

# A Novel Approach For 3D Surface Reconstruction Of LV Endocardium Based On SPCNN And MC

Ruoming Lei, Yurun Ma, Kemin Wang, Yide Ma\*

School of Information Science and Engineering, Lanzhou University

Lanzhou, Gansu 730000, China

E-MAIL: ruomlei@163.com, wysysanab@126.com, kimwangs@163.com

**Keywords:** 3D surface reconstruction, Left ventricle, Cubic Spline Interpolation, SPCNN segmentation, MC

**ABSTRACT.** The auxiliary diagnosis based on three-dimensional (3D) modeling techniques can improve the diagnosis efficiency of heart disease. In this paper, we propose a new method for the 3D modeling of left ventricle. The novel approach incorporates two key points: one is LV endocardium segmentation via the simplified pulse-coupled neural network (SPCNN), and the other is interpolation between slices. To ensure the accuracy of modeling, we firstly use the cubic spline interpolation to increase the resolution between slices. Then, the endocardial contours of all slices are automatically obtained based on SPCNN. Finally, Marching Cubes (MC) algorithm is implemented to reconstruct the LV endocardial surface. We test the new method on four sets of 3D data and obtain effective results. The APD and ODM are respectively 1.7950mm and 91.34% for the endocardium segmentation, which makes the surface reconstruction of LV endocardium accurate and smooth.

## INTRODUCTION

It is highly subjective for physicians to obtain clinical diagnostic information from a large number of two-dimensional slices of the heart. Considering that these subjective imaginations and judgments may lead to uncertainty, 3D reconstruction and visualization based on two-dimensional slices was put forward to provide a more objective and realistic morphogenesis for doctors [1]. In addition, the LV reconstruction is the basis for the analysis of cardiac electrophysiology and motion. Thus, LV reconstruction is always a research hot spot.

Recovering three-dimensional structure of object from 3D data includes two main methods: volume rendering [2] and surface rendering [3], which can be implemented by a lot of algorithms. The volume rendering can draw image directly from 3D data field, which is the dataset of uniformly spaced sample points in three-dimensional space, and the minimum unit is voxel. The central idea of volume rendering is to assign an opacity to every voxel and consider the effects of transmission, reflection and emission that voxels will have on light [4]. The method can be divided into four steps in principle: projection, blanking, rendering and compositing [5]. Commonly used algorithms for volume rendering are Ray-Casting [6], Splatting [7], and Shear-Wrap [8]. Dealing with the relationship between light and voxel directly, the volume rendering has the advantage of retaining much detail information of the object and improving the quality of image. Its drawback is that computation cost is too large. The basic idea of surface rendering is to conduct edge segmentation from a series of two-dimensional image slices, extract relevant surface information and restore three-dimensional model of the object through lighting and shading [9]. Surface rendering mainly includes two categories. One is the contour reconstruction based on boundary [10], which requires the contour of the particular area within every two-dimensional image plane. It connects the contour vertices via triangle, and piles up the triangles to form the surface of the object. The other is isosurface reconstruction based on the voxel, which extracts isosurface from data field. Isosurface refers to a surface in space, and voxel is the basic hexahedron unit that constitutes the three-dimensional image. Algorithms of isosurface reconstruction mainly comprise Marching Cubes [11] and Dividing Cubes [12]. Although contour reconstruction has the advantage of simple and

facilitating real-time rotation operation, the quality of its result is poor and not intuitive enough. Isosurface reconstruction, in contrast, behaves better in image quality, but it is relatively slow in image drawing.

Methods proposed in recent years about LV reconstruction by many researchers. Zhang et al.[13] proposed a method that creates a 3D object mesh from a 2D stack of contours, based on geometry processing algorithms and a semi-constrained deformation method. Wang et al.[14] presented a method for left ventricle (LV) endocardium motion reconstruction using high resolution CT data and tagged MRI.

In this paper, we derive a framework that creates the 3D surface of LV endocardium based on SPCNN segmentation method and cubic spline interpolation. Firstly, we use the cubic spline interpolation to eliminate the drawback that the resolution along z-axis is far less than that of X-Y plane. Secondly, SPCNN model was used to extract LV endocardium contour. Finally, we finish the 3D surfacing reconstruction by applying MC algorithm. Rest of this paper is organized as follows: Section 2 introduces the theories employed in this paper; Section 3 describes the LV endocardium segmentation and surface reconstruction in detail; Section 4 shows the experiment results; Section 5 conclude our work.

## THEORY

### SPCNN model

PCNN is a bio-inspired neural network, which was developed in light of synchronous dynamics of neuronal activity in cat visual cortex[15]. PCNN has great potential for developing image segmentation algorithm [16-18]. The SPCNN model used in our work is described as follows:

$$U_{ij} = e^{-\alpha_f} U_{ij}[n-1] + S_{ij} (1 + \beta V_L \sum_{kl} W_{ij,kl} Y_{kl}[n-1]) \tag{1}$$

$$E_{ij}[n] = e^{-\alpha_e} E_{ij}[n-1] + V_E Y_{ij}[n] \tag{2}$$

$$Y_{ij}[n] = \begin{cases} 1, & \text{if } U_{ij}[n-1] > E_{ij}[n-1] \\ 0, & \text{else} \end{cases} \tag{3}$$

$F_{ij}$  is the feeding inputs, which is the normalized gray intensity of  $S_{ij}$ ,  $Y_{kl}$  and  $L_{ij}$  represent the outputs pulse and linking inputs around the neural cell respectively. Linking strength is  $\beta$ . The linking inputs are biased and then multiplied together, and further multiplied with the feeding input to form the total internal activity  $U_{ij}$ .  $E_{ij}$  is the neuromime threshold,  $v_o$  and  $\alpha_o$  set the operation of the neuromime. Input an image into SPCNN, it will produce a series of binary images by using different parameters.

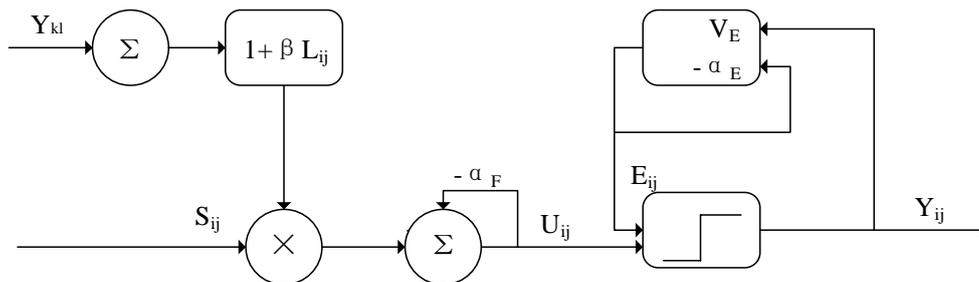


Figure 1 Flowchart of SPCNN

### Cubic spline interpolation

In order to produce a smooth surface after 3D reconstruction, we firstly increase the resolution between slices via recovering interlayer images. Considering the accurate and smoothness of the image interpolation result, here, we use the cubic spline interpolation[19]which is a kind of piecewise interpolation. When providing function values from 0 to  $n$  nodes and two boundary conditions, we can get interpolation function having continuous first and second order derivative at the nodes. Using it can better adapt to different requirements of smoothness and restore the interlayer missing images accurately.

We assume the gray value of point  $(u, v)$  on the  $i^{\text{th}}$  image to be  $f_i(u, v)$ , the cubic spline interpolation function  $y(x)$  calculated by three moment method[20] is as follows:

$$y(x) = M_i \frac{(x_{i+1} - x)^2}{6h_i} + M_{i+1} \frac{(x - x_i)^2}{6h_i} + \left( f_i(u, v) - \frac{M_i h_i^2}{6} \right) \left( \frac{x_{i+1} - x}{h_i} \right) + \left( f_{i+1}(u, v) - \frac{M_{i+1} h_i^2}{6} \right) \left( \frac{x - x_i}{h_i} \right)$$

$$y''(x_i) = M_i \quad y''(x_{i+1}) = M_{i+1} \quad h_i = x_{i+1} - x_i \quad (i = 0, 1, 2, \dots, n-1)$$

$$x \in [x_i, x_{i+1}] \quad (i = 0, 1, 2, \dots, n-1)$$
(4)

$x_i$  is the actual physical coordinate of the  $i^{\text{th}}$  image,  $x$  is the actual physical coordinate of the image to be interpolated.

### Marching Cubes

The basic idea of MC algorithm is to process the cubes in the data field one by one (Figure 2).

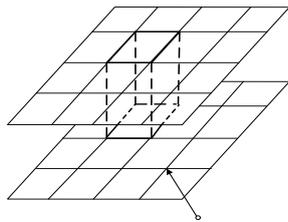


Figure 2. Marching Cube

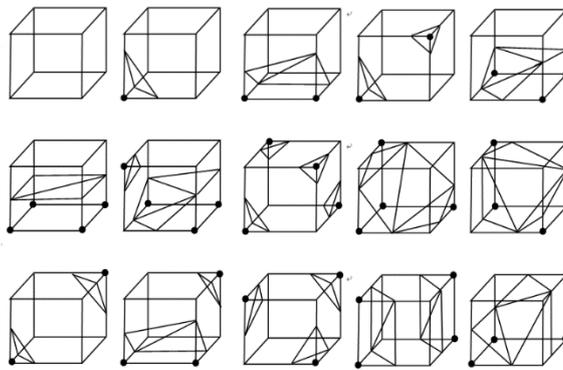


Figure 3. Triangulated Cubes

If the gray value at a vertex is greater than or equals to the value of the surface that we are constructing, we specify this vertex as one. These vertexes are inside (or on) the surface. On the contrary, the vertexes whose gray value are lower than surface will be specified as zero and are outside the surface. The surface will intersect with the cube's edge only when one vertex is greater than the value of the surface and the other is less than the surface. This assumption is the foundation of determining the topology of the surface within the cube and intersection location later.

Since a cube has eight vertexes and each vertex has two states: inside the surface and outside the surface, a cube can have  $2^8=256$  states in total. Applying complementary and rotational symmetry[21], we reduce the 256 states to 14 patterns, Figure 3 show the triangulation for the 14 patterns.

We can create an edge table which represents the connection mode of triangle by numbering for 8 vertexes and 12 edges of the cube, as shown in Figure 4. The edge table includes 256 items, and each of them is a 12-bit binary value. Each bit of binary number corresponds to an edge of the cube.

Zero indicates that the edge doesn't intersect with the isosurface, while one represents intersection. The eight-bit binary index corresponds to the cube's 8 vertexes which are specified as one or zero by comparing with the isosurface. Index= $v_7v_6v_5v_4v_3v_2v_1v_0$ . This index will be used as a pointer into the edge table, from which we can tell which edge the surface intersects.

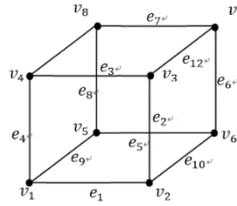


Figure 4. Cube Numbering

After we determine the topology of the surface within the cube, the final step is to calculate the intersection location that the triangle intersects with the cube's edge. In order to get the image with reality, the normal for each triangle vertex is also necessary [21].

### METHOD

LV surface reconstruction involves three steps: 1) cubic spline interpolation between slices 2) automatic segmentation of LV endocardium via SPCNN 3) LV endocardium surface reconstruction using Marching Cubes. Figure 5 shows the workflow of the SPCNN and MC-based algorithm and the details are described in the following sections

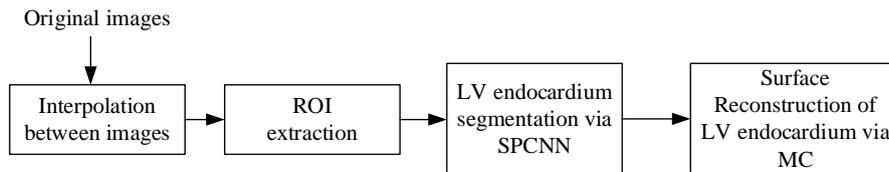


Figure 5. Workflow of the SPCNN and MC-based algorithm

### Cubic spline interpolation between slices

The number of original image slices used in this text is relatively small, leading to aliasing and edge blurring. In order to obtain better three-dimensional effect, interpolation between original images was done before reconstruction.

The research data used in this text are DICOM image sequences, the images of this kind of format contain many other information besides gray values. The number of slices to be interpolated between two adjacent layers  $n$  is determined by layer spacing  $\Delta_z$  and pixel spacing  $\Delta$ , which can

be acquired from DICOM information. The calculation formula is  $n = \text{int} \left\{ \frac{\Delta_z}{\Delta} \right\} - 1$ .

After knowing the number of slices to be inserted, we can get  $n$  intermediate layers between two adjacent slices. The original data sets are a stack of parallel slices, and each of them are consist of same rows and columns of pixels. So the method of getting the interpolated slices are as follows:

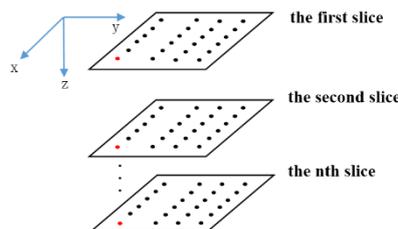


Figure 6. N slices of original data sets

Step 1: we select the pixels whose coordinate are  $(i, j)$  in each slice from the first to  $n^{\text{th}}$  to form a one-dimensional vector along the z-axis. As shown in Figure 6, the red dots along z-axis constitute a one-dimensional vector.

Step 2: we can conduct the cubic spline interpolation on this one-dimensional vector to form new sampling points along z-axis.

Step 3: Traversing the pixels on the x-y plane of all slices as step 1 and step 2, we will get all the intermediate slices. Among the images shown in Figure 7, (a) and (d) are two adjacent raw images, (b) and (c) are the images formed by cubic spline interpolation.

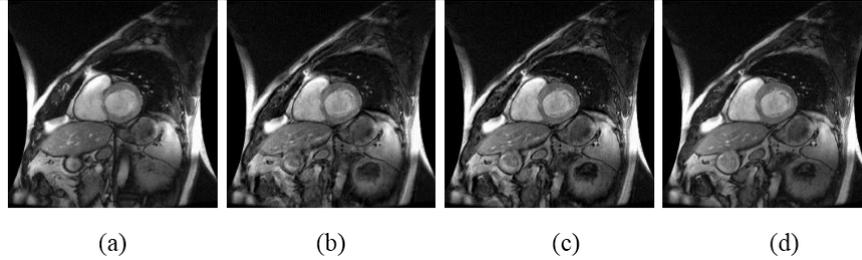


Figure 7. Raw images and Images by cubic spline interpolation

### LV endocardium segmentation via SPCNN

For endocardium segmentation, we have proposed an SPCNN based method in [22]. Firstly, the ROI is extracted based on the time-domain characteristics of 3D data to decrease the computation and complexity of our algorithm. Secondly, we input each slice image into SPCNN model and set the SPCNN parameters as follows:  $W = [0.5, 1, 0.5; 1, 0, 1; 0.5, 1, 0.5]$ ,  $\beta = 0.1$ ,  $V_E = 20$ , and  $\alpha_E$  is adaptively set by using  $\alpha_E = C/\text{mean}(S)$  [23],  $C$  is 0.34, and “mean(S)” represents the calculation of the average gray level. After obtaining the binary image from SPCNN segmentation, we detect the LV cavity and delineate the fine endocardium by using some post-processing procedures. The endocardium segmentation results are shown in Figure 8. The red closed curve is the obtained endocardial contour.

Using the endocardial contour, we can extract the LV blood pools (shown in Figure 9) which will be used in the next 3D reconstruction.

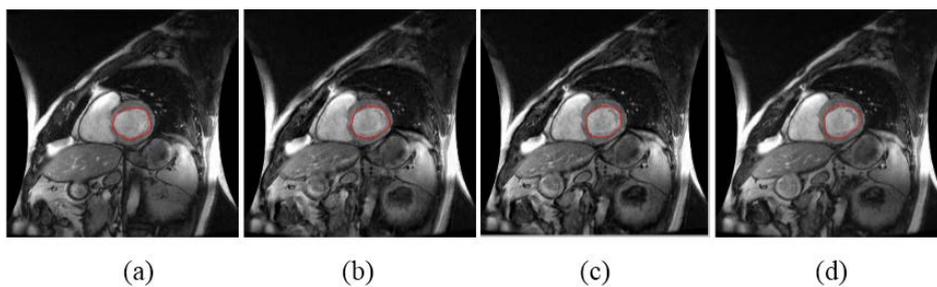


Figure 8. Segmentation result of LV endocardium

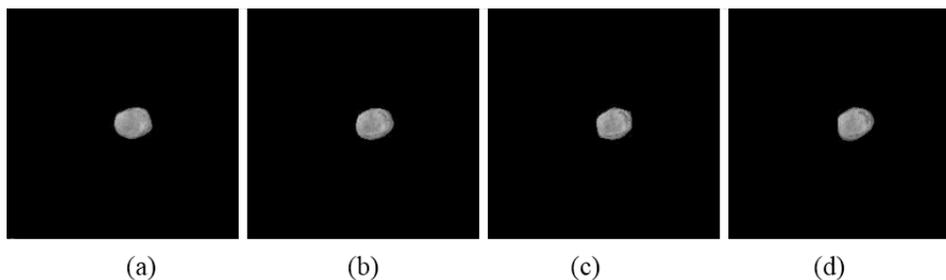


Figure 9. Result of extraction of endocardial area

Once we obtain the endocardial area of all the slices, MC algorithm can be implemented on all slice sequence to reconstruct the surface of LV endocardium.

## RESULTS

The database we use are provided by MICCAI 2009, containing 45 cases of data. Each case has 6~12 slices and every slice corresponds to 20 different phases. Our experiment is implemented on a part of cases from training datasets by using PC(2.8GHZ CPU,4GB RAM) and Matlab R2013b.

In order to indicate the accuracy of SPCNN based endocardium segmentation, we compare four sets of segmentation results with the ground truths which are manually delineated by expert. And, the obtained APD and ODM is listed in table 1. The definition of APD and ODM are introduced in[24].

Table 1 APD and ODM

Patient id	APD[mm]	ODM
SC-N-02	2.5286	0.8317
SC-HF-I-05	1.4648	0.9397
SC-HF-NI-07	1.8823	0.9282
SC-HF-I-10	1.3042	0.9559
Average value	1.7950	0.9134

We take the dataset of SC-N-02 as an example to show the result of LV endocardium surfacereconstruction. From the DICOM information, we know that layer spacing between two slices is 5.2mm, and pixel spacing is 1.3672mm. According to  $n = \text{int} \left\{ \frac{\Delta_z}{\Delta} \right\} - 1$ , the number of layers needed to be interpolated between two adjacent slices is 2 and we know the number of original images is 9. So the total number of slices after interpolation is 25, and the endocardium areas obtained by SPCNN segmentation are shown in Figure 10.

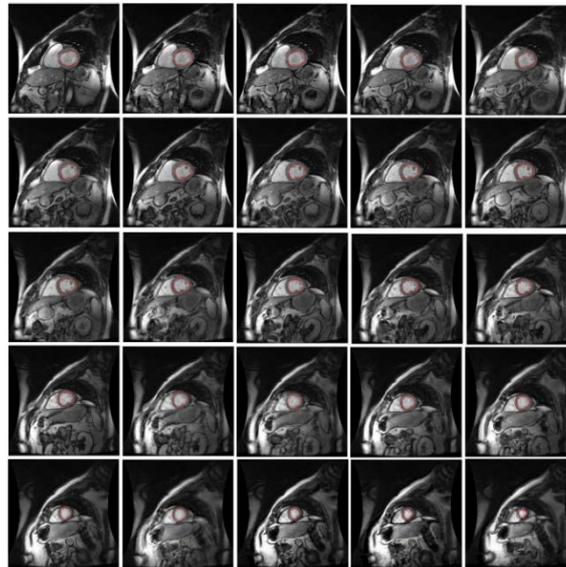


Figure 10. Endocardium areas encircled by red contour-line

The results of 3D surface reconstruction of LV endocardium based on SPCNN and MC algorithm and contour algorithm are illustrated in Figure 11 and Figure 12 respectively. From Figure 10 and Figure 11, we know that the surface obtained by SPCNN and MC algorithm is more intuitive and better in quality than that by the contour algorithm. Surface reconstruction based on contour algorithm is largely influenced by the number of contour points. The denser the contour points are, the better reconstruction result we will obtain.

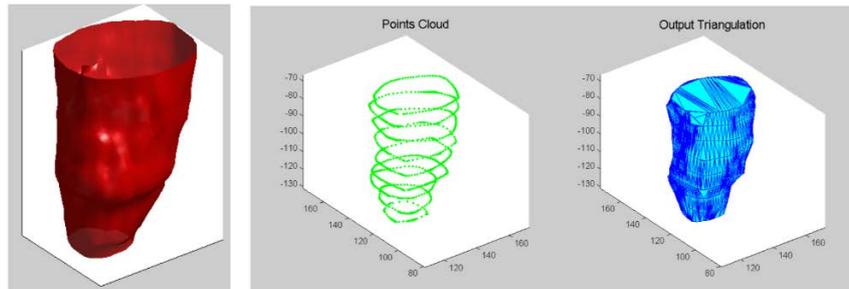


Figure 11 SPCNN

Figure 12 MC

## CONCLUSION

In this paper, a complete process of LV surface reconstruction was introduced, involving three key steps, namely cubic spline interpolation between slices, segmentation of LV endocardium via SPCNN and LV surface reconstruction based on MC. The final result demonstrates that 3D surface reconstruction based on SPCNN and MC algorithm has a better reconstruction performance. In the future, we will carry 3D reconstruction on more data, facilitating the real-time display of the left ventricular endocardial surface in a cardiac cycle. Meanwhile, systolic function of left ventricular will be also analyzed and evaluated according to the results of 3D reconstruction.

## REFERENCES

- [1] Y. Huang, Z. Qiu, and Z. Song, "3D reconstruction and visualization from 2D CT images." 153-157.
- [2] R. A. Drebin, L. Carpenter, and P. Hanrahan, "Volume rendering," *Acm Siggraph Computer Graphics*, 22(4), 65-74 (1988).
- [3] U. Tiede, K. H. Hoehne, M. Bomans *et al.*, "Surface Rendering," *IEEE Computer Graphics & Applications*, 10(2), 41-53 (1990).
- [4] N. Max, "Optical Models for Direct Volume Rendering," *IEEE Transactions on Visualization & Computer Graphics*, 1(2), 99-108 (1995).
- [5] H. U. Junhua, "New Research Progress of Medical Image 3d Reconstruction Technology," *China Science & Technology Information*, (2013).
- [6] M. Levoy, "Display of surfaces from volume data," *IEEE Computer Graphics & Applications*, 8(3), 29-37 (1988).
- [7] Z. G. Sun, J. W. Zhang, and J. Z. Sun, "An Improvement of Volume Splatting Algorithm," *Journal of Tianjin University*, (2003).
- [8] P. Lacroute, "Fast volume rendering using a shear-warp factorization of the viewing transformation." 451-458.
- [9] Y. Z. Zhang, Y. B. Li, Y. H. Jiang *et al.*, "Three-dimensional reconstructive methods in the visualization of anterolateral thigh flap," *Surgical & Radiologic Anatomy Sra*, 30(1), 77-81 (2008).
- [10] Z. Bin, H. Q. Feng, H. Q. Zhou *et al.*, "Fast 3D Surface Rendering for CT or MR Image on a Personal Computer," *Space Medicine & Medical Engineering*, 15(4), 296-299 (2002).
- [11] L. Zhong, L. I. Lei, J. H. Sun *et al.*, "Preoperative diagnosis of gastric cancer using 2-D magnetic resonance imaging with 3-D reconstruction techniques," *Chinese Journal of Digestive Diseases*, 6(4), 159-164 (2005).
- [12] H. E. Cline, S. Ludke, and W. E. Lorensen, [Dividing cubes system and method for the display of surface structures contained within the interior region of a solid body] US, (1988).

- [13] S. Zhang, M. Uzunbas, Z. Yan *et al.*, "Construction of Left Ventricle 3D Shape Atlas from Cardiac MRI." 88-94.
- [14] X. Wang, V. Mihalef, Z. Qian *et al.*, "3D cardiac motion reconstruction from CT data and tagged MRI." 4083-6.
- [15] R. Eckhorn, H. J. Reitboeck, M. Arndt *et al.*, "Feature Linking via Synchronization among Distributed Assemblies: Simulations of Results from Cat Visual Cortex," *Neural Computation*, 2(3), 293-307 (1990).
- [16] D. Zhou, H. Zhou, C. Gao *et al.*, "Simplified parameters model of PCNN and its application to image segmentation," *Formal Pattern Analysis & Applications*, 1-13 (2015).
- [17] W. Xie, Y. Li, and Y. Ma, "PCNN-based level set method of automatic mammographic image segmentation," *Optik - International Journal for Light and Electron Optics*, 127(4), 1644-1650 (2015).
- [18] A. K. Helmy, and G. S. El-Taweel, "Image segmentation scheme based on SOM-PCNN in frequency domain," *Applied Soft Computing*, 40, 405-415 (2016).
- [19] S. H. Hong, L. Wang, T. K. Truong *et al.*, "Novel approaches to the parametric cubic-spline interpolation," *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, 22(3), 1233-41 (2013).
- [20] X. U. Xiao-Yong, and T. Y. Zhong, "Construction and Realization of Cubic Spline Interpolation Function," *Ordnance Industry Automation*, (2006).
- [21] W. E. Lorensen, and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," *Acm Siggraph Computer Graphics*, 21(4), 163-169 (1987).
- [22] Y. Ma, W. Li, Y. Ma *et al.*, "An SPCNN-GVF-based approach for the automatic segmentation of left ventricle in cardiac cine MR images," *International Journal of Computer Assisted Radiology & Surgery*, 1-14 (2016).
- [23] S. Wei, Q. Hong, and M. Hou, "Automatic image segmentation based on PCNN with adaptive threshold time constant," *Neurocomputing*, 74(9), 1485-1491 (2011).
- [24] P. Radau, Y. Lu, K. Connelly *et al.*, "Evaluation framework for algorithms segmenting short axis cardiac MRI," *The MIDAS Journal-Cardiac MR Left Ventricle Segmentation Challenge*, 49, (2009).