Journal of X-Ray Science and Technology 19 (2011) 429–442
DOI 10.3233/XST-2011-0305
IOS Press

A novel method for 3D crack edge extraction in CT volume data

Bi Bi[a,b], Li Zeng[a,c,*] and Haina Jiang[a,b]
[a]*ICT Research Center, Key Laboratory of Optoelectronic Technology and System of the Education Ministry of China, Chongqing University, Chongqing, China*
[b]*College of Optoelectronic Engineering, Chongqing University, Chongqing, China*
[c]*College of Mathematics and Statistics, Chongqing University, Chongqing, China*

Received 29 December 2010
Revised 23 May 2011
Accepted 13 July 2011

Abstract. To reduce time cost and improve the performance of edge extraction in CT volume data which is often in large size, we propose a novel method of 3D crack edge extraction using two fusion steps, one is fusion on Finite Line Integral Transform (FLIT) values in spatial directions called SD-FLIT and another is fusion on Local Binary Pattern (LBP) values on spatial planes called SP-LBP. By analyzing the S function of LBP operator, we find that value “0” of this function can describe the change between two equivalence planes. However, this property is sensitive to point difference, thus SD-FLIT is introduced to smooth noises and artifacts before the application of SP-LBP to extract 3D edge on binary volume data. Besides, fusions on directions and planes are aimed at extracting enough spatial information. Experimental results show that, owing to the sufficiency of information extraction and the simplicity of computation, our method can get continuous, thin and occlusive 3D edge, including the crack tip. Furthermore, it can be used to complicate volume data. Compared with 3D wavelet and Facet model, our method cost less time, saving at least 89% of that.

Keywords: CT volume data, 3D crack edge, data fusion, LBP, FLIT

1. Introduction

With wavelengths in the nanometer or picometer scales, X-rays are energetic enough to penetrate most materials including castings and bones. It makes X-rays ideal for noninvasive industrial detecting [1] and medical probing [2,3]. Being one of such nondestructive detection methods, Computed Tomography (CT) acquires slice images synthesized from the collected X-ray signals along directions. Thanks to its friendly nature to detected objects, as well as its unique capability in showing the section plane of objects, CT has witnessed broad applications in industrial detection [4,5] and medical diagnosis [6,7]. In order to improve the correction rate of diagnosis or recognition, corresponding processes are applied to CT images, such as image enhancement, edge extraction, object segmentation and image measuring.

Recently, applications of CT volume data have become more and more popular in automatic detection. The edge extraction of CT volume data is the bridge between image enhancement and object recognition.

*Corresponding author. Tel.: +86 23 86394828; Fax: +86 23 65103562; E-mail: drlizeng@hotmail.com.

Almost trivial to human vision, this edge extraction still remains one of the most challenging and most studied problems in real time detection, defects measuring and object recognition, due to the inevitable appearance of noises and artifacts, the large size of volume data, the efficiency request of automatic detection and the precision request of defects location. Traditional 3D edge extraction algorithms include methods based on wavelet analysis, methods based on gradient and methods based on partial differential equation theory, such as 3D wavelet [8], Facet model [9] and level set algorithm [10], etc., have specified results for specified parameters, thus users have to modify parameters whenever the segmentation results are not satisfactory. Additionally, some of these methods are tough to keep the balance between noise elimination and small edge preservation, as they may lose the tip of slender edge, or improperly extract sharp corners as fillet corners. Moreover, because of the complication of algorithm or the exhaustion of iteration, these methods are time-consuming. These shortcomings have decoupled the application of CT volume data in real time automatic detection.

In this paper we aim at finding a simple method of 3D edge extraction for crack CT volume data, which can not only reduce time cost but also achieve a good extraction performance. The proposed novel method uses two fusions: one is fusion on Finite Line Integral Transform (FLIT) values in spatial directions called SD-FLIT; another one is fusion on Local Binary Pattern (LBP) values on spatial planes called SP-LBP. Essentially, LBP is a comparison between pixels in a certain region, so that value "1" of S function in LBP operator presents the smooth among pixels and value "0" of that presents the leap of pixels. From this point, LBP can be used to describe 3D edge in CT volume. But also due to the comparison, LBP is sensitive to noises, thus FLIT which calculates the integral along line segment is introduced, since it can weaken effects of noises and artifacts when extracting direction information. The experimental results show that, with plenty of spatial information guaranteed by fusions, our method can obtain occlusive and thin 3D edge in a short time. Worthwhile, our method can preserve the tip of crack, which is an important component of crack expansion analysis. Thus our method is an efficient edge extraction way with respect to time cost and image quality.

The remainder of this paper is organized as follows: Section 2 introduces LBP operator and FLIT operator. Section 3 details the new method of fusions on local information extracted by SD-FLIT and SP-LBP respectively. Experimental results are presented and discussed in section 4. And section 5 concludes the paper.

2. Overview

2.1. LBP

Local Binary Patten (LBP) was first mentioned by Ojala et al. [11] to describe the texture information for objects recognition. The current pixel is assumed to be the center of a circular field with a radius R, and P points are sampled on the circle. Let f_c denotes the pixel value of the current pixel, and f_i represents the pixel value of the point on the circle, $i = 1, \dots, P$. Then the LBP value of current pixel can be given as:

$$LBP_{P,R} = \sum_{i=1}^{P} S(f_i - f_c)2^{i-1} \tag{1}$$

Where, $S(x) = \begin{cases} 1 & x \geqslant 0 \\ 0 & x < 0 \end{cases}$. As shown in Fig. 1, when $P = 8$ and $R = 1$, the circular field is just the 3×3 neighborhood. Thus the texture descriptor is generated by histograms of LBP images. Then

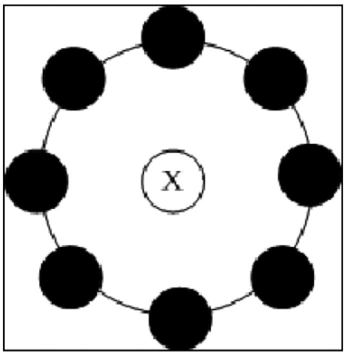

Fig. 1. The illustration of $\mathrm{LBP}_{8,1}$ neighborhood.

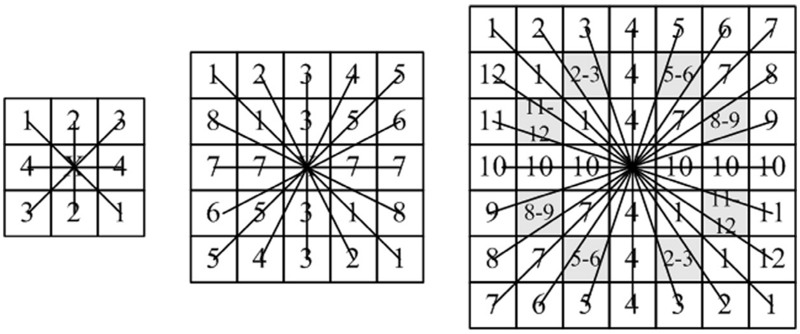

Fig. 2. Lines of FLIT in different 2D templates, from left to right: 3×3, 5×5 and 7×7.

uniform pattern and rotation invariance were introduced into LBP [12–14], assuring this descriptor more robust and less redundant when being applied in recognition and classification. Later, LBP was modified in dimension reduction [15], neighborhood selection [16], S functions quantization [17], histograms integration [18], and in the selection of 3D LBP samples [19] for a better performance, further more, in features matching [20–22].

However, LBP operators mentioned above are not suitable for our proposition, because these operators were treated as texture descriptors and their final forms were histograms. Here we need to keep the grayscale description and edge information.

2.2. FLIT

Finite Line Integral Transform (FLIT) was proposed by Yang et al. [23], using lines in the square template to analyze images. Given a template of $p \times p$ (p is an odd number) pixels, $2(p-1)$ lines can be obtained. These lines are oriented in different directions obeying several rules. When $p \geqslant 7$, the definition of lines in the template may not be unique. Detail lines in different templates can be seen in Fig. 2, where different numbers mean the point sets lying in different direction lines.

Given k different directions, L_k indicates the line in direction k. Then FLIT of pixel (i,j) with gray value $f(i,j)$ is defined as:

$$FLIT(i,j,k) = \sum_{(i,j)\in L_k} f(i,j), 1 \leqslant k \leqslant 2(p-1) \tag{2}$$

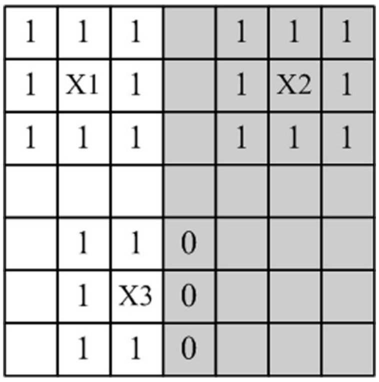

Fig. 3. Examples of S function.

Zeng and Li extended the unique condition to make every line in the template to be unique no matter what size it is, and called this method as IFLIT. They developed IFLIT to 3D forms as well [24], where lines were defined as follows: Each line goes through the centric voxel of the template; All lines intersect only at the centric voxel; Every voxel in the template belongs to only one of these line, except the centric voxel; Each edge voxel and its centric symmetrical voxel in the template lie in the same line. For a $p \times p \times p$ template, p is an odd number, $3p^2 - 6p + 4$ lines can be obtained. Then 3D IFLIT was used for the extraction of linear feature in CT volume. Li and Zeng [25] also extended the line integral to plane integral called Finite Plane Integral Transform (FPIT), and combined FPIT with planelet to extract crack surface. This method is based on planes, so the extraction is a process of plane approximating curve surface. As a result, there are some discontinuous regions on curve surface more or less.

In this paper, to get a smooth and occlusive 3D edge, FLIT is used not for line extraction but for smoothing. Besides, not all the lines of 3D IFLIT in [24] are used in our method, details see section 3.

3. Our method

Following the basic knowledge prepared in the preceding section, we further explore the rule of LBP as an efficient tool for edge detection. Firstly, we begin by analyzing the S function in LBP operator, $S(x) = \begin{cases} 1 & x \geqslant 0 \\ 0 & x < 0 \end{cases}, x = f_i - f_c$. Where the S function values of surrounding pixels are "1" if the image is an equivalence plane, otherwise, the S function values of the surroundings may be "0". Figure 3 gives us an example of image formed by two equivalence planes, X1, X2 are points in planes and X3 is a point near the intersection line, and their function values can be seen clearly in the figure. In general, value "0" of S function describes the edge and value "1" presents the plane. Besides, LBP has the properties of grayscale invariance, computational simplicity and tolerance against monotonic gray-scale changes. So we use LBP to detect 3D edge of CT volume data.

Before detecting, we notice that LBP is a comparison between points essentially, so it magnifies noises and artifacts, see Fig. 4. The left image is the first slice of original CT volume data of crack, and the right one is the result directly processed by LBP operator. So, to smooth noises and artifacts, FLIT is used by integrating the voxel values in the template neighborhood. Thus a new simple method is presented, which contains two fusion steps:

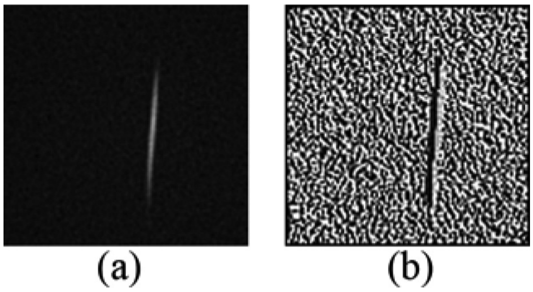

Fig. 4. The noises magnification of LBP operator. (a) The first slice of crack CT volume data. (b) The result by LBP.

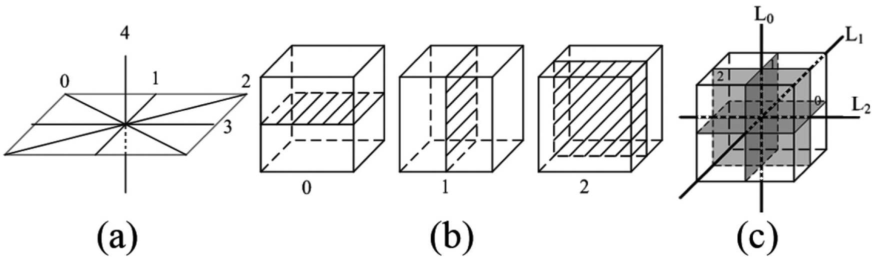

Fig. 5. Directions in SD-FLIT and planes in SP-LBP. (a) Five lines in SD-FLIT. (b) Three planes in SP-LBP. (c) Three intersection lines in SP-LBP.

3.1. SD-FLIT

For the whole volume data, we compute FLIT value for every voxel along lines in five directions in the $3 \times 3 \times 3$ neighborhood:

$$FLIT(x,y,z,k) = \sum_{(x,y,z)\in L_k} f(x,y,z), \quad k = 0,\ldots,4. \tag{3}$$

$f(x,y,z)$ is the gray value of current voxel, and L_k is the line in the kth direction. Note that the selected five directions are not lying in the same plane, see Fig. 5 (a).

Then, for each direction of the whole data, the mean value of FLIT m_k is computed. Later m_k is compared with FLIT value of each voxel in such direction to obtain a binary value:

$$b_k(x,y,z) = \begin{cases} 1 & FLIT(x,y,z,k) > m_k \\ 0 & \text{otherwise} \end{cases}, \quad k = 0,\ldots,4. \tag{4}$$

And then we fuse the directional binary data. Considering the defects are bright in our images, we extract defects information by the maximum fusion rule:

$$Fd(x,y,z) = \max_{k=0,\ldots,4}\{b_k(x,y,z)\} \times 255. \tag{5}$$

3.2. SP-LBP

After the first fusion, we compute $LBP_{8,1}$ on three spatial planes, see Fig. 5 (b). These planes lie in the same $3 \times 3 \times 3$ neighborhood, and share the same centric point which is the current voxel. The

$LBP_{8,1}$ value on plane n is calculated as:

$$g_n(x,y,z) = LBP_{8,1}(Fd(x,y,z),n) = \sum_{i,c \in P_n} S(Fd_i - Fd_c)2^{i-1}, n = 0,1,2, \quad i = 1,\ldots,8 \quad (6)$$

Where $Fd_c = Fd(x,y,z)$ denotes the binary value of the centric voxel, Fd_i denotes the binary value of surrounding voxel in the circularity neighborhood, P_n is the nth plane in Fig. 5 (b).

Labeling the intersection lines of these planes as L_n, see Fig. 5 (c), FLIT values of these three lines are used to weight LBP results:

$$w_n(x,y,z) = FLIT(x,y,z,n) / \sum_{n=0}^{2} FLIT(x,y,z,n) = \frac{\sum\limits_{(x,y,z)\in L_n} Fd(x,y,z)}{\sum\limits_{n=0}^{2} \sum\limits_{(x,y,z)\in L_n} Fd(x,y,z)},$$

$$n = 0,1,2 \quad (7)$$

Finally, LBP values on spatial planes are fused as:

$$Fp(x,y,z) = \sum_{n=0}^{2} (w_n(x,y,z) \cdot g_n(x,y,z)) \quad (8)$$

By turning the edge to be white and the background to be black, the extracted edge can be highlighted.

The pseudo code of our method can be listed as follows:

Step 1: Read CT volume data $f(x,y,z)$ with size Width $\times$ Height $\times$ Layer. And assume $x = 0$, $y = 0$, $z = 0$, $k = 0$, $n = 0$.

Step 2: SD-FLIT.

For $z = 1$: Layer-2

For $y = 1$: Height-2, $x = 1$: Width-2, $k = 0$:4

$$FLIT(x,y,z,k) = \sum_{(x,y,z)\in L_k} f(x,y,z)$$

End for k, x, y

For $k = 0$: 4

$$m_k = \frac{\sum\limits_{y=1}^{Height-2} \sum\limits_{x=1}^{Width-2} FLIT(x,y,z,k)}{(Height-2)(Width-2)}$$

For $y = 1$: Height-2, $x = 1$: Width-2

$$b_k(x,y,z) = \begin{cases} 1 & FLIT(x,y,z,k) > m_k \\ 0 & \text{otherwise} \end{cases}$$

End for x, y

End for k

For $y = 1$: Height-2, $x = 1$: Width-2

$$Fd(x,y,z) = \max\{b_k(x,y,z), k = 0,\ldots,4\} \times 255$$

End for x, y
End for z

Step 3: SP-LBP.
For $z = 1$: Layer-2, $y = 1$: Height-2, $x = 1$: Width-2
For $n = 0$: 2

$$g_n(x,y,z) = LBP_{8,1}(Fd(x,y,z), n)$$

$$w_n(x,y,z) = \frac{\sum_{(x,y,z)\in L_n} Fd(x,y,z)}{\sum_{n=0}^{2} \sum_{(x,y,z)\in L_n} Fd(x,y,z)}$$

End for n

$$Fp(x,y,z) = \sum_{n=0}^{2} (w_n(x,y,z) \cdot g_n(x,y,z))$$

End for x, y, z
Note that $k = 0$, 1, 2, 3 and $n = 0$, if $z = 0$ or $z =$ Layer-1.

Step 4: Turn the obtained 3D edge to be white and the background to be black.

In order to maintain the simplicity of computation, five lines and three planes would be appropriate respectively for most of CT volume data. But they are not absolutely fixed. In order to avoid the edge fracture and area shrinkage, the number of lines in SD-FLIT should be more than four. Furthermore, to prevent the edge extraction from totally reliance on SD-FLIT, the number of planes in SP-LBP should be more than one.

By using line integrals, SD-FLIT can smooth noises and artifacts. Also by integrating along line segments, SD-FLIT can preserve the tips of cracks. And after fusing binary data in five directions, changes that maintain in several voxels such as object edge can be kept and changes that occurred incidentally like noises would be eliminated. As a result, the edge information is reserved after the first fusion step. Besides, SD-LBP compares voxels in the neighborhood, so it can extract both voxel-level information and region-level information to get continuous and occlusive edge. With edge information fused from three planes, some of the lost information in the first fusion step can be complemented in the second fusion step. All of these have made our method work as we expect.

4. Results and discussion

4.1. Phantoms and data

All the experiments are tested on a PC equipped with Intel(R) Pentium(R) 4, CPU 2.93 GHz, EMS memory 1GB, and the procedures are written by VC++ 6.0 language in a windows XP environment.

Our method is tested on CT volume data of crack ($100 \times 100 \times 18$), which comes from real castings. To validate the applicability of our method, the method is also experimented on CT volume data of engine ($256 \times 256 \times 108$) [26] and CT volume data of carburetor ($512 \times 512 \times 167$). A simulation data ($128 \times 128 \times 20$) is used to evaluate the robustness and the performance of our method to various cases. The efficiency is presented in two ways: one is image quality, another is time cost.

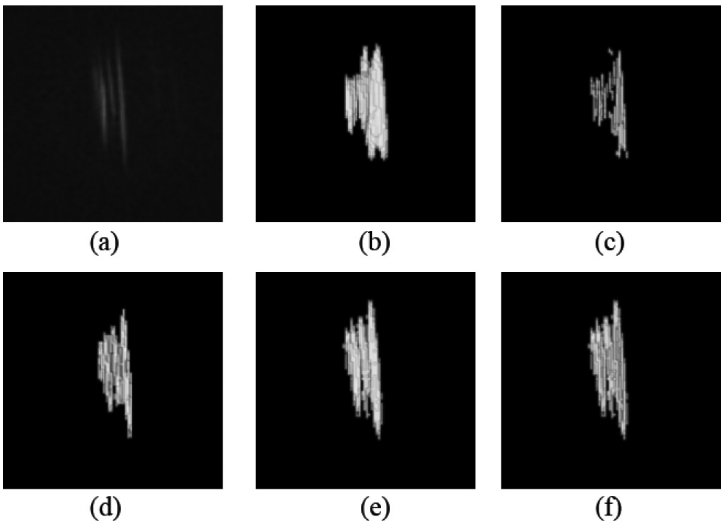

Fig. 6. Snapshots for 3D volume data. (a) Original CT volume data of crack. (b) Edge obtained by 3D wavelet. (c) Edge obtained by Facet model. (d) Edge obtained by Li's method. (e) Edge obtained by SD-FLIT. (f) Edge obtained by our method.

4.2. Image quality

4.2.1. CT volume data of crack

Firstly, we show the extracted 3D edges in 3D forms snapshot by 3D Med [27]. Then, slices of the volume data are shown to give a detailed view, including slices where crack is curve and slices where crack is surrounded with heavy noises and artifacts.

Snapshots of 3D edges extracted by 3D wavelet, Facet model [9], method of Li [25], SD-FLIT and our method are presented in Fig. 6 from (b) to (f). Shown in Fig. 6 (b), 3D edge obtained by 3D wavelet is thicker, or say coarser, than other methods. As known to all, wavelet is influenced by parameters like threshold or convolution kernel size. In this experiment, considering edge continuity and time cost, the convolution kernel size is 9 and the threshold for wavelet coefficients is 2000. The edge obtained by Facet model is thin enough but has some burrs, see Fig. 6 (c). Besides, edges extracted by 3D wavelet and Facet model are both shorter than the original crack, since they cannot extract the tips of cracks. Figure 6 (d) illustrates that 3D edge detected by Li's method is thin, and has the tip of crack. But Li's method needs morphology methods to smooth the edge. While our method can obtain continuous 3D edge, meanwhile maintain the crack tip, shown in Fig. 6 (f). Figure 6 (e) gives a view of the middle result of our method, which is the binary data of SD-FLIT. Snapshots of edges obtained by LBP along and LBP with Gaussian filter are not presented, because the crack has been shadowed by magnified noises.

Figure 7 (a) shows the second slice of original CT volume data, Fig. 8 (a) presents the fourth slice of original CT volume data in which the crack is surrounded by noises and artifacts, and Fig. 9 (a) displays the twelfth slice of original volume data where the crack is a bit curve. Slices of 3D edge extracted by wavelet are illustrated in Fig. 7 (b), Fig. 8 (b), and Fig. 9 (b), where the edge is coarse and discontinuous. Slices of 3D edge extracted by Facet model is single-pixel wide but discontinuous, see Fig. 7 (c), Fig. 8 (c), and Fig. 9 (c). Especially can be seen in Fig. 8 (c), the extracted edge is fractured and distorted. Compared with these two methods, by means of the smooth strategy, Li's method can get thin, continuous and occlusive edge, seen from Fig. 7 (d), Fig. 8 (d), and Fig. 9 (d). While Fig. 7 (f), Fig. 8 (f) and Fig. 9 (f) are results directly by LBP operator, which are terribly influenced by noises and artifacts, as

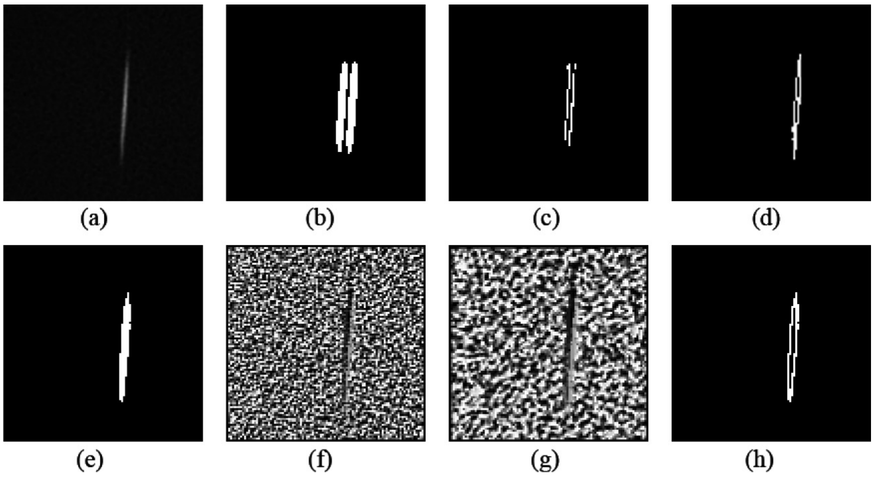

Fig. 7. The second slices of 3D volume data. (a) Original slice of crack. (b) Edge slice of 3D wavelet. (c) Edge slice of Facet model. (d) Edge slice of Li's method. (e) The slice of binary data. (f) The slice of data processed by LBP. (g) The slice of data processed by LBP with Gaussian filter. (h) Edge slice of our method.

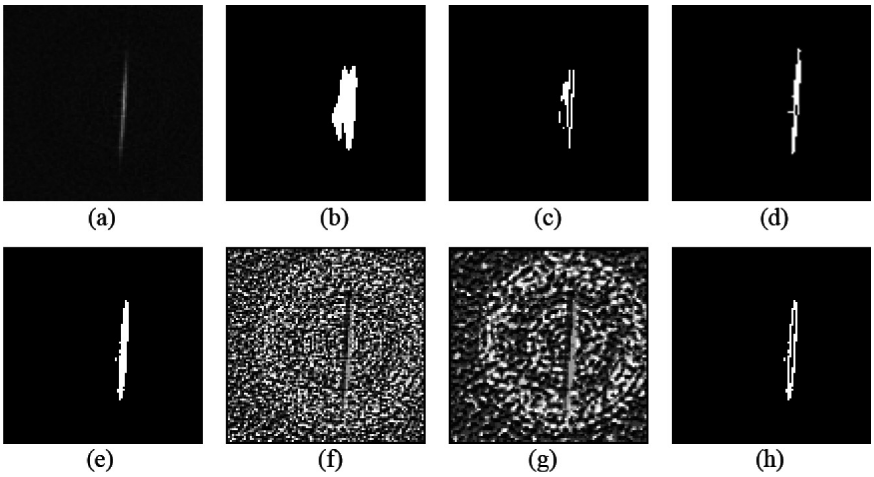

Fig. 8. The fourth slice of 3D volume data. (a) Original slice of crack. (b) Edge slice of 3D wavelet. (c) Edge slice of Facet model. (d) Edge slice of Li's method. (e) The slice of binary data. (f) The slice of data processed by LBP. (g) The slice of data processed by LBP with Gaussian filter. (h) Edge slice of our method.

LBP magnifies the difference among voxels. Even with the aid of Gaussian filter, the magnification of point singularity of LBP operator still remains, as shown in Fig. 7 (g), Fig. 8 (g), and Fig. 9 (g). This is why we use SD-FLIT to obtain binary volume data before using SP-LBP to extract 3D edge. Slices of the binary data obtained by SD-FLIT are shown in Fig. 7 (e), Fig. 8 (e), and Fig. 9 (e), from which we can find that SD-FLIT can extract the target object overcoming the disturbance of noises and artifacts. As shown in Fig. 7 (h), Fig. 8 (h), and Fig. 9 (h), based on the fusions on spatial directions and spatial planes, our method can conquer the sensitiveness to point singularity, so as to obtain thin, continuous and occlusive edge. Compared with Li's result in Fig. 9 (d), edge of our method shown in Fig. 9 (h) is smoother, though the extracted edge is formed by small line segments at the curve part as a result of line integral.

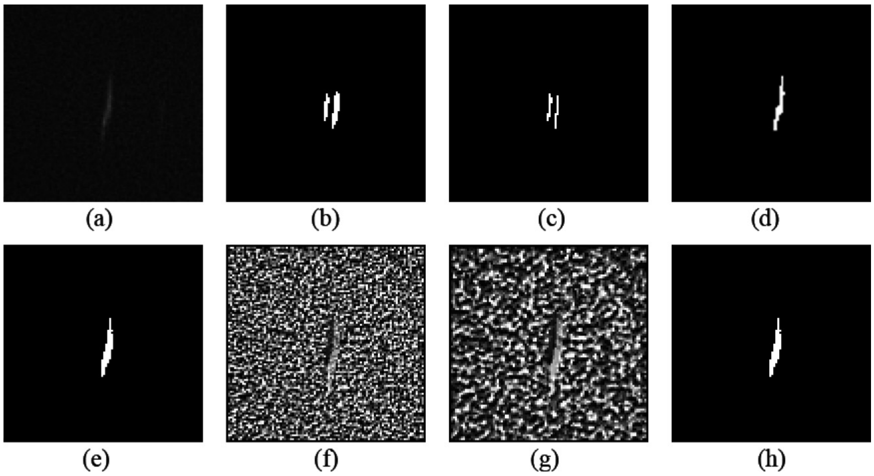

Fig. 9. The ninth slice of 3D volume data. (a) Original slice of crack. (b) Edge slice of 3D wavelet. (c) Edge slice of Facet model. (d) Edge slice of Li's method. (e) The slice of binary data. (f) The slice of data processed by LBP. (g) The slice of data processed by LBP with Gaussian filter. (h) Edge slice of our method.

4.2.2. CT volume data of complicate data

Experiments above have verified the effectiveness of our method for simple volume data, in order to further demonstrate its availability for complicate ones, CT volume data of engine ($256 \times 256 \times 108$) and CT volume data of carburetor ($512 \times 512 \times 167$) are applied in this experiment. The performances of our method applying on these two data are shown in Figs 10 and 11 respectively. The extracted edges presented in 3D snapshots and slices indicate that our method can be used to extract various edges. As can be seen from slices, there is a little difference between the edge of binary data and final edge data. The reason for the difference is that fusion on planes can extract spatial information again so as to correct edge to a certain extent. Please pay attention to the two occlusive edge curves inside the engine, our method also works for curve lines.

From these figures, we know that due to the binarization step, the method proposed in this paper is more suitable for binary phase data. For multiple phase one, we have to modify the binarization step. For example, if we want to extract the brightest region in engine volume data, we may change formula (4) to the following form:

$$b_k(x,y,z) = \begin{cases} 1 & FLIT(x,y,z,k) > m_k + \sigma_k \\ -1 & FLIT(x,y,z,k) < m_k - \sigma_k\,, \quad k = 0,\ldots,4. \\ 0 & \text{otherwise} \end{cases}$$

where σ_k stands for the standard deviation.

4.2.3. Simulation data

In order to know whether the proposed algorithm would work robustly in all cases, our method is also tested on a simulation volume data ($128 \times 128 \times 20$) with Poisson noise. Inside the simulated data, there is an imitated curve crack and an imitated straight crack. Figure 12 (a) shows the 3D snapshot of noisy simulation data, (b) gives a sight of the ninth slice of data without any noise and (c) displays the ninth slice of simulation data with Poisson noise. After being processed by our method, 3D edge of simulated data is shown in Fig. 12 (d). To see more details, the ninth slice of binary data by SD-FLIT is

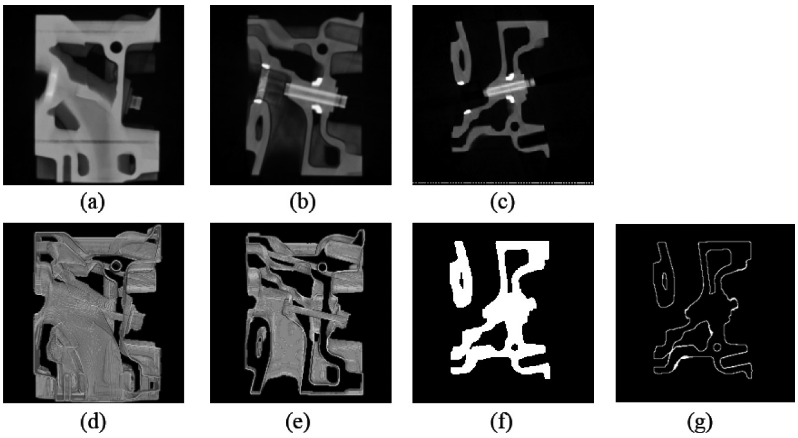

Fig. 10. Edge extraction of CT volume data of engine. (a) Original CT volume data of engine (slices 1–108). (b) Part of CT volume data of engine (slices 1–31). (c) The thirty-first slice of original engine data. (d) 3D edge obtained by our method (slices 1–108). (e) 3D edge obtained by our method (slices 1–31). (f) The thirty-first slice of binary data. (g) The thirty-first slice of 3D edge.

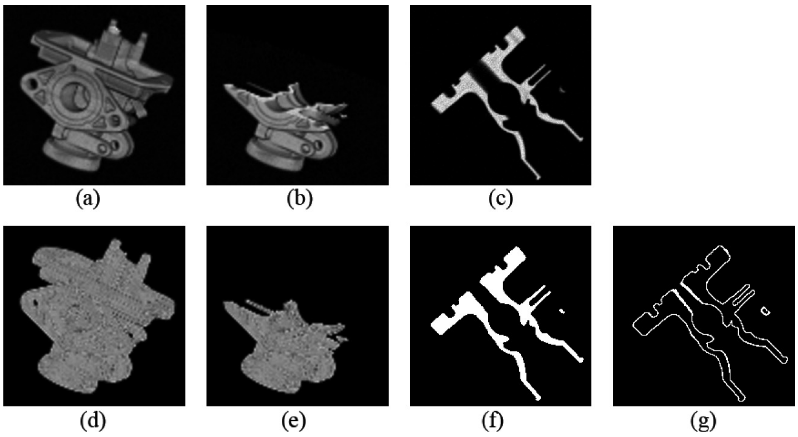

Fig. 11. Edge extraction of CT volume data of carburetor. (a) Original CT volume data of carburetor (slices 1–167). (b) Part of CT volume data of carburetor (slices 1–75). (c) The seventy-third slice of original carburetor data. (d) 3D edge obtained by our method (slices 1–167). (e) 3D edge obtained by our method (slices 1–75). (f) The seventy-third slice of binary data. (g) The seventy-third slice of 3D edge.

presented in Fig. 12 (e) and the ninth slice of final 3D edge is demonstrated in Fig. 12 (f). The volume of region inside obtained 3D edge and the volume of simulated crack are compared in Table 1. Seen from the simulation experiment, our method is robust to noises and has good performance on both curve crack and straight crack.

However, if the noise is too heavy to have appropriate binary data, then the binarization step needs further modification, such as change the mean values to a larger threshold

4.3. Time cost

Table 2 demonstrates a comparison of time consumed by 3D wavelet method, Facet modal and our method. Time costs of LBP alone and LBP with Gaussian filter are not presented because of their terrible

Table 1
Comparison on the Volumes of simulated crack

Simulated crack	The volume of original data (voxel ^{3})	The volume inside 3D edge (voxel ^{3})	Relative error
Curve crack	3380	3400	0.59%
Straight crack	2400	2394	0.25%

Table 2
Time cost for three methods

CT volume data	Size (voxel)	Time (s)		
		3D wavelet	Facet model	Our method
Crack	$100 \times 100 \times 18$	4.55 ± 0.10	6.59 ± 0.05	$\mathbf{0.44 \pm 0.05}$
Engine	$256 \times 256 \times 108$	183.80 ± 0.10	690.10 ± 1.00	$\mathbf{17.53 \pm 0.06}$
Carburetor	$512 \times 512 \times 167$	1119.81 ± 3.06	3733.56 ± 3.64	$\mathbf{122.15 \pm 0.13}$
Simulation	$128 \times 128 \times 20$	8.79 ± 0.03	23.90 ± 0.07	$\mathbf{0.82 \pm 0.04}$

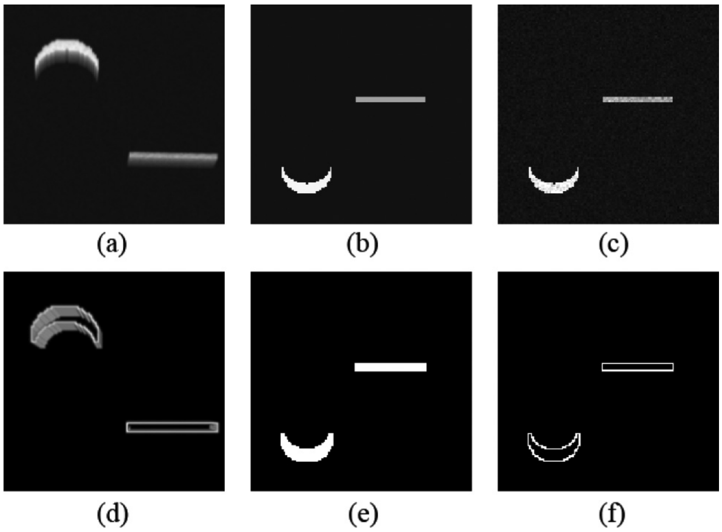

Fig. 12. Edge extraction of simulation volume data. (a) Simulated volume data. (b) The ninth slice of simulated data without noises. (c) The ninth slice of simulated data with Poisson noise. (d) 3D Edge of simulated data by our method. (e) The ninth slice of binary data. (f) The ninth slice of 3D edge data.

image quality. The time cost of Li's method is not listed in the table as well, owing to the absence of being applied to complicate data. We record the time of eight times executions for three methods respectively, then calculate the average time and the limit error. Our method uses the least time, almost 11% of that by 3D wavelet, and 7% of that by Facet models.

To sum up, our method can get occlusive, continuous and thin 3D edge in a rather short time without being affected by noises and artifacts. Whatever efficiency we considering, image quality or time cost, our method performs better than other means. Unfortunately, as the edge description capacity of S function depends on equivalence plane, 3D edge by our method relies on the binary results of SD-FLIT mostly, even with the assistance of fusions on spatial information. Besides, the crack edge gained is approached by small line segments at the curve part, so it is not smooth enough more or less. Moreover, the binarization by SD-FLIT has taken almost half of the time. So SD-FLIT needs further modification or substituted by a method that can use less time to obtain binary data when handling the disturbance of noises and artifacts.

5. Conclusion

To obtain 3D edge of large size CT volume data, we propose a new simple method. This method has made full use of local region information and spatial information, by fusing FLIT values in spatial directions to obtain binary data when smoothing noises and artifacts, and then fusing LBP values on spatial planes to obtain the final 3D edge. Experiments show that 3D edge gained by our method is occlusive, continuous and thin, which provide a good platform for latter processing as defect recognition or image measuring. Owing to the simplicity of algorithm, our method can save at least 89% of the time when compared with 3D wavelet and Facet model. Moreover, experiments on CT volume data of engine and carburetor show that our method is also available for CT volume data with complex structure.

However, the result edge depends on SD-FLIT to a certain extent, besides, all the local information is extracted voxel by voxel, which is not necessary and takes lots of time. So we will go ahead to modify this method to be more independent and efficient.

Acknowledgment

We would like to thank Xiaoyan Liu and Yan Gao for their constructive comments on the manuscript. This work is supported by the National Natural Science Foundation of China under Grant No. 60972104.

References

[1] K. Shen, Y.M. Wen and Y.F. Cai, Efficient x-ray image enhancement algorithm using image fusion, *Journal of X-Ray Science and Technology* **17**(3) (2009), 207–220.
[2] Y. Yang, D.S. Park, S.Y. Huang and J. Yang, Fusion of CT and MR images using an improved wavelet based method, *Journal of X-Ray Science and Technology* **18**(2) (2010), 157–170.
[3] Z.Q. Zhao, D. Steinmann and D. Muller-Zivkovic, A lung area estimation method for analysis of ventilation inhomogeneity based on electrical impedance tomography, *Journal of X-Ray Science and Technology* **18**(2) (2010), 171–182.
[4] M. Celenk, M.L. Farrell, H. Eren et al., Upper airway detection and visualization from cone beam image slice, *Journal of X-Ray Science and Technology* **18**(2) (2010), 121–135.
[5] Z. Yin, K. Khare and B. De Man, Parametric boundary reconstruction algorithm for industrial CT metrology application, *Journal of X-Ray Science and Technology* **17**(2) (2009), 115–133.
[6] L. Papp, R. Juhasz, S. Travar et al., Automatic detection and characterization of funnel chest based on spiral CT, *Journal of X-Ray Science and Technology* **18**(2) (2010), 137–144.
[7] X.H. Zhang, R.L. Ning and D. Yang, Cone beam breast CT noise reduction using 3D adaptive Gaussian filtering, *Journal of X-Ray Science and Technology* **17**(4) (2009), 319–313.
[8] P. Korfiatis, S. Skiadopoulos, P. Sakellaropoulos et al., Automated 3D segmentation of lung fields in thin slice exploiting wavelet preprocessing, *CAIP* **4673** (2007), 237–244.
[9] L. Zeng, H.Y. Guo and R. Ma, Edge detection of industrial CT 3D image based on wavelet locating and Facet model, *Optics and Precision Engineering* **18**(2) (2010), 443–450.
[10] P. Gamage, S.Q. Xie, P. Delmas and W.L. Xu, Segmentation of radiographic images under topological constraints: application to the femur, *Int J Comput Assist Radiol Surg* **5**(5) (2010), 425–435.
[11] T. Ojala, M. Pietikäinen and D. Harwood, A comparative study of texture measures with classification based on feature distributions, *Pattern Recognition* **29**(1) (1996), 51–59.
[12] M. Pietikäinen, T. Ojala and Z. Xu, Rotation-invariant texture classification using feature distributions, *Pattern Recognition* **33**(1) (2000), 43–52.
[13] T. Ojala, M. Piätikainen and T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**(7) (2002), 971–987.
[14] T. Ahonen, A. Hadid and M. Pietikäinen, Face recognition with local binary patterns, *The European Conference on Computer Vision, LNCS* **3021** (2004), 469–481.
[15] W.H. Ma, L. Huang and C.P. Liu, Advanced local binary pattern descriptors for crowd estimation, *Pacific/Asia Workshop on Computational Intelligence and Industrial Application* (2008), 958–962.

[16] Y.C. Fang and Z. Wang, Improving LBP features for gender classification, *Proceedings of the 2008 International Conference on Wavelet Analysis and Pattern Recognition*, HongKong, (2008), 373–377.
[17] L. Setia, A. Teynor, A. Halawani and H. Burkhardt, Grayscale medical image annotation using local relational features, *Pattern Recognition Letters* **29** (2008), 2039–2045.
[18] H.H. Shen, Q.S. Zhang and D.K. Yang, Adaptive local binary patterns for 3D face recognition, *Proceedings of the 2009 Chinese Conference on Pattern Recognition and the First CJK Joint Workshop on Pattern Recognition* (2009), 533–536.
[19] A. Hadid and M. Pietikäinen, Combining appearance and motion for face and gender recognition from videos, *Pattern Recognition* **42** (2009), 2818–2827.
[20] D. Huang, M. Ardabilian, Y. Wang and L. Chen, Automatic asymmetric 3D-2D face recognition, *ICPR* (2010), 1225–1228.
[21] H.L. Tang, Y.F. Sun, B.C. Yin et al., Expression-robust 3D face recognition using LBP representation, *ICME* (2010), 334–339.
[22] R. Gupta, H. Patil and A. Mittal, Robust order-based methods for feature description, *CVPR* (2010), 334–341.
[23] M.Q. Yang, Y.H. Peng and Y.X. Liu, The algorithm and application of finite line integral transform, *IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies Proceedings* (2005), 411–414.
[24] Z.J. Li and L. Zeng, 3D-FLIT and its application to linear feature extraction of ICT cubic data, *Optics and Precision Engineering* **17**(12) (2009), 3089–3095.
[25] Z.J. Li, L. Zeng, X.B. Zou and C.B. Xiang, Crack surface extraction of industrial CT volume data using FPIT and planelet, *Journal of X-Ray Science and Technology* **19**(1) (2011), 1–12.
[26] http://www.volren.org/[EB/OL].
[27] Institute of Automation Chinese Academy of Science, Medical Image Processing Group. 3DMed Software. http://www.3dmed.net.