Fast Reprojection using Tree-Structured Filter Bank from Cone Beam Projections

Hiroaki MORIMOTO[1], Kazuhiro UEDA[1] and Yoshitaka MORIKAWA[1]
[1]Graduate School of Natural Science and Technology, Okayama University
[1]3-1-1, Tsushima-naka, Okayama, Okayama 700-8530, Japan
[1]Tel: +81-086-251-8128, Fax: +81-086-251-8127
[1]E-mail: {morimoto, ueda, morikawa}@trans.cne.okayama-u.ac.jp

Abstract: This paper proposes a fast reprojection method using Tree-Structured Filter Bank from cone beam projection data and shows that we can calculate 4.6 times faster than the conventional interpolation method.

1. Introduction

Since we can investigate internal structures without any harm to objects by X-ray CT, it is used in medical area for detection of abnomal regions and in industrial area for non destructive testing. Recently, 3D X-ray CT which reconstructs 3D image receives much attention. The strict method of 3D image reconstruction from cone beam projections was proposed by Grangeat[1]. In this method, we compute the first derivative of the Radon Transform from cone beam projections, followed by backprojection. However it requires much computation time to reconstruct 3D distribution. In this paper, the authors proposes a method using Tree-Structured Filter Bank (TSFB) to reduce processing time of reprojection which requires many calculations in Grangeat's method. Additionally, we overcome boundary effect by extending the detector region.

2. Fundamentals of 3D reconstruction and Reprojection

Let$f(\mathbf{r})(\mathbf{r} = (x, y, z)^T)$ be an absorption distribution in 3D space. Let, $\mathbf{n}$ and ρ be the plane tangent vector and its distance from the origin. Then, the 3D Radon transform is expressed in the following equation;

$$Rf(\rho, \mathbf{n}) = \int f(\mathbf{r})\delta(\rho - \mathbf{r}\cdot\mathbf{n})d\mathbf{r} \quad (1)$$

where $\delta()$ is the Dirac delta and $\cdot$ represents the inner product of vectors. Eq. (1) means that the 3D Radon transform is surface integrals for multiple planes defined by $\mathbf{n}$ and ρ.

On the other hand, 3D inverse Radon transform is expressed in the following equation;

$$f(\mathbf{r}) = -\frac{1}{8\pi^2}\int \frac{\partial^2 Rf}{\partial\rho^2}(\mathbf{r}\cdot\mathbf{n}, \mathbf{n})d\mathbf{n} \quad (2)$$

This equation means that the functions differentiated twice with respect to ρ are backprojected through the respective planes determined by $\mathbf{r}\cdot\mathbf{n}$ $(= \rho)$. Therefore, if we obtain the 3D Radon transform value of a certain object, then we can strictly reconstruct the 3D object distribution.

In the cone beam CT setting, a planer detector records the attenuated radiations in 2D form, as shown in Fig. 1. Grangeat showed that the first derivative of the line integral with respect to t is equal to $\partial Rf/\partial\rho$. So, for implementation of Grangeat method, we have to implement the line integration. As an example, let us calculate the first derivative of

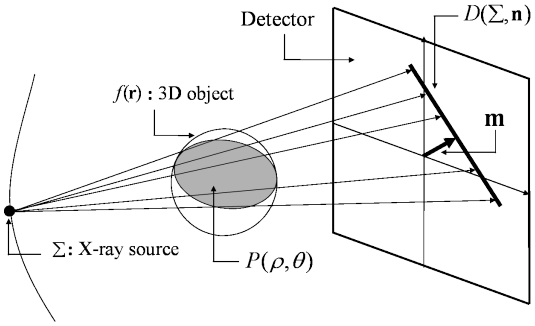


Figure 1. A relationship Cone Beam Projections and the first derivative of 3D Radon transform

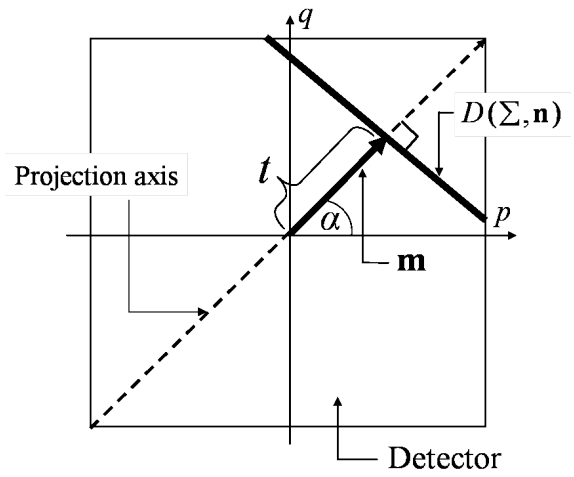


Figure 2. Illustration of line integral process along $D(\Sigma, \mathbf{n})$ on the detector

3D Radon transform value of the plane expressed by $P(\rho, \mathbf{n})$ in Fig.1. $P(\rho, \mathbf{n})$ is also determined by a source position Σ and a line $D(\Sigma, \mathbf{n})$ in the detector plane. We can regard cone beam as assemblage of linear beams. Therefore, if we sum up the intensities on the line $D(\Sigma, \mathbf{n})$ and take its derivative with respect to t, we can get the $\partial Rf/\partial\rho$. This line integral operation is called reprojection. Fig. 2 illustrates the line integral operation along $D(\Sigma, \mathbf{n})$ on the detector plane.

Let the distribution of detected values be $Yf(\mathbf{r})(\mathbf{r} = (p, q)^T)$ on the detector plane, normal vector be $\mathbf{m}(\alpha) = (\cos\alpha, \sin\alpha)^T$, and the distance from the origin of the detector plane to $D(\Sigma, \mathbf{n})$ be $t(= \mathbf{r}\cdot\mathbf{m})$ the reprojection is expressed in the following equation;

$$q(t, \alpha) = \iint_{-\infty}^{\infty} Yf(p, q)\delta(p\cos\alpha + q\sin\alpha - t)dpdq \quad (3)$$

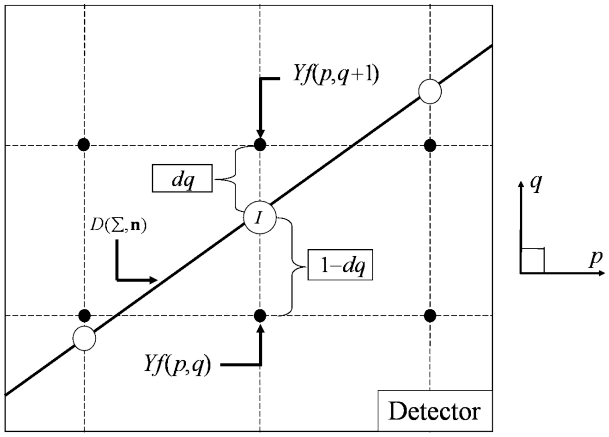


Figure 3. Illustration of implementation of line integral process

Eq. (3) shows that reprojection is line integral along the segment with various distances from the origin on detector plane.

Because the distribution of detected values is dealt on discrete lattice, we have to employ interpolation to implement this line integral operation. Fig. 3 illustrates implementation of line integral process by linear interpolation. For instance, we consider the value of the point Ⓘ shown in Fig. 3. We can calculate the value by using the following linear interpolation;

$$Ⓘ \quad = \quad dq \cdot yf(p,q) + (1-dq) \cdot yf(p,q+1) \qquad (4)$$

We approximate the line integral along $D(\Sigma, \mathbf{n})$ by summing up the values calculated at point of every vertical line $x = q$.

3. Fast Reprojection using TSFB

3.1 Tree structure and a design method of parallelogram filter

Fourier slice theorem [2] says that the projection perpendicular to a normal vector $\mathbf{n}$ is equivalent to the lowpass operation with filter of narrow pass filter along to $\mathbf{n}$ as shown in Fig. 4.

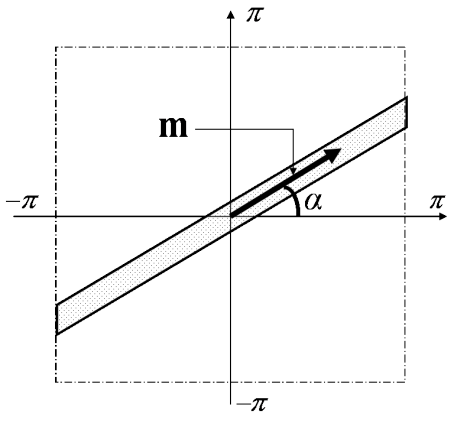


Figure 4. Passband of narrowband filter of α-degree angle

We can get the projection data by re-sampling middle of the line of 2D detected value after such a filtering operation. Such a narrowband filter is designed by connecting some parallelogram filters with different inclinations shown in Fig. 5 [3],

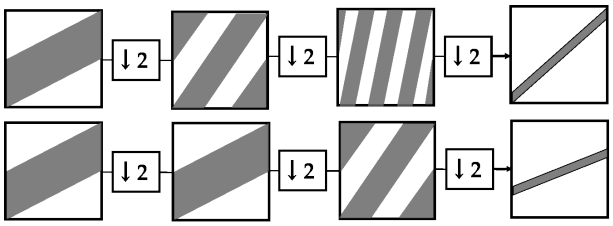


Figure 5. Example of designing a narrowband filter

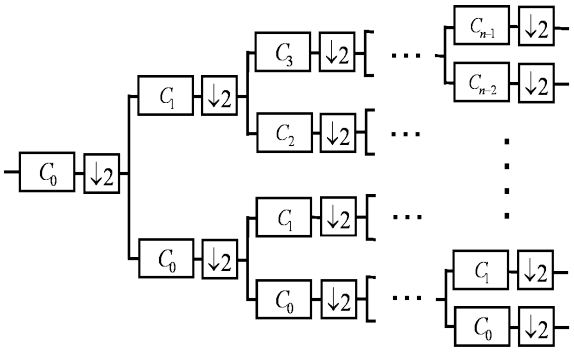


Figure 6. Tree-Structured Filter Bank system

where $\boxed{\downarrow 2}$ is the downsampler of factor 2 in the vertical direction.

Our objective to implement the reprojection as such a filtering operations shown in Fig. 5. In the figure, the first stages of the two filters are the same, so these can be replaced by one operation. We can perform such a simplification between all the directional filters. Fig. 6 shows the result of the extreme simplification. Such a filter structure is called tree structure. Note that $\boxed{H_n}$ shows the transfer function of parallelogram filter. The amount of calculation of the filter bank system decreases by making it tree-structure, and reprojection can be performed faster[4].

The parallelogram filter shown in Fig. 5 is composed as follows. For this end, consider the linear phase filter $H(z)$ as [4];

$$H(z) \quad = \quad \frac{zA(z) + A(z^{-1})}{2} \qquad (5)$$

$$\because \ A(z) \quad = \quad \frac{1 + 0.5z^{-1}}{1 + 0.5z} \qquad (6)$$

where $A(z)$ is the all-pass filter shown in Fig. 7. Note its low calculation cost of this filter. We can compose the parallelo-

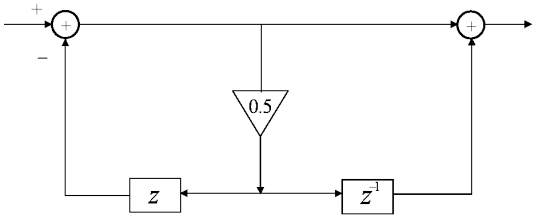


Figure 7. Processing unit of $A(z)$

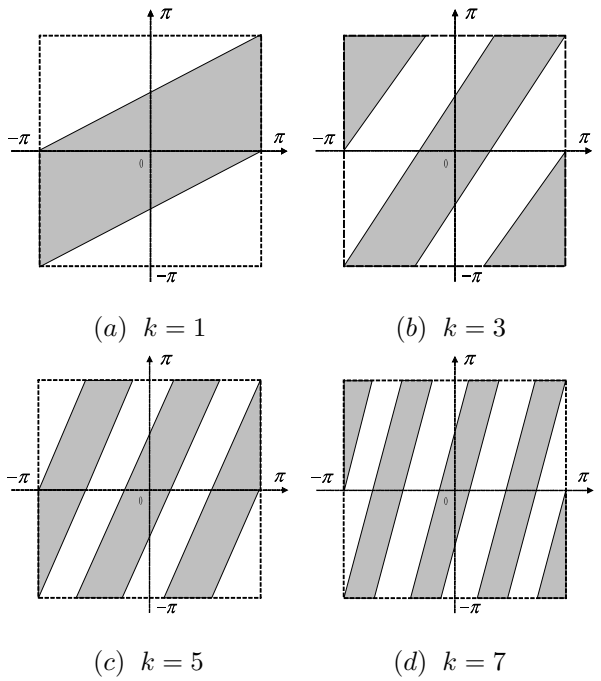


Figure 8. Example of pass bands of parallelogram filter

gram filter using $H(z)$ as follows [3];

$$C_k(z_0, z_1) = \frac{1}{2}[1 + z_0^{-\frac{1-k}{2}} z_1^{-1} H(z_0) H(z_0^{-k} z_1^2)] \qquad (7)$$

where k is a odd-numbered parameter which represents inclination of the parallelogram, and a inclination of centerline of the parallelogram is equivalent to $k/2$. Fig. 8 shows example of pass bands of parallelogram filter $C_k(z_0, z_1)(k = 1, 3, 5, 7)$.

3.2 Considerations of filtering process

Fig. 9 is a block diagram of parallelogram filter. The processing is interpreted as follows;

1. Divide 2D signal into even and odd numbered rows.
2. Apply $H(z)$ in an oblique direction of $-1/k$ between the even numbered rows.
3. Apply $H(z)$ in a column direction to the even number rows.
4. Delay the processed results by $(1-k)/2$ samples, followed by their addition to odd numbered row signal.
5. Multiply the result by $1/2$.

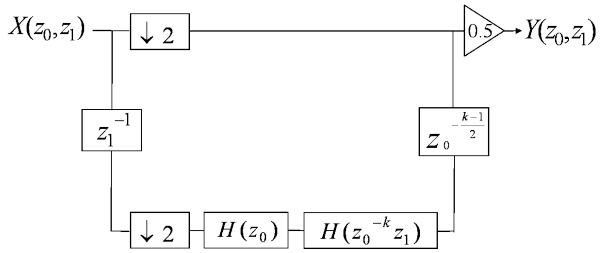


Figure 9. Processing of parallelogram filter

There are some considerations for parallelogram filter implementation. First, let us consider boundary problem. For instance, let $k = 1$, we apply $H(z)$ in an oblique direction of

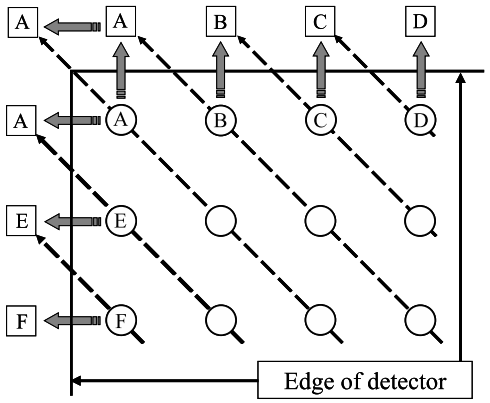


Figure 10. Illustration of boundary process ($k = 1$)

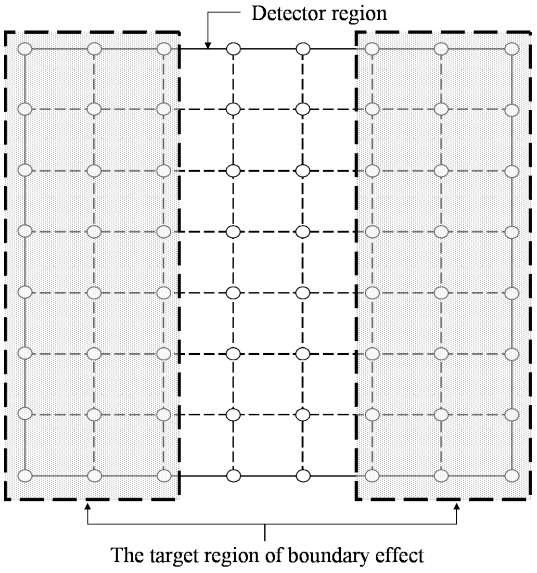


Figure 11. A example of the region which influenced by boundary effect ($k = 3$)

-1 in step 2. Then, we confront a problem that we must use a value of outside of the detector region. In this case, we overcome the problem by copying the nearest neighbor of the required value. Fig.10 illustrates the process.

However, the values which are needed outside of the detector region are not correct values, these are simply assumed for calculation. Therefor, a part of the filtering results include errors caused by supposed boundary values. Fig. 11 shows a example of the region which influenced by the boundary effect. The influenced region may increase and decrease by the value of k. Consequently, in order to overcome this effect, we extend the region of the input 2D signal and followed by implementing the filtering process including the extension region. Fig. 12 illustrates the extension. This method pads the zero outside the object region. In this manner, we suppress region effect without influencing the region of interest.

3.3 Comparison of calculation cost

We will show comparison of calculation cost. Let t, α shown in Eq. (3) be sampled to N samples respectively, pixel size of detector be N^2, and extend input 2D signal twice ($2N$)

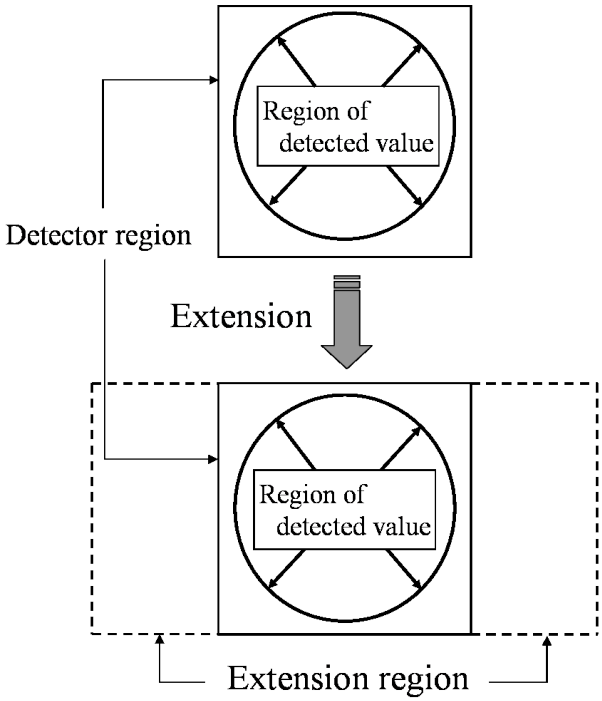


Figure 12. Illustration of extend process

in horizontal direction. In proposed method, it is necessary to multiply 7 times multiplication per 1 pixel at one parallelogram filter processing. The target number of samples for filtering process is $2N^2$ and the number of stages of filter-bank system is $\log_2 N$, so the amount of calculation of the proposed method is $14N^2 \log_2 N$. In the conventional interpolation method, the calculation cost is $2N$ per 1 pixel. Because the target number of samples for processing is N^2, the amount of calculation of the method is $2N^3$. Let $N = 256$, the proposed method can calculate $2N^3/14N^2 \log_2 N = 4.6$ times faster than interpolation method.

4. Simulation

We compared the reprojection of Grangeat method with that of the proposed method i.e. TSFB method. As a simulated object, we used the 3D head phantom whose pixel size is $256 \times 256 \times 256$, in which small elliptic balls exist. Trajectory of X-ray source is circular orbit of the radius $128 \cdot 4$, and pixel size of detector is 256×256.

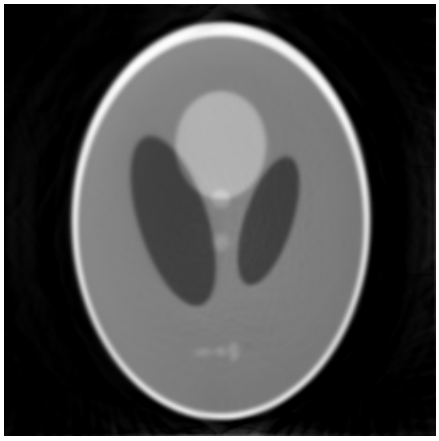
Table 1 shows the comparison of computation time. We see TSFB method, works 4.9 times faster than Grangeat method. This result is close to expected ratio. Fig. 14 show the respective crosssection reconstructions at $z = 0$. We see that there are hardly differences between them.

Table 1. Comparison of computation time

Interpolation	TSFB	Ratio of calculation cost
257*[sec]	52*[sec]	4.9

*CPU: Intel Core2 CPU 6400, 2.13GHz

(a) Interpolation
(PSNR: 23.5dB)



(b) TSFB
(PSNR: 22.5dB)

Figure 13. Comparison of reconstructed crosssections at $z = 0$

5. Conclusion

In this paper, we have shown fast reprojection using Tree-Structured Filter Bank(TSFB). According to this method, We can calculate more than about 4 times fater than Gangeat's method. As a future work, it is required that we investigate more effective implementation of process of parallelogram filter .

References

[1] P.Grangeat:"Mathematical framework of cone beam 3D reconstruction via the first derivative of the Radon transform in Tomography.Lecture notes in Mathematics", Vol.1497,pp.66-97,Springer,1991.

[2] G.T.Herman, Image Reconstruction From Projections: Implementation and Applications, Academic Press, New York, 1980.

[3] J.Murakami, K.Mizowaki and Y.Morikawa: "Reconstruction Algorithm for Computerized Tomography Using Tree-Structured Filter Bank", IEICE (D-II), vol.J84-D-II, no.3, pp.580-589.

[4] P.P.Vaidyanathan: "Multirate, Systems And Filter Banks", P T R Prentice Hall, New Jersey, 1993.