HeNLM-3D: A 3D COMPUTER TOMOGRAPHY IMAGE DENOISING ALGORITHM*

N. Mamaev, A. Lukin, D. Yurin
Laboratory of Mathematical Methods of Image Processing,
Faculty of Computational Mathematics and Cybernetics,
Lomonosov Moscow State University, Moscow, Russia
http://imaging.cs.msu.su
e-mails: nikolay.mamaev93@mail.ru, lukin@ixbt.com, yurin@cs.msu.su

A 3D filtering algorithm for CT images is proposed (HeNLM-3D). It is an extension of the known Local Jets method (LJNLM-LR) and our previous 2D algorithm (HeNLM). The proposed algorithm is based on the expansion of pixel neighborhoods into Hermite functions which form the orthonormal system. Tests on real CT images have shown that HeNLM-3D performs filtering better in high-detailed areas in comparison with LJNLM-LR and classical NLM method.

Introduction

Computed tomography (CT) is a method for studying the internal structure of the body by using a Radon transform [1]. Noise is always present in obtained CT images, and it becomes higher when the X-ray radiation dose is reduced. So, the effective noise reduction in CT images allows reducing the radiation dose [2][3]. Noise in CT images is close to Gaussian [4]. CT images comprise the series of slices which forms a 3D image. It is important to note that these slices are a feature of the registration procedure, not of the object itself. For example, a blood vessel is not required to lie in a plane of the slice or to be perpendicular to it. Obviously, in this case the preferred direction of filtering should be the local direction of the vessel, not a slice plane or a perpendicular to the slice. Therefore, fully 3D algorithms – isotropic or with locally adaptive anisotropy – are needed.

Today some of the most effective image denoising algorithms are those in which each pixel of the output image is formed as a weighted sum of pixels of the source image. At the same time the weights depend on the similarity of whole pixels neighborhoods (patches) [5][6][7]. In a non-local means (NLM) algorithm [5] weights depend on the Euclidian distance between patches of pixels. LJNLM-LR [6] and GFNLM [7] algorithms are the extension of [5] and weights depend on the Euclidian distance between the feature vectors which characterize the patches. In [6] components of the feature vector are values of Taylor series expansion coefficients, which, in turn, are the values of image convolution with derivatives of the Gaussian function. One of advantages of [6] is the invariance of features to rotation. In [7] features are based on Gabor functions.

In this article a 3D method for noise reduction is proposed based on Hermite functions expansion which forms the orthonormal system. Inheriting all advantages of [6] the HeNLN-3D method is fully three-dimensional. It can better distinguish textures due to higher independence of feature vector components and better description of high-frequency components of the pixel neighborhood.

1. Hermite Functions

Hermite functions of order n are defined as

$$\psi_n(x) = \frac{1}{c_n} e^{\frac{x^2}{2}} \frac{d^n(e^{-x^2})}{dx^n} = \frac{(-1)^n}{c_n} H_n(x) e^{-\frac{x^2}{2}}, \text{ where } c_n = \sqrt{\sqrt{\pi}\, 2^n n!}, \tag{1}$$

$$H_n(x) = (-1)^n \frac{d^n e^{-x^2}}{dx^n} e^{x^2} \text{ is a Hermite polynomial}$$

* The research was supported by the RFBR grant 13-07-00584.

They form the orthonormal system in $L_2(-\infty,+\infty)$ [8]:

$$\int_{-\infty}^{+\infty} \psi_n(x)\psi_m(x)dx = \delta_{nm} \tag{2}$$

Multiscale Hermite functions are defined as follows:

$$\psi_n^\sigma(x) = \frac{1}{\sigma}\psi_n(\frac{x}{\sigma}) \tag{3}$$

The multiplier $1/\sigma$ in (3) is introduced for similarity of the filter response to similar patterns across different scales.

Some of the first Hermite functions and derivatives of the Gaussian function are shown in the Fig. 1. We can see that unlike Gaussians, Hermite functions have approximately the same overall amplitude and higher values at the sides of their support interval.

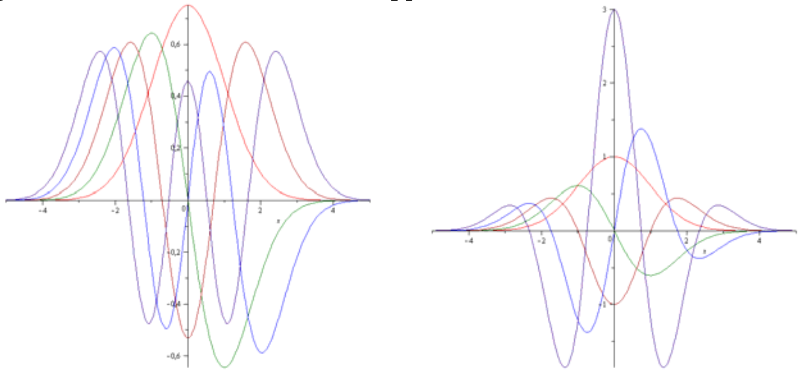


Fig. 1. Hermite functions (on the left) and derivatives of the Gaussian function (on the right).

We define 3D Hermite functions as

$$\psi_{nmk}^\sigma(x,y,z) = \psi_n^\sigma(x)\psi_m^\sigma(y)\psi_k^\sigma(z) \tag{4}$$

2. 3D Hermite Functions based Non-Local Means

In [9] we have proposed a modification of LJNLM-LR algorithm [6] based on Hermite functions instead of derivatives of the Gaussian function. In this article, we propose a 3D extension of [9] (HeNLM-3D), i.e. our pixel neighborhood is now a 3D cube and elements of the feature vector are values of a convolution of the source image series with 3D Hermite functions:

$$h_{nmk}^\sigma(x,y,z) = I(x,y,z) * \left(\frac{1}{N}\psi_{nmk}^\sigma(x,y,z)\right), \tag{5}$$

where N is the number of different Hermite functions of $n+m+k$ order.

As well as in [6], the obtained features are transformed to the new coordinate system (see (6)) (ξ_1,ξ_2,ξ_3) for invariance to rotation. In [6] and [9] the new 2D coordinate system was chosen as a gradient direction and the normal to it. In the case of a 3D coordinate system, two directions have to be chosen. Therefore, the transfer of rotation formulas from methods [6], [9] is impossible. In this article we propose using the eigenvectors of a structure tensor [10] as a new coordinate system:

$$M = \left\langle g^\sigma \cdot g^{\sigma^T} \right\rangle, \text{ where } g^\sigma = (g_x^\sigma, g_y^\sigma, g_z^\sigma)^T \text{ is the image gradient} \tag{6}$$

Averaging in (6) means averaging of matrix elements by a Gaussian filter with $\sigma_M = 3\sigma$ which is 3 times higher than σ used for the derivatives computation. It is important to note that, for example, in a small neighborhood of the vessel with a radius on the order of σ, the eigenvector corresponding to the minimal eigenvalue is directed along the vessel axis which has a favorable effect on image filtering.

To obtain an explicit expression for components of the feature vector in the new coordinate system, we consider two coordinate systems: the original one (x_1, x_2, x_3) related to rows, columns and slices of the 3D image, and the rotated coordinate system (ξ_1, ξ_2, ξ_3):

$$\begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix} = R \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, \; R = \begin{pmatrix} \mathrm{a}_{11} & \mathrm{a}_{12} & \mathrm{a}_{13} \\ \mathrm{a}_{21} & \mathrm{a}_{22} & \mathrm{a}_{23} \\ \mathrm{a}_{31} & \mathrm{a}_{32} & \mathrm{a}_{33} \end{pmatrix}, \; \mathrm{a}_{\mathrm{ij}} = (\vec{x}_i, \vec{\xi}_j), \text{ where } i, j = 1,2,3, \tag{7}$$

$\vec{x}_i, \vec{\xi}_j$ are basis unit vectors of basis $(x_1, x_2, x_3) \equiv (x, y, z)$ and (ξ_1, ξ_2, ξ_3) respectively

Then the differential operator in this new coordinate system can be expressed through the linear combination of operators in the old coordinate system:

$$\frac{d}{d\xi_i} = \sum_{j=1,2,3} R_{ij} \frac{d}{dx_j} \tag{8}$$

Therefore the expression for the Gaussian function derivatives can be written as

$$\frac{dG_\sigma}{d\xi_1^n d\xi_2^m d\xi_3^k} = \left(\sum_{j=1}^{3} R_{1j} \frac{d}{dx_j} \right)^n \left(\sum_{j=1}^{3} R_{2j} \frac{d}{dx_j} \right)^m \left(\sum_{j=1}^{3} R_{3j} \frac{d}{dx_j} \right)^k G_\sigma =$$

$$= \left(\sum_{i=0}^{n} \sum_{j=0}^{i} C_{i-j,j}^{n} \left(R_{11} \frac{d}{dx_1} \right)^{n-i} \left(R_{12} \frac{d}{dx_2} \right)^{i-j} \left(R_{13} \frac{d}{dx_3} \right)^{j} \right) \cdot$$

$$\left(\sum_{i=0}^{m} \sum_{j=0}^{i} C_{i-j,j}^{m} \left(R_{21} \frac{d}{dx_1} \right)^{m-i} \left(R_{22} \frac{d}{dx_2} \right)^{i-j} \left(R_{23} \frac{d}{dx_3} \right)^{j} \right) \cdot \tag{9}$$

$$\cdot \left(\sum_{i=0}^{k} \sum_{j=0}^{i} C_{i-j,j}^{k} \left(R_{31} \frac{d}{dx_1} \right)^{k-i} \left(R_{32} \frac{d}{dx_2} \right)^{i-j} \left(R_{33} \frac{d}{dx_3} \right)^{j} \right) G_\sigma$$

$$\text{where } C_{pq}^{n} = \frac{n!}{p!q!(n-p-q)!}$$

A 3D Hermite function is a product of the Gaussian function derivative, the multiplier $\exp\left((x^2 + y^2 + z^2)/2\right)$ and the coefficient c_i. Therefore, from (7) and the radial symmetry of the exponential function it follows that

$$c_n c_m c_k \tilde{\psi}_{nmk}^{\sigma} = \sum_{i=0}^{r} \sum_{j=0}^{i} \alpha_{r-i,i-j,j} c_{r-i} c_{i-j} c_j \psi_{r-i,i-j,j}^{\sigma}, \text{ where } r = n+m+k$$

$$\alpha_{abc} = \sum_{i=\max\{0,a-n\}}^{\min\{a,m+k\}} \; \sum_{j=\max\{0,a-n-m\}}^{\min\{i,k\}} \; \sum_{\substack{p=\max\{0,b-n,\\ a-n-i-b\}}}^{\min\{b,m+k\}} \; \sum_{\substack{q=\max\{0,b-n-m,\\ p-m+i-j\}}}^{\min\{p,k-j\}} R_{11}^{a-i} R_{21}^{i-j} R_{31}^{j} R_{12}^{b-p} \cdot \tag{10}$$

$$\cdot R_{22}^{p-q} R_{32}^{q} R_{13}^{n-a+i-b+p} R_{23}^{m-i+j-p+q} R_{33}^{k-j-q} C_{b-p,n-a+i-b+p}^{n} C_{p-q,m-i+j-p+q}^{m} C_{q,k-j-q}^{k}$$

In (10) $\tilde{\psi}_{nmk}^{\sigma}$ is the Hermite function in a new coordinate system. We note that formula (10) suits for Gaussian function derivatives by setting coefficients $c_i = 1$.

Finally, the feature vector is determined as follows:

$$\vec{\mathbf{v}} = \left\{ \tilde{h}_{nmk}^{\sigma}; n + m + k \le r, \sigma \in S \right\} \tag{11}$$

Here r is the maximal order of the Hermite function, and S is a set of different scales. And the value of the output pixel (voxel) $f(\vec{\mathbf{p}})$ is the weighed sum of source image pixels (voxels) $I(\vec{\boldsymbol{p}})$ from the neighborhood Q:

$$f(\vec{\boldsymbol{p}}) = \frac{1}{\sum_{\vec{\xi} \in Q} w} \sum_{\vec{\xi} \in Q} w(\vec{\boldsymbol{p}}, \vec{\xi}) I(\vec{\boldsymbol{p}} + \vec{\xi}) \tag{12}$$

The weights depend on the similarity of feature vectors (11):

$$w(\vec{\boldsymbol{p}}, \vec{\xi}) = \exp\left(- \frac{\left\| \vec{\boldsymbol{v}}(\vec{\boldsymbol{p}}) - \vec{\boldsymbol{v}}(\vec{\boldsymbol{p}} + \vec{\xi}) \right\|_2^2}{2\rho^2} \right) \tag{13}$$

The use of Hermite functions instead of derivatives of the Gaussian function provides better characterization the pixel neighborhood because their orthogonality means less dependence between feature vector components and, accordingly, their greater significance. In addition, the localization region is expanded (see Fig. 1) and the peripheral data of a local neighborhood is better taken into account.

3. Results

Fig. 2 shows the source CT image, and Fig. 3 shows the enlarged fragment of it and the results of filtering by two methods: LJNLM-LR and HeNLM-3D. The size of a neighborhood for LJNLM-LR method has been set to 21×21 pixels. Because in our CT images the physical size of a voxel along x and y axes is about 2.1 times less than its size along z axis, we set the size of a neighborhood for the HeNLM-3D method to 21×21×9 voxels. For the same reason, the parameter σ for Gaussian and Hermite functions has been set to 2.1 for x and y axes and 1.0 for z axis. The maximal order of Hermite functions and Gaussian function derivatives has been set to 4. The parameter ρ for all methods has been manually adjusted, so that PSNR between noisy and filtered images would be same for all methods in comparison.

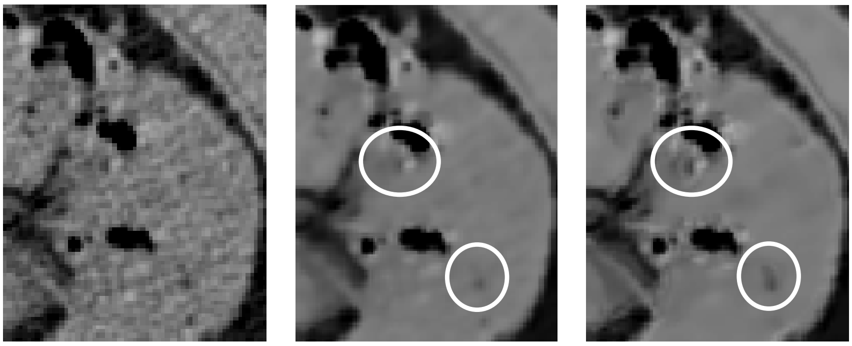
Fig. 3 shows that HeNLM-3D method performs filtration better in fragments with small details (shown by ellipses in Fig. 3b and 3c), and keeps some details which LJNLM-LR method blurs.

Conclusion

A three-dimensional method for CT image filtering is proposed. A test on real images shows that HeNLM-3D method performs filtering better than LJNLM-LR algorithm in areas that contain small details.

The research was supported by the RFBR grant 13-07-00584.

Fig. 2. The original CT image. A white border shows the fragment enlarged in Fig. 3.



a) *b)* *c)*

Fig. 3. The original CT image fragment (a) and the results of filtering by LJNLM-LR (b) and HeNLM-3D (c) algorithms.

References

1. Natterer, F. The Mathematics of Computerized Tomography. // SIAM: Society for Industrial and Applied Mathematics. – 2001. – P. 222.

2. Can Noise Reduction Filters Improve Low-Radiation-Dose Chest CT Images? Pilot study. / M.K. Kalra [et al.] // Radiology. – 2003. – Vol. 228. – P. 257–264.

3. 2.5D Extension of Neighborhood Filters for Noise Reduction in 3D Medical CT Images. / M.V. Storozhilova [et al.] // Lecture Notes in Computer Science. – 2013. – Vol. 7870. – P. 1–16.

4. Kijewski, M.F. The Noise Power Spectrum of CT Images. / M.F. Kijewski, P.F. Judy // Phys. Med. Biol. – 1987. – Vol. 32. – No 5 – P. 565–575.

5. Buades, A. A Non-Local Algorithm for Image Denoising. // IEEE Computer Society Conference on Computer Vision and Pattern Recognition. – 2005. – Vol. 2. – P. 60–65.

6. Manzanera, A. Local Jet Based Similarity for NL-Means Filtering. // 20th International Conference on Computer Vision and Pattern Recognition (ICPR). – 2010. –P. 2668–2671.

7. Gabor Feature Based Nonlocal Means Filter for Textured Image Denoising. / S. Wang [et al.] // Journal of Visual Communication and Image Representation. – 2012. – Vol. 23. – P. 1008–1018.

8. Abramowitz, M. Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. / M. Abramowitz, I.A. Stegun // U.S. National Bureau of Standards. – 1965.

9. Hermite Functions Expansion Based Non-Local Means Algorithm for CT Applications. / N. Mamaev [et al.] // Proc. of The 11-th International Conference “Pattern Recognition and Image Analysis: New Information Technologies”. – 2013. – Vol. 2. – P. 638–641.

10. Knutsson, H. Representing Local Structure Using Tensors. // Proceedings 6th Scandinavian Conf. on Image Analysis – 1989. – P. 244–251.