Optics & Laser Technology 49 (2013) 166–172

Contents lists available at SciVerse ScienceDirect

Optics & Laser Technology

journal homepage: www.elsevier.com/locate/optlastec

An improved method of 3D contours extraction from industrial computed tomography volume data

Haina Jiang [a,c,d], Li Zeng [a,b,*], Bi Bi [a,c], Xiangyu Yang [e]

[a] ICT Research Center, Key Laboratory of Optoelectronic Technology and System of the Education Ministry of China, Chongqing University, Chongqing 400044, China
[b] College of Mathematics and Statistics, Chongqing University, Chongqing 401331, China
[c] College of Optoelectronic Engineering, Chongqing University, Chongqing 400044, China
[d] College of Electronic Engineering, Chongqing 401331, China
[e] Mathematics & Econometrics College of Hunan University, Changsha 410082, China

A R T I C L E I N F O

Article history:
Received 2 November 2012
Received in revised form
21 December 2012
Accepted 2 January 2013
Available online 31 January 2013

Keywords:
Contour
Fuse
Stationary wavelet transform

A B S T R A C T

An improved method of 3D contours extraction is presented in this paper. To simplify and improve the two steps operation of traditional wavelet algorithm, an improved method for 2D contours extraction called 2D-IMCE is developed based on stationary wavelet transform. This method can gain complete, fine and continuous contours by only one step through automatically percolating non-contour points without thresholding and human–machine interactions. To inherit superiority of 2D-IMCE, an extended way of it for 3D contours extraction named Fuse-2D-IMCE is derived. It utilizes satisfactory 2D contours to approximate 3D ones based on fusion operation. Firstly, industrial CT volume data are divided into three groups of slices corresponding to three mutual perpendicular directions. Secondly, in each slice group, 2D contours are extracted by 2D-IMCE. Finally, 3D contours are obtained by fusing all 2D ones of three directions. Experimental results demonstrate that the Fuse-2D-IMCE method can effectively extract 3D contours from industrial CT volume data.

1. Introduction

Technology of industrial computed tomography (CT) adopts X-rays to penetrate tested materials, and reflects the construction or inner constitution by various features extracted from CT volume data [1]. For example, in the reverse engineering based on industrial CT, the work-piece manufacture error can be calculated by registering contours [2] extracted from CT volume data with that of CAD model. In this paper, we will focus on contours extraction.

The study of contours extraction is significant in measuring technique, pattern recognition and computer vision [3–5]. In the former work, we have studied 2D contours [6]. With the increase of volume data, the research on 3D ones becomes necessary. At present, there are many methods of 3D contours extraction. For example, Bi et al. [7] combined improved finite line integral transform (FLIT) and local binary pattern (LBP) to extract 3D contours from volume data. Xu and Zhao [8] employed a novel 3D B-spline active contour to converge to the object's 3D surface boundary. Quan et al. [9] used the optical fringe projection method to extract 3D surface profile. The methods above are appealing for their own special characteristics. However, the direct extraction technology for 3D contours is young though more attention is being paid to it [10]. Compared with it, the technique of 2D ones is more mature. Then, many methods for 3D contours extraction based on 2D ones appeared. For instance, Gamage et al. [11] achieved the radiographic image segmentation under topological control by incorporating 2D image analysis. Zeng et al. [12] obtain 3D linear feature of volume data based on 2D Wedgelet. Wen et al. [13] identified features of volume data based on 2D engineering drawings. Leandro et al. [14] described a methodology for automatic contour extraction from 2D images of 3D neurons.

By drawing inspiration from the methods above, an improved method of 2D contours extraction based on stationary wavelet transform (SWT) called 2D-IMCE (an improved method of 2D contour extraction) is developed. This method is motivated by simplifying and improving the traditional wavelet algorithm, and then obtaining satisfactory 2D contours.

The traditional wavelet algorithm [15] often extracts contours by two steps. The first one is thresholding wavelet coefficient and the second is edge tracking. It is famous for good time–frequency property, less noise susceptibility and wide application. However, the threshold can affect the accuracy of contour position because a large threshold could lead to discontinuous contours, while

* Corresponding author. Tel.: +86 23 8639 4828;
fax: +86 23 6510 3562.
E-mail address: drlizeng@hotmail.com (L. Zeng).

0030-3992/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.optlastec.2013.01.004

a small one may get rough ones. Moreover, additional edge tracking is necessary for continuous contours, while the resulting contours may not always be satisfactory due to different schemes of tracking and limited information of binary image. To overcome such shortages and considering that industrial computed tomography images have a little noise, the 2D-IMCE method is developed. It removes process of thresholding to weaken the influence of subjective factor. Then, it merges contour extraction and edge tracking into one step to improve the algorithm efficiency. Through that, 2D-IMCE can automatically gain satisfactory contours by only one step but without thresholding and human–machine interactions. To further obtain good 3D contours, we combine 2D-IMCE with fusion operation named Fuse-2D-IMCE to present a way of 3D contours extraction.

The rest of this paper is organized as follows: in Section 2, 2D-IMCE is presented. Section 3 shows fusion operation. And detailed Fuse-2D-IMCE is exhibited in Section 4. In Section 5, some experiments are conducted on real industrial CT images. And Section 6 concludes this paper.

2. Presentations of 2D-IMCE

2.1. Stationary wavelet transforms (SWT)

SWT [16], which was presented by Nason and Silverman, is achieved by applying appropriate high and low pass filters to the input data at each level, then, two sequences at the next level can be produced. There is no decimation so that the two new sequences have the same length as the original one.

Some frames of SWT are shown in Fig. 1. Where, l_i is the low-pass filter at level i, h_i is the high-pass filter at level i. cA_j is the input image. Output cA_{j+1} is approximate sub-band, $cD^{(h)}_{j+1}$ is the horizontal one, $cD^{(v)}_{j+1}$ is the vertical one and $cD^{(d)}_{j+1}$ is the diagonal one. All of them have the same size as that of input image. For Fig. 1(b), l_{i+1} and h_{i+1} are filters of the next level after up sampling l_i and h_i respectively.

SWT is outstanding for its redundancy and time invariance, whose properties are particularly important in detection or estimation problems [17]. Additionally, the corresponding output does not need enlargement or interpolation. Compared with it, discrete time transform (DWT) has no characteristics. Contrast with continuous wavelet transforms (CWTs), SWT can be easily implemented by filter banks and reduce computational complexity. Based on considerations above, we adopt SWT to calculate the modulus matrix of tested image. By borrowing idea of Mallat and Huang [15], the SWT modulus at point (x,y) can be computed by

$$Mod_{SWT}(x,y)=\sqrt{[cD^{(h)}_{j+1}(x,y)]^2+[cD^{(v)}_{j+1}(x,y)]^2} \quad (1)$$

where $cD^{(h)}_{j+1}(x,y)$ and $cD^{(v)}_{j+1}(x,y)$ are horizontal and vertical components of SWT at point (x,y), respectively. They can be calculated by

$$cD^{(h)}_{j+1}(x,y)=\sum_{x=-\infty}^{+\infty}\sum_{y=-\infty}^{+\infty}h(x)l(y)cA_j(x,y) \quad (2)$$

$$cD^{(v)}_{j+1}(x,y)=\sum_{x=-\infty}^{+\infty}\sum_{y=-\infty}^{+\infty}l(x)h(y)cA_j(x,y) \quad (3)$$

where l and h are low-pass filter and high-pass filter respectively. To keep the accuracy of gray level, the modulus matrix will not be normalized. Obviously, as long as pixels not belonging to contours can be percolated, ideal 2D contours can be extracted.

2.2. 2D-IMCE algorithm

Now, we take a slice image I with $Width \times Height$ for example to demonstrate 2D-IMCE. It is given that SWT modulus matrix of image I is Mod_{SWT}. *Num_contour* is the contour number of image I, which needs to be specified by us. *Num_one* is the number of pixel with gray value '1'.

Step 1: Initialize parameters: circulating counter: $n=0$; contour points recording matrix: $contour\_2D=0_{Width\times Height}$, for contour points, we mark them with gray value '1', otherwise, with '0'; recording array for coordinate of the current point: $current\_point=[0,0]$; recording array for coordinate of the next point: $next\_point=[0,0]$.

Step 2: Search candidate starting contour point: the pixel with the highest modulus value is assumed as candidate starting contour point denoted as P_{start}. If the number of P_{start} is more than one, we choose any one as P_{start}. It is supposed that (x_0,y_0) is the coordinate. Then, we mark $Mod_{SWT}(x_0,y_0)=0$ to prevent P_{start} from being chosen again.

Step 3: Obtain satisfactory starting contour point: if candidate P_{start} is at or over the image margin, we abandon it and turn to step 2 to find other. Otherwise, we judge *Num_one* in the eight-neighbor of candidate P_{start} in matrix *contour_2D*:

(a) If *Num_one* is not smaller than three, candidate P_{start} is redundant. We abandon it and turn to step 2 to find another.

(b) If not, candidate P_{start} is satisfactory one. We record it by $contour\_2D(x_0,y_0)=1$ and $current\_point=[x_0,y_0]$.

Step 4: Choose the next point in the eight-neighbor of determined *current_point*, we choose one point with *largest_Mod* (the largest Mod_{SWT} value) to be the next point. Sometimes, the number maybe more than one, so some decisions have to be made:

(a) If *largest_Mod* is zero, it means that there are no contour points any more among the eight-neighbor pixels of determined *current_point* or a piece of contour has been extracted. Then, we stop step 4 and go to step 6.

(b) If not, we choose any one as the next point.

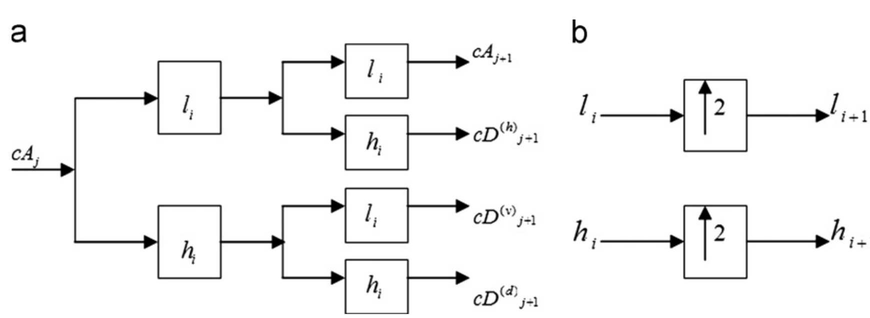

Fig. 1. Frames of SWT: (a) decomposition frame of SWT and (b) upsampling operation of filters.