

Inpainting with 3D sparse transforms

Arnaud WOISSELLE^{1,2}, Jean-Luc STARCK¹, Jalal FADILI³

¹CEA, IRFU, SEDI-Service d'Astrophysique
F-91191 GIF-Sur-YVETTE, France.

²Sagem, groupe SAFRAN
95101 Argenteuil CEDEX, France

³GREYC CNRS UMR 6072, Image Processing Group
ENSICAEN 14050 Caen Cedex, France.

Arnaud.Woiselle@cea.fr, jstarck@cea.fr
Jalal.Fadili@greyc.ensicaen.fr

Résumé – Nous présentons deux nouvelles transformées parcimonieuses en 3D, qui sont les deux extensions 3D des curvelets 2D première génération. Ces transformées ont des atomes qui ressemblent à des plans ou des filaments, et permettent de bien représenter les éléments de co-dimension un et deux dans un volume 3D. Nous appliquons ces transformées du dbruitage et de l'inpainting de données 3D et 2D+temps (vidéo).

Abstract – We present two new sparse 3D transforms, which are the two 3D extensions to the first generation 2D curvelets. These transforms have plane- and filament-like atoms, and aim at representing elements of co-dimension one or two in a 3D volume. We show the applicability of these transforms to 3D data and 2D video inpainting and denoising.

1 Introduction

Since the beginning of the multiscale approach in image processing, an important gap has been crossed with the arrival of directional transforms and the curvelet transform, as their mainly isotropic wavelet predecessors failed to represent the curvilinear edges in images. In fact, the curvelet transform provides atoms which are well localized in position, scale and orientation. This tool is now well known and used in many applications such as denoising [5], inpainting [4] or deconvolution [6].

With the increasing capabilities of computers, it becomes feasible to analyze the 3D data we get as a volume and not only as slices. In order to deal with this data, we need to develop tools, among which the multiscale ones like the wavelets, and to design new directional transforms.

In the second section, we present two new 3D transforms, as extensions of the 2D first generation curvelet transform (of which we skip the explanation for this resume). In the third one, we show that they can be powerful tools in applications like denoising and inpainting on 3D data, whether it is real 3D space or video (2D space plus time).

2 3D extensions to the 2D First generation curvelet transform

2.1 The RidCurvelet transform

The first generation 2D curvelet transform is built from the Radon transform, which aims at representing lines in an image. It is from this core idea that we extend the overall construction to build the 3D Curvelet transforms. There are two ways of extending the Radon projection in three dimensions, which lead to the two transforms described below. The first one is obtained by projecting on a line (3D Radon transform), which leads to the Rid-Curvelets, and the second by projecting *only* on planes (3D partial Radon transform), which leads to the Beam-Curvelets. The first extension of the first generation curvelet transform in 3D is accomplished by using the 3D ridgelet transform [2], a transform based on the 3D Radon transform. A three-dimensional ridge function is given by :

$$\psi_{s,k,\theta_1,\theta_2}(x_1, x_2, x_3) = s^{-1/2}. \quad (1)$$

$$\psi\left(\frac{1}{s}(x_1 \cos \theta_1 \cos \theta_2 + x_2 \sin \theta_1 \cos \theta_2 + x_3 \sin \theta_2 - k)\right),$$

As we can see, a ridge function is parametrized by a scale parameter s , a position k and a direction (θ_1, θ_2) , which is a set of two angles in 3D. The transform consists in summing the cube over planes at every direction and position. For a fixed direction (θ_1, θ_2) , the summation gives us a line. Each point on this line represents a plane in

the original cube. It is a projection of the cube on a line passing through the origin at a given angle. Then we apply a mono-dimensional wavelet transform on each of these lines to obtain the ridgelet transform. The 3D ridge function obtained is useful for representing planes in a 3D space. This 3D Ridgelet is applied blockwise (possibly with overlapping windows) to the scales of an isotropic wavelet transform in order to achieve the Rid-Curvelet transform. The global scheme in real space is illustrated on figure 1. An atom is shown on figure 3a.

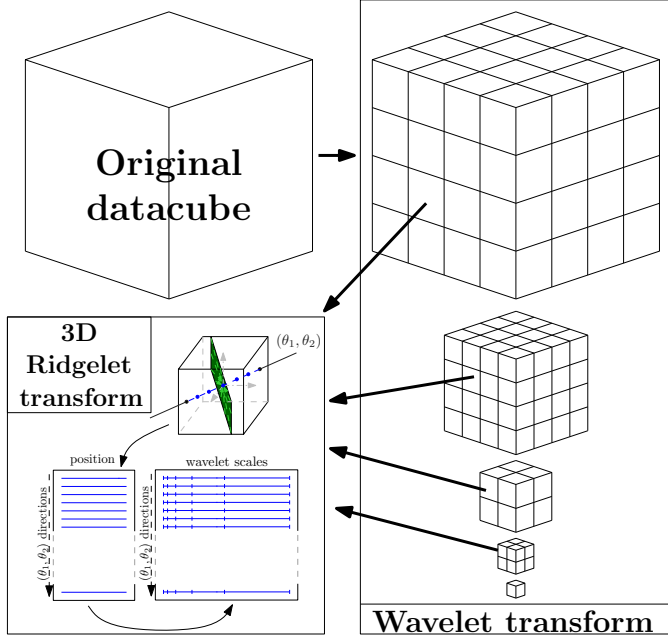


FIG. 1: Global flow graph of a 3D RidCurvelet transform.

2.2 The BeamCurvelet transform

The other extension of the curvelet in 3D is done by using the 3D Beamlet transform [3] instead of the ridgelets. A three-dimensional beam function is given by :

$$\begin{aligned} \psi_{s,k_1,k_2,\theta_1,\theta_2}(x_1,x_2,x_3) &= s^{-1/2}\psi(\\ &(-x_1 \sin \theta_1 + x_2 \cos \theta_1 + k_1)/s, \\ &(x_1 \cos \theta_1 \cos \theta_2 + x_2 \sin \theta_1 \cos \theta_2 - x_3 \sin \theta_2 + k_2)/s). \end{aligned} \quad (2)$$

Compared to 3D ridgelets, which sum over planes, the beamlet sums over the lines (θ_1, θ_2) , which gives us a plane for each direction. The transform consists in summing the cube over lines at every direction and position. For a fixed direction (θ_1, θ_2) , the summation gives us a plane. Each point on this plane represents a line in the original cube. We then apply a two dimensional wavelet transform on each plane to obtain the Beamlet transform. This 3D Beamlet is applied blockwise to the scales of an isotropic wavelet transform in order to achieve the BeamCurvelet (BC) transform. The 3D BeamCurvelet functions aim at

representing filaments in a 3D space. The global scheme is illustrated on figure 2. An atom is shown on figure 3b.

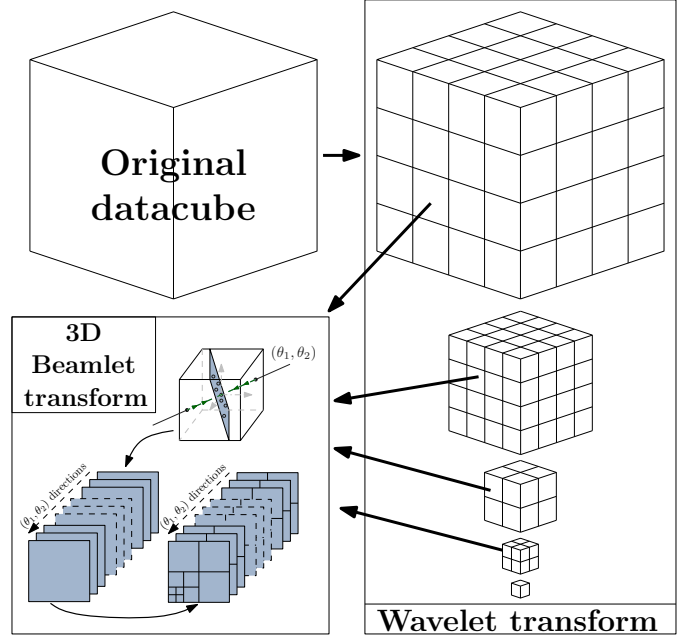


FIG. 2: Global flow graph of a 3D BeamCurvelet (BC) transform.

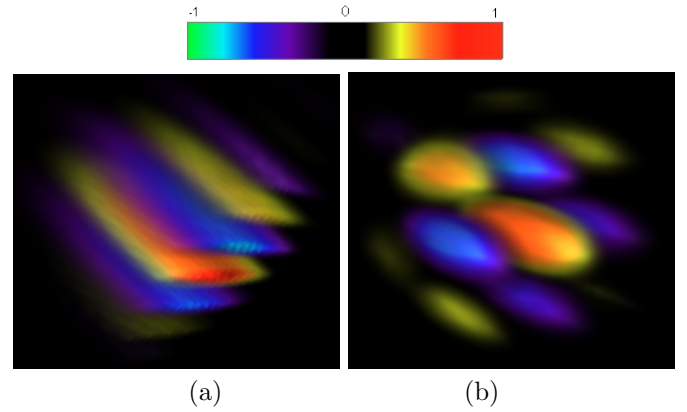


FIG. 3: An example of (a) RidCurvelet and (b) BeamCurvelet atom.

3 Applications

3.1 Denoising

One standard application when using sparse representations is the denoising. It can be done easily using a thresholding operator. The morphological specificity of the BeamCurvelets is interesting when dealing with datasets with strong filamentary structures. An example in spatial 3D density is the λ CDM astrophysical simulation. Figure 4 shows a slice of the perfect cube and the noisy data (addi-

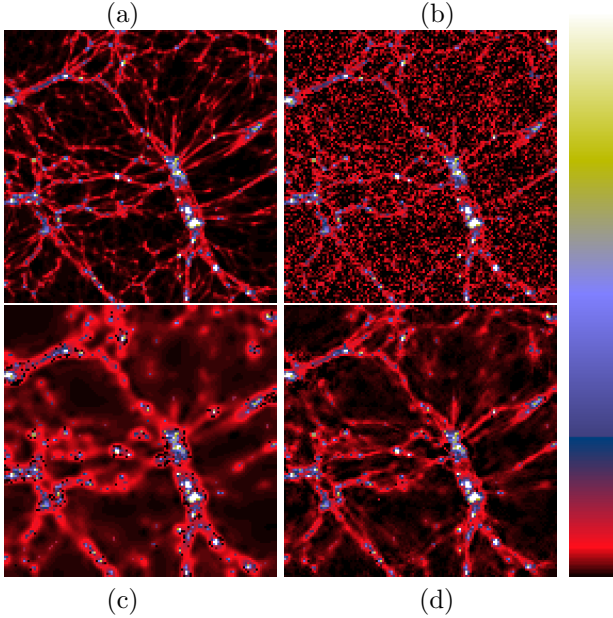


FIG. 4: The central slice of (a) the original λ CDM data cube, (b) the noisy data, (c) the data recovered by Wavelet Thresholding, and (d) recovered by combining the Wavelets and the BeamCurvelets.

Isotropic wavelets	BeamCurvelets & wavelets
76.2	78.6

TABLE 1: PSNR of the filtered λ CDM datacube.

tive gaussian noise). The undecimated wavelet denoising doesn't recover the filaments, while combining it [7] with the new BeamCurvelet transform improves greatly the results, recovering the main filamentary structure. The quantitative PSNRs of the filtered data are found in table 1.

The new transforms perform well on standard video denoising (*mobile*, *silent*, ...), and have PSNR similar to that of the Undecimated Wavelet Transform. In videos containing filaments, like in biomedical imaging, we would expect a significant improvement.

3.2 Inpainting

There are several physical problems that can cause loss of data. In video, we can have pixels with an abnormal behavior, a crashed line, or even a frame lost in transmission or any other problem. In order to simulate these missing pixels, we create a binary mask M with 0 (resp. 1) on missing (known) pixels, with which we multiply the video. Then we apply an inpainting algorithm to recover the video everywhere. The functional we wish to minimize through Alg. 1 is

$$\arg \min_x \sum_k \|\Phi_k^T x\|_0 \text{ s.t. } y = Mx, \quad (3)$$

where Φ and Φ^T are respectively the backward and forward transforms, x our unknown and y the observed data with missing pixels.

The mask we use in our simulations presents a line of dead pixels, 5% random malfunctioning pixels at each frame, three clusters of dead pixels, and a slowly rotating 2-pixel-wide swirl of missing pixels (see figure 5b, in dark red). It also contains three completely lost frames, two of which are consecutive. We apply this mask to the standard video *mobile* found on www.cipr.rpi.edu, and inpaint the data using algorithm 1, similar to the one presented in [4], with either the 3D undecimated tri-orthogonal wavelet transform with CDF 7/9 filters, or the RidCurvelet transform, or both iteratively. The choice of the RidCurvelets against the BeamCurvelets comes from the structure of ordinary (standard) videos, which contain edges, adapted for RidCurvelets, and not 3D filaments as used in the denoising section. Figure 5 shows the central frame of the video, with and without mask, and two frames of the reconstruction using different transforms: the central (normal) frame, and one of the two contiguous missing frames. For the *normal* frames (not entirely missing), we see that the wavelet inpainting reconstructs smoothly the missing areas, while the RidCurvelet one reconstructs the structure but adds some noise and artifacts in the process (see the flowers in the center of the image). Using both transforms leads to a very good compromise, with distinct edges as well as less noise and artifacts. The PSNR of these reconstructions are shown on table 2. As for the missing frames, the RidCurvelets reconstruct it with a lot of details, though a little noisy (which can easily be removed afterwards), while wavelets only get the low frequencies. Using both transforms leads to the best result.

Algorithm 1: The Inpainting Process

Data: A mask M (its complementary \bar{M}), the observed data $x = My$, y being the unknown cube, the number of iterations N , an initial threshold level $\lambda^{(0)}$. K dictionaries Φ_k and the forward transforms associated Φ_k^T , $k \in \llbracket 1, K \rrbracket$

Result: The estimate $x^{(N)}$ of y .

Let $\mathcal{HT}_{\lambda^{(n)}}$ be the hard-thresholding operator with threshold $\lambda^{(n)}$.

begin

```

 $x^{(0)} = y$ 
for  $n = 1$  to  $N$  do
   $\lambda^{(n)} = \lambda^{(0)}(N - 1 - n)/(N - 1)$ 
   $\tilde{x} = x^{n-1}$ 
  for  $k = 1$  to  $K$  do
     $\tilde{x} = \Phi_k \mathcal{HT}_{\lambda^{(n)}} (\Phi_k^T [\tilde{x} + M(y - M\tilde{x})])$ 
     $\tilde{x} = \Phi_k \mathcal{HT}_{\lambda^{(n)}} (\Phi_k^T [y + \bar{M}\tilde{x}])$ 
   $x^{(n)} = \tilde{x}$ 

```

end

	Wavelets	BC	BC & W
Standard frames	30.9	33.6	35.0
Missing frames	22.7	28.3	28.3
Global	28.4	30.4	34.3

TAB. 2: PSNR of the inpainted *mobile* sequence.

4 Conclusion

We presented here two new sparse 3D representations with plane- filament-like atoms, the BeamCurvelet being the only existing transform representing filaments at every scale in a 3D volume. We showed that they can be used to denoise specific data-sets with special morphology where the classical tools are not optimal. We also investigate the inpainting problem in video, and demonstrate the power of this kind of transforms to fill-in random missing pixels as well as clusters or even reconstruct missing frames when conjointly used with wavelets.

References

- [1] E.J. Candès and D.L. Donoho. Curvelets: A surprisingly effective nonadaptive representation for objects with edges. *Stanford University, Dept. of Statistics, Technical Report, (28)*, 1999.
- [2] E.J. Candès and D.L. Donoho. Ridgelets: the key to high dimensional intermittency. *Philosophical Transactions of the Royal Society of London A*, 357:2495–2509, Sep. 1999.
- [3] D.L. Donoho and O. Levi. Fast x-ray and beamlet transforms for three-dimensional data. *Modern Signal Processing, D. Rockmore and D. Healy. Mathematical Science Research Institute Publications*, Cambridge University Press, March 2002.
- [4] M. Elad, J.-L. Starck, P. Querre, and D.L. Donoho. Simultaneous cartoon and texture image inpainting using morphological component analysis. *Computational Harmonic Analysis*, 19:340–358, 2005.
- [5] J.L. Starck, D.L. Donoho, and E. Candès. Very high quality image restoration by combining wavelets and curvelets. In A. Laine, M.A. Unser, and A. Aldroubi, editors, *SPIE conference on Signal and Image Processing: Wavelet Applications in Signal and Image Processing IX, San Diego, 1-4 August*. SPIE, 2001.
- [6] J.L. Starck, M.K. Nguyen, and F. Murtagh. Wavelets and curvelets for image deconvolution: a combined approach. *Signal Processing*, 83:2279–2283, 2003.
- [7] A. Woiselle, J.-L. Starck, J. Fadili. New 3D Data Representations : Applications in Astrophysics. *Computational Harmonic Analysis*, 2009, submitted.

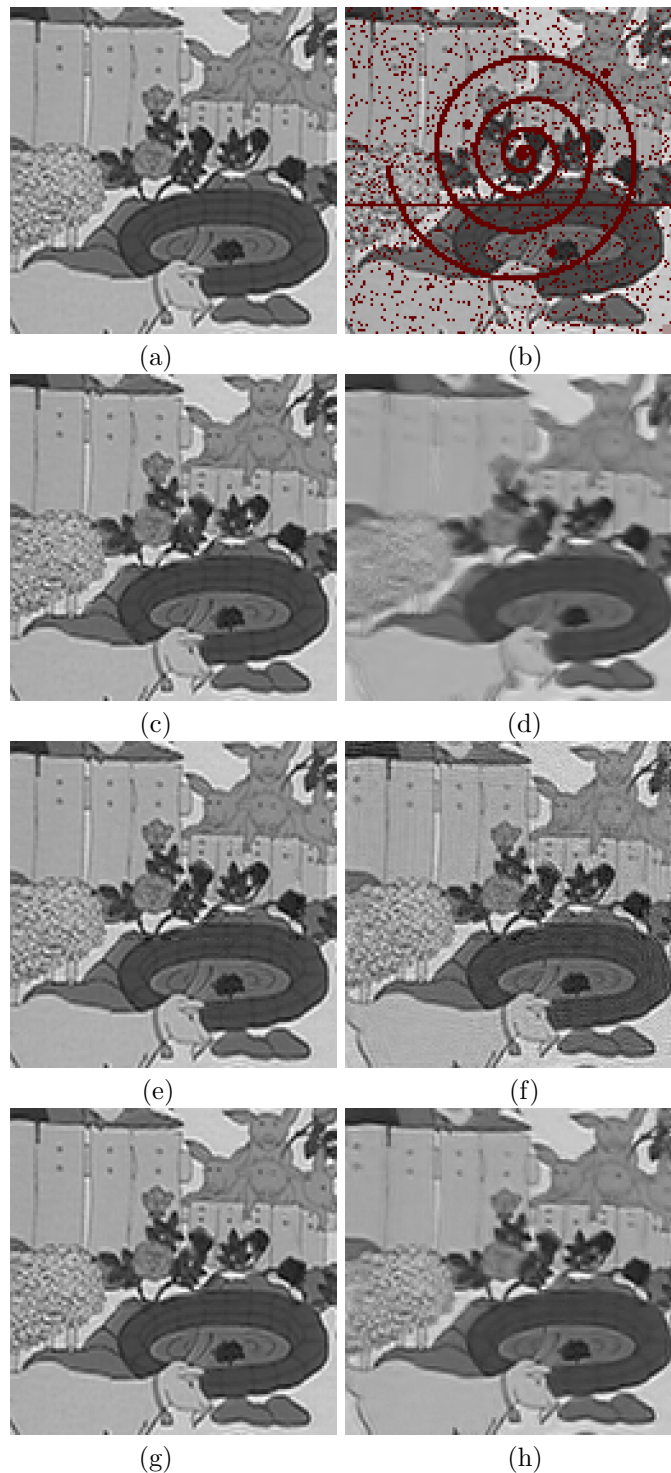


FIG. 5: Central frame of (a,b) the original video without and with the missing pixels mask (in dark red). (c,e,g) Central frame result of inpainting using respectively the Wavelets, the BEamCurvelets or both. (d,f,h) Reconstruction of one of the two consecutive missing frames of the video, in the same order as the left column.