



(12) 发明专利

(10) 授权公告号 CN 101447079 B

(45) 授权公告日 2012. 11. 28

(21) 申请号 200810187167. 8

用》. 2003, (第 23 期),

(22) 申请日 2008. 12. 11

审查员 姚天宇

(73) 专利权人 香港理工大学

地址 中国香港九龙红磡

(72) 发明人 史文中 廖剑锋 黄长青

(74) 专利代理机构 深圳市顺天达专利商标代理

有限公司 44217

代理人 郭伟刚

(51) Int. Cl.

G06T 7/00(2006. 01)

(56) 对比文件

张小萍等. 基于 CT 医学图像的边缘提取研究. 《生物医学工程研究》. 2009, (第 01 期),

郑成勇等. 基于最优分类数及遗传算法的灰度图像多阈值自动分割法. 《计算机工程与应

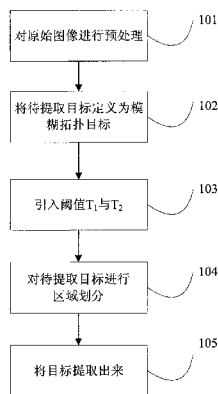
权利要求书 1 页 说明书 7 页 附图 4 页

(54) 发明名称

基于模糊拓扑的图像区域目标提取方法

(57) 摘要

本发明涉及一种基于模糊拓扑的图像区域目标提取方法,包括如下步骤:对原始图像进行预处理;引入最优阈值 T,将待提取目标定义为模糊拓扑目标;以最优阈值 T 诱导两个阈值 T₁与 T₂;对待提取目标进行区域划分;并最后将目标提取出来。本发明所述方法以模糊拓扑方法引入两个阈值以用于确定待提取目标的内部、边界以及外部,从而在一定程度上提高了提取质量,并且通过对边界像素进行模糊拓扑连通性分析以确定其是否属于提取范围,并在分析过程中引入局部二元模式(LBP)算符以提高判断精度。此外,本发明所述基于模糊拓扑的图像区域目标提取方法能够显著提高那些具有较高边界像素判断难度的目标提取精度,从而得到较优的提取结果。



1. 一种图像区域目标提取方法,其特征在于,所述方法包括如下步骤:

- (a) 对原始图像进行预处理;
- (b) 引入最优阈值 T ,将待提取目标定义为模糊拓扑目标;
- (c) 以最优阈值 T 诱导两个阈值 T_1 与 T_2 ;
- (d) 基于上述引入的阈值 T_1 与 T_2 将待提取目标划分为内部、边界以及外部;
- (e) 提取目标;

所述步骤(e)进一步包括:对待提取目标的边界像素进行连通性分析以确定所述像素是否需要提取;以及将上述经连通性分析的边界像素与步骤(d)划分所得的待提取目标的内部像素组合以形成提取目标;

所述连通性分析进一步包括如下步骤:

- S1:搜索每个边界像素的位置;
- S2:搜索每个内部像素的位置;
- S3:计算内部像素范围;

S4:对于每个边界像素,引入局部二元模式算符计算其相邻像素距离偏差值;所述步骤 S4 中引入局部二元模式算符计算相邻像素的距离偏差值,该距离偏差值为平均偏差或标准偏差;

所述局部二元模式算符计算的过程如下:假定每个待判断的边界像素是与其相距 R 的相邻像素的中心像素,取每个边界像素 p_c 的 $R>0$,令 $\{p_1, p_2, \dots, p_n\}$ 为位于

$B(p_i; \text{dist}(p_c, p_i) < R)$ 内灰度值分别为 $\{v_1, v_2, \dots, v_n\}$ 的像素;利用公式 $\mu = \frac{1}{n} \sum_{i=1}^n v_i$ 或

$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (v_i - \mu)^2}$ 计算相邻像素的距离平均偏差或标准偏差;

S5:根据计算得出的距离偏差值判断所述相邻像素是否属于内部像素范围内;

重复执行步骤 S4 与 S5 直至完成对所有边界像素的判断。

2. 根据权利要求 1 所述的方法,其特征在于,所述步骤(a)进一步包括:引入最优阈值 T ,将原始图像定义为模糊拓扑图像并将其划分为目标及背景。

3. 根据权利要求 1 所述的方法,其特征在于,所述步骤(c)进一步包括:基于错误分类总数最小化原则诱导阈值 T_1 与 T_2 。

基于模糊拓扑的图像区域目标提取方法

技术领域

[0001] 本发明涉及图像处理方法,更具体地说,涉及一种基于模糊拓扑的图像区域目标提取方法。

背景技术

[0002] 随着多媒体技术的不断发展,图像作为一种信息载体正逐渐融入到人类生活的每个角落。当今社会中,人类对图像的使用已不仅仅只是记录信息以及浏览信息,对图像更进一步的处理需求也在与日俱增,比如对图像中的目标进行提取。这种技术称作目标提取(Object Extraction),进行提取操作的目标载体可以是各种图像,例如:数字航空图像、卫星图像、医疗图像、或日常生活中的数字照片,提取出来的目标对象可以是自然物,比如池塘、石砾,也可以是人造物,比如建筑物。

[0003] 现有目标提取技术主要采用的是最优阈值方法,该方法利用一个阈值来对提取目标边界区域进行判断。而现有技术的缺陷在于:无法对图像中待提取目标的边界像素进行准确判断来确定这些边界像素是否属于待提取目标,从而导致目标提取的精度不够高。

[0004] 比较本发明后续将要结合附图介绍的系统,现有技术的其它局限性和弊端对于本领域的普通技术人员来说是显而易见的。

发明内容

[0005] 本发明要解决的技术问题在于,针对现有技术的上述无法对图像中待提取目标的边界像素进行准确判断而导致的提取精度不够高的缺陷,提供一种基于模糊拓扑的图像区域目标提取方法。

[0006] 本发明解决其技术问题所采用的技术方案是:提供一种图像区域目标提取方法,包括如下步骤:

[0007] (a) 对原始图像进行预处理;

[0008] (b) 引入最优阈值 T ,将待提取目标定义为模糊拓扑目标;

[0009] (c) 以最优阈值 T 诱导两个阈值 T_1 与 T_2 ;

[0010] (d) 对待提取目标进行区域划分;

[0011] (e) 提取目标。

[0012] 在本发明所述的图像区域目标提取方法中,所述步骤 (a) 进一步包括:引入最优阈值 T ,将原始图像定义为模糊拓扑图像并将其划分为目标及背景。

[0013] 在本发明所述的图像区域目标提取方法中,所述步骤 (c) 进一步包括:基于错误分类总数最小化原则诱导阈值 T_1 与 T_2 。

[0014] 在本发明所述的图像区域目标提取方法中,所述步骤 (d) 进一步包括:将所述待提取目标基于上述引入的阈值 T_1 与 T_2 划分为内部、边界以及外部。

[0015] 在本发明所述的图像区域目标提取方法中,所述步骤 (e) 进一步包括:对待提取目标的边界像素进行连通性分析以确定所述像素是否需要提取。

[0016] 在本发明所述的图像区域目标提取方法中,所述步骤(e)进一步包括:将上述经连通性分析的边界像素与所述内部像素组合以形成提取目标。

[0017] 在本发明所述的图像区域目标提取方法中,所述连通性分析进一步包括如下步骤:

[0018] S1:搜索每个边界像素的位置;

[0019] S2:搜索每个内部像素的位置;

[0020] S3:计算内部像素范围;

[0021] S4:对于每个边界像素,计算其相邻像素距离偏差值;

[0022] S5:根据计算得出的距离偏差值判断所述相邻像素是否属于内部像素范围内。

[0023] 在本发明所述的图像区域目标提取方法中,所述步骤S4中引入局部二元模式算符计算相邻像素的距离偏差值。

[0024] 在本发明所述的图像区域目标提取方法中,所述距离偏差值包括平均偏差和/或标准偏差。

[0025] 在本发明所述的图像区域目标提取方法中,重复执行步骤S4与S5直至完成对所有边界像素的判断。

[0026] 实施本发明的图像区域目标提取方法,具有以下有益效果:

[0027] (1)与现有技术引入一个阈值相比,本发明所述方法引入两个阈值以用于确定待提取目标的内部、边界以及外部,从而在一定程度上提高了提取质量;

[0028] (2)对边界像素进行连通性分析以确定其是否属于提取范围,并在分析过程中引入局部二元模式(LBP)算符以提高判断精度;

[0029] (3)本方法能够显著提高那些具有较高边界像素判断难度的目标提取精度,从而得到较优的提取结果。

[0030] 本发明的各种优点、各个方面和创新特征,以及其中所示例的实施例的细节,将在以下的描述和附图中进行详细介绍。

附图说明

[0031] 下面将结合附图及实施例对本发明作进一步说明,附图中:

[0032] 图1是本发明所述方法一实施例的步骤流程图;

[0033] 图2是本发明所述方法一实施例中进行连通性分析的步骤流程图;

[0034] 图3是局部二元模式(LBP)原理示意图;

[0035] 图4是卫星图像上的石砾示意图;

[0036] 图5是传统目标提取方法对图4进行处理后的结果示意图;

[0037] 图6是本发明所述目标提取方法对图4进行处理后的结果示意图。

具体实施方式

[0038] 本发明所述基于模糊拓扑的图像区域目标提取方法的软件运行平台采用StoneExtra,该软件是用Visual Basic程序设计语言设计的用于从航空或高清晰度卫星图像中提取石砾以及岩石的图像处理系统。利用该系统可以对石砾的尺寸、形状以及高度分布进行提取。该系统可用于高速公路建设、以及其他地形或市政工程项目。此项技术与传统

实地勘测解决方案相比具有高性价比以及更高的安全性,因为该系统中增加了如下功能:

[0039] (1) 图像优化:包括线性优化、直方图调整以及区域空间优化;

[0040] (2) 石砾分布探测:根据图像频谱特性探测石砾的空间分布;

[0041] (3) 石砾信息提取:石砾特性,比如:尺寸、形状以及高度可基于原始图像的探测以及 DEM 来探测;

[0042] (4) 信息分析:根据石砾的空间分布以及特性进行风险分析;

[0043] (5) 石砾勾画:自动生成石砾的空间分布图并将原始图像进行覆盖。

[0044] 图 1 是本发明所述方法一实施例的步骤流程图。如图 1 所示,流程开始于步骤 101,在此步骤中,先对原始图像进行预处理,包括:将原始图像定义为模糊拓扑图像,将其划分为目标及背景两部分,比较目标分布与背景分布;在步骤 102 中,将上步中的目标进一步定义为模糊拓扑目标;在步骤 103 中,引入最优阈值 T ,并基于错误分类总数最小化原则诱导出阈值 T_1 与 T_2 ;在步骤 104 中,根据上述引入的阈值 T_1 与 T_2 将目标进一步划分为内部、边界以及外部三部分;最后在步骤 105 中将确定的目标提取出来。

[0045] 在上述流程的步骤 105 中,如何判断边界像素是否属于需要被提取的内部像素范围是本发明所述方法的关键点所在,在本方法中采用连通性分析来确定边界像素与内部像素的关联度。该连通性分析针对边界上的所有点进行的,需要对边界上的每一个点进行判断,具体流程如图 2 所示。

[0046] 流程开始于步骤 201,在步骤 201 及 202 中首先对每个边界像素以及内部像素的位置进行搜索,以确定具体位置;在步骤 203 中,根据以上对每个内部像素以及边界像素的位置搜索结果来计算出内部像素范围,为后续判断作基础;在步骤 204 中,取定一个边界像素,对其进行连通性分析,即是计算其相邻像素之间的距离偏差值,该偏差值可以是平均偏差也可以是标准偏差,具体计算方法是引入 LBP(局部二元模式)算符来进行;在步骤 205 中,根据上述计算得出的距离偏差值来判断该像素究竟是否属于内部像素范围,也就是说该像素是否应该与内部像素一起形成提取目标,若判断结果显示该边界像素属于内部像素范围,则流程进入步骤 206a 中将此像素与内部像素进行组合,否则流程进入步骤 206b 放弃该边界像素;随后在步骤 207 中,判断是否对所有的边界像素都判断完毕,若还有边界像素没有进行判断,则流程返回步骤 204 重新取定边界像素进行判断,若全部边界像素均已判断完毕,则流程进入步骤 208;在步骤 208 中,根据对每个边界像素的判断结果将适当的边界像素与内部像素进行组合以形成最终的提取目标。

[0047] 图 3 是局部二元模式(LBP)原理示意图。如图所示,31 为待判断的边界像素,32 为其相邻像素,假定每个待判断的边界像素是与其相距 R 的相邻像素的中心像素,计算方法如下:

[0048] 取每个边界像素 p_c 的 $R>0$,令 $\{p_1, p_2, \dots, p_n\}$ 为位于 $B(p_c; \text{dist}(p_c, p_i) < R)$ 内灰

度值分别为 $\{v_1, v_2, \dots, v_n\}$ 的像素;利用公式 $\mu = \frac{1}{n} \sum_{i=1}^n v_i$ or $\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (v_i - \mu)^2}$ 计算相邻

像素的距离平均偏差或标准偏差,若计算结果显示该边界像素的平均偏差或标准偏差在内部像素范围内则指定该边界像素属于内部,否则确定该边界像素为背景像素。

[0049] 图 4 是卫星图像上的石砾示意图;图 5 和图 6 分别是使用传统目标提取方法和本发明所述目标提取方法对图 4 进行处理后的结果示意图。为便于进行比较说明,现定义两

个指标值:正确率=正确提取的石砾数/总石砾数;错误率=正确提取的石砾数/总石砾数=1-正确率。两种方法的具体实施结果比较见表1。其中,图4的实际总石瓦数为201。

[0050]

	总石砾数	正确提取数	正确率	错误提取数	错误率	漏提取数	漏提取率
传统方法	288	209	72.6%	79	27.4%	36	17.9%
本方法	235	184	78.3%	51	21.7%	15	6.4%

[0051] 表1 传统目标提取方法与本发明方法的实施效果比较

[0052] 如表1所示,其中的正确率计算方法如前所述,正确率=正确提取的石砾数/总石砾数,因而,对传统方法来说,正确率=209/288=72.6%,因而其错误率=1-正确率=1-72.6%=27.4%;对于本发明方法,正确率=184/235=78.3%,错误率=1-78.3%=21.7%。同理可得出两种方法的相应“漏提取率”,如表所示。比较表1中相应数据可得出如下结论:采用基于模糊拓扑的图像区域目标提取方法较传统方法而言提取精度显著提高。

[0053] 主要程序代码如下:

[0054] Public Function ImgBinary WithFuzzy ThresholdhnSelArea(hSrcBmp As Long, hDestBmp As

[0055] Long, interiorThreshold As Integer, exteriorThreshold As Integer,

[0056] selBorder As Border)As Boolean

[0057] Dim tSBmpInfo As BITMAP, tDBmpInfo As BITMAP

[0058] Dim sBits()As Byte, dBits() As Byte

[0059] Dim BytesPixel As Long

[0060] Dim bmHeight As Long, bmWidth As Long

[0061] Dim flag As Boolean

[0062] Dimw As Long, h As Long

[0063] Dim pColor As PALETTEENTRY

[0064] Dim colorGray As Integer

[0065] Dim startPoint As CPoint

[0066] Dim aStone As CStone

[0067] Call GetObject(hSrcBmp, Len(tSBmpInfo), tSBmpInfo)

[0068] Call GetObject(hDestBmp, Len(tDBmpInfo), tDBmpInfo)

[0069] ReDim sBits(1 To tSBmpInfo.bmWidthBytes, 1 To tSBmpInfo.bmHeight)

[0070] ReDim dBits(1 To tDBmpInfo.bmWidthBytes, 1 To tDBmpInfo.bmHeight)

[0071] Call GetBitmapBits(hSrcBmp, tSBmpInfo.bmWidthBytes*tSBmpInfo.bmHeight, sBits(1,

[0072] 1))

[0073] Call GetBitmapBits(hDestBmp, tDBmpInfo.bmWidthBytes*tDBmpInfo.bmHeight, dBits(1,

[0074] 1))

[0075] BytesPixel = tSBmpInfo.bmBitsPixel/8

```
[0076] ' start applying the stone algorithm
[0077] Dim beginW As Long, endW As Long, beginH As Long, endH As Long
[0078] begin W = IIf(selBorder.minX >= 1, selBorder.minX, 1)
[0079] endW = IIf(selBorder.maxX <= tSBmpInfo.bmWidth, selBorder.maxX,
[0080]     tSBmpInfo.bmWidth)
[0081] beginH = IIf(selBorder.minY >= 1, selBorder.minY, 1)
[0082] endH = IIf(selBorder.maxY <= tSBmpInfo.bmHeight, selBorder.maxY,
[0083]     tSBmpInfo.bmHeight)
[0084] For h = beginH To endH
[0085]     frmProgress.ProgressProcessing.Value = Round((h-beginH)/
[0086]         (endH-beginH)*100,
[0087]         0)
[0087]     DoEvents
[0088]     For w = beginW To end W
[0089]         If pointInside(w, h, selBorder) Then
[0090]             Call CopyMemory(pColor, sBits((w-1)*BytesPixel+1, h), BytesPixel)
[0091]             colorGray = grayColor(rgbColor(pColor))
[0092]             If colorGray > interiorThreshold Then
[0093]                 Call CopyMemory(dBits((w-1)*BytesPixel+1, h), pBlack,
[0094]                     BytesPixel)
[0095]                 Call CopyMemory(sBits((w-1)*BytesPixel+1, h), pBlack,
[0096]                     BytesPixel)
[0097]             ElseIf colorGray >= exteriorThreshold Then
[0098]                 Call CopyMemory(dBits((w-1)*BytesPixel+1, h), pWhite,
[0099]                     BytesPixel)
[0100]                 Call CopyMemory(sBits((w-1)*BytesPixel+1, h), pBlack,
[0101]                     BytesPixel)
[0102]             Else
[0103]                 Call CopyMemory(dBits((w-1)*BytesPixel+1, h), pWhite,
[0104]                     BytesPixel)
[0105]                 Call CopyMemory(sBits((w-1)*BytesPixel+1, h), pWhite,
[0106]                     BytesPixel)
[0107]             End If
[0108]         End If
[0109]     Next w
[0110] Next h
[0111] For h = beginH To endH
[0112]     frmProgress.ProgressProcessing.Value = Round((h-beginH)/
```

```
[0113]         0)
[0114]         DoEvents
[0115]         For w = begin W To endW
[0116]             If pointInSide(w, h, selBorder) Then
[0117]                 Call CopyMemory(pColor, sBits((w-1)*BytesPixel+1, h), BytesPixel)
[0118]                 If rgbColor(pColor) = vbBlack Then
[0119]                     Call CopyMemory(pColor, dBits((w-1)*BytesPixel+1, h),
[0120]                                     BytesPixel)
[0121]                     If rgbColor(pColor) = vbBlack Then
[0122]                         Set startPoint = New CPoint
[0123]                         startPoint.X = w
[0124]                         startPoint.Y = h
[0125]                         Set aStone = New CStone
[0126]                         aStone.maxPoint.X = w
[0127]                         aStone.maxPoint
[0128]                         aStone.minPoint.X = w
[0129]                         aStone.minPoint.Y = h
[0130]                         aStone.pointCount = 0
[0131]                         aStone.LineQueue.Enqueue startPoint
[0132]                     extractOneStone tSBmpInfo, sBits, BytesPixel, aStone, vbBlack,
[0133]                                     pBlue, True
[0134]                     End If
[0135]                 End If
[0136]             End If
[0137]         Next w
[0138]     Next h
[0139]     For h = beginH To endH
[0140] frmProgress.ProgressProcessing.Value = Round((h-beginH)/(endH-beginH)*100,
[0141]         0)
[0142]         DoEvents
[0143]         For w = beginWTo endW
[0144]             Call CopyMemory(pColor, sBits((w-1)*BytesPixel+1, h), BytesPixel)
[0145]             If rgbColor(pColor) = vbBlue Then
[0146]                 Call CopyMemory(dBits((w-1)*BytesPixel+1, h), pBlack, BytesPixel)
[0147]             Else
[0148]                 Call CopyMemory(dBits((w-1)*BytesPixel+1, h), pWhite, BytesPixel)
[0149]             End If
[0150]         Next w
[0151]     Next h
```


[0152] Call SetBitmapBits(hDestBmp, tDBmpInfo.bmWidthBytes*tDBmpInfo.bmHeight, dBits(1,

[0153] 1))

[0154] End Function

[0155] 上面结合附图对本发明的实施例进行了描述,但是本发明并不局限于上述的具体实施方式,上述的具体实施方式仅仅是示意性的,而不是限制性的,本领域的普通技术人员在本发明的启示下,在不脱离本发明宗旨和权利要求所保护的范围情况下,还可做出很多形式,这些均属于本发明的保护范围。

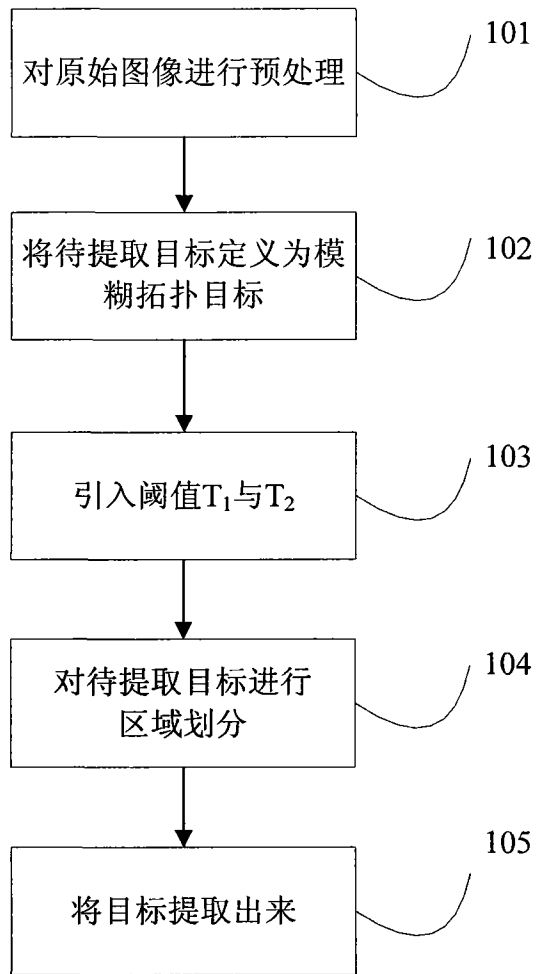


图 1

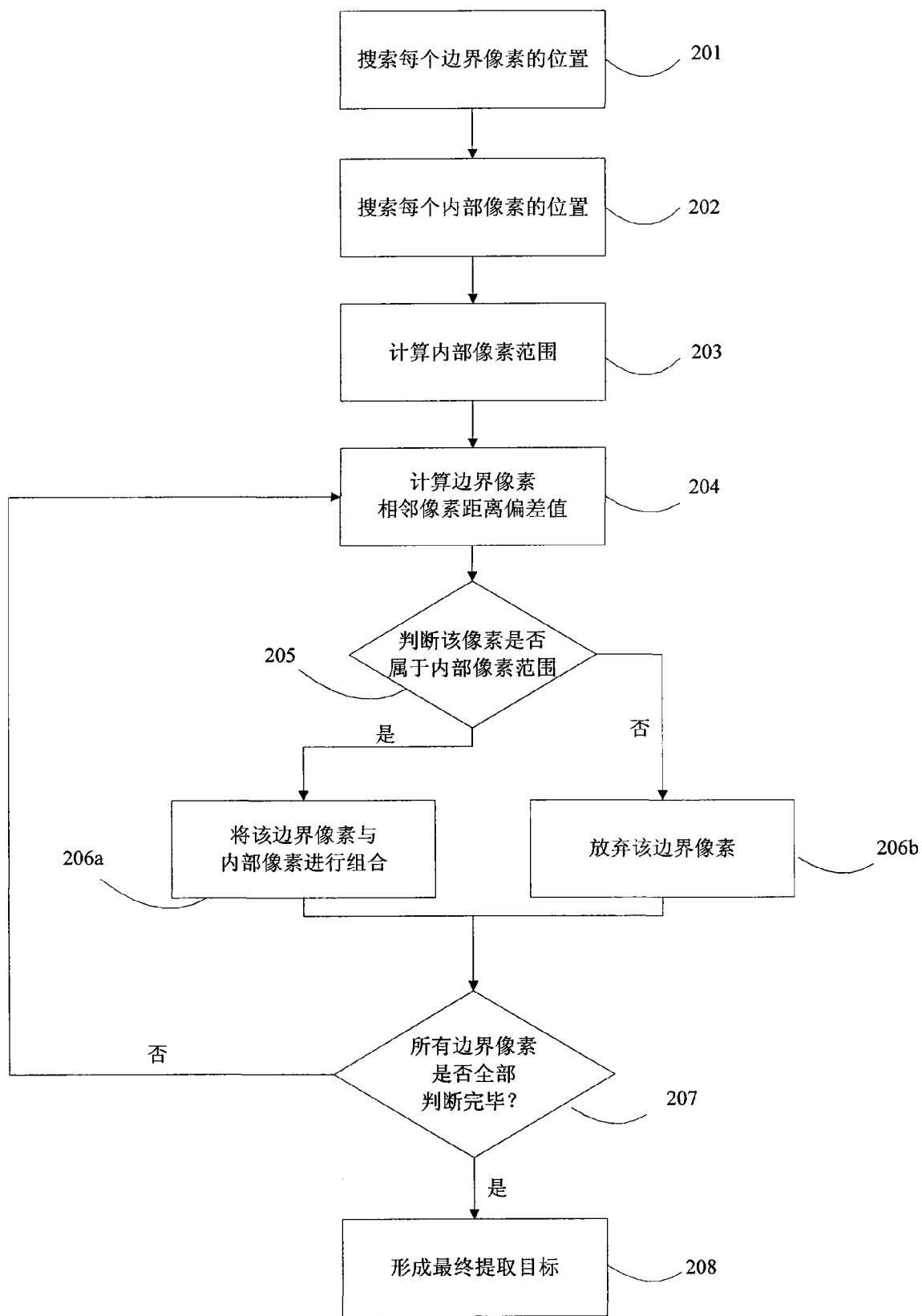


图 2

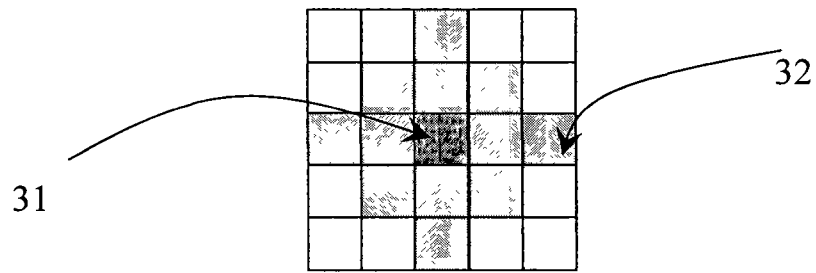


图 3

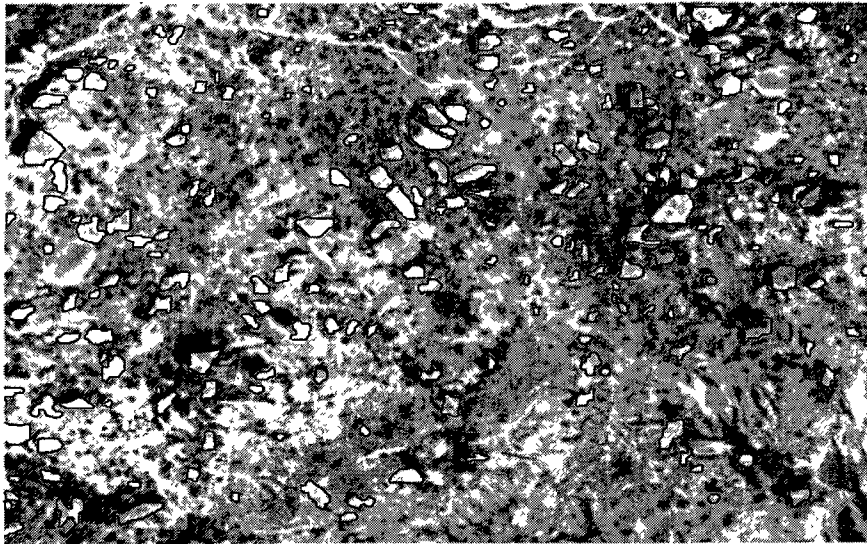


图 4

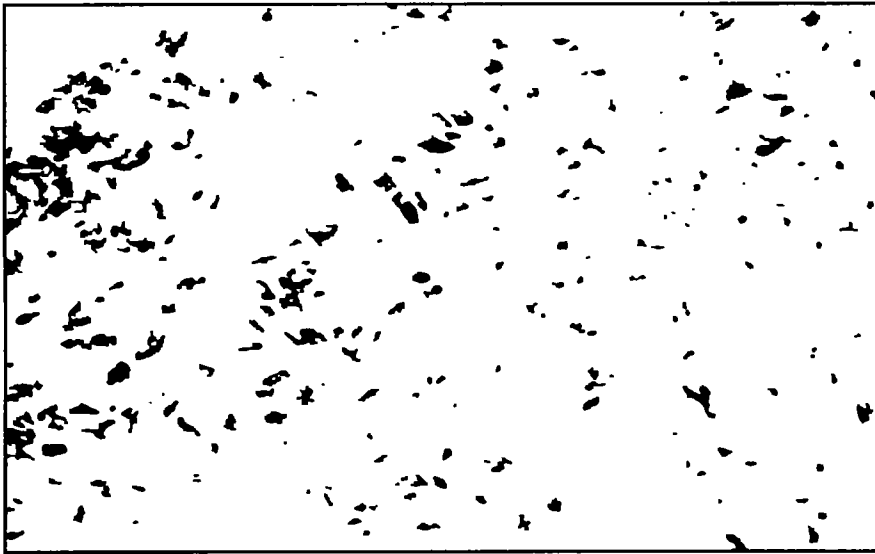


图 5

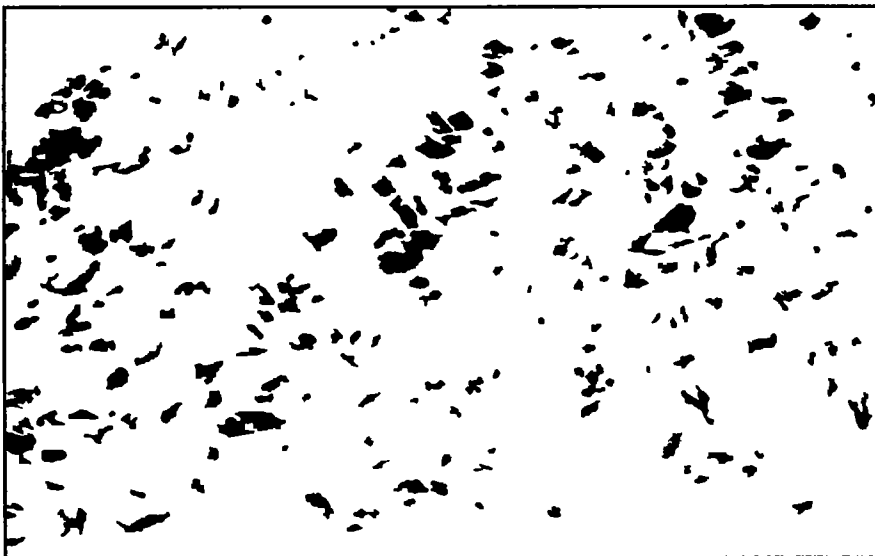


图 6