of a noisy symbol z when the underlying clean symbol is x. Throughout the paper, $\mathbf{\Pi}$ is assumed to be known and fixed, and of full row rank. When a reconstruction $\hat{x}$ is made for a clean symbol x, the goodness of the reconstruction is measured by a loss function $\Lambda : \mathcal{X} \times \hat{\mathcal{X}} \to [0, \infty)$. Upper case letters denote random variables; lower case letters denote either individual deterministic quantities or specific realizations of random variables. Without loss of generality, the elements of any finite alphabet $\mathcal{V}$ will be identified with $\{0, 1, \cdots, |\mathcal{V}| - 1\}$. For $\mathcal{V}$-valued sequence, we let $v^n = (v_1, \cdots, v_n)$, $v_m^n = (v_m, \cdots, v_n)$, and $v^{n \backslash t} = v^{t-1} v_{t+1}^n$. $\mathbb{R}^{\mathcal{V}}$ is a space of $|\mathcal{V}|$-dimensional column vectors with real-valued components indexed by the elements of $\mathcal{V}$.

Now, consider the set $\mathcal{S} = \{s : \mathcal{Z} \to \hat{\mathcal{X}}\}$, which is the (finite) set of mappings that take $\mathcal{Z}$ into $\hat{\mathcal{X}}$. We refer to elements of $\mathcal{S}$ as "single-symbol denoisers", since each $s \in \mathcal{S}$ can be thought of as a rule for estimating $x \in \mathcal{X}$ on the basis of $z \in \mathcal{Z}$. Then, for any $s \in \mathcal{S}$, we can *always* devise an estimated loss $\ell(Z, s)$ with the knowledge of $\mathbf{\Pi}$, which is an unbiased estimate of the true expected loss $E_x \Lambda(x, s(Z))$, i.e. satisfying

$$E_x \ell(Z, s) = E_x \Lambda(x, s(Z)) \quad \forall x \in \mathcal{X}. \tag{1}$$

The expectation in (1) is with respect to the conditional distribution on Z given x, $\Pi(x, \cdot)$. For more details on the motivation for using a loss function ℓ satisfying (1), and on its explicit form, readers may refer to [2, Section II-A].

B. DUDE and S-DUDE for 1-D data

Here, we review and summarize the results from [1] and [2], and collect the ideas that will be needed for this paper.

For one dimensional data, an n-*block denoiser* is a collection of n mappings $\hat{\mathbf{X}}^n = \{\hat{X}_t\}_{1 \le t \le n}$, where $\hat{X}_t : \mathcal{Z}^n \to \hat{\mathcal{X}}$. The performance of the denoiser $\hat{\mathbf{X}}^n$ on the individual sequence pair (x^n, z^n) is measured by its normalized cumulative loss $L_{\hat{\mathbf{X}}^n}(x^n, z^n) = \frac{1}{n} \sum_{t=1}^n \Lambda(x_t, \hat{X}_t(z^n))$. As argued in [2, Section II-B], the n-block denoiser $\hat{\mathbf{X}}^n = \{\hat{X}_t\}_{1 \le t \le n}$ can be identified with $\mathbf{F}^n = \{F_t\}_{1 \le t \le n}$, where $F_t : \mathcal{Z}^{n \backslash t} \to \mathcal{S}$ is defined as follows: $F_t(z^{n \backslash t}, \cdot)$ is the single-symbol denoiser in $\mathcal{S}$ satisfying

$$\hat{X}_t(z^n) = F_t(z^{n \backslash t}, z_t), \quad \forall z_t. \tag{2}$$

One special class of widely used n-block denoisers is that of the k-th order "sliding window" denoisers, which we denote by $\hat{\mathbf{X}}^{n, \mathcal{S}_k}$. Such denoisers are of the form

$$\hat{X}_t^{s_k}(z^n) = s_k(z_{t-k}^{t+k}) = s_k(\mathbf{c}_t, z_t) \tag{3}$$

for $t = k+1, \cdots, n-k$, where s_k is an element of $\mathcal{S}_k = \{s_k : \mathcal{Z}^{2k+1} \to \hat{\mathcal{X}}\}$, the (finite) set of mappings from $\mathcal{Z}^{2k+1}$ into $\hat{\mathcal{X}}$, and $\mathbf{c}_t \triangleq (z_{t-k}^{t-1}, z_{t+1}^{t+k})$ is the (two-sided) k-th order context for z_t. We refer to $s_k \in \mathcal{S}_k$ as a "k-th order denoiser". Note that $\mathcal{S}_0$ equals to $\mathcal{S}$ in the previous section. $\mathbf{C}_k \triangleq \{(u_{-k}^{-1}, u_1^k) : (u_{-k}^{-1}, u_1^k) \in \mathcal{Z}^{2k}\}$ is the set of all possible k-th order contexts, and for given z^n and each $\mathbf{c} \in \mathbf{C}_k$, $\mathcal{T}(\mathbf{c}) \triangleq \{\tau : \mathbf{c}_\tau = \mathbf{c}, \quad k+1 \le \tau \le n-k\}$ is further defined to be the set of indices where the context of z_i equals $\mathbf{c}$. From the association in (2) and the definition (3), we can deduce that for each $\mathbf{c} \in \mathbf{C}_k$, the k-th order sliding window denoiser employs a time-invariant single-symbol denoiser, $s_k(\mathbf{c}, \cdot)$, at all points $t \in \mathcal{T}(\mathbf{c})$.

In [1], the performance target of the denoising is $D_k(x^n, z^n) \triangleq \min_{s_k \in \mathcal{S}_k} \frac{1}{n-2k} \sum_{t=k+1}^{n-k} \Lambda(x_t, s_k(\mathbf{c}_t, z_t))$, the minimum normalized loss on (x^n, z^n) that can be attained by a k-th order sliding window denoiser. It is shown in [1, Theorem 1] that, despite the lack of knowledge of x^n, $D_k(x^n, Z^n)$ is essentially achievable by the Discrete Universal DEnoiser (DUDE), which accesses only Z^n and is implementable with linear complexity in n. The essence of DUDE was to use $\ell(z, s)$ in (1), in lieu of the true loss $\Lambda(x, s(z))$, and employ $\arg \min_{s_k \in \mathcal{S}_k} \frac{1}{n-2k} \sum_{t=k+1}^{n-k} \ell_t(z_t, s(\mathbf{c}_t, \cdot))$.

Whereas the DUDE competed with the best *fixed* k-th order denoiser, [2] competes with the best among $\mathcal{S}_{k,m}^n$, a set of "combinations" of k-th order denoisers $\{s_{k,t}\}_{t=k+1}^{n-k}$ that allow at most m shifts within $t \in \mathcal{T}(\mathbf{c})$ for each $\mathbf{c} \in \mathbf{C}_k$. Thus, [2] sets a more ambitious performance target

$$D_{k,m}(x^n, z^n) \triangleq \min_{\mathbf{S} \in \mathcal{S}_{k,m}^n} \frac{1}{n-2k} \sum_{t=k+1}^{n-k} \Lambda(x_t, s_{k,t}(\mathbf{c}_t, z_t)), \tag{4}$$

the minimum normalized loss on (x^n, z^n) that can be achieved by the sequence of k-th order denoisers that allow at most m shifts (changes) within each context. It is clear that $D_{k,m}(x^n, z^n) \le D_k(x^n, z^n)$ for all (x^n, z^n). The new algorithm devised in [2] was called the (k, m)-Shifting Discrete Universal DEnoiser (S-DUDE), $\hat{\mathbf{X}}_{\text{univ}}^{n,k,m}$, and, as shown in [2, Theorem 4], was able to asymptotically achieve $D_{k,m}(x^n, Z^n)$ on the basis of Z^n only, as long as m grows sub-linearly in

n. The trick was again to work with the estimated loss $\ell(z, s)$ to obtain

$$\hat{\mathbf{S}}_{k,m} = \arg \min_{\mathbf{S} \in \mathcal{S}_{k,m}^n} \frac{1}{n - 2k} \sum_{t=k+1}^{n-k} \ell(z_t, s_{k,t}(\mathbf{c}_t, \cdot)), \qquad (5)$$

and employ it throughout the sequence. Unlike the case of DUDE, obtaining (5) efficiently is not straightforward since the size of the set $\mathcal{S}_{k,m}^n$ is exponential in both n and m. Another key component of [2] was developing an algorithm that can attain (5) efficiently, i.e., with complexity linear in both n and m. The detailed algorithm can be found in [2, Section V-A].

III. S-DUDE FOR 2-D DATA

As noted in the Introduction, extending S-DUDE to two-dimensional (2-D) data case is not straightforward. The main reason is that, whereas the output of DUDE is independent of the ordering of data within each context and only requires the empirical distribution of the data, the ordering of said data is very consequential for S-DUDE's output and its performance. Before describing how we approach to address this issue in detail, we introduce additional notation in Section III-A through Section III-C. Then, in Section III-D, we derive our scheme and present its complexity and theoretical guarantees of performance.

A. 2-D data and contexts

We represent the 2-D data with the coordinate of each data point. For simplicity, we assume the 2-D data is always in the square form[2]. Then, denote $\mathcal{T}_N \triangleq \{t \in \mathbb{Z}^2 : t = (t_1, t_2), 1 \le t_1 \le N, 1 \le t_2 \le N\}$ as the set of coordinates of the given 2-D data. Also, let $n = |\mathcal{T}_N| = N \times N$ be the total size of the data. For $t \in \mathcal{T}_N$, z_t will denote the noisy symbol at location $t = (t_1, t_2)$, and $x^{N \times N}$ and $z^{N \times N}$ will denote the total clean and noisy 2-D data, respectively. In addition, for $t \in \mathcal{T}_N$, $z^{N \times N \setminus t}$ will denote $\{z_i : i \in \mathcal{T}_N, i \ne t\}$. With this notation, notions of the 2-D n-block denoiser $\hat{\mathbf{X}}_{2D}^n$, the normalized cumulative loss $L_{\hat{\mathbf{X}}_{2D}^n}(x^{N \times N}, z^{N \times N}) = \frac{1}{n} \sum_{t \in \mathcal{T}_N} \Lambda(x_t, \hat{X}_{t,2D}(z^{N \times N}))$, and the association in (2) follow naturally in parallel to the 1-D data case.

The 2-D k-th order sliding window denoisers can be understood similarly. First, consider the sequence of coordinates, $\mathcal{I} = (i_1, i_2, i_3, \cdots)$, in the 2-D lattice of integers, in which the coordinates are enumerated in the order of increasing distance to the origin as in Figure 1. Then, 2-D k-th order context for z_t is defined to be $\mathbf{c}_{t,2D} = (z_{t+i_1}, \cdots, z_{t+i_{2k}})$, where $(i_1, \cdots, i_{2k})$ are the first $2k$ coordinates of $\mathcal{I}$, and the additions of coordinates simply boil down to the translation of the coordinates. We also denote $\mathbf{C}_{k,2D}$ as a set of all possible 2-D k-th order contexts. Then, 2-D k-th order sliding window denoiser at location t is again of the form $\hat{X}_{t,2D}^{s_k}(z^n) = s_k(\mathbf{c}_{t,2D}, z_t)$, with $\mathbf{c}_{t,2D} \in \mathbf{C}_{k,2D}$, parallelling (3).

[2]For other cases, we can simply fill in remaining regions with dummy symbols.

Fig. 1. The ordering coordinates of the 2-D integer lattice plane

B. The quadtree (QT) decomposition

A quadtree (QT) is a tree of which every node is either a leaf or a parent node with four children. This QT structure can be used to segment the 2-D data as follows. Each node of a QT at depth d represents a quadrant of size $\frac{N}{2^d} \times \frac{N}{2^d}$ (assuming $N = 2^r$ and $d \le r$), and a child node represents one of the four quadrants inside the parent node's quadrant. The four children of a parent node associate with the four quadrants of the parent node's quadrant in the order of (upper-left)→(upper-right)→(lower-right)→(lower-left). Obviously, the root node is associated with the whole 2-D data. The leaves of a QT represent the final segmentation of the 2-D data for given QT. Thus, if a QT has m leaves, the resulting segmentation has m distinct regions. For example, in Figure 2(a), the two dimensional data is segmented into 13 different regions, and the corresponding QT in Figure 2(b) has $m = 13$ leaves. We also denote Q_m as a set of all QTs that have m leaves,

(a) 2-D decomposition   (b) QT for the decomposition

Fig. 2. The decomposition of a 2-D data and its corresponding QT ($m = 13$)

and for given $q \in Q_m$, define $R_q : \mathcal{T}_N \to \{1, \cdots, m\}$ as a mapping that maps each coordinate of the 2-D data to the one of m leaves in q corresponding to the region containing the coordinate.

C. Peano-Hilbert (PH) Curve

The Peano-Hilbert (PH) curves are well known as space-filling curves. They possess the property that, for any level of quadrants, the curves never leave a quadrant before visiting all the sites within the quadrant. The details of PH curves and scanning orders can be found in [4, Section II, Figure 4]. The PH curve naturally defines an ordering of 2-D according to the order in which the PH curve fills the plane. Then, for noisy 2-D data $z^{N \times N}$, we denote z_{PH}^n as its PH scanned noisy 1-D sequence and x_{PH}^n as the corresponding clean 1-D sequence. In addition, we denote the ith index according to the PH scan by PH_i. Note that $\text{PH}_i \in \mathcal{T}_N$, $i = 1 \ldots, n$. Thus, for example,

z_{PH_i} stands for the ith component of the n-tuple z_{PH}^n, and $\mathbf{c}_{\mathrm{PH}_i,\mathrm{2D}} \in \mathbf{C}_{k,\mathrm{2D}}$ is the 2-D k-th order context at that location.

D. Derivation of the scheme and performance guarantees

Equipped with the definitions and notation in the previous subsections, we first focus on the case in which the window size $k = 0$. The main idea for $k = 0$ case could be easily generalized to $k > 0$ case.

For simplicity, we assume $N = 2^r$, and $n = N \times N = 2^{2r}$. Consider a 2-D n tuple of single-symbol denoisers $\mathbf{S} = \{s_t : t \in \mathcal{T}_N\} \in \mathcal{S}_0^n$. For such $\mathbf{S}$, we can associate the 2-D n-block denoiser $\hat{\mathbf{X}}_{\mathrm{2D}}^{n,\mathbf{S}}$ as $\hat{X}_{t,\mathrm{2D}}^{\mathbf{S}}(z^{N \times N}) = s_t(z_t)$ for all $t \in \mathcal{T}_N$. In order to construct the reference class based on QT decomposition, we define $\mathcal{S}_{0,Q_m}^n$ as a set of all possible configurations of single-symbol denoising rules confined to be constant in regions defined by all possible QTs with m leaves. For a more rigorous definition, refer to [13]. Following simple lemma shows that the size of $\mathcal{S}_{0,Q_m}^n$ grows exponentially in terms of the number of segmented regions m.

Lemma 1. *The set $\mathcal{S}_{0,Q_m}^n$ defined above satisfies* $|\mathcal{S}_{0,Q_m}^n| = \Omega\big(3^{\frac{m}{3}}\big)$.

Proof: The proof follows from enumerating all possible quadtrees in $\mathcal{S}_{0,Q_m}^n$. See [13] for details. ∎

Given the reference class $\mathcal{S}_{0,Q_m}^n$, we define the performance target for given 2-D data $(x^{N \times N}, z^{N \times N})$ as

$$D_{0,m}^{2D}(x^{N \times N}, z^{N \times N}) = \min_{\mathbf{S} \in \mathcal{S}_{0,Q_m}^n} \frac{1}{n} \sum_{t \in \mathcal{T}_N} \Lambda(x_t, s_t(z_t)), \quad (6)$$

i.e., the best denoising performance attainable among all combinations of single-symbol denoisers in $\mathcal{S}_{0,Q_m}^n$. In order to asymptotically achieve (6) based only on $Z^{N \times N}$, we may again use the idea of utilizing the estimated loss (1) in place of the true loss as in [2] to find

$$\arg \min_{\mathbf{S} \in \mathcal{S}_{0,Q_m}^n} \frac{1}{n} \sum_{t \in \mathcal{T}_N} \ell(Z_t, s_t). \quad (7)$$

However, the naive brute-force algorithm to find the achiever of (7) requires the exhaustive search over the set $\mathcal{S}_{0,Q_m}^n$, which results in *exponential* complexity in m as specified in Lemma 1. In contrast to the 1-D case, an efficient algorithm that directly finds the best combination of single-symbol denoisers in $\mathcal{S}_{0,Q_m}^n$ does not appear to exist. To circumvent this issue, the PH scanning comes into play and serves as a key component for devising an efficient algorithm to attain performance essentially at least as good as (6). To this end, we define another set of combinations of single-symbol desnoisers

$$\mathcal{S}_{0,m}^{\mathrm{PH}(n)} \triangleq \Big\{\mathbf{S} \in \mathcal{S}_0^n : \sum_{i=1}^n \mathbf{1}_{\{s_{\mathrm{PH}_{i-1}} \neq s_{\mathrm{PH}_i}\}} \leq m\Big\}. \quad (8)$$

In words, $\mathcal{S}_{0,m}^{\mathrm{PH}(n)}$ is a set of combinations of single-symbol denoisers that have at most m switches when the denoisers are ordered according to the PH scanning order. Equation (8)

is identical to [2, eq. (20)] except that it is for the PH scanned 1-D data of the original 2-D data. We can now define

$$\hat{\mathbf{S}} = \arg \min_{\mathbf{S} \in \mathcal{S}_{0,m}^{PH(n)}} \frac{1}{n} \sum_{i=1}^n \ell(Z_{\mathrm{PH}_i}, s_{\mathrm{PH}_i}), \quad (9)$$

which can be attained with *linear* complexity both in n and m by the two-pass dynamic programming algorithm established in [2, Section V]. We denote our 2-D $(0, m)$-S-DUDE as $\hat{\mathbf{X}}_{\mathrm{2D\ univ}}^{n,0,m}$, and define it to be $\hat{\mathbf{X}}_{\mathrm{2D}}^{n,\hat{\mathbf{S}}}$. Before stating the performance guarantee of our scheme, we have following lemma.

Lemma 2. *Define a quantity*

$$D_{0,m}^{PH}(x^{N \times N}, z^{N \times N}) \triangleq \min_{\mathbf{S} \in \mathcal{S}_{0,m}^{PH(n)}} \frac{1}{n} \sum_{t \in \mathcal{T}_N} \Lambda(x_t, s_t(Z_t)).$$

Then, $D_{0,m}^{PH}(x^{N \times N}, Z^{N \times N}) \leq D_{0,m}^{2D}(x^{N \times N}, z^{N \times N})$ *for every* $(x^{N \times N}, z^{N \times N})$.

Proof: The proof follows from the observation of $\mathcal{S}_{0,Q_m}^n \subseteq \mathcal{S}_{0,m}^{PH(n)}$. For details, see [13]. ∎

The following theorem shows that the performance of $\hat{\mathbf{X}}_{\mathrm{2D\ univ}}^{n,0,m}$ is concentrated around $D_{0,m}^{2D}(x^{N \times N}, z^{N \times N})$ with high probability.

Theorem 1. *For $\hat{\mathbf{X}}_{\mathrm{2D\ univ}}^{n,0,m}$ defined in (9), and for all $\epsilon > 0$ and $x^{N \times N} \in \mathcal{X}^{N \times N}$, we have*

$$\Pr\Big(L_{\hat{\mathbf{X}}_{\mathrm{2D\ univ}}^{n,0,m}}(x^{N \times N}, Z^{N \times N}) - D_{0,m}^{2D}(x^{N \times N}, Z^{N \times N}) > \epsilon\Big)$$

$$\leq 2 \exp\Big(-n\Big[\frac{\epsilon^2}{2L_{\max}^2} - 2\Big\{h\Big(\frac{m}{n}\Big) + \frac{(m+1)\ln|\mathcal{S}|}{n}\Big\}\Big]\Big) \quad (10)$$

where $h(x) = -x \ln x - (1 - x) \ln(1 - x)$ for $0 \leq x \leq 1$, $L_{\max} = \max_{x \in \mathcal{X}, z \in \mathcal{Z}} \Lambda(x, z) + \max_{z \in \mathcal{Z}, s \in \mathcal{S}} \ell(z, s)$, and $|\mathcal{S}| = |\hat{\mathcal{X}}|^{|\mathcal{Z}|}$, the size of the set of single-symbol denoisers.

Proof: The proof follows from adding and subtracting $D_{0,m}^{\mathrm{PH}}(x^{N \times N}, z^{N \times N})$, the union bound, Lemma 2, and [2, Theorem 2]. For details, refer to the full paper [13]. ∎

Now, for general $k > 0$, we can essentially run the algorithm for $k = 0$ case for each of the subsequence points associated with each 2-D context $\mathbf{c} \in \mathbf{C}_{k,2D}$. Due to the lack of space, we just give the abridged description of the general algorithm and readers may refer to [13] for rigorous arguments and details. Let $\tilde{k} = \lceil k/4 \rceil$ and $n_k = (N - \tilde{k}) \times (N - \tilde{k})$. Define $\mathcal{T}_{N_k} \triangleq \{t \in \mathbb{Z}^2 : t = (t_1, t_2), \tilde{k} + 1 \leq t_1 \leq N - \tilde{k}, \tilde{k} + 1 \leq t_2 \leq N - \tilde{k}\}$, which is a subset of $\mathcal{T}_N$ with size $|\mathcal{T}_{N_k}| = n_k$. For a reference class to compete with, we define $\mathcal{S}_{k,Q_m}^n$ as all combinations of (k-th order denoiser induced) single-symbol denoisers that, within each subsequence associated with each $\mathbf{c} \in \mathbf{C}_{k,2D}$, is confined to remain constant within each of the regions determined by QTs with m leaves. Then, given the 2-D data $(x^{N \times N}, z^{N \times N})$, the k-th order performance target is set to be $D_{k,m}^{2D}(x^{N \times N}, z^{N \times N}) = \min_{\mathcal{S}_{k,Q_m}^n} \frac{1}{n_k} \sum_{t \in \mathcal{T}_{N_k}} \Lambda(x_t, s_{k,t}(\mathbf{c}_{t,2D}, z_t))$. In order to attain

above efficiently, we again define $\mathcal{S}_{k,m}^{\mathrm{PH}(n)}$ that parallels (8) and [2, (28)] as a set of combination of k-th order sliding window denoisers that shift at most m times along the PH-scanned subsequence for each 2-D context $\mathbf{c} \in \mathbf{C}_{k,\mathrm{2D}}$. Then, we define

$$\hat{\mathbf{S}}_{k,m} = \arg \min_{\mathbf{S}_k \in \mathcal{S}_{k,m}^{\mathrm{PH}(n)}} \frac{1}{n_k} \sum_{\{i:\mathrm{PH}_i \in \mathcal{T}_{N_k}\}} \ell(Z_{\mathrm{PH}_i}, s_{k,\mathrm{PH}_i}(\mathbf{c}_{\mathrm{PH}_i,\mathrm{2D}}, \cdot))$$

and the 2-D (k,m)-S-DUDE, $\hat{\mathbf{X}}_{\mathrm{2D\ univ}}^{n,k,m}$, as $\hat{\mathbf{X}}_{\mathrm{2D}}^{n,\hat{\mathbf{S}}_{k,m}}$. Similarly as in $k = 0$ case, we can apply the same dynamic programming algorithm of the 1-D case [2] for each PH-scanned subsequence defined by each context $\mathbf{c} \in \mathbf{C}_{k,\mathrm{2D}}$ and obtain $\hat{\mathbf{S}}_{k,m}$ with complexity *linear* in both n and m. Again, directly finding the minimizer from $\mathcal{S}_{k,Q_m}^n$ requires exponential complexity in m. A subtle point to emphasize here is that we apply our algorithm on are defined by the 2-D k-th order contexts. In this way, our scheme still competes with the set of 2-D k-th order sliding window denoisers specified by the quadtree decomposition, $\mathcal{S}_{k,Q_m}^n$, with high probability. That is, we obtain a concentration theorem similar to Theorem 1 and following result, of which proof is given in [13].

Theorem 2. *Suppose $k = k_n$ and $m = m_n$ are such that they satisfy the summable condition specified in [13], e.g., $k = c_1 \log n$ with $c_1 < \frac{1}{2\log|\mathcal{Z}|}$ and $m = n^\alpha$ with $\alpha < 1$. Then, with probability 1, for all $\mathbf{x} \in \mathcal{X}^{\infty \times \infty}$, the sequence of denoisers $\{\hat{\mathbf{X}}_{\mathrm{2D\ univ}}^{n,k,m}\}$ satisfies*

$$\lim_{N \to \infty} \left[L_{\hat{\mathbf{X}}_{\mathrm{2D\ univ}}^{n,k,m}}(x^{N \times N}, Z^{N \times N}) - D_{k,m}^{\mathrm{2D}}(x^{N \times N}, Z^{N \times N}) \right] = 0.$$

Remark: One may conceive of a denoising algorithm that heuristically finds a QT decomposition by greedily merging child nodes as in [6], with the estimated loss. This scheme is practical, and may be competitive with the best shifting sliding window denoisers based on QT decomposition, but is difficult to analyze and obtain rigorous performance guarantee as our scheme.

IV. EXPERIMENTAL RESULTS

We now show the performance of our scheme on a sample image and compare with baseline schemes. Figure 3(a) shows the clean image that was constructed by pasting four binary sub-images that have different characteristics. This image was corrupted by a binary symmetric channel (BSC) with crossover probability $\delta = 0.1$. The total image size is 512×512 and each sub-image has size 256×256. We compare our 2-D S-DUDE with three different baselines; (i) **2-D DUDE** that applies regular DUDE with the 2-D contexts (ii) **1-D S-DUDE after horizontally raster scanning the data** as in [2, Section V-A], and (iii) **1-D DUDE after raster scanning the data** which is identical with (ii) with $m = 0$. Figure 3(b) shows that our scheme dominates all baselines significantly, which confirms the additional gain we get for denoising heterogeneous data. Note that (ii) does not perform well since too many shifts (linear in n) are required for a raster scanned 1-D data. On the other hand, 2-D S-DUDE PH scans the subsequence points with respect to the 2-D contexts and manages to identify the

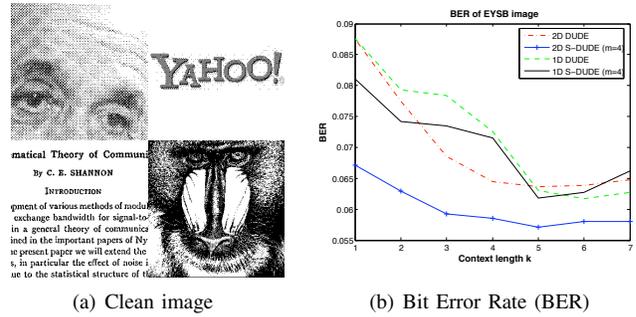(a) Clean image   (b) Bit Error Rate (BER)

Fig. 3. Synthetic test image and denoising results

heterogeneous regions and further reduces BER. Furthermore, one may think that the gain in this experiment would be coming from artificially generated image that seems to favor quadtree decomposition. However, it is shown in [13] that 2-D S-DUDE still shows gains for more realistic heterogeneous images.

V. CONCLUDING REMARKS

We have generalized the S-DUDE proposed in [2] to 2-D data by making a novel combination of QT decomposition techniques for designing a reference class and PH scanning for unfolding the 2-D data into 1-D in a desirable order. Among other related lines of inquiry, future work will investigate the effectiveness of combining more general data segmentation and scanning techniques for denoising problems.

REFERENCES

[1] T. Weissman, E. Ordentlich, G. Seroussi, S. Verdú, and M. Weinberger, "Universal discrete denoising: Known channel," *IEEE Trans. Inform. Theory*, vol. 51, no. 1, pp. 5–28, 2005.

[2] T. Moon and T. Weissman, "Discrete denoising with shifts," *IEEE Trans. Inform. Theory*, vol. 55, no. 11, pp. 5284–5301, 2009.

[3] E. Ordentlich, G. Seroussi, S. Verdú, M. Weinberger, and T. Weissman, "A discrete universal denoiser and its application to binary images," *Proc. IEEE Int. Conf. on Image Processing*, p. 117-120, vol. 1, Sept. 2003.

[4] A. Lempel and J. Ziv, "Compression of two-dimensional data," *IEEE Trans. Inform. Theory*, vol. 32, no. 1, pp. 2–8, 1986.

[5] H. Sagan, *Space-Filling Curves*. Springer-Verlag, 1994.

[6] E. Shusterman and M. Feder, "Image compression via improved quadtree decomposition algorithms," *IEEE Trans. Image Processing*, vol. 3, no. 2, pp. 207–215, 1994.

[7] G. Sullivan and R. Baker, "Efficient quadtree coding of images and video," *IEEE Trans. Image Processing*, vol. 3, no. 3, pp. 327–331, 1994.

[8] J. Smith and S.-F. Chang, "Quad-tree segmentation for texture-based image query," *Proceedings of the second ACM international conference on Multimedia*, pp. 279–286, 1994.

[9] R. Shukla and M. Vetterli, "Geometrical image denoising using quadtree segmentation," *IEEE Int. Conf. on Image Processing*, vol. 2, pp. 1213–1216, 2004.

[10] T. Weissman and S. Mannor, "Universal compression of multi- dimensional data arrays using self-similar curves," *Proc. 38th Annu. Allerton Conf. Communication, Control, and Computing*, pp.470-479, October 2000.

[11] A. Cohen, T. Weissman, and N. Merhav, "Scanning and sequential decision making for multi-dimensional data - Part I: the Noiseless case," *IEEE Trans. Inform. Theory*, vol. 53, no. 9, pp. 3001–3020, 2007.

[12] A. Cohen, T. Weissman, and N. Merhav, "Universal scanning and sequential decision making for multi-dimensional data, Part II: the Noisy case," *IEEE Trans. Inform. Theory*, vol. 54, no. 12, pp. 5609–5631, 2008.

[13] T. Moon, T. Weissman, and J. Kim, "Discrete denoising for heterogeneous two-dimensional data," submitted to *IEEE Trans. Inform. Theory*, available at http://arxiv.org/abs/1007.1799