

# Application of Marching Square Algorithm in 2D Tensor Voting

Xiaofang Shao, Dalong Li, Yan Tang

Qingdao Branch of NAEI, Qingdao Shandong  
Email: [pugongying\\_0532@163.com](mailto:pugongying_0532@163.com)

Received: Apr. 3<sup>rd</sup>, 2015; accepted: Apr. 23<sup>rd</sup>, 2015; published: Apr. 29<sup>th</sup>, 2015

Copyright © 2015 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

Marching Cube is a classic algorithm applied in 3D medical visualization for its simpleness and practicality. Marching Square algorithm is the 2D version of Marching cube, which is used mainly in extracting isolines. Tensor voting algorithm introduces this algorithm to detect general curves. This paper presents the application of Marching Square in 2D tensor voting. Experimental results show its efficiency and the influence of scale parameter.

## Keywords

Marching Square, Perceptual Completion, Curve Detection

---

# 移动正方形算法在2D张量投票中的应用

邵晓芳, 李大龙, 汤 燕

海军航空工程学院青岛校区, 山东 青岛  
Email: [pugongying\\_0532@163.com](mailto:pugongying_0532@163.com)

收稿日期: 2015年4月3日; 录用日期: 2015年4月23日; 发布日期: 2015年4月29日

---

## 摘 要

移动立方体(Marching Cube)算法是三维医学可视化领域应用的经典算法, 因其简单实用而得到广泛推

广。移动正方形算法是移动立方体算法的二维版本，传统应用主要体现在等值线提取方面，如今在二维张量投票中得到了扩展，可用于曲线检测。该文介绍了移动正方形算法在二维张量投票中的应用，并通过实验展示了应用的效果及尺度参数的影响。

## 关键词

移动正方形算法，感知修复，曲线检测

## 1. 引言

根据输入的数据类型，我们可将以往有关曲线、区域和三维表面等特征提取的研究工作归为以下几类：正则化方法(Regularization) [1]、松弛标记法(Relaxation Labeling) [2]、鲁棒性方法(Robust Methods) [3]、水平集方法(Level Set Methods) [4]、聚类(Clustering)方法[5]、基于结构显著性(Structural Saliency)的方法[6] [7]、基于视觉竞争合作机制的组织(Grouping with Cooperative and Inhibitive Fields)方法和用于图像修复的一些方法[8]等。张量投票[7]是一种基于结构显著性的方法，与相关研究工作相比，张量投票方法数据表示方式更为丰富，而且不同结构类型可以在张量投票的计算框架下进行一体化的处理。张量投票方法因其简单实用而获得广泛的应用。

然而，在张量投票方法的特征提取环节，如果仅做投票解释，得到的会是曲线及其周围一定邻域范围内点的曲线特征显著度；如果进行简单的极值搜索，会因忽略各点取向信息而增大曲线点检测的误差。另一方面，移动正方形算法[9]自提出以来，其应用一直局限在等值线提取方面。本文主要介绍移动正方形算法在二维张量投票中的应用，并通过实验说明应用效果及尺度参数的影响。本文主要内容安排如下：第一节是对移动正方形算法的介绍；第二、节描述张量投票方法中应用移动正方形方法的过程和一些细节问题；第三节是实验结果；最后是结束语。

## 2. 移动正方形算法

移动正方形法是一种二维等值线算法[9]，它是移动立方体法(Marching Cube)的基础和依据。移动立方体法正是移动正方形法的三维引申发展起来的，因其简单实用而得到广泛推广，在生物化学、生物医学、可变形建模、数字雕刻、环境科学、机械动力学、三维可视化及算法分析等领域均有应用[10]。

移动正方形法首先找四个相邻的像素，图 1 给出了阈值为 5 的一个例子。每个像素值有大于阈值和小于阈值两种情况，如果像素值大于阈值用实心圆点表示，如果小于阈值就用圆圈表示。四个点就有 16 种组合形式，图 2 列出了所有的可能组合形式。每一种形式就是等值线与正方形边之间的一种拓扑关系。图 2 中的连线就是等值线的路径。没有连线的形式说明等值线不与正方形相交。以图 2 中 case 1 为例，该图中左下角的像素值大于给定值，其它三个像素小于给定值，那么可以推断出等值线的一侧是圆圈代表的像素，另一侧是另外三个像素，等值线只能以图中连线所示的这种方式与正方形相交。等值线与正方形边的交点坐标可以用线性插值求得。这样当一幅图像中的所有正方形都求出了各自的一段等值线后，这些线段自然而然的就连成一个闭合的等值线。移动正方形算法如下：

- 1) 在一幅图像中求出所有四个相邻像素点构成的正方形。
- 2) 判断四个像素值与阈值的关系，生成 0101 的代码。
- 3) 由上步生成的代码按照图 3 所示的关系求出等值线与四个像素点间的拓扑关系。
- 4) 由拓扑关系，用线性插值法求出等值线与正方形边的交点。
- 5) 顺序连接等值线段得到等值线。

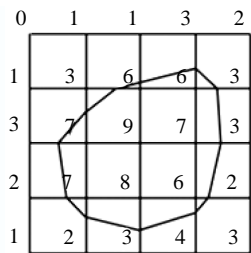


Figure 1. An example for Marching Square's isoline extraction  
图 1. 移动正方形等值线提取示例

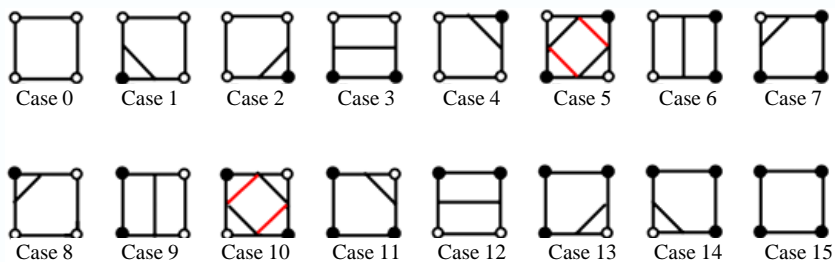


Figure 2. Sixteen cases of Marching Square's iso-surface  
图 2. 移动正方形等值面的 16 种情况

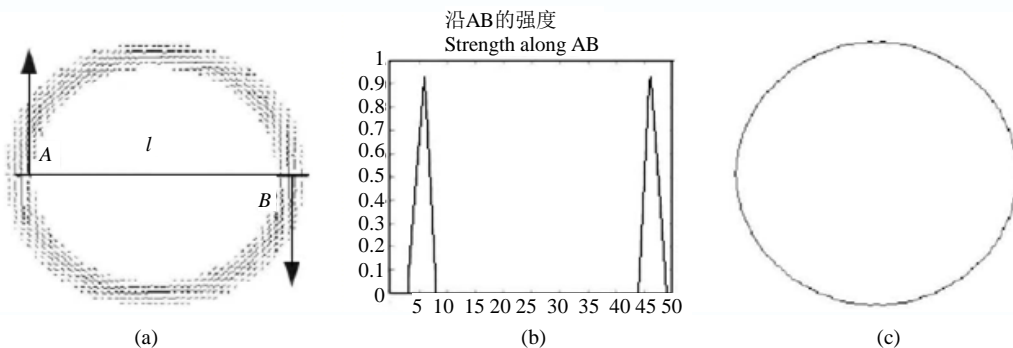


Figure 3. An example for curve maximum extraction  
图 3. 曲线极点提取示例

### 3. 张量投票的特征提取

张量投票的主要思想是：曲线和曲面的插值都可以通过收集局部邻域及其取向信息来实现，不连续点和边界也可以通过对邻域取向信息的一致性度量来检测。在这一方法中，邻域信息通过投票的方式聚集到一起，而取向信息通过矢量场的计算规则进行估计，允许使用类似卷积的方式进行有效的计算，实现数据之间的通信。这种方法的新颖性在于采用二阶矩来聚集和解释矢量投票。

基于这一思想，张量投票方法主要由两部分组成：数据的张量表示和数据间以非线性投票方式进行信息传递，这里的数据由空间点组成，空间的维数不限。张量投票方法还涉及到显著性的概念，其显著性的主要涵义是视觉上能够迅速引起观察者的注意的一种“突出性”。从模糊数学角度来看，对表示图像数据的张量而言，其具有某一特征的显著性度量可以理解为该张量具有这种特征的置信度。

然而，在张量投票方法的特征提取环节，如果仅做投票解释，得到的会是曲线及其周围一定邻域范围内点的曲线特征显著度；如果进行简单的极值搜索，会因忽略各点取向信息而增大曲线点检测的误差。因此，可在其特征提取环节，引入移动正方形算法，辅助完成曲线检测。

为便于说明, 首先给出以下定义:

定义 1 曲线极点[1]: 给定某一投票结果的曲线特征显著程度图, 设其中每一元素可以用一个二元组  $(s, \hat{t})$  表示, 其中  $s$  表示强度大小(即对应于  $\lambda_2 - \lambda_3$ ),  $\hat{t}$  表示切向(即对应于  $e_3$ ),  $u, v$  表示与  $\hat{t}$  垂直的平面,  $R$  表示将某一三维矢量旋转到  $u-v$  平面的操作(具体来说就是使三维矢量  $q = (q_x, q_y, q_z)$  满足  $q_x \leftrightarrow \frac{ds}{du}$ ,  $q_y \leftrightarrow \frac{ds}{dv}$ ,  $q_z = 0$ ), 令  $g = [\partial s / \partial x \quad \partial s / \partial y \quad \partial s / \partial z]^T$ , 我们称某一点为曲线极点当且仅当该点满足以下条件:

- 1)  $s$  是局部极值点
- 2)  $\partial s / \partial u = \partial s / \partial v = 0$
- 3)  $q = R(\hat{t} \times g) = 0$

即曲线上的点的显著性  $s$  应在该点的切线方向上有局部极大值, 而在与切线垂直的平面上分量最小。

曲线即是由一个个的曲线极点组成的, 张量投票应用移动正方形算法搜索曲线极点。图 3 给出了曲线极点的示例说明[1], 其中图 3(a)为对计算机产生的圆环点集的投票计算结果; 图 3(b)是沿 3(a)中 A、B 两点的法线方向计算直线  $l$  上各点矢量域强度的大小, 可见 A、B 两点处各有一个峰值, 根据极值搜索即可得到曲线极点的位置; 图 3(c)是将所有曲线极点连接起来而成的曲线。图 4 给出了曲线极点的微分求解过程, 曲线极点即为两个方向上微分过零点连线的交线[1]。

下面首先给出离散情况下提取曲线极点时需计算的参量, 然后给出算法流程。

离散情况下,  $q$  的计算公式为:

$$q_{i,j,k} = R(\hat{t}_{i,j,k} \times g_{i,j,k}) \quad (1)$$

式中,  $R$  定义了一个将某一平面旋转至平行于  $u-v$  平面的旋转矩阵。从上式可以看出, 所有的  $q_{i,j,k}$  的集合  $\{q_{i,j,k}\}$  组成一个矢量数组, 而这个域可以用文献[2]提出的算法改进后进行处理, 在这里我们与张量投票的提出者保持一致, 将这一算法简称为 SingleSubVoxelCMatch [1], 其原理如图 3, 图 4 所示, 而曲线极值搜索的整个算法流程如图 5 [2]所示, 算法首先根据  $s$  值的大小挑选出可能构成曲线的“种子”元素, 然后对超过高门限的元素应用 SingleSubVoxelCMatch 算法进行处理, 判断其是否曲线极点, 直到被处理元素的  $s$  值小于低门限则停止搜索, 再退回到种子点处向相反方向重复同样处理过程, 直至所有“种子”元素均被处理过为止, 最终输出图像中存在的曲线。与曲面极点的搜索过程类似, 门限只是用来提高计算效率的, 因而这一参数设置在一定范围变化时对算法运行结果的影响不大。搜索出的所有曲线极点就构成了待检测的曲线。

## 4. 实验结果

应用移动正方形算法辅助张量投票检测曲线的实验结果如图 6, 图 7 所示, 此处为减小曲线提取的计算量, 我们在投票之后对图像中的显著性测度进行了归一化并直接删除了显著性测度小于整体均值的点。图 6 是对投票点较为密集的圆形轮廓的实验结果, 其中图 6(a)为原始图像; 图 6(x\_1,2){x = b, c, ..., g} 分别是不同尺度下的投票结果及极值曲线的检测结果, 尺度参数依次为 5、10、18、40、50、100; 从这组实验结果中可以看出, 当尺度参数取得较小时 ( $\sigma = 5$ ), 轮廓线刚好得到修复, 投票产生的杂散点很少, 曲线提取可以较快得到修复后的曲线, 只是由于实验中保留了投票结果所得的各像素灰度值, 轮廓线能看到明显的分段痕迹; 尺度参数  $\sigma = 10 \sim 18$  时, 圆形轮廓曲线得到了很好的修复, 变得更加平滑; 尺度参数较大 ( $\sigma \geq 40$ ) 后, 修复的曲线开始发生局部变形, 所得曲线平滑程度下降, 这主要是因为参与投票的点过多, 增加了数据点之间的干扰致使规则的形状发生拉伸。

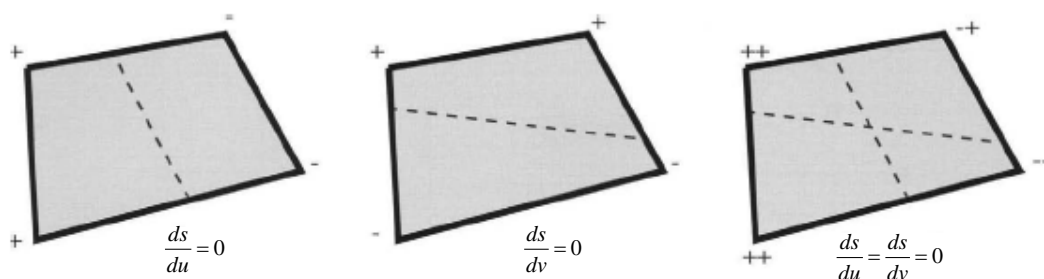


Figure 4. Differential computation for curve maximum extraction

图 4. 曲线极点的微分求解

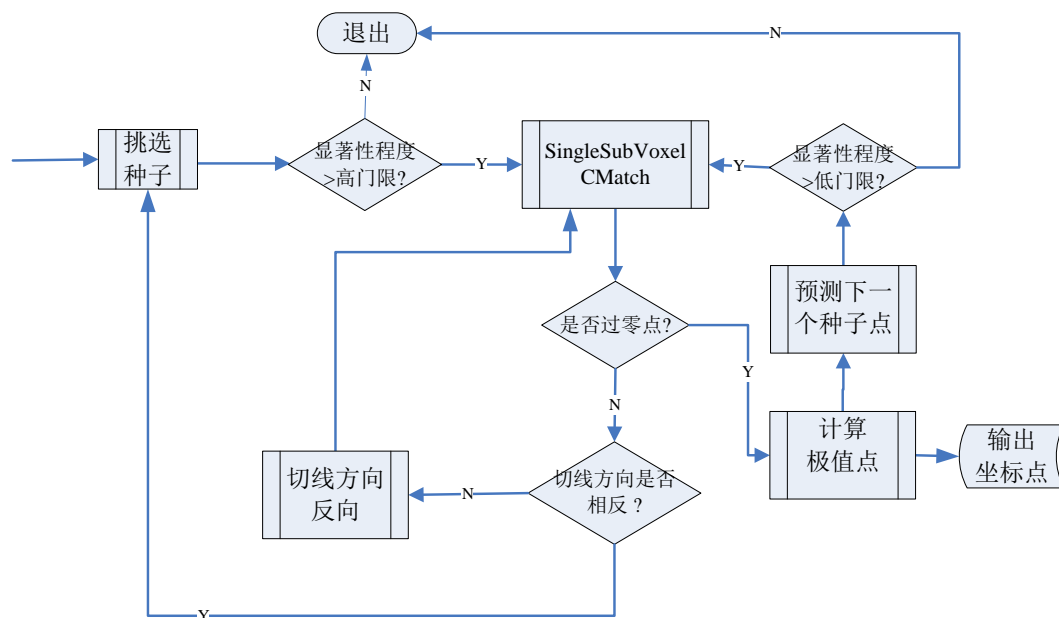


Figure 5. Flow chart for curve maximum extraction

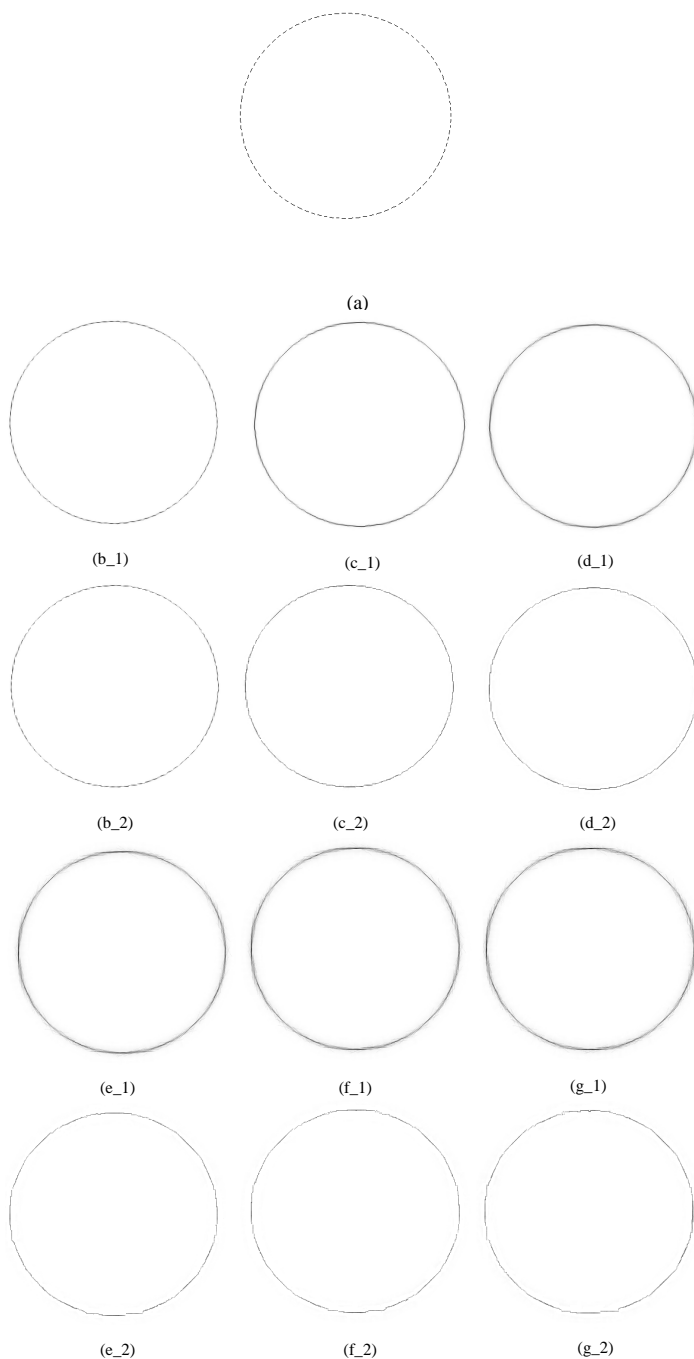
图 5. 曲线极点的搜索算法流程

图 7 是对投票点密度近似为图 6 的原始图像的二分之一的圆形轮廓点处理的结果。其中图 7(a)为原始图像；图 7(x<sub>1,2</sub>){x = b, c, ..., g}分别是不同尺度下的投票结果及极值曲线的检测结果，尺度参数依次为 5、10、18、40、50、100；从这组实验结果中可以看出，当尺度参数取得较小时 ( $\sigma = 5, 10, 18$ )，轮廓线仍然存在缺口，随尺度变大，缺口有缩小的趋势；尺度参数  $\sigma = 40$  时，轮廓线刚好得到修复，投票产生的杂散点很少，极值曲线提取可以较快得到修复后的曲线，只是由于实验中保留了投票结果所得的各像素灰度值，轮廓线能看到明显的分段痕迹；随着尺度参数增大 ( $\sigma \geq 50$ ) 后，修复的曲线开始发生局部变形，分段直线呈现接连趋势。

总体来说，这两组实验除了说明尺度参数的影响之外，还说明了加入移动正方形算法，可以有效地从投票结果中提取出单像素曲线。

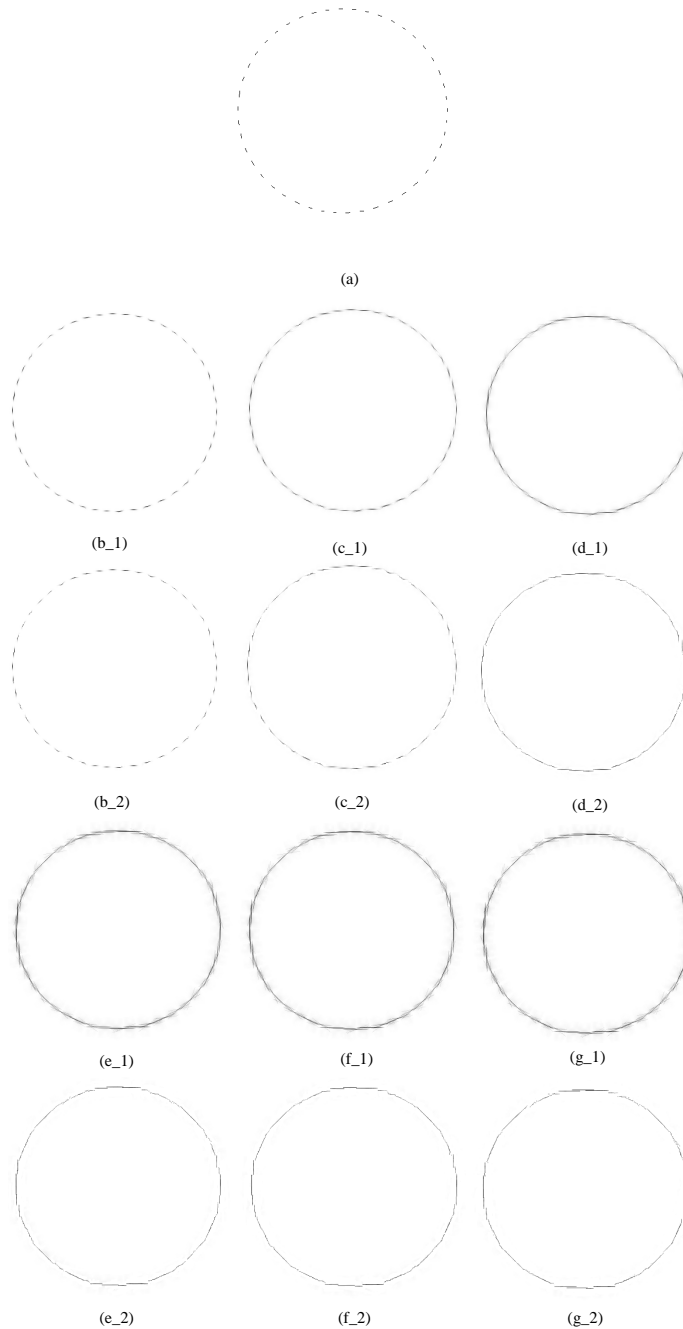
## 5. 结束语

本文介绍了移动正方形算法在二维张量投票中的应用，并通过实验展示了应用的效果及尺度参数的影响。长期以来，移动正方形算法一直被扩展到三维用于等值线提取[11][12]，而张量投票方法赋予移动正方形方法一个新的应用，也使得我们可以从一个新的角度观察这一算法。移动正方形算法在 2D 张量



**Figure 6.** An example of 2D curve extraction. (a) Original image; (b\_1) Voting result for  $\sigma = 5$ ; (b\_2) Maximum curve for  $\sigma = 5$ ; (c\_1,2) voting result and maximum curve for  $\sigma = 10$ ; (d\_1,2) Voting result and maximum curve for  $\sigma = 18$ ; (e\_1,2) Voting result and maximum curve for  $\sigma = 40$ ; (f\_1,2) Voting result and maximum curve for  $\sigma = 50$ ; (g\_1,2) Voting result and maximum curve for  $\sigma = 100$

**图 6.** 二维曲线提取实验。(a) 原始图像; (b\_1)  $\sigma = 5$  投票结果; (b\_2)  $\sigma = 5$  极值曲线检测结果; (c\_1,2)  $\sigma = 10$  投票结果及极值曲线检测结果; (d\_1,2)  $\sigma = 18$  投票结果及极值曲线检测结果; (e\_1,2)  $\sigma = 40$  投票结果及极值曲线检测结果; (f\_1,2)  $\sigma = 50$  投票结果及极值曲线检测结果; (g\_1,2)  $\sigma = 100$  投票结果及极值曲线检测结果



**Figure 7.** The effect of scale parameter on 2D curve extraction. (a) Original image; (b<sub>1</sub>) Voting result for  $\sigma = 5$ ; (b<sub>2</sub>) Maximum curve for  $\sigma = 5$ ; (c<sub>1,2</sub>) Voting result and maximum curve for  $\sigma = 10$ ; (d<sub>1,2</sub>) Voting result and maximum curve for  $\sigma = 18$ ; (e<sub>1,2</sub>) Voting result and maximum curve for  $\sigma = 40$ ; (f<sub>1,2</sub>) Voting result and maximum curve for  $\sigma = 50$ ; (g<sub>1,2</sub>) Voting result and maximum curve for  $\sigma = 100$

**图 7.** 尺度参数对二维曲线提取的影响实验。(a) 原始图像; (b<sub>1</sub>)  $\sigma = 5$  投票结果; (b<sub>2</sub>)  $\sigma = 5$  极值曲线检测结果; (c<sub>1,2</sub>)  $\sigma = 10$  投票结果及极值曲线检测结果; (d<sub>1,2</sub>)  $\sigma = 18$  投票结果及极值曲线检测结果; (e<sub>1,2</sub>)  $\sigma = 40$  投票结果及极值曲线检测结果; (f<sub>1,2</sub>)  $\sigma = 50$  投票结果及极值曲线检测结果; (g<sub>1,2</sub>)  $\sigma = 100$  投票结果及极值曲线检测结果

投票中的应用使得该算法的曲线提取环节可以有效地提取出单像素取向，从而可以便于后续处理和高层识别，这是其他相关工作没有考虑的问题。本文通过不同尺度下圆形轮廓线提取的实验，既说明了移动正方形在曲线提取中的应用，也验证了尺度参数的影响。我们将在后续研究工作中进一步提高算法的运行效率，并与相关工作进行量化比较。

## 参考文献 (References)

- [1] Bogdan, M., van den Berg, E., Su, W.J. and Candes, E.J. (2013) Statistical estimation and testing via the ordered L1 norm. arXiv preprint, arXiv:1310.1969v2.
- [2] Yan, X.-W., Wang, W., Zhao, J., Hu, J.-M., Zhang, J. and Wan, J.-W. (2013) Relaxation labeling for non-rigid point matching under neighbor preserving. *Journal of Central South University*, **20**, 21-26.
- [3] Zhang, Z. (1997) Parameter estimation techniques: A tutorial with application to conic fitting. *Image and Vision Computing Journal*, **15**, 59-76.
- [4] Enright, D., Fedkiw, R.P., Ferziger, J.H. and Mitchell, I. (2002) A hybrid particle level set method for improved interface capturing. *Journal of Computational Physics*, **183**, 83-116.
- [5] Zhang, W., et al. (2012) Graph degree linkage: Agglomerative clustering on a directed graph. *12th European Conference on Computer Vision*, Florence, October 2012, 7-13.
- [6] Sha'ashua, A. and Ullman, S. (1988) Structural saliency: The detection of globally salient structures using a locally connected network. *International Conference on Computer Vision*, Tampa, 5-8 December 1988, 321-327.
- [7] Medioni, G., Lee, M.S. and Tang, C.K. (2000) A computational framework for feature extraction and segmentation. Elsevier Science, The Netherlands, 75-113.
- [8] Lorenzi, L., Melgani, F. and Mercier, G. (2011) Inpainting strategies for reconstruction of missing data in VHR images. *IEEE Geoscience and Remote Sensing Letters*, **8**, 914-918.
- [9] Newman, T.S. and Yi, H. (2006) A Survey of the marching cubes algorithm. *Computers & Graphics*, **30**, 854-879.
- [10] Lorensen, W.E. and Cline, H.E. (1987) Marching cubes: A high resolution 3D surface construction algorithm. *Computer Graphics*, **21**, 163-169.
- [11] [http://en.wikipedia.org/wiki/Marching\\_cubes](http://en.wikipedia.org/wiki/Marching_cubes)
- [12] 周筠 (2012) 面向生物医学仿真的表面重建和四面体化技术研究. 中南大学, 长沙.