



(12)发明专利

(10)授权公告号 CN 104463859 B

(45)授权公告日 2017.07.04

(21)申请号 201410709348.8

(22)申请日 2014.11.28

(65)同一申请的已公布的文献号
申请公布号 CN 104463859 A

(43)申请公布日 2015.03.25

(73)专利权人 中国航天时代电子公司
地址 100094 北京市海淀区丰滢东路1号

(72)发明人 向永红 张国勇 姜梁 孙浩惠
马祥森 郭茜 王家星

(74)专利代理机构 北京金智普华知识产权代理
有限公司 11401

代理人 刘震

(51)Int.Cl.

G06T 7/00(2017.01)

G06T 5/00(2006.01)

(56)对比文件

US 2011/0090337 A1,2011.04.21,
吴垠 等.基于尺度不变特征的光流法目标
跟踪技术研究.《计算机工程与应用》.2013,第49
卷(第15期),全文.

戴斌 等.基于光流技术的运动目标检测和
跟踪方法研究.《科技导报》.2009,第27卷(第12
期),第57页左栏第1段.

审查员 马晋涛

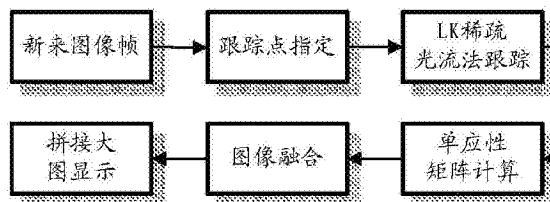
权利要求书2页 说明书5页 附图2页

(54)发明名称

一种基于跟踪指定点的实时视频拼接方法

(57)摘要

本发明提供一种基于跟踪指定点的实时视频拼接方法,所述视频拼接方法直接从图像中选取某些像素点作为跟踪点,并且通过控制跟踪点的个数和分布来控制拼接的计算时间和效果。本发明与现有技术相比的优点在于:没有特征点提取过程,直接从图像中选取某些像素点作为跟踪点;跟踪点的个数和分布可以控制,从而可以部分控制计算时间和效果。



1. 一种基于跟踪指定点的实时视频拼接方法,其特征在于,所述视频拼接方法直接从图像中选取某些像素点作为跟踪点,并且通过控制跟踪点的个数和分布来控制拼接的计算时间和效果,

所述方法包括以下步骤:

1) 指定跟踪点,从图像中自动、均匀选取一定数量的点作为跟踪点;

2) 利用LK稀疏光流法计算指定跟踪点在当前图像帧中的位置,将所得到的点利用LK稀疏光流法反向计算得到已拼好的图像中的位置;

3) 求取指定点与反向计算得到的对应点之间的距离,按从小到大排序后选取前n个点计算单应性矩阵;

4) 图像融合并显示拼接图。

2. 根据权利要求1所述的视频拼接方法,其特征在于,所述步骤1)中:设图像宽为w,高为h,从图像中均匀地选取 $m \times m$ 个点作为跟踪点,两个相邻跟踪点水平距离为 $w/(m+1)$,垂直距离为 $h/(m+1)$ 。

3. 根据权利要求2所述的视频拼接方法,其特征在于,所述跟踪点的数目是 28×28 个跟踪点。

4. 根据权利要求1所述的视频拼接方法,其特征在于,所述步骤2)中具体是:将上一帧图像 f_{i-1} 指定的 $m \times m$ 个跟踪点用二维列向量表示为: $(x_s, y_s)^T$,其中 $s \in \{1, 2, \dots, m \times m\}$;采用金字塔分层和LK结合的方法,以图像 f_i 中的点 $(x_s, y_s)^T$ 为起始点计算 f_{i-1} 中的跟踪点 $(x_s, y_s)^T$ 在 f_i 中的光流;设计算得到在 f_i 中跟踪点的坐标为 $(x_s^{i,i}, y_s^{i,i})^T$;类似地,以 f_i 中点 $(x_s^{i,i}, y_s^{i,i})^T$ 为跟踪点,计算其在 f_{i-1} 中的光流,得到反向跟踪之后, f_i 中点 $(x_s^{i,i}, y_s^{i,i})^T$ 在图像 f_{i-1} 上的坐标 $(x_s^{i-1,i}, y_s^{i-1,i})^T$ 。

5. 根据权利要求1所述的视频拼接方法,其特征在于,所述步骤3)中具体是:

根据上一帧图像 f_{i-1} 指定的跟踪点 $(x_s, y_s)^T$ 与经过正、反向跟踪之后得到的 f_{i-1} 中对应点 $(x_s^{i-1,i}, y_s^{i-1,i})^T$,使用如下的公式计算每个特征点的误差值errors,

$$\text{errors} = \sqrt{(x_s - x_s^{i-1,i})^2 + (y_s - y_s^{i-1,i})^2}$$
 根据每个点的误差值进行排序,选择其中误差最小的n对匹配点计算两帧图像之间的单应性矩阵。

6. 根据权利要求5所述的视频拼接方法,其特征在于,进一步采用随机抽样一致性算法对匹配点对进行过滤,并消除浮点精度带来的误差。

7. 根据权利要求6所述的视频拼接方法,其特征在于,所述消除浮点精度带来误差的方法是:通过 f_{i-1} 和 f_i 在全局坐标系下四个角点坐标之间的对应关系计算得到计算 f_i 与 f_{i-1} 的变换关系 $H_{i-1,i}$,从而计算出 $H_{i,i} = H_{i-1,i}^{-1} \cdot H_{i-1,i}$ 。

8. 根据权利要求7所述的视频拼接方法,其特征在于,所述图像融合的基本步骤为:

(a) 利用 $H_{i,i}$ 计算 f_i 四个角点在拼接好的大图g中的坐标,得到 f_i 在g中的包围盒B;

(b) 对B中每个像素点 $(x, y)^T$,利用 $H_{i,i}$ 计算得到其对应于 f_i 中的像素点 $(x', y')^T$,取最临近点的像素值进行计算;

(c) 如果包围盒B中某点 (x, y) 的亮度值 $IB(x, y) \neq 0$,那么进行融合计算,更新 $IB(x, y)$ 为 $\alpha If_i(x', y') + \beta IB(x, y)$,设置 $\alpha = 0.4 \beta = 0.6$;如果 $IB(x, y) = 0$,则更新 $IB(x, y)$ 为 $If_i(x', y')$,

$y')$,

其中, M^{-1} 表示 M 的逆矩阵 $(M)^{-1}$ 。

一种基于跟踪指定点的实时视频拼接方法

技术领域

[0001] 本发明属于视频领域,具体涉及一种通过对指定点进行跟踪来计算图像配准矩阵的方法而实现的实时视频拼接技术。

背景技术

[0002] 视频拼接就是将视频图像帧拼接在一起形成大图,可以看到视频所对应拍摄区域的全景。无人机视频拼接是无人机信息处理的基本步骤之一,是后续如地图产品输出等步骤的基础。视频拼接往往面临拼接效果控制和实时性之间的矛盾,要实现好的拼接效果就难以做到实时拼接;否则,连续拼接时间短,效果差。小型无人机由于容易受外界环境干扰,姿态变化比较剧烈,同时小型无人机搭载的摄像机图像质量较低,小型无人机实时视频拼接一直以来都是一个难题。

[0003] 基于频域的图像拼接主要通过傅里叶变换将图像信息转换到频域后再进行拼接,如Fourier-Mellin算法以及相位相关法。目前主要的图像配准算法是基于空域的,包括基于灰度和基于特征两大类。基于特征点匹配的图像拼接方法主要步骤包括:图像预处理,图像特征点提取,特征点匹配,误匹配点去除,图像配准(计算变换矩阵),图像融合,生成拼接图。其中,核心步骤是图像配准,即计算新图像帧相对于参考图像帧的变换矩阵。通常要求计算特征点的算法对噪声不敏感,具有平移变换、旋转变换、尺度变换、仿射变换等的不变性。比较典型的计算图像特征点的方法有Harris角点,SIFT特征,SURF特征等。好的特征点计算算法往往存在计算量大,难以实现实时计算的问题。鉴于此,有研究者提出基于改进的Harris特征匹配或者SIFT特征匹配的拼接算法,或者利用GPU进行加速计算,也有研究者提出了通过跟踪角点,SIFT特征点,SURF特征点等进行图像配准计算的方法来拼接图像。

发明内容

[0004] 本发明的技术解决问题:和传统特征点配准的方法相比,利用跟踪的方法配准特征点减少了特征点配准的搜索范围,降低了计算复杂度。但即使利用特征点跟踪进行图像配准,仍然难以实现实时拼接,关键原因在于图像特征点提取(在图像中搜索特征明显的点)同样要求较大的计算量。通过指定图像中特定位置的点作为跟踪点,从而省去特征点提取过程,大大降低了图像特征点提取的计算量,最终实现实时拼接。

[0005] 本发明的技术解决方案:主要工作流程(参见图1)如下:(1)指定跟踪点,从图像中自动、均匀选取一定数量的点;(2)利用LK稀疏光流法计算指定跟踪点在当前图像帧中的位置,将所得到的点利用LK稀疏光流法反向计算得到已拼好的图像中的位置;(3)求取指定点与反向计算得到的对应点之间的距离,按从小到大排序后选取前n个点计算单应性矩阵;(4)图像融合并显示拼接图。

[0006] 本发明与现有技术相比的优点在于:和已有图像拼接方法相比,本发明提出的图像拼接方法具有以下两个特点:

[0007] (1)没有特征点提取过程,直接从图像中选取某些像素点作为跟踪点;

[0008] (2) 跟踪点的个数和分布可以控制,从而可以部分控制计算时间和效果。

[0009] 通过对大量不同型号、不同载荷、在不同地点和不同时段拍摄的无人机视频进行试验,证明所提出的该方法实现实时拼接,获得了较好的拼接效果。

附图说明

[0010] 图1为本发明视频拼接基本计算流程图;

[0011] 图2为本发明跟踪点指定方法;

[0012] 图3为本发明拼接效果比较图:(a) SURF算法拼接效果比较图,(b) SIFT算法拼接效果比较图,(c) 本发明提出的指定点跟踪法拼接效果比较图。

具体实施方式

[0013] 下面从跟踪点指定,LK稀疏光流法跟踪,单应性矩阵计算,图像融合和实验结果比较分析等几个方面,结合附图对本发明做进一步介绍。

[0014] 1. 跟踪点指定

[0015] 设图像宽为 w ,高为 h ,从图像中均匀地选取 $m \times m$ 个点作为跟踪点(参见图2,图中所示交叉点即为所指定的跟踪点)。由于边界点不作为跟踪点,则所选取的点的集合为

$$P = \left\{ \left(\frac{w}{m+1}, \frac{h}{m+1} \right), \left(\frac{2 \cdot w}{m+1}, \frac{h}{m+1} \right), \dots, \left(\frac{m \cdot w}{m+1}, \frac{h}{m+1} \right), \dots, \left(\frac{w}{m+1}, \frac{2 \cdot h}{m+1} \right), \left(\frac{2 \cdot w}{m+1}, \frac{2 \cdot h}{m+1} \right), \dots, \left(\frac{m \cdot w}{m+1}, \frac{2 \cdot h}{m+1} \right), \dots, \left(\frac{m \cdot w}{m+1}, \frac{m \cdot h}{m+1} \right) \right\}$$

。两个相邻跟踪点水平距离为 $w/(m+1)$,垂直距离为 $h/(m+1)$ 。

这种指定方法与通常的特征点计算和选取方法相比,所得到的跟踪点分布范围更大,跟踪点间距离不会太近、太密集,有利于得到更精确的配准矩阵;这种方式也省去了特征点选取的复杂运算。

[0016] 通过对尺寸(以像素为单位)为 576×384 和 1024×768 的视频实验测试,发现对于 26×26 及更少个跟踪点,跟踪效果不太理想,而对于 30×30 及以上个跟踪点,虽然跟踪效果不错,但计算时间较长。实验结果显示, 28×28 个跟踪点是一个比较理想的选择。

[0017] 2. LK稀疏光流法跟踪

[0018] 光流场的计算一般分为四类:基于梯度的方法(如Horn-Schunck和Lucas-Kanade即LK算法);基于匹配的方法;基于能量的方法;基于相位的方法等。光流法有以下假设:

[0019] (1) 颜色一致性,一个跟踪点在帧 f_{i-1} 和帧 f_i 中的颜色值是一样的(对于灰度图像来说,就是亮度相同)。

[0020] 即对于图像上点 $p = (x, y)^T$ 在时刻 $t = t_{i-1}$ (拍摄图像帧 f_s 对应的时刻为 t_s)的灰度值为 $I(x, y, t_{i-1})$,经时间间隔 $\Delta t = t_i - t_{i-1}$ 后,对应点的灰度为 $I(x + \Delta x, y + \Delta y, t + \Delta t)$,有

$$\frac{dI(x, y, t)}{dt} = 0, \text{ 则 } I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$$

。假设 $u = \frac{dx}{dt}$, $v = \frac{dy}{dt}$ 分别代表该点

光流矢量在 x, y 方向上的分量,将等式左边使用泰勒公式展开,忽略二阶以上的高次项,如果令 $\Delta t = 0$,则有

[0021] $I_x u + I_y v + I_t = 0$

[0022] 其中 $I_x = \frac{\partial I}{\partial x}$, $I_y = \frac{\partial I}{\partial y}$, $I_t = \frac{\partial I}{\partial t}$ 分别代表图像灰度相对于 x, y, t 的偏导。

[0023] (2) 两幅图像间像素位移比较小。

[0024] (3) 空间一致性,即邻近像素的运动是一致的。

[0025] (4) 不能识别垂直于局部梯度的运动:光流约束方程包含u和v两个未知量,显然由一个方程并不能唯一确定,这就是孔径问题,要解决这个问题,必须找新的约束。

[0026] 由于无人机拍摄的视频重叠率高,基本符合上述的四点假设,应用LK光流法进行跟踪具有合理性。LK算法是基于局部约束的,假定以点p为中心的一个小邻域内各点的光流相同,对区域内不同的点给予不同的权重,这样光流的计算就转化为最小化如下方程:

$$[0027] \quad \sum_{(x,y) \in \Omega} W^2(x,y) [\nabla I(x,y,t) \cdot V + I_x(x,y,t)]^2$$

[0028] 其中 Ω 代表以p点为中心的一个小邻域, ∇ 是梯度算子, $V=(u,v)$ 为光流, $W(x,y)$ 为窗口函数,代表区域中点 (x,y) 的权重,离p点越近,权重越高。

[0029] 为了计算运动较为剧烈视频的光流场,可以采用两种方法。(1) 扩大 Ω 的搜索范围;(2) 采用高斯金字塔分层与LK方法结合起来。第一种方法易于实现,但是会引入较大的计算量,难以适应实时性的要求。第二种方法采用了由粗到精的分层策略将图像分解成不同的分辨率,并将在粗尺度下得到的结果作为下一个尺度的初始值,该方法是计算大运动速度和较大姿态变化的有效技术手段。实验中采用的高斯金字塔层次为3,能较好地处理大运动速度的视频。

[0030] 将上一帧图像 f_{i-1} 指定的 $m \times m$ 个跟踪点用二维列向量表示为: $(x_s, y_s)^T$,其中 $S \in \{1, 2, \dots, m \times m\}$ 。采用金字塔分层和LK结合的方法,以图像 f_i 中的点 $(x_s, y_s)^T$ 为起始点计算 f_{i-1} 中的跟踪点 $(x_s, y_s)^T$ 在 f_i 中的光流。设计算得到在 f_i 中跟踪点的坐标为 $(x_i^{curr}, y_i^{curr})^T$ 。类似地,以 f_i 中点 $(x_i^{curr}, y_i^{curr})^T$ 为跟踪点,计算其在 f_{i-1} 中的光流,得到反向跟踪之后, f_i 中点 $(x_i^{curr}, y_i^{curr})^T$ 在图像 f_{i-1} 上的坐标 $(x_i^{back}, y_i^{back})^T$ 。如果跟踪比较精确,则 $(x_i, y_i)^T$ 与 $(x_i^{back}, y_i^{back})^T$ 的欧氏距离会比较小。

[0031] 3. 单应性矩阵计算

[0032] 根据上一帧图像 f_{i-1} 指定的跟踪点 $(x_s, y_s)^T$ 与经过正、反向跟踪之后得到的 f_{i-1} 中对应点 $(x_s^{back}, y_s^{back})^T$,可以使用如下的公式计算每个特征点的误差值 $error_s$ 。 $error_s$ 值越小表示该点被跟踪得越精确,它也表示了 f_{i-1} 中的跟踪点 $(x_s, y_s)^T$ 与 f_i 中跟踪点 $(x_s^{curr}, y_s^{curr})^T$ 的匹配程度。

$$[0033] \quad error_s = \sqrt{(x_s - x_s^{back})^2 + (y_s - y_s^{back})^2}$$

[0034] 根据每个点的误差值进行排序,选择其中误差最小的n对匹配点计算两帧图像之间的单应性矩阵。采用误差值的排序可以有效剔除误匹配点。

[0035] 采用误差值排序方法可以剔除大量的误匹配点,但是为了得到更精确的单应性矩阵,我们还采用了随机抽样一致性(Random Sample Consensus, RANSAC)算法对匹配点对进行过滤。该算法在误匹配点较多的情况下仍能保持有效,缺点是计算量大,速度慢。计算所得到的单应性矩阵为图像 f_i 到图像 f_{i-1} 的变换关系,用 H_{i-1}^i 表示。

[0036] 将当前图像 f_i 变换到第一帧图像 f_1 的视角,完成基于第一张视角的全景图拼接。给人感觉是基于第一帧图像的视角看到的全景图。使用如下的公式计算第一帧图像 f_1 与当前图像 f_i 之间的变换关系 H_1^i 。

$$[0037] \quad H_1^i = \prod_2^{k=i} H_{k-1}^k$$

[0038] 直接采用公式可以将之前计算出来的各个矩阵进行累乘,从理论上可以得到图像 f_i 到图像 f_1 的变换关系。但由于每次矩阵乘法运算都涉及到双精度浮点数的精度问题,这会导致在累乘多个变换矩阵之后产生较大误差。为消除浮点精度带来的计算误差,采用如下方式计算图像 f_i 到图像 f_1 之间的变换关系:通过 f_{i-1} 和 f_1 在全局坐标系下四个角点坐标之间的对应关系计算得到 H_1^{i-1} ,计算 f_i 与 f_{i-1} 的变换关系 H_{i-1}^i 。从而可以计算出 $H_1^i = H_1^{i-1} * H_{i-1}^i$ 。

[0039] 通过 H_1^i ,可以将图像 f_i 变换至全景图上。采用该方法可以有效消除浮点精度带来的误差,可以连续拼接更多图片。

[0040] 4. 图像融合

[0041] 完成 f_i 与 f_{i-1} 之间跟踪点匹配,并且得到了变换矩阵 H_{i-1}^i 之后,为了得到无缝拼接的大图,需要选用合适的图像融合算法对两幅图像进行融合。图像融合是一种特殊的数据融合,指将多源信道所采集到的关于同一目标的图像数据经过图像处理和计算机技术等,将两幅图像上的信息综合为一幅信息量更大的图,提高图像信息量。图像融合有很多方法,如直接平均值法,加权平均法,距离权重法,多分辨率融合法等。本发明采用加权平均法实现图像融合。为了消除图像拼接空洞问题,通常采用反向求对应像素点,这要求计算 H_{i-1}^i 的逆矩阵 $(H_{i-1}^i)^{-1}$,用 H_i^{i-1} 表示。图像融合的基本步骤为:

[0042] (1) 利用 H_i^{i-1} 计算 f_i 四个角点在 g (拼接好的大图)中的坐标,得到 f_i 在 g 中的包围盒 B ;

[0043] (2) 对 B 中每个像素点 $(x, y)^T$,利用 H_i^{i-1} 计算得到其对应于 f_i 中的像素点 $(x', y')^T$,由于得到的像素坐标不一定是整数,本发明取最临近点的像素值进行计算;

[0044] (3) 如果包围盒 B 中某点 (x, y) 的亮度值 $I_B(x, y) \neq 0$,那么进行融合计算,更新 $I_B(x, y)$ 为 $\alpha I_{f_i}(x', y') + \beta I_B(x, y)$,实验中设置 $\alpha = 0.4, \beta = 0.6$;如果 $I_B(x, y) = 0$,则更新 $I_B(x, y)$ 为 $I_{f_i}(x', y')$ 。

[0045] 在步骤2中,也可采用线性插值或者更复杂的方法进行计算,但是这些方法同时也会增加计算时间。采用最临近法可能会导致拼接后的图像出现颜色突变,步骤3的融合在某种程度上弥补了这种不足。

[0046] 5. 实验结果比较分析

[0047] 所提出的算法在Windows 7操作系统,Intel i7 CPU处理器,4GB内存,主要将本发明提出的指定点跟踪法与基于SIFT和SURF(基于OpenCV库)的拼接方法进行了对比,得到如表1所示结果。

[0048] 从表中可见,在给定相应参数的情况下,指定点跟踪法拼接的计算时间约为SIFT的1/4,SURF的1/6,其帧率达到了约17帧/秒。通过抽帧处理,实现了较好、较长时间稳定的实时拼接效果。由于无人机视频帧间重叠率较高,每隔一帧取一帧基本不影响拼接效果。

[0049] 图3给出了基于三个配准算法得到拼接图的效果比较,图中左边列的拼接图效果较好,右边列的图则都存在一定的问題。其中SURF算法(图3(a))得到的配准矩阵出现了较大的误差,1号圈中同一条路拼接后出现了分叉。SIFT算法(图3(b))的拼接出现了断裂(2号圈)和鬼影(4号圈),而3号圈则道路消失了一部分。而基于指定点跟踪法的拼接(图3(c))虽然出现了断裂(5号圈)现象,但总体上是接受的。更重要的是,本发明由于采用了新的单应性矩阵计算策略,没有SURF和SIFT算法的拼接图像后期变形严重。

[0050] 本发明虽然没有选择特征明显的点进行跟踪,而是指定跟踪点,并且使用了较简单的融合算法,但是由于(1)跟踪点个数可以指定,并且通过RANSAC以后,通常还会减少,通过选取约40个最好的跟踪点,可以计算得到比较好的配准矩阵;(2)采用指定点跟踪法大大提高了计算速度,在进行实时视频拼接时,配准速度的提高使得抽帧数目减少,一般不会出现连续抽掉两帧或多帧的情况,因此帧间位移较小,减少了计算RANSAC的迭代次数,并且得到了更好的配准矩阵,从而实现了较好的实时视频图像拼接。

[0051] 表1

[0052]

匹配算法	匹配用时(毫秒)	RANSAC用时(毫秒)	最大拼接次数	特征点数量	参数
本发明方法	55.4	0.54	362	784	28×28个特征点 最大迭代次数5 选取前40对特征点
SURF	305.209	1.161	515	1872	nOctaves=2 nOctaveLayers=4 hessianThreshold=50
SIFT	205.35	1.21	185	1763	nOctaves=2 nOctaveLayers=4 contrastThreshold=0.01

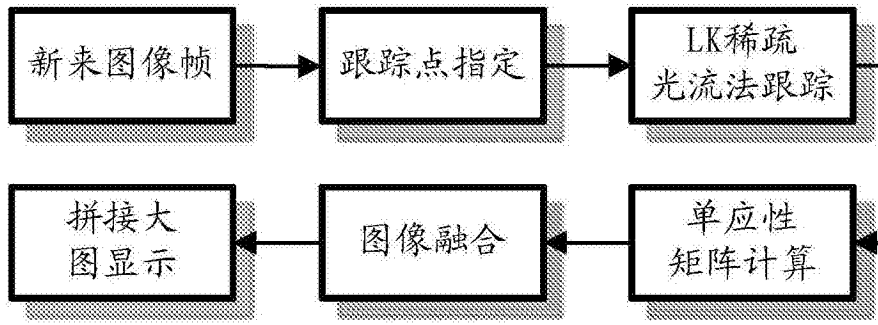


图1

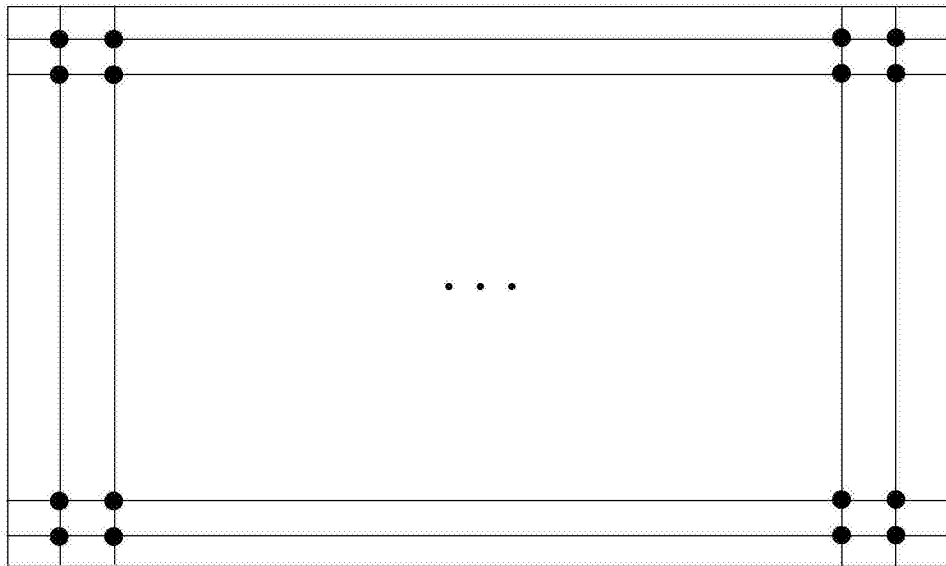
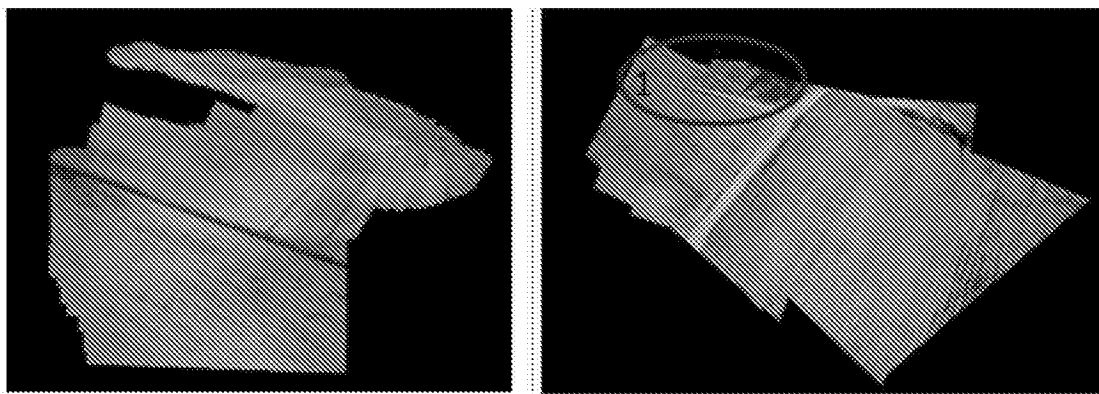
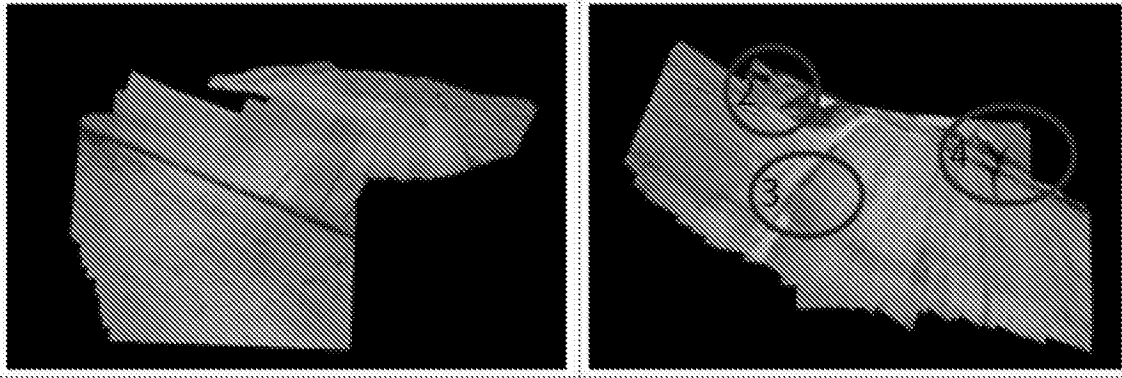


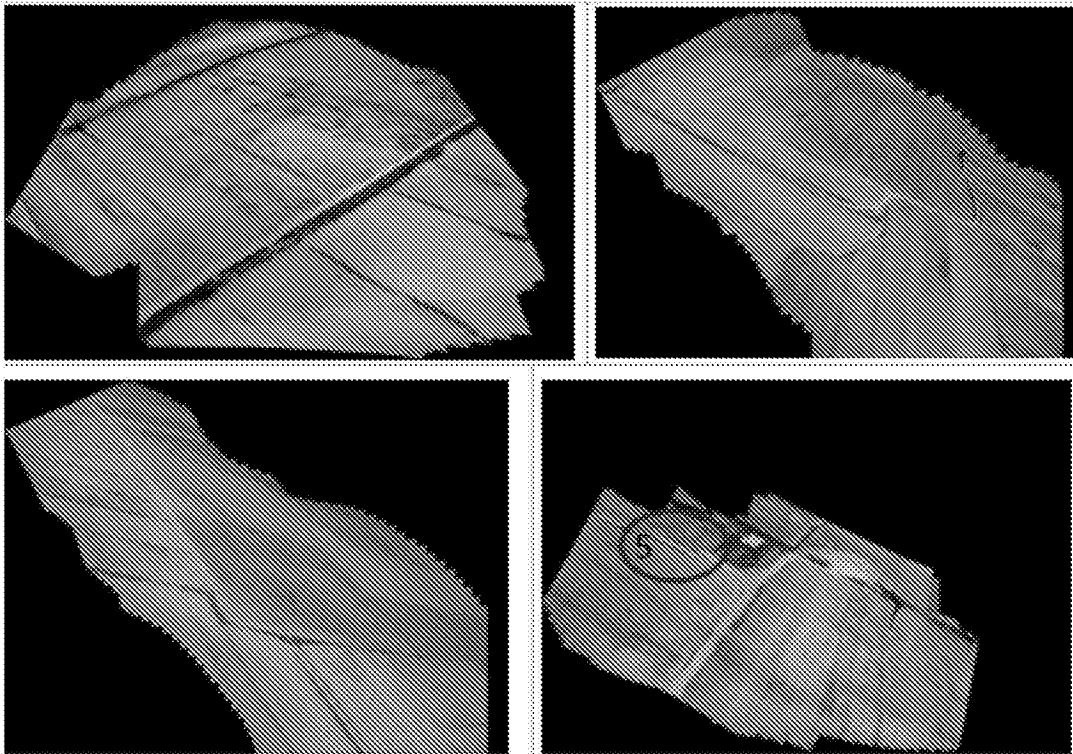
图2



(a)



(b)



(c)

图3