Some Results on the Combination of Linear Transforms with Order-Statistic Filters

Amir Said
Imaging Systems Laboratory
HP Laboratories Palo Alto
HPL-2004-8
January 15th , 2004*

imaging, coding, signal processing

Since the most efficient waveform coding methods use linear transforms before quantization and entropy coding, the methods designed to allow random access to compressed data normally have to deal with the result of these transforms. However, in some specialized technical applications the information needed is the result of non-linear operations. For instance, it is useful to have fast access to the minimum and maximum values in the compression of elevation maps. In this document we show that some linear transforms have the property of preserving order if certain conditions are satisfied. We provide a proof that this property can be used not only for maximum and minimum, but also for the very general class of non-linear order-statistic filters (which includes median filters). We show that this result is valid for a set of commonly used transforms, including the discrete cosine, Walsh-Hadamard, and dyadic Haar transforms, and also valid for any type of order-statistic filter output.

* Internal Accession Date Only
Approved for External Publication

Some Results on the Combination of Linear Transforms with Order-Statistic Filters

Amir Said
Hewlett Packard Laboratories
Palo Alto, CA

Abstract

Since the most efficient waveform coding methods use linear transforms before quantization and entropy coding, the methods designed to allow random access to compressed data normally have to deal with the result of these transforms. However, in some specialized technical applications the information needed is the result of non-linear operations. For instance, it is useful to have fast access to the minimum and maximum values in the compression of elevation maps. In this document we show that some linear transforms have the property of preserving order if certain conditions are satisfied. We provide a proof that this property can be used not only for maximum and minimum, but also for the very general class of non-linear order-statistic filters (which includes median filters). We show that this result is valid for a set of commonly used transforms, including the discrete cosine, Walsh-Hadamard, and dyadic Haar transforms, and also valid for any type of order-statistic filter output.

1 Introduction

Current image compression methods that use multi-resolution representations, like wavelets, enable users to efficiently browse or extract parts of large images without the need to decompress the whole image [3]. While this is very convenient for natural images, it cannot be directly applied to those applications in which the pixel range is needed. The problem originates from the fact that low-resolution images are commonly obtained through averages of pixels from the original image, and the range is not conserved.

The use of lifting with some nonlinear filters ("nonlinear wavelets" [1, 2]) provides one type of solution. However, this simple use of nonlinear steps during lifting is not very effective. It provides very little flexibility, and it is detrimental to compression performance.

In this paper we prove that we can obtain nonlinear transforms that are modifications of commonly used linear transforms. The results are surprisingly general, showing that we can obtain the desired conservation of order statistics, while having the other transform coefficients equal to those used for coding in standard methods. We show that because the order is preserved, the result generalizes to any type of order-statistic filter output.

2 Order-Preserving Linear Transforms

Throughout this section we assume that all vectors are defined in an N-dimensional space, and consider the linear transformations that are defined by an $N \times N$ non-singular matrix $\mathbf{T}$. To study the properties of a special class of transforms we first need to define the N-dimensional vectors

$$\mathbf{u} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad \text{and} \quad \mathbf{v} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{1}$$

We say that matrix $\mathbf{T}$ defines an *order-preserving linear transform* (OPLT) if

$$\mathbf{T}^{-1}\,\mathbf{v} = \gamma\,\mathbf{u}. \tag{2}$$

which means that all elements of the first column of $\mathbf{T}^{-1}$ are equal to a constant value $\gamma \neq 0$. Note that we restrict our definition of OPLTs to those with a constant *first* column only to simplify notation. The most general case requires simply the multiplication by a permutation matrix.

Many commonly used transforms have property (2). For example, the discrete cosine transform (DCT), the Walsh-Hadamard transform, and the dyadic Haar transform [6]. It is important to observe that all the multidimensional extensions of these linear transforms have the same property. For instance, the 8×8 2-D DCT has the same property, but in a space with dimension $N = 64$.

Let $\Omega = \{1, 2, \ldots, N\}$ be the set of integers from 1 to N, and let the order of the elements of a vector $\mathbf{x}$ be defined by the vector function $\mathbf{s} : \mathbb{R}^N \to \Omega^N$, such that,

$$\bigcup_{i \in \Omega} s_i(\mathbf{x}) = \Omega, \tag{3}$$

$$s_i(\mathbf{x}) < s_j(\mathbf{x}) \implies (x_i < x_j) \text{ or } ((x_i = x_j) \text{ and } (i < j)). \tag{4}$$

For instance, $s_1(\mathbf{x})$ is the index of the smallest element of $\mathbf{x}$, $s_2(\mathbf{x})$ the second smallest, up to $s_N(\mathbf{x})$, which is the index of the largest element of $\mathbf{x}$. Note that this definition of $\mathbf{s}(\mathbf{x})$ is unique, since it also specifies the order when more than one element has the same value.

The reason we call these transforms order-preserving is because changes in the first component of the transform do not change the order when the inverse transform is computed.

Lemma 2.1 *Let* $\mathbf{T}$ *be a non-singular matrix that defines an order-preserving linear transform. For all* α *and* $\mathbf{x}$*, if* $\mathbf{a} = \mathbf{T}\mathbf{x}$ *and* $\mathbf{y} = \mathbf{T}^{-1}(\mathbf{a} + \alpha\,\mathbf{v})$ *then* $\mathbf{x}$ *and* $\mathbf{y}$ *have the same order, i.e.,* $\mathbf{s}(\mathbf{x}) = \mathbf{s}(\mathbf{y})$*.*

Proof: From the definition of $\mathbf{a}$ and $\mathbf{y}$, and (2), we have

$$\mathbf{y} = \mathbf{T}^{-1}\,\mathbf{a} + \alpha\mathbf{T}^{-1}\mathbf{v} = \mathbf{x} + \alpha\gamma\,\mathbf{u}. \tag{5}$$

thus, for any $i, j \in \Omega$

$$\begin{array}{lcl} x_i < x_j & \iff & y_i = x_i + \alpha\gamma < x_j + \alpha\gamma = y_j, \\ x_i = x_j & \iff & y_i = x_i + \alpha\gamma = x_j + \alpha\gamma = y_j, \end{array} \tag{6}$$

which means that the order is identical. ∎

We define an order-statistic filter (OSF) [4, 5] with coefficient vector $\mathbf{c}$ as

$$\mu_{\mathbf{c}}(\mathbf{x}) = \sum_{i=1}^{N} c_i\, x_{s_i(\mathbf{x})}. \tag{7}$$

This linear combination of the sorted elements of $\mathbf{x}$ defines several commonly used filters. For instance, if only c_1 (c_N) is different from zero, then $\mu_{\mathbf{c}}(\mathbf{x})$ is proportional to the minimum (maximum) element of $\mathbf{x}$. If N is an odd number and only $c_{(N+1)/2}$ is different from zero, then $\mu_{\mathbf{c}}(\mathbf{x})$ is proportional to the median of the elements of $\mathbf{x}$.

Lemma 2.2 *For any* δ*, any vector* $\mathbf{x}$*, and any order-statistic filter with coefficient vector* $\mathbf{c}$ *we have*

$$\mu_{\mathbf{c}}(\mathbf{x} + \delta\mathbf{u}) = \mu_{\mathbf{c}}(\mathbf{x}) + \delta\,\mathbf{c}'\mathbf{u}. \tag{8}$$

Proof: As shown in the proof of Lemma 2.1, order is preserved with the addition of vectors proportional to $\mathbf{u}$, i.e., $\mathbf{s}(\mathbf{x}) = \mathbf{s}(\mathbf{x} + \delta\mathbf{u})$. Thus,

$$\mu_{\mathbf{c}}(\mathbf{x} + \delta\mathbf{u}) = \sum_{j=1}^{N} c_j\,[x_{s_j(\mathbf{x}+\delta\mathbf{u})} + \delta] = \sum_{j=1}^{N} c_j\, x_{s_j(\mathbf{x})} + \delta \sum_{j=1}^{N} c_j = \mu_{\mathbf{c}}(\mathbf{x}) + \delta\,\mathbf{c}'\mathbf{u}. \tag{9}$$

∎

Using the definitions and results above we can define a family of reversible non-linear transforms as follows.

Proposition 2.3 *Let* $\mathbf{c}$ *be any vector such that* $\mathbf{c}'\mathbf{u} \neq 0$*, and let* $\mathbf{T}$ *be a non-singular matrix that defines an order-preserving linear transform, with* $\mathbf{t}_i$ *being the* i*-th row of* $\mathbf{T}$*. The nonlinear transformation* $\mathbf{f}(\mathbf{x})$ *defined by*

$$f_i(\mathbf{x}) = \begin{cases} \mu_{\mathbf{c}}(\mathbf{x}), & i = 1, \\ \mathbf{t}_i\, \mathbf{x}, & i = 2, 3, \ldots, N, \end{cases} \tag{10}$$

is reversible, and the inverse is defined by

$$\mathbf{x} = \mathbf{T}^{-1}\mathbf{f} + \frac{f_1 - \mu_{\mathbf{c}}(\mathbf{T}^{-1}\mathbf{f})}{\mathbf{c}'\mathbf{u}}\, \mathbf{u}. \tag{11}$$

Proof: Let us define

$$\mathbf{y} = \mathbf{T}^{-1}\mathbf{f}(\mathbf{x}). \tag{12}$$

From the definition of $\mathbf{f}(\mathbf{x})$ we know that

$$\mathbf{f}(\mathbf{x}) = \mathbf{T}\,\mathbf{x} + [\mu_{\mathbf{c}}(\mathbf{x}) - \mathbf{t}_1\,\mathbf{x}]\;\mathbf{v}, \tag{13}$$

and using (2) we obtain

$$\mathbf{y} = \mathbf{x} + [\mu_{\mathbf{c}}(\mathbf{x}) - \mathbf{t}_1\,\mathbf{x}]\;\mathbf{T}^{-1}\,\mathbf{v} = \mathbf{x} + \gamma[\mu_{\mathbf{c}}(\mathbf{x}) - \mathbf{t}_1\,\mathbf{x}]\;\mathbf{u} = \mathbf{x} + \delta\,\mathbf{u}. \tag{14}$$

The value of δ is not directly known for computation of the inverse transform because $\mathbf{t}_1\,\mathbf{x}$ is not in the transform vector (10). However, we can compute the value of δ from $\mathbf{f}(\mathbf{x})$ using the fact that $\mathbf{T}$ defines an order-preserving linear transform, since

$$\mu_{\mathbf{c}}(\mathbf{y}) = \mu_{\mathbf{c}}(\mathbf{x} + \delta\mathbf{u}) = \mu_{\mathbf{c}}(\mathbf{x}) + \delta\mathbf{c}'\mathbf{u}. \tag{15}$$

and thus

$$\delta = \frac{\mu_{\mathbf{c}}(\mathbf{y}) - \mu_{\mathbf{c}}(\mathbf{x})}{\mathbf{c}'\mathbf{u}} = \frac{\mu_{\mathbf{c}}(\mathbf{y}) - f_1}{\mathbf{c}'\mathbf{u}} \tag{16}$$

In conclusion, to compute the inverse transform we first compute $\mathbf{y}$ using (12), then compute $\mu_{\mathbf{c}}(\mathbf{y})$, and finally recover $\mathbf{x}$ using

$$\mathbf{x} = \mathbf{y} - \frac{\mu_{\mathbf{c}}(\mathbf{y}) - f_1}{\mathbf{c}'\mathbf{u}}\, \mathbf{u}, \tag{17}$$

which is equal to the desired result. ∎

3 Extension to Groups of Coefficients

In the last section we defined a transform that is linear except for one of its elements. We can generalize the order-preserving linear transforms to obtain transforms with similar properties, and with a larger number of non-linear coefficients.

First, we should define sets of indexes $\{\sigma_1, \sigma_2, \dots, \sigma_P\}$ which define a partition of the set $\Omega = \{1, 2, \dots, N\}$ in P disjoint groups, i.e.,

$$\begin{array}{l} \bigcup_{p=1}^{P} \sigma_p = \Omega, \\ p \neq q \iff \sigma_p \cap \sigma_q = \varnothing, \quad p, q = 1, 2, \dots, P. \end{array} \tag{18}$$

We also define the set of vectors $\mathbf{u}^{(p)}$ and $\mathbf{v}^{(p)}$, $p = 1, 2, \dots, P$, such that

$$u_i^{(p)} = \begin{cases} 1, & i \in \sigma_p \\ 0, & i \in \Omega - \sigma_p \end{cases} \tag{19}$$

and

$$v_i^{(p)} = \begin{cases} 1, & i = p \\ 0, & i \in \Omega - \{p\} \end{cases} \tag{20}$$

We say that a non-singular matrix $\mathbf{T}$ defines an *group-order-preserving linear transform* (GOPLT) if

$$\mathbf{T}^{-1}\,\mathbf{v}^{(p)} = \gamma_p\,\mathbf{u}^{(p)}, \tag{21}$$

with $\gamma_p \neq 0,\ p = 1, 2, \dots, P$.

The advantage of this type of transform is that it preserves the order inside each of the groups defined by sets σ_p. For a more formal presentation of this property we have to define new notation to describe the order inside each group. This can be done by defining a set of vector functions $\mathbf{s}^{(p)} : \mathbb{R}^N \to \Omega^N$, $p = 1, 2, \dots, P$ such that,

$$\bigcup_{i \in \Omega} s_i^{(p)}(\mathbf{x}) = \Omega, \tag{22}$$

$$s_i^{(p)}(\mathbf{x}) = i, \quad \forall\, i \notin \sigma_p, \tag{23}$$

$$s_i^{(p)}(\mathbf{x}) < s_j^{(p)}(\mathbf{x}) \implies (x_i < x_j) \text{ or } ((x_i = x_j) \text{ and } (i < j)), \quad \forall\, i, j \in \sigma_p. \tag{24}$$

Note that to be compatible with the notation we use later, the order is defined using the indexes of each group. For instance, if we have $\sigma_p = \{4, 5, 7, 9\}$ then $s_4^{(p)}(\mathbf{x})$ ($s_9^{(p)}(\mathbf{x})$) is the index of the smallest (largest) element of $\{x_4, x_5, x_7, x_9\}$. In the definition of $\mathbf{s}^{(p)}(\mathbf{x})$ the order outside the group defined by σ_p does not depend on $\mathbf{x}$, but we chose to define it as $s_i^{(p)}(\mathbf{x}) = i$, so that we can still use (22), and have each index used only once.

Using this new notation we can generalize Lemma 2.1 as follows.

Lemma 3.1 *Let $\mathbf{T}$ be a non-singular matrix that defines a group-order-preserving linear transform for groups defined by partition $\{\sigma_1, \sigma_2, \dots, \sigma_P\}$. In addition, let $\mathbf{s}^{(p)}(\mathbf{x})$ be the order defined only on the elements of $\mathbf{x}$ belonging to σ_p. For all α, $\mathbf{x}$, and all $p, q \in \{1, 2, \dots, P\}$, if $\mathbf{a} = \mathbf{T}\mathbf{x}$ and $\mathbf{y} = \mathbf{T}^{-1}(\mathbf{a} + \alpha\,\mathbf{v}^{(q)})$ then $\mathbf{x}$ and $\mathbf{y}$ have the same order in the group defined by σ_p, i.e., $\mathbf{s}^{(p)}(\mathbf{x}) = \mathbf{s}^{(p)}(\mathbf{y})$.*

Proof: We start using definition (21) to obtain

$$\mathbf{y} = \mathbf{T}^{-1}\,\mathbf{a} + \alpha \mathbf{T}^{-1}\mathbf{v}^{(q)} = \mathbf{x} + \alpha\gamma_q\,\mathbf{u}^{(q)}. \tag{25}$$

The proof for case $q \neq p$ is straightforward: we have $\mathbf{s}^{(p)}(\mathbf{x}) = \mathbf{s}^{(p)}(\mathbf{y})$ because $u_i^{(q)} = 0$ for all $i \in \sigma_p$ and thus $x_i = y_i$ for all $i \in \sigma_p$.

The proof for case $q = p$ is similar to the proof of Lemma 2.1: for all $i, j \in \sigma_p$ we have

$$\begin{array}{lcl} x_i < x_j & \Longleftrightarrow & y_i = x_i + \alpha\gamma_p < x_j + \alpha\gamma_p = y_j, \\ x_i = x_j & \Longleftrightarrow & y_i = x_i + \alpha\gamma_p = x_j + \alpha\gamma_p = y_j, \end{array} \tag{26}$$

which means that the order is identical in the subset defined by σ_p. The order outside this subset does not change either because $x_i = y_i$ for all $i \notin \sigma_p$. ∎

In our definition of new nonlinear transforms we assume that we use P different order-statistic filters, one for each group, and each using as input only the elements belonging to that group. One important property of such filters is defined as follows.

Lemma 3.2 *Given a set of group indexes* $\{\sigma_1, \sigma_2, \ldots, \sigma_P\}$, *and a particular set index* $1 \leq p \leq P$, *for any vector* $\mathbf{x}$, *and any order-statistic filter with coefficient vector* $\mathbf{c}^{(p)}$ *such that*

$$(\mathbf{c}^{(p)})'\mathbf{u}^{(p)} \neq 0, \tag{27}$$

$$c_i^{(p)} = 0, \quad \forall\, i \in \Omega - \sigma_p, \tag{28}$$

we have

$$\mu_{\mathbf{c}^{(p)}}\left(\mathbf{x} + \sum_{q=1}^{P} \delta_q \mathbf{u}^{(q)}\right) = \mu_{\mathbf{c}^{(p)}}(\mathbf{x}) + \delta_p\,(\mathbf{c}^{(p)})'\mathbf{u}^{(p)}. \tag{29}$$

Proof: From the proof of Lemma 3.1 we know that order is preserved with the addition of vectors proportional to $\mathbf{u}^{(q)}$, and thus

$$\mu_{\mathbf{c}^{(p)}}\left(\mathbf{x} + \sum_{q=1}^{P} \delta_q \mathbf{u}^{(q)}\right) = \mu_{\mathbf{c}^{(p)}}(\mathbf{x}) + \sum_{q=1}^{P} \delta_q\,(\mathbf{c}^{(p)})'\mathbf{u}^{(q)}. \tag{30}$$

From the pre-defined condition (28) we conclude that

$$p \neq q \;\Longleftarrow\; (\mathbf{c}^{(p)})'\mathbf{u}^{(q)} = 0, \tag{31}$$

and consequently

$$\sum_{q=1}^{P} \delta_q\,(\mathbf{c}^{(p)})'\mathbf{u}^{(q)} = \delta_p\,(\mathbf{c}^{(p)})'\mathbf{u}^{(p)}, \tag{32}$$

which completes the proof. ∎

At this point we can use Lemmas 3.1 and 3.2 to generalize Proposition 2.3.

Proposition 3.3 *Let* $\mathbf{T}$ *be a non-singular matrix that defines an group-order-preserving linear transform, for groups defined by sets* $\{\sigma_1, \sigma_2, \ldots, \sigma_P\}$, *and with* $\mathbf{t}_i$ *being the i-th row of* $\mathbf{T}$. *Let* $\mathbf{c}^{(p)}$, $p = 1, 2, \ldots, P < N$ *be any set of vectors such that*

$$(\mathbf{c}^{(p)})'\mathbf{u}^{(p)} \neq 0, \tag{33}$$

$$c_i^{(p)} = 0, \quad \forall\, i \in \Omega - \sigma_p. \tag{34}$$

The nonlinear transformation $\mathbf{f}(\mathbf{x})$ *defined by*

$$f_i(\mathbf{x}) = \begin{cases} \mu_{\mathbf{c}^{(i)}}(\mathbf{x}), & i = 1, 2, \ldots, P, \\ \mathbf{t}_i\,\mathbf{x}, & i = P+1, P+2, \ldots, N, \end{cases} \tag{35}$$

is reversible, and the inverse is defined by

$$\mathbf{x} = \mathbf{T}^{-1}\mathbf{f} + \sum_{p=1}^{P} \frac{f_p - \mu_{\mathbf{c}^{(p)}}(\mathbf{T}^{-1}\mathbf{f})}{(\mathbf{c}^{(p)})'\mathbf{u}^{(p)}}\, \mathbf{u}^{(p)}. \tag{36}$$

Proof: This proof is similar to the proof of Proposition 2.3, but now we have

$$\mathbf{f}(\mathbf{x}) = \mathbf{T}\,\mathbf{x} + \sum_{i=1}^{P} [\mu_{\mathbf{c}^{(i)}}(\mathbf{x}) - \mathbf{t}_i\,\mathbf{x}]\, \mathbf{v}^{(i)}. \tag{37}$$

Multiplying the transform $\mathbf{f}(\mathbf{x})$ by $\mathbf{T}^{-1}$ now yields

$$\begin{aligned} \mathbf{y} = \mathbf{T}^{-1}\mathbf{f}(\mathbf{x}) &= \mathbf{x} + \sum_{i=1}^{P} [\mu_{\mathbf{c}^{(i)}}(\mathbf{x}) - \mathbf{t}_i\,\mathbf{x}]\, \mathbf{T}^{-1}\, \mathbf{v}^{(i)} \\ &= \mathbf{x} + \sum_{i=1}^{P} \gamma_i [\mu_{\mathbf{c}^{(i)}}(\mathbf{x}) - \mathbf{t}_i\,\mathbf{x}]\, \mathbf{u}^{(i)} \\ &= \mathbf{x} + \sum_{i=1}^{P} \delta_i\, \mathbf{u}^{(i)}. \end{aligned} \tag{38}$$

The values δ_i are now computed using Lemma 3.2

$$\mu_{\mathbf{c}^{(p)}}(\mathbf{y}) = \mu_{\mathbf{c}^{(p)}}(\mathbf{x} + \sum_{i=1}^{P} \delta_i \mathbf{u}^{(i)}) = \mu_{\mathbf{c}^{(p)}}(\mathbf{x}) + \delta_p (\mathbf{c}^{(p)})'\mathbf{u}^{(p)}. \tag{39}$$

and thus

$$\delta_i = \frac{\mu_{\mathbf{c}^{(i)}}(\mathbf{y}) - \mu_{\mathbf{c}^{(i)}}(\mathbf{x})}{(\mathbf{c}^{(i)})'\mathbf{u}^{(i)}} = \frac{\mu_{\mathbf{c}^{(i)}}(\mathbf{y}) - f_i}{(\mathbf{c}^{(i)})'\mathbf{u}^{(i)}}. \tag{40}$$

∎

4 Quantization of the Transform Coefficients

Since the objective of the last sections was only to introduce this new class of non-linear transforms, we did not consider some details of practical implementations. For instance, it is also important to consider that for waveform compression the transform coefficients are quantized before coding. However, the main results also apply to the combination of transforms and quantization.

Let $\mathbf{z} = \mathbf{T}\,\mathbf{x}$ represent the transform of $\mathbf{x}$, and let $\hat{\mathbf{z}} = Q(\hat{\mathbf{z}})$ be the vector with quantized transform coefficients. We just have to change definition (10) so that the OSF output is computed not from the original vector $\mathbf{x}$ but from the recovered data available at the decoder, $\mu_{\mathbf{c}}(\mathbf{T}^{-1}\hat{\mathbf{z}})$.

$$f_i(\mathbf{x}) = \begin{cases} \mu_{\mathbf{c}}(\mathbf{T}^{-1}\hat{\mathbf{z}}) & i = 1, \\ \hat{z}_i, & i = 2, 3, \ldots, N. \end{cases} \tag{41}$$

The inverse transform is the same

$$\hat{\mathbf{x}} = \mathbf{T}^{-1}\mathbf{f} + \frac{f_1 - \mu_{\mathbf{c}}(\mathbf{T}^{-1}\mathbf{f})}{\mathbf{c}'\mathbf{u}}\,\mathbf{u}. \tag{42}$$

Different from the linear transform, this requires the encoder to compute an inverse transform $(\mathbf{T}^{-1}\hat{\mathbf{z}})$ before encoding.

5 Integer-to-Integer Transforms

Transforms for lossless compression do not have the same type of quantization used by lossy compression, but may have nonlinear truncation [7]. In this case lossless transform as long as a property similar to (2) is valid. For example, there is a unitary Walsh-Hadamard transform for lossy compression, while a version for lossless compression uses integer coefficients and truncation. The modified Walsh-Hadamard transform applied to groups of 2×2 pixels $(x_{00}, x_{01}, x_{10}, x_{11})$ is defined as

$$\begin{aligned} f_{00} &= \lfloor (x_{00} + x_{01} + x_{10} + x_{11})/4 \rfloor, && (43)\\ f_{01} &= \lfloor (x_{00} + x_{01} - x_{10} - x_{11})/2 \rfloor, && (44)\\ f_{10} &= \lfloor (x_{00} - x_{01} + x_{10} - x_{11})/2 \rfloor, && (45)\\ f_{11} &= x_{00} - x_{01} - x_{10} + x_{11}. && (46) \end{aligned}$$

The inverse transform is

$$\begin{aligned} x_{00} &= f_{00} + \lfloor (2f_{01} + 2f_{10} + 4f_{11} - 6\lfloor f_{11}/2 \rfloor + 3)/4 \rfloor, && (47)\\ x_{01} &= f_{00} + \lfloor (2f_{01} - 2f_{10} - f_{11} + 3)/4 \rfloor, && (48)\\ x_{10} &= f_{00} + \lfloor (2f_{10} - 2f_{01} - f_{11} + 3)/4 \rfloor, && (49)\\ x_{11} &= f_{00} + \lfloor (f_{11} - 2f_{01} - 2f_{10} + 2)/4 \rfloor. && (50) \end{aligned}$$

Note that the order of the recovered values is not altered by changes in f_{00}, so it too can be replaced with the output of a proper order-statistic filter (OSF).

6 Conclusion

In this report we present proofs of some properties of a family of nonlinear transforms based on the property of certain linear transforms, called order-preserving linear transforms (OPLTs). Examples of OPLTs include the discrete cosine, Walsh-Hadamard, and dyadic Haar transforms. We first define the new transforms by replacing a single OPLT coefficient with a nonlinear component—the result of an order-statistics filter applied to the original data. This is advantageous for coding purposes because most of the the well-known properties of these transforms are preserved, together with the nonlinear component. We present a proof of the perfect reversibility of these non-linear transforms by showing how to compute the inverse transform.

Next, we generalize these results to transforms that contain several nonlinear coefficients, computed from disjoint subsets of coefficients of certain linear transforms (called group-order-preserving, GOPLT). The proof of the reversibility of these transforms is also done by showing how the inverse is computed. We observe that while the nonlinear components of these transforms must be computed using only some linear transform coefficients in a pre-defined set, the other linear components do not need to satisfy any restriction other than the reversibility of the original linear transform.

In the last sections we explain how the properties can be preserved if we apply some simple nonlinear transformations, like quantization, to the coefficients of these nonlinear transforms. We discuss the changes required by the quantization of coefficients, as required in coding applications, for both lossy and lossless compression.

References

[1] C.D. Creusere, "Compression of digital elevation maps using nonlinear wavelets," *Proc. IEEE Int. Conf. Image Processing*, vol. 3, pp. 7–10, Oct. 2001.

[2] H.J.A.M. Heijmans and J. Goutsias, "Nonlinear multiresolution signal decomposition schemes — Part II: morphological wavelets," *IEEE Trans. on Image Processing,* vol. 9(11), pp. 1897–1913, Nov. 2000.

[3] D.S. Taubman and M.W. Marcellin, *JPEG 2000 Image Compression Fundamentals, Standards and Practice,* Kluwer Acad. Pub., 2002.

[4] H. A. David, *Order Statistics*, Wiley, Toronto, 1981.

[5] A. C. Bovik, T. S. Huang, and D. C. Munson, "A generalization of median filtering using linear combinations of order statistics," *IEEE Trans. Acoustics, Speech, Signal Proc.*, vol. 31(6), pp. 1342–1349, Dec. 1983.

[6] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, Addison-Wesley, Reading, MA, 1992.

[7] A. Said and W.A. Pearlman, "An image multiresolution representation for lossy and lossless compression," *IEEE. Trans. Image Processing*, vol. 5, pp. 1303–1310, Sept. 1996.