NONSUBSAMPLED CONTOURLET TRANSFORM: FILTER DESIGN AND APPLICATIONS IN DENOISING

Arthur L. da Cunha, J. Zhou and Minh N. Do

University of Illinois at Urbana-Champaign
Department of Electrical and Computer Engineering
Coordinated Science Laboratory, Urbana, IL 61801
Email:{cunhada,jzhou2,minhdo}@uiuc.edu

ABSTRACT

In this paper we study the nonsubsampled contourlet transform. We address the corresponding filter design problem using the McClellan transformation. We show how zeroes can be imposed in the filters so that the iterated structure produces regular basis functions. The proposed design framework yields filters that can be implemented efficiently through a lifting factorization. We apply the constructed transform in image noise removal where the results obtained are comparable to the state-of-the art, being superior in some cases.

1. INTRODUCTION

A number of image processing tasks are efficiently carried out in a domain other than the pixel domain, often by means of an invertible linear transformation. This linear transformation can be redundant or not, depending on whether the set of basis functions is linear independent. By allowing redundancy, it is possible to enrich the set of basis functions so that the representation is more efficient in capturing some signal behavior. Imaging applications such as edge detection, contour detection, denoising and image restoration can greatly benefit from redundant representations.

In the context of multiscale expansions implemented with filter banks, dropping the basis requirement offers the possibility of an expansion that is shift-invariant, a crucial property in a number of applications. For instance, in image denoising via thresholding in the wavelet domain, the lack of shift-invariance causes pseudo-Gibbs phenomena around singularities [1]. As a result, most of the state-of-the-art wavelet denoising routines (see e.g. [2], [3]) use an expansion with less shift sensitivity than the standard maximally decimated wavelet decomposition.

In addition to shift-invariance, an efficient image representation has to account for the geometry pervasive in natural scenes. The contourlet transform [4] is a directional multiscale transform that is constructed by combining the Laplacian pyramid (LP) and the directional filter bank (DFB) [5]. Due to downsamplers and upsamplers present in both the LP and DFB, the contourlet transform is not shift-invariant.

The non-subsampled contourlet transform (NSCT) is obtained by coupling a nonsubsampled pyramid structure with the nonsubsampled DFB (Section 2). In this paper we address the filter design problem of the NSCT (Section 3) and show its effectiveness in image denoising (Section 4).

Thanks to CAPES and NSF for funding.

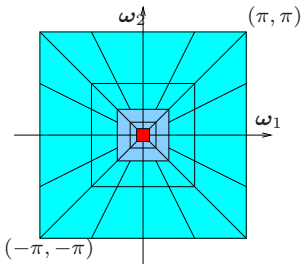


Fig. 1. The idealized frequency partitioning obtained with the NSCT.

2. THE NONSUBSAMPLED CONTOURLET TRANSFORM

The idea behind a fully shift invariant multiscale directional expansion similar to contourlets is to obtain the frequency partitioning of Figure 1 without resorting to critically sampled structures that have periodically time-varying units such as downsamplers and upsamplers. The NSCT construction can thus be divided into two parts: (1) A nonsubsampled pyramid structure which ensures the multiscale property and (2) A nonsubsampled DFB structure which gives directionality. Next we describe each part in detail.

2.1. The Nonsubsampled Pyramid

The shift sensitivity of the LP can be remedied by replacing it with a 2-channel nonsubsampled 2-D filter bank structure. Such expansion is similar to the 1-D *à trous* wavelet expansion [6] and has a redundancy of $J + 1$ when J is the number of decomposition stages. The ideal frequency support of the low-pass filter at the j-th stage is the region $[-\frac{\pi}{2^j}, \frac{\pi}{2^j}] \times [-\frac{\pi}{2^j}, \frac{\pi}{2^j}]$. Accordingly, the support of the high-pass filter is the complement of the low-pass support region on the $[-\frac{\pi}{2^{j+1}}, \frac{\pi}{2^{j+1}}] \times [-\frac{\pi}{2^{j-1}}, \frac{\pi}{2^{j-1}}]$ square. The proposed structure is thus different from the tensor product *a tròus* algorithm. It has $J + 1$ redundancy. By contrast, the 2-D *à trous* algorithm has $3J + 1$ redundancy.

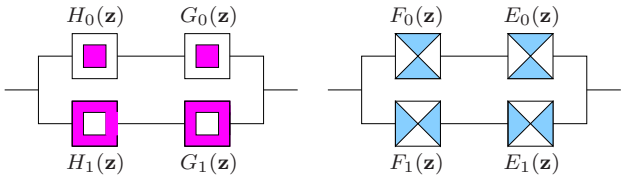


Fig. 2. Two kinds of desired responses. (a) The pyramid desired response. (b) The fan desired response.

2.2. The Nonsubsampled Directional Filter Bank

The directional filter bank [5] is constructed by combining critically sampled fan filter banks and pre/post re-sampling operations. The result is a tree-structured filter bank which splits the frequency plane into directional wedges.

A fully shift-invariant directional expansion is obtained by simply switching off the downsamplers and upsamplers in the DFB equivalent filter bank. Due to multirate identities, this is equivalent to switching off each of the downsamplers in the tree structure, while still keeping the re-sampling operations that can be absorbed by the filters. This results in a tree structure composed of two-channel nonsubsampled filter banks.

The NSCT is obtained by carefully combining the 2-D nonsubsampled pyramid and the nonsubsampled DFB (NSDFB) [7]. The resulting filtering structure approximates the ideal partition of the frequency plane displayed in Figure 1. It must be noted that, different from the contourlet expansion, the NSCT has a redundancy given by $R = \sum_{j=0}^{J} 2^{l_j}$ where 2^{l_j} is the number of directions at scale j.

3. FILTER DESIGN

The filter design problem in the proposed NSCT construction can be split into two related parts: (1) design of the pyramid filters in the NS pyramid and (2) design of the nonsubsampled fan filter bank which constitute the basic building block of the NSDFB. Figure 2 illustrates these nonsubsampled filter banks and their idealized frequency responses. The goal is to design a set of filters that satisfy the Bezout identity

$$H_0(\mathbf{z})G_0(\mathbf{z}) + H_1(\mathbf{z})G_1(\mathbf{z}) = 1, \tag{1}$$

and, in addition, approximate the ideal frequency responses (Fig. 2.) A nonsubsampled filter bank underlies a frame expansion in $\ell_2(\mathbf{Z}^2)$ and the frame is tight whenever $G_i(\mathbf{z}) = H_i(\mathbf{z}^{-1})$, $i = 0, 1$ [8]. Tight FIR designs often require spectral factorization [8] which is hard in 2-D. If we relax the tightness constraint then the design becomes more flexible. In addition, non-tight FIR designs can be linear-phase, thus avoiding phase distortion and allowing for symmetric extension at the boundaries.

A set of 2-D filters satisfying (1) can be constructed with the aid of the McClellan transformation. The design can be summarized in the following steps:

1. Construct a set of 1-D polynomials $\{H_0^{(1D)}, H_1^{(1D)}, G_0^{(1D)}, G_1^{(1D)}\}$ that satisfy the Bezout identity.
2. Construct a mapping function $f(\mathbf{z})$ so that the 2-D filters $H_0^{(1D)}(f(\mathbf{z}))$, $H_1^{(1D)}(f(\mathbf{z}))$, $G_0^{(1D)}(f(\mathbf{z}))$, $G_1^{(1D)}(f(\mathbf{z}))$ satisfy the Bezout identity, have desired frequency response and satisfy the "zeros" condition (see below).

The 1-D filters are called prototype filters. We restrict our attention to zero phase FIR designs only. In this case we write $f(\mathbf{z}) = \tilde{f}(x, y)$ with $x = (z_1 + z_1^{-1})/2$ and $y = (z_2 + z_2^{-1})/2$.

3.1. Zeros of the mapped filters

It is desirable to have zero moments in the mapped filters. For the pyramid case, high order zeros at $\omega_1 = \pi$ and $\omega_2 = \pi$ result in smoothness of the scaling function and wavelet associated with the iterated filter bank. We point out that for the approximation of polynomial surfaces, point zeros at $(\pm\pi, \pm\pi)$ would suffice. However, point zeros alone do not guarantee a "reasonable" frequency response of the pyramid filters. Furthermore, to obtain a high degree of smoothness, a high number of point zeros is often needed. By contrast, a small number of line zeros at $\omega_1 = \pi$ and $\omega_2 = \pi$ yield a fairly good degree of smoothness. The following proposition characterizes the mapping function that generates such zeros.

Proposition 1 *Let $G(z)$ be a n-th order polynomial with $n \geq 1$ and roots $\{z_i\}_{i=1}^{n}$ where each z_i has multiplicity n_i. Suppose we want a mapping function $f(x, y)$ such that*

$$G\left(f(x,y)\right) = (x-c)^{N_x}\ (y-d)^{N_y}\ \ r(x,y) \tag{2}$$

where $r(x, y)$ is a bivariate polynomial. Then $G(f(x, y))$ has the form in (2) if and only if $f(x, y)$ takes the form

$$f(x,y) = z_j + (x-c)^{N'_x}\ \ (y-d)^{N'_y}\ \ r_f(x,y) \tag{3}$$

for some root z_j and with $r_f(x, y)$ a bivariate polynomial and N'_x, N'_y such that $N'_x n_i \geq N_x$ and $N'_y n_i \geq N_y$.

Proof: See [7].

3.2. Implementation through lifting.

If a set of filters $\{H_0(\mathbf{z}), H_1(\mathbf{z}), G_0(\mathbf{z}), G_1(\mathbf{z})\}$ satisfy (1), then a different solution can be obtained by setting $G'_0(\mathbf{z}) = G_0(\mathbf{z}) + r(\mathbf{z})$ and $H'_1(\mathbf{z}) = H_1(\mathbf{z}) - s(\mathbf{z})$ in which case the Bezout's relation is satisfied provided that

$$H_0(\mathbf{z})r(\mathbf{z}) = G_1(\mathbf{z})s(\mathbf{z}) \tag{4}$$

Consequently, we see that $r(\mathbf{z})$ is an FIR function that has $G_1(\mathbf{z})$ as a factor so that $r(\mathbf{z}) = G_1(\mathbf{z})f(\mathbf{z})$ and similarly we have $s(\mathbf{z}) = H_0(\mathbf{z})g(\mathbf{z})$. Substituting in (4) we conclude that $g(\mathbf{z}) = f(\mathbf{z})$. We then have the following:

Proposition 2 *Let $H_0(\mathbf{z})$, $H_1(\mathbf{z})$, $G_0(\mathbf{z})$, $G_1(\mathbf{z})$ be FIR 2-D filters satisfying the Bezout relation (1). Then, the new filters defined as*

$$G'_0(\mathbf{z}) = G_0(\mathbf{z}) - G_1(\mathbf{z})P(\mathbf{z}), \quad H'_1(\mathbf{z}) = H_1(\mathbf{z}) + H_0(\mathbf{z})P(\mathbf{z}) \tag{5}$$

or

$$H_0'(\mathbf{z}) = H_0(\mathbf{z}) + H_1(\mathbf{z})U(\mathbf{z}), \quad G_1'(\mathbf{z}) = G_1(\mathbf{z}) + G_0(\mathbf{z})U(\mathbf{z}) \tag{6}$$

where $P(\mathbf{z})$ and $U(\mathbf{z})$ are 2-D FIR filters also satisfy Bezout's relation.

Proposition 2 characterizes the NSFB in terms of lifting steps in the same way as in the critically sampled filter bank. In this context, (5) is a predict operation whereas (6) is the update operation. Thus, one can construct NS filter banks starting with the *Lazy* nonsubsampled filters given by $G_0(\mathbf{z}) = K$, $H_1(\mathbf{z}) = 1-K$ and then adding update and predict operators. Figure 3 shows the resulting ladder structure with one predict and one update step.

In the 1-D case, one can show the above factorization is complete, as a consequence of the Euclidean algorithm for Laurent polynomials. Moreover, given $H_0(z)$ and $H_1(z)$, the factorization is easily computed. For the 2-D case, computing a factorization is hard as a Euclidean algorithm is absent. When the filters are

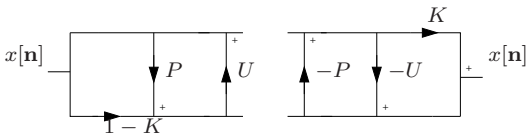


Fig. 3. Lifting structure for the nonsubsampled filter bank.

designed via mapping then a factorization can be obtained from the 1-D filters. Assume without loss of generality that the degree of the highpass prototype filter $H_1^{(1D)}(x)$ is smaller than the degree of $H_0^{(1D)}(x)$. Since there are synthesis filters $G_0^{(1D)}(x)$ and $G_1^{(1D)}(x)$ such that Bezout's identity is satisfied, it follows that $\gcd\{H_0^{(1D)}, H_1^{(1D)}\} = 1$. The Euclidean algorithm then allows us to write [9]

$$\begin{pmatrix} H_0^{(1D)}(x) \\ H_1^{(1D)}(x) \end{pmatrix} = \prod_{i=0}^{N} \begin{pmatrix} 1 & 0 \\ p_i(x) & 1 \end{pmatrix} \begin{pmatrix} 1 & q_i(x) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

for q_i, p_i polynomials. As a result we can obtain a 2-D lifting factorization by replacing x with the mapping function $f(x, y)$. In general, the implementation using predict/update stages halves the number of multiplications/additions of the direct form. The complexity can be reduced further if the mapping filter is separable. The next example illustrates the main points of this section.

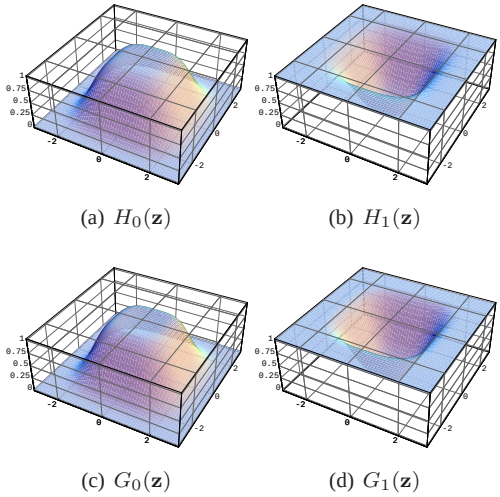
Example 1 *We construct the prototypes to each have one zero at $x = -1$ and minimize the mean square distance of their coefficients. This ensures that the underlying frame in the NSFB is close to being tight. We get*

$$H_0^{(1D)}(x) = \frac{1}{2}(x+1)\left(\sqrt{2} + (1-\sqrt{2})x\right)$$

and

$$G_0^{(1D)}(x) = \frac{1}{2}(x+1)\left(\sqrt{2} + (4-3\sqrt{2})x + (2\sqrt{2}-3)x^2\right).$$

To guarantee the response of the high-pass filters we impose $H_1^{(1D)}(x) = H_0^{(1D)}(-x)$ and $G_1^{(1D)}(x) = G_0^{(1D)}(-x)$. The lifting



(a) $H_0(\mathbf{z})$ (b) $H_1(\mathbf{z})$

(c) $G_0(\mathbf{z})$ (d) $G_1(\mathbf{z})$

Fig. 4. Design example 1 with maximally flat filters. The nonsubsampled pyramid filter bank is almost tight.

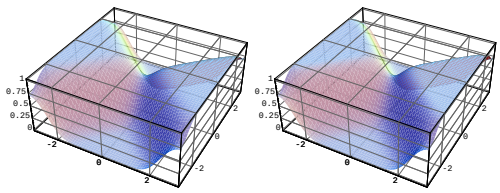
factorization of the prototype filters is given by

$$\begin{pmatrix} H_0^{(1D)}(x) \\ H_1^{(1D)}(x) \end{pmatrix} = \begin{pmatrix} 1 & \alpha x \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \beta x & 1 \end{pmatrix} \begin{pmatrix} 1 & \gamma x \\ 0 & 1 \end{pmatrix} \begin{pmatrix} K_1 \\ K_2 \end{pmatrix} \tag{7}$$

with $\alpha = \gamma = 1 - \sqrt{2}$, $\beta = \frac{1}{\sqrt{2}}$, $K_1 = K_2 = \frac{1}{\sqrt{2}}$. This implementation uses 5 multiplies/sample whereas the direct one yields 10 multiplies/sample. Following Proposition 1, we set $f(x, y) = -1 + 2P_{2,4}(x)P_{2,4}(y)$ using the maximally flat polynomials:

$$P_{N,L}(x) := (1+x)^N \sum_{l=0}^{L-1-N} \binom{N+l-1}{l} 2^{-N-l}(1-x)^l.$$

From Proposition 1, each of the resulting highpass filters has a 4-th order zero at $\omega_1 = \pi$ and $\omega_2 = \pi$. The sizes of the filters $H_0(\mathbf{z})$ and $G_0(\mathbf{z})$ are 13×13 and 19×19 respectively. Figure 4 shows the obtained frequency responses.



(a) $F_0(\mathbf{z})$ (b) $E_0(\mathbf{z})$

Fig. 5. Fan filters designed with prototype filters of Example 1 and diamond maximally flat mapping filters.

3.3. Fan Filter Design

Similar to the pyramid case, due to lack of factorization tools we resort to non-tight solutions obtained through mapping. Thus, the methodology is similar to the pyramid case, the distinction being

"Lena"	PSNR (dB)			
σ	Noisy	SI-AdaptShr [2]	BivShrink [3]	NSCT
10	28.13	-	35.34	**35.38**
15	24.65	33.37	33.67	33.67
20	22.13	32.08	32.40	**32.43**
25	20.17	31.13	31.40	**31.45**
30	18.63	-	30.54	**30.64**
"Barbara"	PSNR (dB)			
σ	Noisy	SI-AdaptShr [2]	BivShrink [3]	NSCT
10	28.17	-	33.35	**33.98**
15	24.65	31.11	31.31	**31.94**
20	22.15	29.49	29.80	**30.51**
25	20.22	28.30	28.61	**29.40**
30	18.63	-	27.65	**28.49**

Table 1. Denoising results for various soft-threshold estimators.

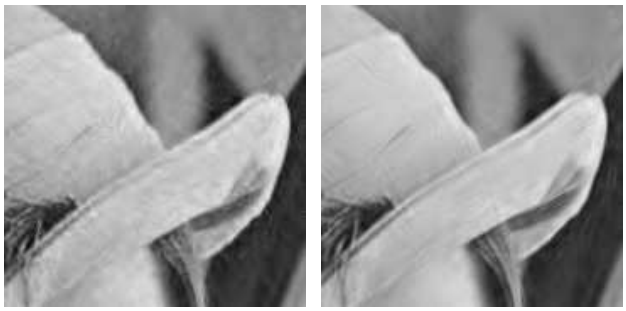
the mapping function. A useful family of mapping functions in this context is the maximally flat diamond one. The fan shaped mapping can be obtained from the diamond one by a simple change of variables. The maximally flat diamond mapping polynomial is obtained by imposing flatness at the points $(x, y) = (1, 1)$ and $(x, y) = (0, 0)$. For instance, with the same prototypes of Example 1, we use a maximally flat mapping to obtain the fan filters $F_0(\mathbf{z})$ and $E_0(\mathbf{z})$ shown in Figure 5. The other filters $F_1(\mathbf{z})$, $E_1(\mathbf{z})$ are modulated versions of $F_0(\mathbf{z})$, $E_0(\mathbf{z})$. The sizes of $F_0(\mathbf{z})$ and $E_0(\mathbf{z})$ are 21×21 and 31×31 respectively.

4. DENOISING EXPERIMENTS

In other to illustrate the potential of the NSCT we attempt to remove additive Gaussian Noise (AWGN) of images by means of a threshold estimator. We perform soft thresholding independently in each high pass filter output of the NSCT. Following the work in [2] we choose the threshold locally using

$$T_{i,j,\mathbf{n}} = \frac{\sigma^2_{n_{ij}}}{\sigma_{i,j,\mathbf{n}}}$$

where $\sigma_{i,j,\mathbf{n}}$ denotes the variance of the $\mathbf{n}$-th coefficient at the i-th direction in the j-th scale and $\sigma^2_{n_{ij}}$ the noise variance in the corresponding subband. We estimate the variances locally using the neighboring coefficients and a maximum likelihood estimator. The noise variance in each subband is inferred using a Monte-Carlo technique where the variances are computed for a few normalized AWGN images and then averaged to stabilize the results. To benchmark the performance of the NSCT we have used the Bivariate shrinkage estimator of [3] which uses a redundant complex wavelet transform and the context adaptive thresholding estimator of [2] which uses the nonsubsampled wavelet transform (NSWT). Table 1 shows the PSNR results obtained for the various methods. The NSCT performs comparably to the BivShrink estimator for the "Lena" image and is superior for the Barbara image. Figure 6 show the denoised "Lena" image obtained with the NSCT and the NSWT using the same denoising routine. Notice that the NSCT exhibits a much better reconstruction of edge features thus attesting the effectiveness of the proposed transform. Further results and comparisons can be found in [7].



(a) (b)

Fig. 6. Image Denoising with the NSCT. (a) Denoised "Lena" image using the NSWT, PSNR = 31.92dB. (b) Denoised with NSCT, PSNR=32.43dB.

5. CONCLUSION

In this paper we have addressed the filter design problem of the NSCT and applied the construction in image denoising. The design methodology proposed also allows for a fast implementation through lifting factorization which halves the number of operations when compared to the direct form. The proposed transform structure with the new designed filters has proven very efficient in noise removal surpassing the undecimated discrete wavelet transform and being comparable (sometimes superior) to other state-ofthe- art denoising methods.

6. REFERENCES

[1] R. R. Coifman and D. L. Donoho, "Translation invariant de-noising," in *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim, Eds. New York: Springer-Verlag, 1995, pp. 125–150.

[2] S. G. Chang, B. Yu, and M. Vetterli, "Spatially adaptive wavelet thresholding with context modeling for image denoising," *IEEE Trans. Img. Proc.*, vol. 9, no. 9, pp. 1522–1531, September 2000.

[3] L. Sendur and I. W. Selesnick, "Bivariate shrinkage with local variance estimation," *IEEE Signal Processing Letters*, vol. 9, no. 12, pp. 438–441, December 2002.

[4] M. N. Do and M. Vetterli, "The contourlet transform: An efficient directional multiresolution image representation," *IEEE Trans. Img. Processing, to appear*, 2005.

[5] R. H. Bamberger and M. J. T. Smith, "A filter bank for the directional decomposition of images: Theory and design," *IEEE Trans. Signal Processing*, vol. 40, no. 4, pp. 882–893, April 1992.

[6] M. J. Shensa, "The discrete wavelet transform: Wedding the à trous and Mallat algorithms." *IEEE Trans. Signal Proc.*, vol. 40, no. 10, pp. 2464–2482, October 1992.

[7] A. L. Cunha, J. Zhou, and M. N. Do, "The nonsubsampled contourlet transform: Theory, design, and applications," *IEEE Trans. Img. Proc., submitted*, 2005.

[8] Z. Cvetkovic and M. Vetterli, "Oversampled filter banks," *IEEE Transactions on Signal Processing*, vol. 46, no. 5, pp. 1245–1255, May 1998.

[9] R. E. Blahut, *Fast Algorithms for Digital Signal Processing*. Addison-Wesley, 1985.