Multifrequential Algorithm for Fast 3D Reconstruction

Thomas Rodet, Pierre Grangeat and Laurent Desbat

Abstract— **Some recent medical imaging applications, such as functional imaging (PET and SPECT) or interventional imaging (CT fluoroscopy) involves dynamic data. The image reconstruction time must be reduced. For that purpose, we developed a new fast algorithm for dynamic reconstruction based on frequential hierarchical reconstruction.**

Our algorithm performs an indirect subband decomposition of the image f to be reconstructed ($f = \sum f_j$) through the filtering of the projection Rf. The subband images f_j can be reconstructed on an undersampled grid without information suppression. In order to reduce the computation time, we undersample the number of projections and we choose them in accordance with the undersampled grid. But image compression can also be made directly in our algorithm by elimination of some frequential components with low information content.

Keywords— **Fast reconstruction algorithm, frequential decomposition, angular undersampling, computation compression.**

I. Introduction

CT acquisition systems involve increasing amounts of data. Moreover, for many medical applications, the computation time must be lowered. There are even some applications for which real time is required. Two main fields of medical imaging need dynamic reconstructions, namely functional imaging (PET and SPECT) and interventional imaging (CT fluoroscopy, 3D guidance) [1], [2], [3]. Fast tomographic reconstruction is currently a very active research domain [4], [5], [6].

In this paper, we present a fast algorithm for dynamic reconstruction. In our reconstruction algorithm, we implement the two following ideas to speed-up the reconstruction time: the first is to reconstruct some frequentials components with reduced number of projections, the second is to achieve computation compression. The frequential reconstruction allows to reconstruct each frequential component on an undersampled grid, and to use only a lower number of undersampled projections. Our algorithm associates this undersampling approach with computation compression (see [7]). The computation compression is based on the data compression principle. Our algorithm can generate directly compressed data.

In the first section, we introduce our notations. In the second section, we present the frequential reconstruction principle. Then, we expose the speed up factors of our algorithm: the angular undersampling, the computation compression and the block processing. The last section shows numerical applications of our algorithm on both a static 3D phantom and a dynamic 2D phantom.

T.Rodet and L.Desbat are with the TIMC-IMAG, UMR CNRS 5525, IAB, Faculte de medecine, UJF 38706 La Tronche Cedex, FRANCE. E-mail:Laurent.Desbat@imag.fr

P.Grangeat are with the Laboratoire d'Electronique et de Technologie de l'Information (LETI), Departement Systemes pour l'Information et la Sante (DSIS), Direction des Recherches Technologique (DRT), Commissariat a l'Energie Atomique (CEA), 17 Rue des Martyrs, F 38054 GRENOBLE Cedex 9, FRANCE. E-mail: Thomas.Rodet@cea.fr, Pierre.Grangeat@cea.fr

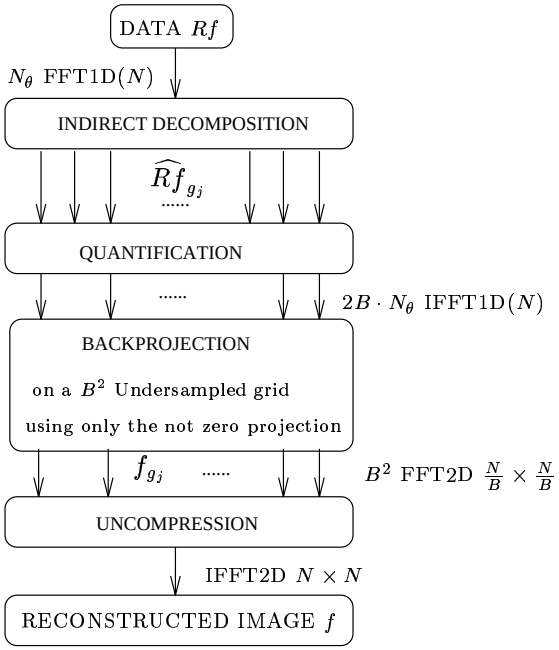

Fig. 1. *Our algorithm*

II. Notations

We define the Fourier transform $\hat{f}$ of a function $f \in L^1(\mathbb{R}^2)$ by:

$$\hat{f}(\nu) = \frac{1}{2\pi} \int_{\mathbb{R}^2} f(t) e^{-i\nu.t} dt$$

Let $s \in \mathbb{R}$, and $\theta \in S^1$, where S^1 is the unit circle, we denote the Radon transform R by:

$$Rf(\theta, s) = \int_{y \in \theta^\perp} f(s\theta + y) dy.$$

We denote by $R^\sharp$ the backprojection operator and by I the ramp filter. We define the functions $g_j \in \mathcal{S}(\mathbb{R}^2)$ as the inverse Fourier transform of adjacent band indicators such that $\sum_{j=1}^{n} \hat{g_j}$ is the indicator function of a domain covering the essential support of the function $\hat{f}$. Then, we denote each elementary frequential component of f: $f_{g_1} = f * g_1, \ldots, f_{g_n} = f * g_n$.

III. Reconstruction algorithm using a decomposition step

The main idea of our approach is to reconstruct B^2 frequential component f_{g_j} of f independently using $f_{g_j} = f * g_j = R^\sharp I(Rf * Rg_j)$ (see [7]). This allows us to undersample the number of projection and to compress the computation like in data compression.
We want to reconstruct B^2 frequential components of f without knowing the image f, but only its projections Rf. Also we must do an indirect decomposition of f through Rf. The projection slice theorem (see [8]) yields a direct relation between $\hat{f}$ and $\widehat{Rf}$. Thus we can find a filter Rg_j to apply to Rf which leads to the decomposition of f into the sum of f_{g_j}(for a theoretical justification see [7]).

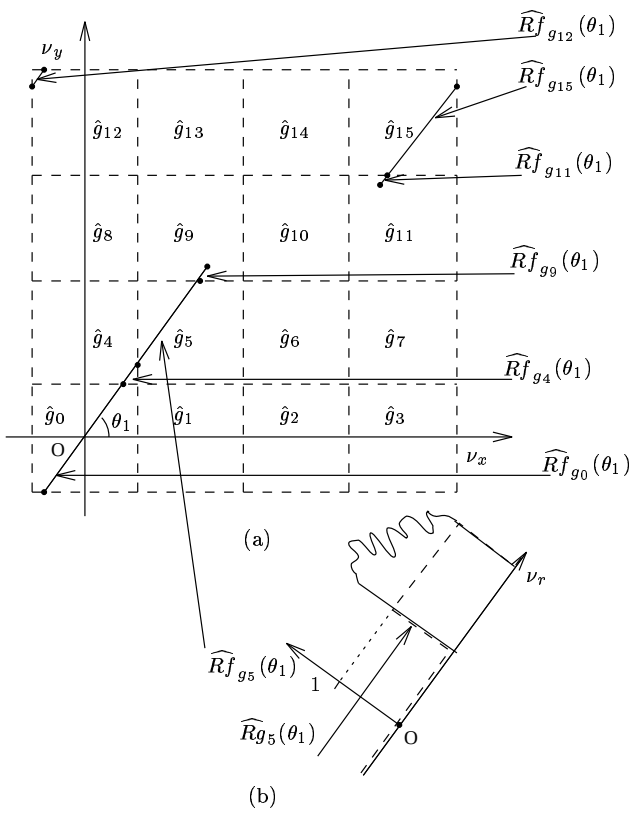

Fig. 2. *Indirect decomposition scheme*

A. Indirect decomposition step

The step of indirect decomposition (see figure 1) consists of generating the Rf_{g_j} with $j \in [0, B^2]$ by filtering Rf. We implement this filtering in the Fourier space to reduce the computation time of this step. We can see on the figure 2a the Fourier transform of one projection ($\widehat{Rf}(\theta_1)$) is localized on a line in the Fourier space. The decomposition of this projection on the indicator function $\hat{g}_j$ is the intersection between the line and the square $\hat{g}_j$ (see the figure 2a). On the figure 2b , we can see the result of the indirect decomposition $\widehat{Rf}_{g_j}$. Let us underline that $\widehat{Rf}_{g_j}(\theta)$ is equal to zero when the line $\widehat{Rf}(\theta)$ do not intersect the square $\hat{g}_j$. We will use this property in the next section to reduce the computation time.

B. Backprojection step

In this step we backproject the B^2 sets of projection Rf_{g_j} to obtain the frequential components f_{g_j}. According to the sampling theory, the function f_{g_j} can be represented by B^2 times less pixel than f. Indeed the undersampling by B^2 of f_{g_j} in the direct space leads to a periodisation of $\hat{g}_j$ in the Fourier space. The function $\hat{g}_j$ is localized on a little square. When the sampling rate is equal to B pixels along each axis, squares do not overlap. Thus the undersampling by B^2 preserves all the information contained in the function f_{g_j}. Thus we can backproject Rf_{g_j} on an undersampled grid. Finally, as we make B^2 backprojections on B^2 undersampled grids, the backprojection step computation cost is the same as for a classical algorithm.

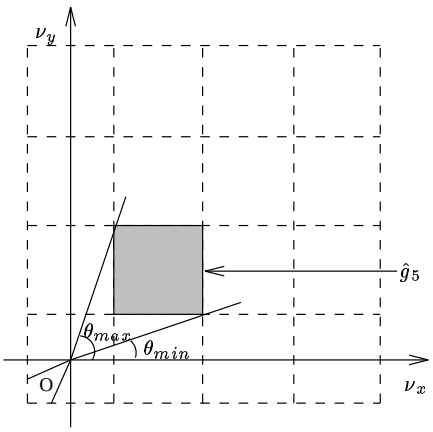

Fig. 3. *Angular interval*

IV. Analyze of the speed up

A. Angular undersampling

The projections of the frequential components f_{g_j} are usually equal to zero for θ outside of an relative small interval $\theta \notin [\theta_{min}, \theta_{max}]$ (see figure 3, $\forall \theta \notin [\theta_{min}, \theta_{max}], \widehat{f_{g_5}} = 0$). We use this property to reduce the backprojection time. In our algorithm when we reconstruct the frequential components f_{g_j}, we backproject only the angle between the corresponding θ_{min} and θ_{max}. The speed up obtained depends on the frequential component. For example, we backproject all angle for f_{g_0} and only approximatively $\frac{N\theta}{B}$ angles for the frequential component $f_{g_{B-1}}$, if B is large (N_θ denotes the number of projections). To analyze the speed up factor obtained by this angular undersampling we count the contribution of the projection to the backprojection of each frequential component. Each projection $Rf(\theta)$ contributes to less than $2B$ frequential components (see figure 2). Thus the number of contributions per bloc is less than $2BN_\theta/B^2 = 2N_\theta/B$. Thus we win at least a factor $\frac{B}{2}$ compare to the backprojection seen in the previous paragraph. The elimination of the zero projection can be considered as an angular undersampling. If we respect the sampling condition to reconstruct f, and if we reconstruct the frequential component f_{g_j} on a B^2 undersampled grid, we can backproject B times less projections.

B. Computation compression

The computation compression will have the same fundamental steps as the data compression. We want also to preserve only the pertinent information and we want to code it with as few bytes as possible. To achieve this result, we decompose the image f in frequential components f_{g_j}, and we compute only the components containing pertinent informations. We denote "quantification", the elimination step of frequential components (see figure 1). This step corresponds to $||\hat{f} \cdot \hat{g}_j||_2 < \epsilon$, where the parameter $\epsilon > 0$ is given and controls the compression rate. Thus, the number of components is reduced from factor depending on the value ϵ. As for data compression, information will be lost if ϵ becomes high.

C. Block processing and parallelism

Most of uniprocessor systems have hierarchical memory. It is composed of various levels of cache memory, a main memory and disk storage. Movement of data between two levels in the hierarchy represents latency time cost. To compute efficiently, we must reduce data movement. Our approach allows to reconstruct some undersampled pictures. If an undersampled picture can be contained in cache memory, the data movement are minimized. Our approach allows to adapt the reconstruction computation to the size of the cache memory and thus to the computer. It is the well known block effect. Because our algorithm is naturally divided on B^2 elementary reconstructions, it can be easily adapted on multiprocessor systems. Owing to the block structure data movement between processor is minimized. In shared memory system, data movement between processor is made through the main memory. These access are limited band width, thus efficiency is improved by reducing memory access in this case too. Our algorithm natural block structure yields efficient computations.

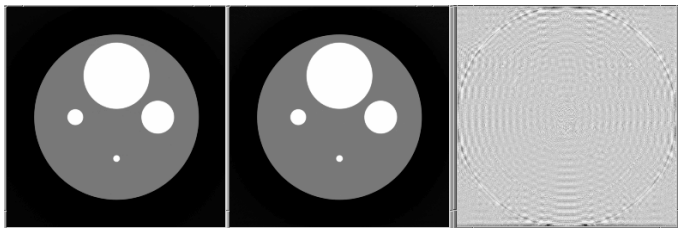

Fig. 4. *Reconstruction of a volume* $512 \times 512 \times 32$: *reference (left), our algorithm with 64 frequential components (center), difference (right).*

V. Application

In this section, we make numerical experiment on two types of phantom: a 3D static phantom to evaluate the optimal number of frequential components and a 2D dynamic phantom to obtain the maximum performance of our algorithm.

A. Static phantom

For our first test, we reconstruct a 3D static phantom f (see phantom definition Table I). The reconstructions are calculated on a $512 \times 512 \times 32$ voxel grid (with x, y and z resolution equal to 1,04 mm), using a subset of the projections collected on 512 angles, uniformly spaced over $[0, \pi[$. We compare our reconstruction with a classical filtered back projection (FBP) algorithm (fig. 4). We observe some artifact on the difference image caused by numerical approximation, but in the case of 64 frequential components, the level on these artifacts is less or equal to 1 %. To highlight the angular undersampling effect, we execute our algorithm with different numbers B^2 of frequential components without quantification. Indeed, we have seen in part IV-A that the speed up factor is proportional to B. Theoretically, the CPU time becomes shorter with big numbers of frequential components. The CPU time decreases up to 64 frequential components but increases afterwards (see figure 5). The first part of the curve is explained by angular undersampling described in part IV-A. The second part of the curve is explained by supplementary computations induced by our approach. Indeed, we can see on the figure 1, that the number of inverse Fourier transforms is equal to $N_\theta \cdot 2B$ instead of N_θ in a classical algorithm. This effect becomes dominating for $B^2 > 64$.

Sphere number	Center (mm)			Radius(mm)	Attenuation
	x	y	z	r	At
1	0.0	0.0	0.0	200	15
2	0.0	-100	0.0	10.0	35
3	-100	0.0	0.0	20	35
4	100	0.0	0.0	40	35
5	0.0	100	0.0	80.0	35

TABLE I
Definition of our 3D phantom

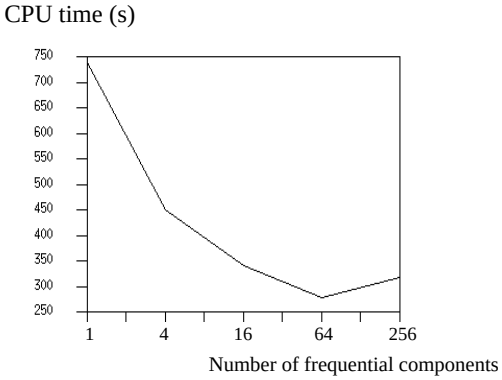

Fig. 5. *Reconstruction CPU time of a volume* $512 \times 512 \times 32$

B. 2D dynamic phantom

For our second test, we reconstruct a 2D dynamic phantom composed of 32 images f_i with $i \in [0, 31]$ (see phantom definition Table II). The phantom motion is a translation of 3 mm per frame along the y axis. In this case, the pertinent information is restricted to one oblic plane in the spatio-temporal Fourier domain [9]. Thus, this dynamic data contains a low number of pertinent information: a lot of frequential components can be eliminated. The

Circle number	Center (mm)		Radius(mm)	Attenuation
	x	y	r	At
1	0.0	$-45 + 3 \times i$	200	15
2	0.0	$-145 + 3 \times i$	10.0	35
3	-100	$-45 + 3 \times i$	20	35
4	100	$-45 + 3 \times i$	40	35
5	0.0	$55 + 3 \times i$	80.0	35
6	0.0	$-45 + 3 \times i$	2.0	60

TABLE II

Definition of our 2D dynamic phantom: with $i \in [0, 31]$

32 reconstructions are calculated on a 512×512 pixel grid (with x and y resolution equal to 1,04 mm), using the same parameter as the previous experiment. During the reconstruction we decompose each f_i in 64 frequential components. To highlight the quantification effect we reconstruct the image sequence with two configurations: the first reconstruction with 38% of components (see figure 6b and c) and the second one with 22% of components (see figure 6d and e). The results are summarized on the table III.

Types of algorithm	% of components	CPU time	speed up	relative L_1 error
FBP	100	754	1	0%
Our	100	290	2.6	1.09%
	38	200	3.8	1.17%
algorithm	22	169	4.5	1.54%

TABLE III

CPU time and quality of dynamic reconstruction

We can observe that the first configuration allows to eliminate 62% of frequential components without deterioration of the image quality. If 78% of components are eliminated, pertinent information is lost (see figure 6e): some artifacts appear in images. Even if a large number of components are eliminated, the speed up factor is not very high. The computation compression is thus only a secondary speed up factor. This can still be interesting for some real time applications, because we can adapt image quality for cpu resource.

VI. CONCLUSION

Our subband Fourier decomposition speeds up the 3D reconstruction. Moreover a compromise between speed up and image quality can be improved through the compression factor. Alternatively decompositions such as Cosinus, Wavelet or Karhunen Loève decomposition could also be used.

REFERENCES

[1] B. Daly and P. A. Templeton, "Real-time ct fluoroscopy: evolution of an interventional tool," *Radiology*, vol. 211, pp. 309–315, 1999.

[2] M. Zhang and M. Kono, "Solitary pulmonary nodules : Evaluation of blood flow paterns with dynamic ct," *Radiology*, pp. 471–478, 1997.

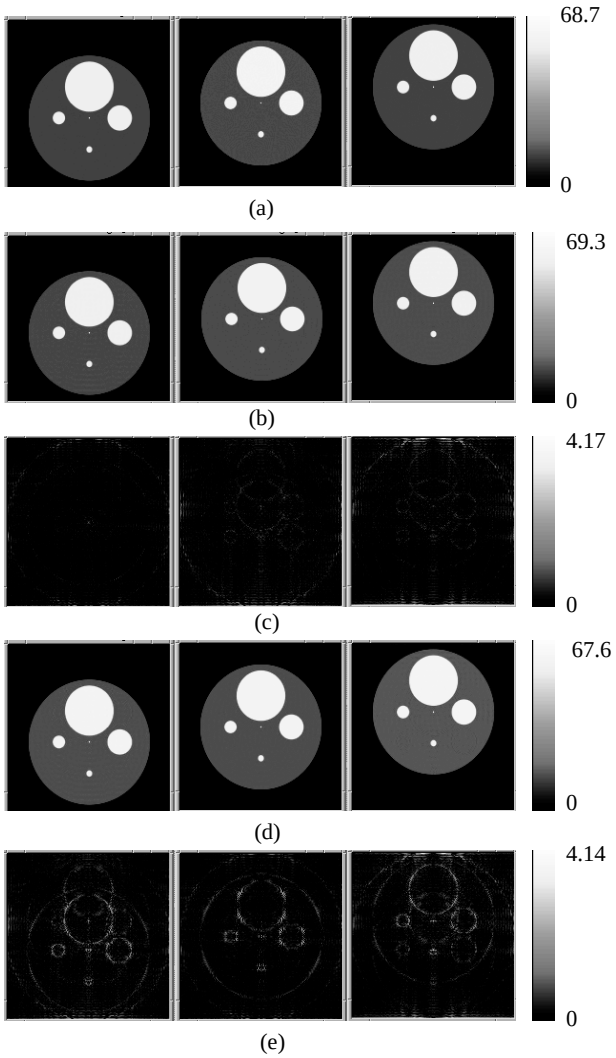

Fig. 6. *(a) Frame 0,15,31 reconstruct by classical FBP, (b) our algorithm with 38 % of frequential components, (c) absolute value of (a)-(b), (d) our algorithm with 22 % of frequential components, (e) absolute value of (c)-(d)*

[3] L. Desbat, M. Fleute, G. Champleboux, and O. Desaint, "3D reconstruction using the PIXIUM 4600 digital X-ray detector ," in *3D-1999*, 1999, pp. 149–152.

[4] M. L. Brady, "A fast discrete approximation algorithm for the radon transform," *SIAM*, vol. 27, no. 1, pp. 107–119, 1998.

[5] H. Turbell, *Three-dimensional image reconstruction in circular and helical computed tomography*, Ph.D. thesis, Linköping University, April 1999.

[6] P. E. Danielsson and M. Ingerhed, "implementation of backprojection in $o(n^2 \log n)$ time," Technical Report LiTH-ISY-R-2003, Linköping University, 1998.

[7] T. Rodet, P. Grangeat, and L. Desbat, "A new computation compression scheme based on a multifrequential approch," in *Conf. Rec. 2000 IEEE Med. Imag. Conf.*, 2000, number IEEE350.

[8] F. Natterer, *The Mathematics of Computerized Tomography*, Wiley, 1986.

[9] Bernd Jähne, *Digital Image Processing*, Springer Verlag Berlin, 1997.