The Discrete Orthonormal Stockwell Transform and Variations, with Applications to Image Compression

J. Ladan and Edward R. Vrscay

Department of Applied Mathematics, Faculty of Mathematics,
University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
{jladan,ervrscay}@uwaterloo.ca

Abstract. We examine the so-called Discrete Orthonormal Stockwell Transform (DOST) and show that a number of quite simple modifications can be made to obtain various desired properties. For example, we introduce a real-valued Discrete Cosine-based DOST (DCST). The coefficients of the DOST and its variations are shown to exhibit a directed graph structure as opposed to the tree-like structure demonstrated by wavelet coefficients. Finally, we employ the DOST and DCST in a series of simple compression experiments and compare the results to those obtained with biorthogonal wavelets and the DCT.

1 Introduction

The Stockwell transform (ST), introduced in [3], is a time-frequency decomposition that provides absolutely-referenced frequency and phase information of a function (the phase is referenced with respect to time $t = 0$). Given a function $h(t)$ supported over the real line $\mathbb{R}$, its ST is defined as follows,

$$\mathcal{H}(\tau, f) = \frac{|f|}{\sqrt{2\pi}} \int_{-\infty}^{\infty} h(t) \exp\left(-\frac{1}{2}(t-\tau)^2 f^2\right) \exp(-i2\pi f t)\, dt, \tag{1}$$

where f is the frequency and t and τ are both time (or spatial) variables. When compared with the usual definition of the Fourier transform (FT) of h,

$$H(f) = \int_{-\infty}^{\infty} h(t)\, \exp(-i2\pi f t)\, dt, \tag{2}$$

we see that the ST is a kind of windowed Fourier transform, where the width of the (normalized) window, centered at τ, is inversely proportional to the frequency f. The ST is a highly redundant representation of a function since the FT of the function may be recovered by integration over the τ-components, i.e.,

$$\int_{-\infty}^{\infty} \mathcal{H}(\tau, f)\, d\tau = H(f). \tag{3}$$

M. Kamel and A. Campilho (Eds.): ICIAR 2013, LNCS 7950, pp. 235–244, 2013.

The ST may be viewed as a linear transform that bridges the gap between Fourier transforms and wavelets. It has been used in a variety of applications, but primarily in signal processing. (A discussion of these applications is beyond the scope of this paper.) Discrete versions of the ST have been formulated but naturally suffer from the high redundancy of the continuous transform mentioned above: Signals of length N are transformed into N^2 data points. For images and long signals, this is much more than most computers can contain in RAM.

This excessive redundancy led to the development of a non-redundant transform, the so-called Discrete Orthonormal Stockwell Transform (DOST) [4]. It has an orthonormal basis, multiple scales, and an absolutely referenced phase. There have also been a couple alternatives to the original DOST, namely the conjugate symmetric DOST [5] and what has been called the general Fourier-family transform (GFT) [1]. research on DOST for images

While the DOST has been studied for texture recognition [2] and some other signal analysis [4], there has been very little investigation into compression and filtering applications. Some very preliminary work examining compression was done by Wang and Orchard [6].

The goal of this paper is to clarify the structure of the DOST and how it is related to the alternatives proposed. From this insight into the structure of the DOST, we then introduce a Discrete Cosine-based DOST which produces a real-valued transform.

Some properties of these transforms, including their basis functions as well as the directed-graph structures of coefficients are also investigated. We then examine these transforms in a couple of simple compression experiments, and extend the work of [6]. The results are compared against established transforms such as biorthogonal wavelets and the discrete cosine transform (DCT).

2 Discrete Orthonormal Stockwell Transform (DOST)

The DOST and its variants can all be understood as combinations of Fourier transforms. Each variation changes minor aspects in order to gain certain properties such as conjugate symmetry of coefficients, or more concentrated basis functions.

To demonstrate how transforms are combined to form the DOST, we first follow the same path as Wang and Orchard [7] in their derivation of the fast DOST algorithm. Consider the following basis vector as defined by Stockwell [4], being constructed as a sum of Fourier basis vectors which are shifted in time and then phase-corrected:

$$D_{\nu\beta\tau}[k] = \frac{e^{i\pi\tau}}{\sqrt{\beta}} \sum_{f=\nu-\beta/2}^{\nu+\beta/2-1} \exp(-i\frac{2\pi}{N}kf)\exp(+i\frac{2\pi}{\beta}\tau f) \ . \tag{4}$$

Because the DOST is an orthogonal transform, each coefficient is obtained by taking an inner product of the signal vector with a basis vector,

$$
\begin{aligned}
< h, D_{\nu\beta\tau} > &= \sum_{k=0}^{N-1} h[k] \frac{e^{i\pi\tau}}{\sqrt{\beta}} \sum_{f=\nu-\beta/2}^{\nu+\beta/2-1} \exp(-i\frac{2\pi}{N}kf) \exp(+i\frac{2\pi}{\beta}\tau f) \\
&= \frac{e^{i\pi\tau}}{\sqrt{\beta}} \sum_{f=\nu-\beta/2}^{\nu+\beta/2-1} \exp(+i\frac{2\pi}{\beta}\tau f) \sum_{k=0}^{N-1} h[k] \exp(-i\frac{2\pi}{N}kf) \\
&= \frac{e^{i\pi\tau}}{\sqrt{\beta}} \sum_{f=\nu-\beta/2}^{\nu+\beta/2-1} \mathcal{F}(h)[f] \exp(+i\frac{2\pi}{\beta}\tau f) \ .
\end{aligned}
$$

The final step above follows from the definition of the Fourier Transform. Next, observe that the final summation is an inverse Fourier transform applied to a subband of the Fourier transform of the signal. We must take care in shifting the indices properly, but after everything is complete we have the result,

$$S[k] = \sqrt{\beta_k} e^{i\pi\tau_k} (\mathcal{F}_{\Omega_j}^{-1} \mathcal{F} h)[k] \ , \tag{5}$$

where $\beta_k, \tau_k, \Omega_k$ are, respectedly, the bandwidth, time-index and band of the kth basis vector.

The DOST is equivalent to the general Fourier-family transform (GFT) presented in [1], except with a rectangular window instead of a truncated Gaussian. This is also the fast algorithm derived in [7], and as noted in [7], it can be simply modified to produce the conjugate-symmetric DOST [5].

Continuing our examination of the DOST, we can also consider it in the framework of matrices. The DOST can be factored as a product of matrices,

$$\text{DOST} = \left(\bigoplus_{i=1}^{k} \mathcal{D}_i \right) \text{DFT} \ , \tag{6}$$

where $\mathcal{F}$ is the Fourier transform, and the direct sum of the matrices $\mathcal{D}_i$ forms a block-diagonal matrix, with each subblock ($\mathcal{D}_i$) being an altered inverse Fourier transform. As noted in [7], this factoring of the DOST allows us to achieve $\mathcal{O}(N \log N)$ running times, using the Fast Fourier transform.

The parameters ν, β, τ in Stockwell's definition [4] can be related to each subblock: β is the dimension of $\mathcal{D}_i$, τ is the index within $\mathcal{D}_i$, and ν is the centre of the subblock in the overall matrix. The conditions for orthogonality given by Stockwell [4] amount to requiring the dimensions of all subblocks sum to the length of the signal N. Let us now examine the matrices $\mathcal{D}_i$ more closely.

In the original DOST, each block is composed of a phase-correction, coming from the factor of $e^{i\pi\tau}$, and a β-dimensional inverse Fourier transform. Because $\tau = 0, 1, \ldots, \beta$, the phase correction can be simplified to $(-1)^\tau$. This yields the form,

$$\mathcal{D}_i = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & -1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & (-1)^n \end{bmatrix} \mathcal{F}^{-1} \,. \tag{7}$$

The conjugate symmetric DOST given in [5] includes a phase shift by 1/2 before and after, while the general Fourier-family transform [1] applies a Gaussian window before the inverse Fourier transform. The algorithm for the DOST then becomes:

```
H = FFT of the signal
for each subband in H
    adjust coefficients/ordering on subband
    apply inverse FFT on subband
```

The differences between each implementation come from how the coefficients are modified on the subband.

The DOST transform is separable [7], so higher dimensional transforms can be performed by applying the transform on each axis one after the other. We can also, for instance, treat the entire image as a vector and use 2-dimensional Fourier transforms. The latter method is used in [2] and will be necessary when implementing a Discrete Cosine-based DOST.

3 Discrete Cosine Stockwell Transform (DCST)

Considering the DOST as defined in Sect. 2, we can also modify it by using a different transform instead of the FFT. The Discrete Cosine Transform, used in many applications throughout image processing, is a real-valued transform, which makes it better suited to compression and filtering. Its relation to the Fourier transform leads to an easy adaptation of the DOST algorithm. A DCT-based DOST (DCST) may be defined by simply replacing the DFT in Eqs. (6) and (7) with a DCT:

$$\mathrm{DCST} = \left(\bigoplus_{i=1}^{k} \mathrm{DCT}_{n_i}^{-1} \right) \mathrm{DCT} \ .$$

When the DCT is used, all frequencies are positive. As a result, higher frequencies are required, so the partitioning in frequency space has to be adjusted. The most straightforward choice is to continue using the dyadic partitioning as before. Given a signal of length 2^N, the widths of the frequency partitions can be defined as follows:

$$\begin{aligned} n_1 &= 1 \\ n_i &= 2^{i-2}, \quad 2 \le i \le N-1. \end{aligned}$$

Other considerations in implementing the DCST include the magnitude of the basis functions, how a phase correction might be implemented, and separability for higher-dimensional transform. In our MATLAB implementation, we use the "dct" function, which is a unitary version of the DCT used in JPEG, so that magnitude is not a problem.

We have also found that the 2-dimensional version is not separable. It is our belief that the frequency shift by 1/2 is the cause. To overcome this, it is sufficient to perform the 2D-DCT, partition the result into sub-bands, and then perform an inverse 2D-DCT on each subband.

4 Investigations of DOST and DCST

4.1 Basis Functions

The DOST basis functions are compact in frequency, but not space. Because of the rectangular window used in the frequency domain, the basis functions exhibit ringing effects in space. We can view them as modulated sinc functions.

A look at two basis functions with different values of τ (Fig. 1) shows the difference between the DOST and wavelet transforms. The basis functions are not simply shifted and scaled; there is also a phase shift to establish an absolutely referenced phase. This phase property may prove useful for various signal and image processing algorithms.

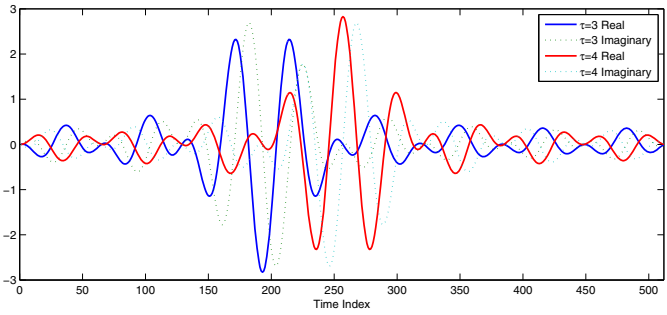
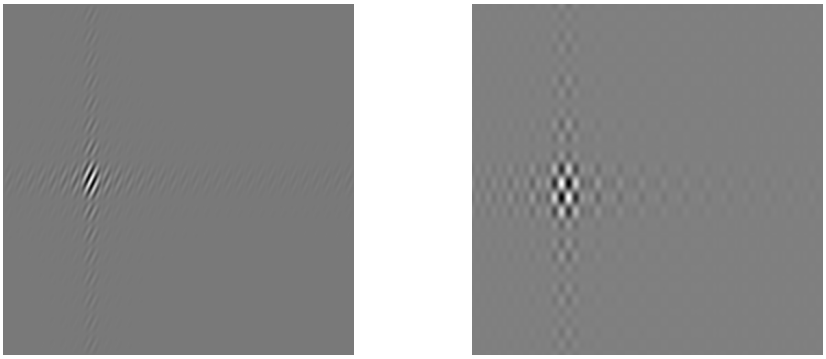


Fig. 1. DOST basis vectors for $\nu = 12$, $\beta = 8$

Examples of basis functions for the 2-dimensional DOST and DCST can be seen in Fig. 2. The fairly local response can be seen in both basis functions, as well as the oscillations on an angle in the DOST. The ringing due to a rectangular window is also quite apparent.

4.2 Directed-Graph Structure of DOST Coefficients

Many wavelet algorithms take advantage of the tree-like structure of the wavelet coefficients. This tree structure comes out of the refinement of horizontal, vertical, and diagonal details. The horizontal detail at the jth level is further refined



(a) (b)

Fig. 2. Examples of 2-Dimensional 512x512 basis functions. For both, $\beta_x = 32, \beta_y = 12$, $\tau_x = \tau_y = 8$. (a) Real part of DOST, (b) DCST.

into four horizontal detail coefficients at the $(j+1)$th level. This also applies to the vertical and diagonal detail components, resulting in 3 different quadtrees.

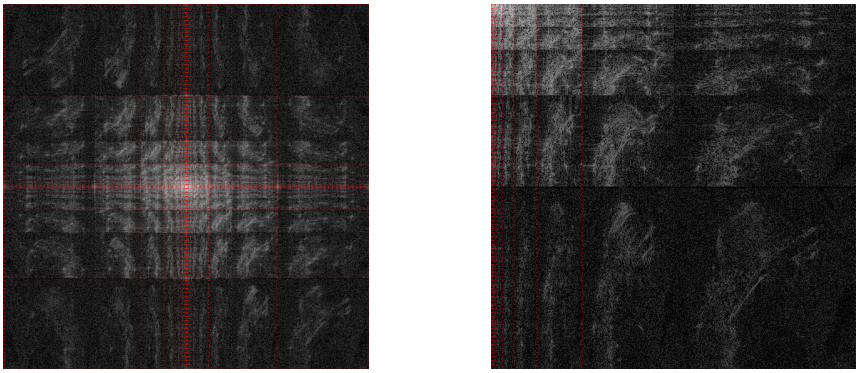
If we examine the structure of the DOST coefficients in the same manner, we do not find a tree-like structure. Rather, a directed graph structure is present. Each band has a corresponding vertical and horizontal frequency resolution. A band with a higher horizontal frequency could be viewed as refining the horizontal detail. The difference is that it is possible to refine either the vertical or horizontal resolution of each band. This means the $(i+1, j+1)$th band is a child of both $(i, j+1)$ and $(i+1, j)$ bands. Thus, it is not possible to directly apply tree-based algorithms on the DOST coefficients.

There is one advantage to having all these wavenumbers: edges/details which are not oriented at 0°, 45°, or 90° are well represented. A coefficient with corresponding wave-vector $\boldsymbol{k} = (k_x, k_y)$ will represent the strength of an edge perpendicular to the wave-vector. As an example, the basis function in Fig. 2(a) would respond more to a 15° feature.

4.3 Examples of Transformed Images

If we look at some examples of transformed images, it is possible to see the nature of the DOST a bit more clearly. Fig. 3 displays the original DOST and the DCST of the test image (512×512 pixel, 8 bits per pixel) *Lena*. The blocked structure can be seen in the little repeating images of the original, in which the edges that are highlighted. The individual bands are delineated to make them more visible.

In the original DOST, the replicas of the original image are flipped along the vertical and horizontal axes, because we use the ordering of coefficients



(a) (b)

Fig. 3. DOST-based transforms of the image *Lena* (512 x 512 pixel, greyscale). The brightness represents the magnitude of each coefficient on a log scale. (a) original DOST, (b) DCST.

proposed in [7] which establishes a conjugate symmetry. The DCST is a real-valued transform, without the concept of negative frequencies, so none of the sub-blocks are mirrored.

5 Structural Similarity

In the compression experiments that follow, we shall employ, in addition to the usual PSNR measure, the structural similarity (SSIM) image quality measure. The SSIM index was proposed [8] as an improved measure of visual distortions between two images. If one of the images is considered perfect, the SSIM can be used to assess image quality, much like PSNR.

The SSIM between two image patches, $\boldsymbol{x}$ and $\boldsymbol{y}$, is usually expressed as a product of two components which measure (i) the similarities of their mean values and (ii) their correlation and contrast distortion. The SSIM between $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as follows:

$$S(\boldsymbol{x},\boldsymbol{y}) = S_1(\boldsymbol{x},\boldsymbol{y})S_1(\boldsymbol{x},\boldsymbol{y}) = \left[\frac{2\bar{\boldsymbol{x}}\bar{\boldsymbol{y}}+\epsilon_1}{\bar{\boldsymbol{x}}^2+\bar{\boldsymbol{y}}^2+\epsilon_1}\right]\left[\frac{2s_{\boldsymbol{xy}}+\epsilon_2}{s_{\boldsymbol{x}}^2+s_{\boldsymbol{y}}^2+\epsilon_2}\right] \quad , \tag{8}$$

where

$$\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} x_i, \;\; s_{\boldsymbol{xy}} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i-\bar{\boldsymbol{x}})(y_i-\bar{\boldsymbol{y}}) \tag{9}$$

and $s_{\boldsymbol{x}}^2$ is obtained by setting $\boldsymbol{x}=\boldsymbol{y}$. The parameters $\epsilon_1, \epsilon_2 \ll 1$ are small constants (relative to the maximum size of the intensities) used to provide numerical stability and to accommodate the human visual system (Weber's Law). The SSIM is symmetric, i.e., $S(\boldsymbol{x},\boldsymbol{y}) = S(\boldsymbol{y},\boldsymbol{x})$. Furthermore, it is bounded: $-1 \le S(\boldsymbol{x},\boldsymbol{y}) \le 1$, with $S(\boldsymbol{x},\boldsymbol{y}) = 1$ if and only if $\boldsymbol{x}=\boldsymbol{y}$.

Here, we employ the *mean-SSIM* to measure the SSIM between two entire images. The SSIM is calculated between corresponding patches of the two images which are windowed by a Gaussian weight function. As its name suggests, the mean-SSIM is the mean of all these SSIM values.

6 Image Degradation Tests

In order examine the potential for DOST in a few simple applications, the effectiveness of some basic algorithms were tested against well-known transforms. In the first test, a percentage of the least significant coefficients are removed. As more coefficients are removed, the quality of the image degrades. We can get a rough idea of the effectiveness of a transform for compression by comparing its rate of degradation with other transforms.

The DOST, DCST, symmetric DOST, GFT and an 8x8 block-DCT were implemented in MATLAB. To ensure valid results, each transform was checked for invertibility and conjugate-symmetry where applicable. For the wavelet transforms, MATLAB's Wavelet Toolbox function "wavedec2" was used.

For each transform, coefficients were removed by setting them to zero via hard-thresholding. The number of coefficients removed ranged from 0% to 100%, with more test points near 100%, in order to capture the rapid variation. After each thresholding, the compressed data was inverted, and compared against the original image using PSNR and mean-SSIM. The code to calculate the mean-SSIM was taken from [9].

For an alternative compression scheme, we also tested a progressive encoding scheme which approximates the image quality obtained from Embedded Zero Tree Wavelet compression. The DCST and Biorthogonal 4.4 wavelets were compared. An initial threshold of $\frac{1}{2}$(maximum coefficient) was chosen. At each pass, coefficients above the threshold are used to approximate the image, and the new threshold is divided by two. The actual encoding scheme using the tree structure was not tested, because there is no clear tree structure in the DOST-based transforms, as explained in Sect. 4.2

The coefficients were not quantized in either test. This is because there are many different schemes available for quantization, and the focus of this paper is on the processing which is applied before that step. All arithmetic was performed in double-precision floating point.

7 Results

We first examine the degradation curves of the various transforms. The conjugate-symmetric DOST presented in [5] proved to be nearly identical in quality to the original DOST, so it was omitted from the plots. The remaining degradation curves are shown in Fig. 4.

The preliminary work in [6] showed DOST to perform better than Daubechies wavelets which we have confimed. However, biorthogonal wavelets perform better than DOST for image compression according to both PSNR and SSIM, but only

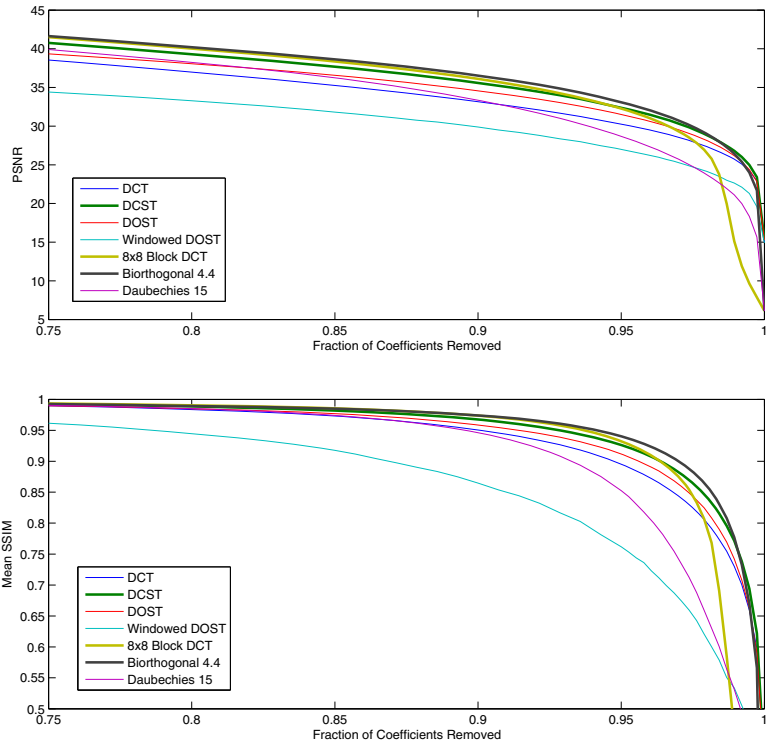


Fig. 4. Average degradation of the test images vs. percentage of coefficients truncated. (a) PSNR, (b) SSIM.

modestly better than the DCST. Among the various DOST transforms, the DCST demonstrated the best performance, while the windowed DOST showed the worst.

In our embedded coding scheme, the DCST provided a higher quality image after fewer steps than the biorthogonal wavelets. The reason for this is seen when one counts the number of coefficients included in each pass. The DCST contains many more coefficients above each threshold after the third pass, so more significant data is sent. This indicates that the compression may not be as good, which agrees with the first test.

8 Conclusion

The discrete orthornormal Stockwell transform can be understood in a few ways. By factoring it in terms of Fourier transforms and basic linear filters, we can make very simple modifications to obtain various desired properties. The connection between the DOST and related transforms amounts to changing one or two steps

in the algorithm. We also used this factoring to design a discrete cosine Stockwell transform, which is real valued and which shows better compression.

While the DOST and DCST perform better compression than some wavelet transforms, the biorthogonal wavelets used in JPEG2000 still perform better in terms of both PSNR and SSIM. The DOST and DCST also lack the tree structure which is crucial to progressive coding routines like zero-tree wavelet compression.

One of the distinguishing properties of DOST is the absolutely referenced phase, but the basic compression algorithms here ignored the phase information. We believe that there is potential for DOST in algorithms which take advantage of the phase information.

Acknowledgements. We gratefully acknowledge the support of this research by the Natural Sciences and Engineering Research Council of Canada (Discovery Grant of ERV) and the Faculty of Mathematics, University of Waterloo (JL).

References

1. Brown, R.A., Lauzon, M.L., Frayne, R.: A general description of linear time-frequency transforms and formulation of a fast, invertible transform that samples the continuous s-transform spectrum nonredundantly. IEEE Transactions on Signal Processing 58(1), 281–290 (2010)
2. Drabycz, S., Stockwell, R.G., Mitchell, J.R.: Image texture characterization using the discrete orthonormal s-transform. Journal of Digital Imaging 22(6), 696–708 (2009)
3. Stockwell, R.G., Mansinha, L., Lowe, R.P.: Localization of complex spectrum: the S transform. IEEE Transactions on Signal Processing 144(4), 998–1001 (1996)
4. Stockwell, R.G.: A basis for efficient representation of the s-transform. Digital Signal Processing: A Review Journal 17(1), 371–393 (2007)
5. Wang, Y., Orchard, J.: Symmetric discrete orthonormal stockwell transform. Numerical Analysis and Applied Mathematics 1048, 585–588 (2008)
6. Wang, Y., Orchard, J.: On the use of the stockwell transform for image compression. Image Processing: Algorithms and Systems Vii 7245, 724504 (2009)
7. Wang, Y., Orchard, J.: Fast discrete orthonormal stockwell transform. Siam Journal on Scientific Computing 31(5), 4000–4012 (2009)
8. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing 13, 600–612 (2004)
9. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: The SSIM Index for Image Quality Assessment, `https://ece.uwaterloo.ca/~z70wang/research/ssim/`