

Débruitage de données de Poisson par BeamCurvelets 3-D

Arnaud WOISELLE¹, Jean-Luc STARCK², Jalal FADILI³

¹Sagem Défense Sécurité, 72-74 rue de la Tour Billy, 95101 Argenteuil CEDEX, France

²CEA, IRFU, SEDI-Service d'Astrophysique, F-91191 GIF-Sur-YVETTE, France.

³GREYC CNRS UMR 6072, Image Processing Group, ENSICAEN 14050 Caen Cedex, France.
arnaud.woiselle@sagem.com, jean-luc.starck@cea.fr, jalal.fadili@greyc.ensicaen.fr

Résumé – Dans cet papier, nous introduisons une façon de stabiliser la variance des transformées décimées en utilisant deux transformées stabilisatrices de variance (VST). Ces VST sont appliquées aux ondelettes de Meyer 3-D qui sont au coeur des curvelets 3-D première génération. Cela nous permet d'étendre ces curvelets 3-D au bruit de Poisson, que nous appliquons ensuite au débruitage d'une simulation cosmologique.

Abstract – In this paper, we introduce a way to stabilize the variance of decimated transforms using two variance stabilizing transforms (VST). These VSTs are applied to the 3-D Meyer wavelet transform which is the core of the first generation 3D curvelets. This allows us to extend these 3-D curvelets to Poisson noise, that we apply to the denoising of a simulated cosmological volume.

1 Introduction

Lors de l'acquisition d'images à faible flux, les données sont corrompues par un bruit de Poisson. Ce phénomène apparait souvent en imagerie médicale avec la microscopie confocale [3], ainsi qu'en astrophysique [4] où l'on cherche à observer des objets toujours plus lointains et plus faibles. Le signal observé $X = (X_i)_i$ est constitué d'un ensemble d'échantillons X_i , réalisations supposées mutuellement indépendantes de paramètres $\lambda_i : X_i \sim \mathcal{P}(\lambda_i)$. La suppression du bruit de poisson dans des données correspond à la recherche de cette densité sous-jacente $(\lambda_i)_i$.

2 La stabilisation de variance multi-échelle

Une transformation stabilisatrice de variance (VST) telle la transformée d'Anscombe [1, 2], transforme des données Poissonniennes en données approximativement stationnaires et gaussiennes. Zhang *et al.* [6] ont étendu ce type de fonctions à la stabilisation de processus filtrés (par un filtre discret h), plus efficaces lorsque le nombre de coups (λ_i) devient très bas (de l'ordre du

photon par pixel, voire moins).

Etant donné un filtre h de moments $\tau_k = \sum_i h[i]$, $k \in [1, 4]$, et un processus de Poisson $X = (X_i)_i$, $X_i \sim \mathcal{P}(\lambda_i)$, nous souhaitons étudier le processus filtré $(Y_j)_j$, $Y_j = \sum_i h[i]X_{j-i}$ obtenu par convolution de X par le filtre h .

La VST proposée est

$$T(Y) = b \cdot \text{sign}(Y + c)|T + c|^{1/2},$$

où b et c sont les paramètres de la transformée, à ajuster au filtre h pour obtenir une convergence optimale vers un bruit gaussien.

En posant $b = \text{sign}(\tau_1)/\sqrt{\tau_1}$ et sous l'hypothèse que le paramètre de Poisson λ est constant sur le support du filtre h , on peut montrer que

$$Y_k \xrightarrow[\lambda \rightarrow \infty]{\mathcal{D}} \mathcal{N}(0, \sigma_h),$$

où σ_h ne dépend que du filtre h , et donc peut être précalculé une fois pour toutes. Pour certains types de filtres h , sa valeur peut être calculée théoriquement, mais dans le cas général, on peut l'estimer facilement à partir de cartes de bruit simulées.

Cette VST a été appliquée aux ondelettes non décimées (UWT), aux ondelettes isotropes non décimées (IUWT), et aux curvelets 2-D première génération.

Ces représentations transforment un signal a_0

en un ensemble de $J + 1$ bandes de coefficients $\{d_1, \dots, d_J, a_J\} = \{\mathcal{W}a_0, a_J\}$, les (d_j) étant les coefficients de détail aux échelles j , et a_J l'approximation basse fréquence. Dans le cas des IUWT, en notant h_j le filtre passe-bas à l'échelle j et T_j la VST associée au filtre global équivalent¹, la transformée en IUWT d'un signal a_0 avec $J + 1$ échelles est

$$\begin{aligned} & \text{IUWT} \begin{cases} a_j = h_{j-1} * a_{j-1} \\ d_j = a_{j-1} - a_j \end{cases} \\ \Rightarrow & \begin{matrix} \text{MS VST} \\ + \\ \text{IUWT} \end{matrix} \begin{cases} a_j = h_{j-1} * a_{j-1} \\ d_j = T_{j-1}(a_{j-1}) - T_j(a_j). \end{cases} \end{aligned}$$

Dans le cas des ondelettes non décimées classiques (UWT), en utilisant les filtres passe-bas et passe-haut h_j et g_j , l'algorithme est le suivant :

$$\begin{aligned} & \text{UWT} \begin{cases} a_j = h_{j-1} * a_{j-1} \\ d_j = g_{j-1} * a_{j-1} \end{cases} \\ \Rightarrow & \begin{matrix} \text{MS VST} \\ + \\ \text{UWT} \end{matrix} \begin{cases} a_j = h_{j-1} * a_{j-1} \\ d_j = g_{j-1} * T_{j-1}(a_{j-1}). \end{cases} \end{aligned}$$

La reconstruction des IUWT est la somme des bandes, et il est facile de voir que dans le cas VST, la reconstruction est faite d'une manière très similaire, menant à $x_{\text{est}} = T_0^{-1} \left(T_J(a_J) + \sum_{j=1}^J d_j \right)$.

Dans le cas des VST+UWT, on ne peut définir de reconstruction directe, et il faut utiliser un algorithme itératif, projetant alternativement le résultat sur l'orthante non négative dans l'espace direct, et sur l'ensemble d'attache aux données défini par :

En notant \mathcal{W} la transformée directe et \mathcal{R} son pseudo-inverse, soit \mathcal{M} l'ensemble des coefficients significatifs du signal a_0 ,

$$\mathcal{M} = \{(i, j) | (\mathcal{W}a_0)_j[i] \text{ is significant}\}.$$

On définit l'ensemble d'attache aux données comme

$$\mathcal{S} = \{d | \forall (i, j) \in \mathcal{M}, d_j[i] = (\mathcal{W}a_0)_j[i]\}.$$

Ces deux projections sont notées $\mathcal{P}_{\mathcal{S}}$ and \mathcal{Q}_+ = $\mathcal{W}\mathcal{P}_+\mathcal{R}$, \mathcal{P}_+ étant la projection sur l'ensemble convexe des signaux positifs. Zhang *et al.* [6] ont proposé l'Algorithme 1 pour obtenir une reconstruction avec ces contraintes et attache aux données. Afin d'obtenir de meilleures performances de reconstruction, notamment en termes de photométrie, cet algorithme est utilisé aussi dans le cas des IUWT.

¹ T_j est associé à $h_j^{\text{global}} = h_j * h_{j-1} * \dots * h_1$, d'où $a_j = h_j^{\text{global}} * a_0$.

Algorithm 1: Algorithme de Reconstruction

Data: a_0 , un nombre d'itérations N .

initialiser $\mathbf{d}^{(0)} = \mathcal{W}\mathcal{P}_+a_0$

for $n = 1$ **to** N **do**

$\hat{\mathbf{d}} = \mathcal{P}_{\mathcal{S}} \circ \mathcal{Q}_+ \mathbf{d}^{(n-1)}$

$\mathbf{d}^{(n)} = \mathcal{S}\mathcal{I}_{\beta_k}(\hat{\mathbf{d}})$, $\mathcal{S}\mathcal{I}_{\beta_k}$ est le seuillage

doux

Result: the estimated density $\mathcal{P}_+\mathcal{R}\mathbf{d}^N$.

3 MS-VST et BeamCurvelets 3-D

La transformée en BeamCurvelets (BC), introduite par Woiselle *et al.* [5] en tant qu'extension des curvelets 2-D première génération est une représentation multi-resolution adaptée aux données filamenteuses. Cette décomposition consiste à appliquer des Beamlet 3-D par blocs sur une transformée en ondelettes isotropes des données. Plus d'informations sur cette représentation sont disponibles dans [5].

A l'image du travail de Zhang *et al.* [6], nous appliquons la VST sur les ondelettes 3-D et laissons le reste de la transformée inchangée à l'exception des coefficients de normalisation. La principale différence avec ses travaux est qu'à notre connaissance, les MS-VST (VST multi-échelles) ne sont définies que sur des transformations non décimées suivant les schémas présentés en section 2, et l'utilisation d'une telle transformée pour les BeamCurvelets n'est pas possible en termes de charge de calcul et d'utilisation mémoire. Il faut utiliser une transformée décimée donc pyramidale. La transformée en ondelettes de Meyer utilisée est un pavage de Fourier en J couronnes dyadiques et une approximation basse fréquence, notées d_1, \dots, d_J, a_J . Soit $h_j : j \in [0, J]$ les filtres passe-bas de cette transformée dont les transformées de Fourier \hat{h}_j ressemblent à des boules L_∞ imbriquées. Les couronnes sont définies comme les complémentaires L_2 de ces boules, $\hat{g}_j = \sqrt{1 - \hat{h}_j}$. Chaque échelle est décimée avec l'opérateur \mathbf{D}_j qui consiste à l'extraction de la partie centrale non nulle en Fourier. Soit \mathbf{U}_j l'opérateur inverse, qui effectue un *zero-padding* de la transformée de Fourier. Cette transformée en ondelettes suit le schéma suivant :

$$a_0 = x, \quad \forall j \in [1, J] : \begin{cases} d_j = g_{j-1} * a_{j-1} \\ a_j = \mathbf{D}_j(h_{j-1} * a_{j-1}), \end{cases}$$

et sa version stabilisée en variance

$$a_0 = x, \quad \forall j \in [1, J] : \begin{cases} d_j = g_{j-1} * T_j(a_{j-1}) \\ a_j = \mathbf{D}_j(h_{j-1} * a_{j-1}). \end{cases}$$

Ce processus n'étant pas inversible, il faut utiliser l'Algorithme 1 pour obtenir une estimée de la densité avec cette représentation.

Pour contrer l'absence d'opérateur de reconstruction, on peut changer la transformée pyramidale, en utilisant une représentation proche des IUWT en 2-D, mais décimée. On définit ainsi les filtres $\hat{\mathbf{h}}_i = \hat{h}_i^2$ et $\hat{\mathbf{g}}_i = \hat{g}_i^2$, qui satisfont la condition d'admissibilité $\hat{\mathbf{h}}_i \hat{\mathbf{h}}_i + \hat{\mathbf{g}}_i \hat{\mathbf{g}}_i = 1$ avec les filtres de reconstruction $\hat{\mathbf{h}}_i$ et $\hat{\mathbf{h}}_i$ identité, donnant une reconstruction par sommation. La transformée en ondelettes devient alors

$$a_0 = x, \quad \forall j \in [1, J] : \begin{cases} \tilde{a}_j = \mathbf{h}_{j-1} * a_{j-1} \\ d_j = a_{j-1} - \tilde{a}_j \\ a_{j+1} = \mathbf{D}_j \tilde{a}_j, \end{cases}$$

la reconstruction étant

$$\forall j \in [1, J] : \begin{cases} \tilde{a}_j = \mathbf{U}_j a_j \\ a_{j-1} = d_j + \tilde{a}_j. \end{cases}, x_{\text{est}} = a_0.$$

Cette autre construction permet d'effectuer une VST :

$$a_0 = x, \quad \forall j \in [1, J] : \begin{cases} \tilde{a}_j = \mathbf{h}_{j-1} * a_{j-1} \\ d_j = T_{j-1}^D(a_{j-1}) - T_j^U(\tilde{a}_j) \\ a_{j+1} = \mathbf{D}_j \tilde{a}_j, \end{cases}$$

où l'on peut noter l'utilisation de deux familles de transformées stabilisatrices T^D and T^U , qui ont des paramètres différents et sont respectivement appliquées aux familles décimées et non décimées des images d'approximation $(a_j)_j$ et $(\tilde{a}_j)_j$. La reconstruction est obtenue par un algorithme légèrement plus complexe que la somme, la décimation et les deux VST devant être prises en compte :

$$\forall j \in [1, J] : \begin{cases} \tilde{a}_j = \mathbf{U}_j a_j \\ a_{j-1} = (T_{j-1}^D)^{-1}(d_j + T_j^U(\tilde{a}_j)). \end{cases}$$

Ce changement dans la transformée en ondelettes mène à des atomes d'analyse très similaires, mais donne une reconstruction directe. Cette représentation peut aussi être utilisée en analyse uniquement, et la reconstruction obtenue par l'algorithme itératif présenté plus haut.

4 Applications to astrophysical data

Les données 3-D issues des simulations de matière noire à très grande échelle contiennent principalement des amas très denses et des filaments beaucoup plus faibles. Ces données étant obtenues

par des codes physiques d'interaction à N corps, le bruit de Poisson y est présent, et un débruitage de ces données permet d'accélérer les temps de calcul. Par ailleurs, les données réelles (relevés de galaxies) suivent naturellement une distribution de Poisson, la matière visible étant interprétable comme une réalisation de la densité de matière noire recherchée. Nous avons donc utilisé la transformée présentée pour débruiter une simulation de matière noire, et les résultats sont présentés dans la Figure 1. Les méthodes utilisant les BC restaurent naturellement mieux les filaments, et l'adaptation des BC au bruit de Poisson améliore les résultats que l'on obtient en stabilisant directement les données.

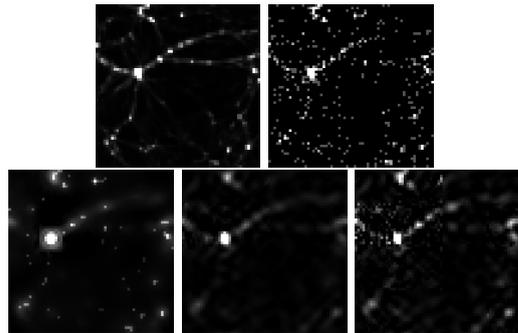


FIG. 1: Dans un ordre de lecture : densité originale, réalisation de Poisson, débruitages utilisant : IUWT3D, Anscombe+BC gaussiennes, BC Poisson.

5 Conclusion

Nous avons présenté une nouvelle construction pour les ondelettes pyramidales utilisées dans les curvelets 3-D première génération (BC), et avons montré comment les stabiliser en variance à l'aide de deux VST. Cette représentation filamentaire adaptée au bruit de Poisson donne de bons résultats en débruitage de simulations astrophysiques. Ces outils seront appliqués par la suite à des relevés réels de galaxies, et les mesures de performances telles le PSNR ou le MAE n'étant pas forcément des indicateurs pertinents pour la mesure de la restauration des structures, nous envisageons une analyse du genus ou des fonctions de Minkowski dans le futur.

References

- [1] F.J. Anscombe. The transformation of poisson, binomial and negative-binomial data.

Biometrika, 35:246–254, 1948.

- [2] D.L. Donoho. Nonlinear wavelet methods for recovery of signals, densities and spectra from indirect and noisy data. *Proc. Symp. Applied Mathematics: Different Perspectives on Wavelets*, 47:173–205, 1993.
- [3] J. Pawley. *Handbook of Biological Confocal Microscopy*, 3rd ed. Springer, 2006.
- [4] J.-L. Starck, F. Murtagh, and A. Bijaoui. *Image Processing and Data Analysis: The Multiscale Approach*. Cambridge University Press, 1998.
- [5] A. Woiselle, J.L. Starck, and M.J. Fadili. 3D curvelet transforms and astronomical data restoration. *Applied and Computational Harmonic Analysis*, 28(2):171–188, March 2010.
- [6] B. Zhang, M.J. Fadili, and J.L. Starck. Wavelets, ridgelets, and curvelets for poisson noise removal. *IEEE Transactions on Image Processing*, 17(7):1093–1108, Jul. 2008.