



# (12)发明专利

(10)授权公告号 CN 104915986 B

(45)授权公告日 2018.04.17

(21)申请号 201510362321.0

CN 103325142 A, 2013.09.25,

(22)申请日 2015.06.26

CN 104299260 A, 2015.01.21,

(65)同一申请的已公布的文献号

申请公布号 CN 104915986 A

肖路昊等. 实体三维模型自动建模技术研究.《中国体视学与图像分析》.2014,第19卷(第3期),参见第243-249页.

(43)申请公布日 2015.09.16

王继东等. 一种新的边折叠网格模型简化算法.《计算机工程与应用》.2013,第49卷(第1期),参见第195-198页.

(73)专利权人 北京航空航天大学

地址 100191 北京市海淀区学院路37号

(72)发明人 肖路昊 龚光红

刘鑫等. 基于GPU和Kinect的快速物体重建.《自动化学报》.2012,第38卷(第8期),参见第1288-1297页.

(74)专利代理机构 北京永创新实专利事务所

11121

代理人 赵文利

张宇. 实时渲染中环境映射技术的研究与应用.《中国优秀硕士学位论文全文数据库信息科技辑》.2013,(第7期),参见第19-22、25-26页.

(51)Int. Cl.

G06T 17/00(2006.01)

审查员 刘娜

(56)对比文件

CN 103279987 A, 2013.09.04,

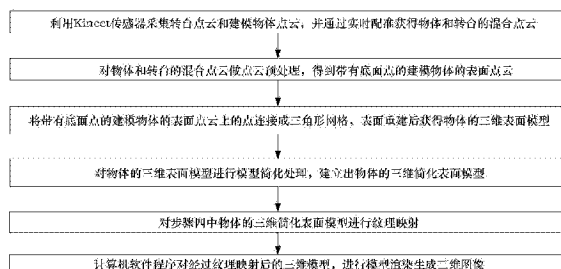
权利要求书4页 说明书13页 附图9页

(54)发明名称

一种实体三维模型自动建模方法

(57)摘要

本发明公开了一种实体三维模型自动建模方法,包括以下步骤:1)、采集转台点云和建模物体点云,并实时配准获得物体和转台的混合点云;2)、点云预处理得到带有底面点的建模物体的表面点云;3)、表面重建获得物体的三维表面模型;4)、对三维表面模型进行模型简化处理,建立出物体的三维简化表面模型;5)、对三维简化表面模型进行纹理映射;6)、对经过纹理映射后的三维模型,进行模型渲染生成二维图像。优点在于:本系统运行稳定可靠,运行过程实现了自动化,重建效果良好,建立出来的多粒度三维模型可以满足不同仿真场景的应用需要。



1.一种实体三维模型自动建模方法,基于Kinect传感器搭建的实体三维模型建模系统,该系统包括Kinect传感器、转台和计算机,其特征在于,包括以下步骤:

步骤一、利用Kinect传感器采集转台点云和建模物体点云,并通过实时配准获得物体和转台的混合点云;

步骤二、对物体和转台的混合点云做点云预处理,得到带有底面点的建模物体的表面点云;

步骤三、将带有底面点的建模物体的表面点云上的点连接成三角形网格,表面重建后获得物体的三维表面模型;

步骤四、对物体的三维表面模型进行模型简化处理,建立出物体的三维简化表面模型;

采用边折叠方法,按照比例删去三维网格模型上的点和边,建立出物体的三维简化表面模型;

步骤五、对步骤四中物体的三维简化表面模型进行纹理映射;

纹理映射分为两部分,首先是将带有彩色信息的点投影,生成彩色纹理点图;然后是对彩色纹理点图进行邻域颜色插值,填充点图上的未着色区域,形成完整的纹理图片;

投影的步骤如下:

步骤501、计算输入点云X的三维质心坐标,得到投射中心点C:

$$\begin{cases} x_c = \sum_{i=1}^n x_i / n \\ y_c = \sum_{i=1}^n y_i / n \\ z_c = \sum_{i=1}^n z_i / n \end{cases}$$

其中,点云X是没有底部的物体点云,投射中心点C坐标为: $(x_c, y_c, z_c)$ ,点云X上的点坐标为 $(x_i, y_i, z_i)$ ,n是点云X的点数,n为整数;

步骤502、点云X投影到以投射中心点C为球心、半径R的球体上,得到点集Ps:

$$\begin{cases} x_{si} = x_c + (x_i - x_c)t_i \\ y_{si} = y_c + (y_i - y_c)t_i \\ z_{si} = z_c + (z_i - z_c)t_i \end{cases}$$

其中, $(x_{si}, y_{si}, z_{si})$ 是点集Ps上的点坐标,

半径R需要满足条件 $R > \text{MAX}\{x_i^2 + y_i^2 + z_i^2\}$ , $t_i$ 是点云X上的点 $(x_i, y_i, z_i)$ 在球体上的距离系数,该系数的数值不同,则点 $(x_i, y_i, z_i)$ 在球体上的投影点 $(x_{si}, y_{si}, z_{si})$ 位置不同;

$$t_i = \frac{R}{\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2}}$$

步骤503、把点集Ps平移到以C为原点的直角坐标系下,得到点集Ps',再转换成球坐标,得到点集Ps'':

$$\begin{cases} x_{si}' = x_{si} - x_c \\ y_{si}' = y_{si} - y_c \\ z_{si}' = z_{si} - z_c \end{cases}$$

$$\begin{cases} r_i = R \\ \phi_i = \arctan\left(\frac{y_{si}'}{x_{si}'}\right) \\ \theta_i = \arccos\left(\frac{z_{si}'}{r_i}\right) \end{cases}$$

其中,  $(x_{si}', y_{si}', z_{si}')$  是点集  $P_s'$  中点  $p_{si}'$  的直角坐标系坐标,  $(r_i, \phi_i, \theta_i)$  是点  $p_{si}'$  的球坐标系坐标,  $r_i$  表示球坐标系原点  $C$  与点  $p_{si}'$  之间的距离,  $\theta_i$  表示有向线段  $Cp_{si}'$  与  $z$  轴正向的夹角,  $\phi_i$  表示从  $z$  轴正方向看自  $x$  轴按逆时针方向转到有向线段  $Cp_{si}'$  在  $xCy$  平面的投影所转过的角度;

步骤504、将点集  $P_s'$  进行等距圆柱投影, 得到纹理坐标点集  $P_c$ :

$$\begin{cases} u_i = r_i \phi_i \\ v_i = r_i \left(\frac{\pi}{2} - \theta_i\right) \end{cases}$$

其中,  $(u_i, v_i)$  是纹理坐标点集  $P_c$  内的点二维坐标系内的坐标,  $u_i$  为水平方向的像素坐标,  $v_i$  为垂直方向的像素坐标;

点云  $X$  中的每个点都对应一个纹理图片上二维坐标系内的坐标  $(u_i, v_i)$ ;

步骤505、新建一张黑色图片, 将纹理坐标点集  $P_c$  中点  $(u_i, v_i)$  的RGB值填充为点云  $X$  的点  $(x_i, y_i, z_i)$  的RGB值  $(red_i, green_i, blue_i)$ , 形成彩色纹理点图;

邻域颜色插值的步骤如下:

步骤I、检索投影得到的彩色纹理点图的每一个像素点, 如果当前像素点  $p_i$  的RGB值是  $(0, 0, 0)$ , 该点为非彩色点, 进入步骤II, 否则, 像素点  $p_i$  的RGB值不是  $(0, 0, 0)$ , 该点为彩色点, 跳到步骤V;

步骤II、在以非彩色点  $p_i$  为中心、边长为  $l_{rect}$  像素的正方形邻域中进行遍历, 判断遍历结束后邻域内是否存在彩色点, 如果至少存在一个彩色点, 则进行步骤III, 否则, 该邻域内不存在彩色点, 进入步骤IV;

步骤III、计算邻域中每个彩色点的权重, 再进入步骤V;

公式如下:

$$\omega_k = \frac{1/d_k}{\sum_{j=1}^m (1/d_j)}$$

$\omega_k$  为当前彩色点  $p_k$  的权重,  $k$  为整数;  $d_k = ||p_k - p_i||$ ,  $d_k$  为当前彩色点  $p_k$  到中心点  $p_i$  的距离;  $d_j$  是邻域中彩色点  $p_j$  距中心点  $p_i$  的距离,  $j = 1 \dots k \dots m$ ;  $m$  是邻域内彩色点的数量,  $m$  为整数;

步骤IV、改变邻域大小为 $l_{rect}=l_{rect}+\Delta l$ ,遍历次数加1,重新进行遍历;如果存在至少一个彩色点,则进入步骤III;

如果遍历次数N达到预设的上限后,没有找到任何彩色点,进入步骤VI;

步骤V、如果 $p_i$ 是步骤I传输过来的彩色点,则直接将 $p_i$ 的RGB值添加到新的点集 $P_t$ 中;如果 $p_i$ 是步骤IV传输过来的点,则利用权重计算点 $p_i$ 的RGB值,并添加到新的点集 $P_t$ ,完成邻域颜色的插值;

$$\begin{cases} red_i = \sum_{k=1}^n red_k \omega_k \\ green_i = \sum_{k=1}^n green_k \omega_k \\ blue_i = \sum_{k=1}^n blue_k \omega_k \end{cases}$$

点 $p_i$ 的RGB值为 $(red_i, green_i, blue_i)$ ,点 $p_k$ 的RGB值为 $(red_k, green_k, blue_k)$ , $\omega_k$ 为彩色点 $p_k$ 的权重;

步骤VI、对下一个像素点 $p_{i+1}$ ,执行步骤I到步骤V,直到彩色纹理点图遍历结束;

步骤六、计算机软件程序对经过纹理映射后的三维模型,进行模型渲染生成二维图像。

2.如权利要求1所述的一种实体三维模型自动建模方法,其特征在于,所述的转台点云是指利用Kinect传感器采集没有放置建模物体时的点云;所述的建模物体点云是指放置建模物体于转台上,在转台旋转过程中利用Kinect传感器采集的点云;所述的实时配准是指对建模物体点云利用三维重建项目Kinect Fusion进行实时配准,得到物体和转台的混合点云。

3.如权利要求1所述的一种实体三维模型自动建模方法,其特征在于,所述的步骤二具体为:步骤201、采用RANSAC平面提取方法,提取并删除物体和转台的混合点云中包含的转台底座点云,得到物体和转台旋转平台的混合点云;

步骤202、采用欧氏距离聚类分割方法,滤除转台点云中所包含的转台底座点云;

步骤203、对步骤201和步骤202得到的点云,均采用体素栅格降采样方法处理,获得物体和转台的混合点云的简化点云,以及转台点云的简化点云;

步骤204、分别计算物体和转台混合点云的简化点云以及转台点云的简化点云的FPFH三维特征,获得物体和转台混合点云的特征点云以及转台点云的特征点云;

步骤205、经过步骤204处理后,将物体和转台的混合点云的特征点云以及转台点云的特征点云进行配准,获得配准点云;

配准方法以点云的FPFH特征作为配准依据,通过比较各自的特征点云的FPFH特征,将两个特征点云统一到同一个坐标系下,得到配准点云;

步骤206、使用RANSAC方法分割配准点云,获得转台旋转平台点云和物体点云;

将进行配准的两个特征点云中重合的属于转台旋转平台部分的点云提取出来,形成转台旋转平台点云,其余部分形成物体点云;

步骤207、用欧氏距离聚类分割方法滤除物体点云中的噪声,得到没有底部的物体点云;

步骤208、从步骤206中的转台旋转平台点云通过基于k-d树的点搜索方法提取构成物

体点云底部的点云；

步骤209、融合步骤208中的物体点云底部的点云与步骤207中的没有底部的物体点云，得到带有底面点的建模物体的表面点云。

4. 如权利要求1所述的一种实体三维模型自动建模方法，其特征在于，所述的步骤三采用Poisson表面重建方法进行表面重建，建立三维网格模型。

5. 如权利要求1所述的一种实体三维模型自动建模方法，其特征在于，所述的步骤II中，当邻域跨越彩色纹理点图的边界时，则超出边界的部分从当前边界相对的另一条边界处起继续遍历像素。

## 一种实体三维模型自动建模方法

### 技术领域

[0001] 本发明涉及点云处理技术、三维建模技术和纹理生成技术,属于虚拟现实领域,具体来说是一种实体三维模型自动建模方法。

### 背景技术

[0002] 图像的建模与渲染技术,简称IBMR(Image-Based Modeling and Rendering),利用物体的二维图像来生成三维模型,物体的几何特征信息和渲染特征信息不需要用户输入,直接从摄像头采集到的图像中获得。

[0003] IBMR技术分为两种方式:主动法和被动法。主动法在图像采集过程中,使用光源向场景中发射光线,如红外线或者可见光,主动地获得场景的反馈,从而建立场景的三维模型;被动法不直接控制光源,通过被动地分析图像来建立模型。主动法直接获得物体表面的各种特征,简化建模过程,随着三维信息处理算法越来越成熟可靠,以及三维信息采集硬件的不断涌现,主动法三维建模技术正在迅速地发展。

[0004] 三维建模技术在工业制造、游戏娱乐和虚拟现实等领域中有广泛的应用。在实际应用中,人们对实体三维模型可视化效果提出了越来越高的要求,例如逆向工程所获得的实体模型是否可以更精确,三维电子游戏中的人物和环境是否可以更真实等等。除了模型的精确度,人们还对三维建模技术的快速性和便利性提出了要求,例如如何缩短建模时间,如何减小建模过程的复杂性,如何降低三维建模技术的使用难度,这些都是三维建模技术研究的方向。

[0005] 目前,国内外已经有多种多样的三维建模系统,包括结构光建模系统、激光三维扫描仪等。现有的建模系统难以在保证性价比的同时实现高效用,并且还存在以下缺点:

[0006] 一方面,高精度的三维建模系统常常基于一些复杂昂贵的传感器,例如结构光传感器、激光测距传感器或者激光雷达;

[0007] 另一方面,三维建模系统的建模过程中往往离不开繁复的人工交互过程,例如参数调整、点云手动配准等等,费时费力。

[0008] 基于上述缺点,研究快速、自动的实体模型建模技术越来越重要,以相对低的成本实现自动化三维建模,使得实体的三维模型可以和照片视频一样方便地获取,满足不同领域对三维建模技术越来越大的需求。

### 发明内容

[0009] 本发明的目的是为了解决三维建模流程中人工交互操作复杂繁琐的问题,以及依赖昂贵传感器的问题,提出了一种实体三维模型自动建模方法。

[0010] 一种实体三维模型自动建模方法,基于Kinect传感器搭建实体三维模型建模系统,该系统包括Kinect传感器、转台和计算机,转台由计算机通过串口进行通讯。该方法具体步骤如下:

[0011] 步骤一、利用Kinect传感器采集转台点云和建模物体点云,并通过实时配准获得

物体和转台的混合点云；

[0012] 转台点云是指利用Kinect传感器采集没有放置建模物体时的点云；

[0013] 建模物体点云是指放置建模物体于转台上,在转台旋转过程中利用Kinect传感器采集的点云；

[0014] 对建模物体点云利用三维重建项目Kinect Fusion进行实时配准,得到物体和转台的混合点云。

[0015] 步骤二、对物体和转台的混合点云做点云预处理,得到带有底面点的建模物体的表面点云。

[0016] 步骤201、采用RANSAC平面提取方法,提取并删除物体和转台的混合点云中包含的转台底座点云,得到物体和转台旋转平台的混合点云。

[0017] 物体和转台的混合点云中包括了建模物体的点云,转台底座点云和转台旋转平台部分的点云,将转台底座点云剔除,获得物体和转台旋转平台的混合点云；

[0018] 步骤202、采用欧氏距离聚类分割方法,滤除转台点云中所包含的转台底座点云。

[0019] 步骤203、对步骤201和步骤202得到的点云,均采用体素栅格降采样方法处理,获得物体和转台的混合点云的简化点云,以及转台点云的简化点云。

[0020] 步骤204、分别计算物体和转台混合点云的简化点云以及转台点云的简化点云的FPFH三维特征,获得物体和转台混合点云的特征点云以及转台点云的特征点云。

[0021] 步骤205、经过步骤204处理后,将物体和转台的混合点云的特征点云以及转台点云的特征点云进行配准,获得配准点云。

[0022] 配准方法以点云的FPFH特征作为配准依据,通过比较各自的特征点云的FPFH特征,将两个特征点云统一到同一个坐标系下,得到配准点云。

[0023] 步骤206、使用RANSAC方法分割配准点云,获得转台旋转平台点云和物体点云；

[0024] 将配准点云使用RANSAC方法提取平面,将进行配准的两个特征点云中重合的属于转台旋转平台部分的点云提取出来,形成转台旋转平台点云,其余部分形成物体点云。

[0025] 步骤207、用欧氏距离聚类分割方法滤除物体点云中的噪声,得到没有底部的物体点云；

[0026] 步骤208、从步骤206中的转台旋转平台点云通过基于k-d树的点搜索方法提取构成物体点云底部的点云。

[0027] 从转台旋转平台点云中获取部分属于建模物体的底面点云,该部分点云的密度明显低于其他部分的点云,通过基于k-d树的点搜索方法去除点密度较大的点集,剩余的点云构成物体点云底部的点云。

[0028] 步骤209、融合步骤208中的物体点云底部的点云与步骤207中的没有底部的物体点云,得到带有底面点的建模物体的表面点云。

[0029] 步骤三、将带有底面点的建模物体的表面点云上的点连接成三角形网格,表面重建后获得物体的三维表面模型；

[0030] 采用Poisson表面重建方法进行表面重建,建立三维网格模型。

[0031] 步骤四、对物体的三维表面模型进行模型简化处理,建立出物体的三维简化表面模型。

[0032] 根据仿真系统的具体要求,采用边折叠方法,按照比例删去三维网格模型上的点

和边,建立出点数较少、三角形面片数较少的三维模型。

[0033] 步骤五、对步骤四中物体的三维简化表面模型进行纹理映射。

[0034] 纹理映射分为两部分,首先是将带有彩色信息的点投影,生成彩色纹理点图;然后是对彩色纹理点图进行邻域颜色插值,填充点图上的未着色区域,形成完整的纹理图片。

[0035] 投影的步骤如下:

[0036] 步骤501、计算输入点云X的三维质心坐标,得到投射中心点C:

$$[0037] \begin{cases} x_c = \sum_{i=1}^n x_i / n \\ y_c = \sum_{i=1}^n y_i / n \\ z_c = \sum_{i=1}^n z_i / n \end{cases}$$

[0038] 其中,点云X是没有底部的物体点云,投射中心点C坐标为:  $(x_c, y_c, z_c)$ ,点云X上的点坐标为  $(x_i, y_i, z_i)$ ,n是点云X的点数,n为整数;

[0039] 步骤502、点云X投影到以投射中心点C为球心、半径R的球体上,得到点集Ps

$$[0040] \begin{cases} x_{si} = x_c + (x_i - x_c)t_i \\ y_{si} = y_c + (y_i - y_c)t_i \\ z_{si} = z_c + (z_i - z_c)t_i \end{cases}$$

[0041] 其中,  $(x_{si}, y_{si}, z_{si})$  是点集Ps上的点坐标,

[0042] 半径R需要满足条件  $R > \text{MAX} \{x_i^2 + y_i^2 + z_i^2\}$ ,  $t_i$  是点云X上的点  $(x_i, y_i, z_i)$  在球体上的距离系数,该系数的数值不同,则点  $(x_i, y_i, z_i)$  在球体上的投影点  $(x_{si}, y_{si}, z_{si})$  位置不同;

$$[0043] t_i = \frac{R}{\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2}}$$

[0044] 步骤503、把点集Ps平移到以C为原点的直角坐标系下,得到点集Ps',再转换成球坐标,得到点集Ps'':

$$[0045] \begin{cases} x_{si}' = x_{si} - x_c \\ y_{si}' = y_{si} - y_c \\ z_{si}' = z_{si} - z_c \end{cases}$$

$$[0046] \begin{cases} r_i = R \\ \phi_i = \arctan\left(\frac{y_{si}'}{x_{si}'}\right) \\ \theta_i = \arccos\left(\frac{z_{si}'}{r_i}\right) \end{cases}$$

[0047] 其中,  $(x_{si}', y_{si}', z_{si}')$  是点集Ps'中点  $p_{si}'$  的直角坐标系坐标,  $(r_i, \phi_i, \theta_i)$  是点  $p_{si}'$



的球坐标系坐标,  $r_i$  表示球坐标系原点  $C$  与点  $p_{psi}'$  之间的距离,  $\theta_i$  表示有向线段  $Cp_{psi}'$  与  $z$  轴正向的夹角,  $\phi_i$  表示从  $z$  轴正方向看自  $x$  轴按逆时针方向转到有向线段  $Cp_{psi}'$  在  $xCy$  平面的投影所转过的角度;

[0048] 步骤504、将点集  $P_s''$  进行等距圆柱投影, 得到纹理坐标点集  $P_c$ :

$$[0049] \quad \begin{cases} u_i = r_i \phi_i \\ v_i = r_i \left( \frac{\pi}{2} - \theta_i \right) \end{cases}$$

[0050] 其中,  $(u_i, v_i)$  是纹理坐标点集  $P_c$  内的点二维坐标系内的坐标,  $u_i$  为水平方向的像素坐标,  $v_i$  为垂直方向的像素坐标;

[0051] 点云  $X$  中的每个点都对应一个纹理图片上二维坐标系内的坐标  $(u_i, v_i)$ 。

[0052] 步骤505、新建一张黑色图片, 将纹理坐标点集  $P_c$  中点  $(u_i, v_i)$  的RGB值填充为点云  $X$  的点  $(x_i, y_i, z_i)$  的RGB值  $(red_i, green_i, blue_i)$ , 形成彩色纹理点图。

[0053] 邻域颜色插值的步骤如下:

[0054] 步骤I、检索投影得到的彩色纹理点图的每一个像素点, 如果当前像素点  $p_i$  的RGB值是  $(0, 0, 0)$ , 该点为非彩色点, 进入步骤II, 否则, 像素点  $p_i$  RGB值不是  $(0, 0, 0)$ , 为彩色点, 跳到步骤V。

[0055] 步骤II、在以非彩色点  $p_i$  为中心、边长为  $l_{rect}$  像素的正方形邻域中进行遍历, 判断遍历结束后邻域内是否存在彩色点, 如果至少存在一个彩色点, 则进行步骤III, 否则, 该邻域内不存在彩色点, 进入步骤IV。

[0056] 遍历次数设为整数  $N$ ; 当邻域跨越彩色纹理点图的边界时, 则超出边界的部分从当前边界相对的另一条边界处起继续遍历像素。

[0057] 步骤III、计算邻域中每个彩色点的权重, 再进入步骤V;

[0058] 公式如下:

$$[0059] \quad \omega_k = \frac{1/d_k}{\sum_{j=1}^m (1/d_j)}$$

[0060]  $\omega_k$  为当前彩色点  $p_k$  的权重,  $k$  为整数;  $d_k = ||p_k - p_i||$ ,  $d_k$  为当前彩色点  $p_k$  到中心点  $p_i$  的距离;  $d_j$  是邻域中每个彩色点  $p_j$  距中心点  $p_i$  的距离,  $j = 1 \dots k \dots m$ ;  $m$  是邻域内彩色点的数量,  $m$  为整数;

[0061] 步骤IV、改变邻域大小为  $l_{rect} = l_{rect} + \Delta 1$ , 遍历次数加1, 重新进行遍历。如果存在至少一个彩色点, 则进入步骤III; 如果遍历次数  $N$  达到预设的上限后, 没有任何彩色点, 进入步骤VI;

[0062] 步骤V、如果  $p_i$  是步骤I传输过来的彩色点, 则直接将  $p_i$  的RGB值添加到新的点集  $P_t$  中; 如果  $p_i$  是步骤IV传输过来的点, 则利用权重计算点  $p_i$  的RGB值, 并添加到新的点集  $P_t$ , 完成邻域颜色的插值;

$$[0063] \quad \begin{cases} red_i = \sum_{k=1}^n red_k \omega_k \\ green_i = \sum_{k=1}^n green_k \omega_k \\ blue_i = \sum_{k=1}^n blue_k \omega_k \end{cases}$$

[0064] 点 $p_i$ 的RGB值为 $(red_i, green_i, blue_i)$ ,点 $p_k$ 的RGB值为 $(red_k, green_k, blue_k)$ , $\omega_k$ 为彩色点 $p_k$ 的权重。

[0065] 步骤VI、对下一个像素点 $p_{i+1}$ ,执行步骤I到步骤V,直到彩色纹理点图遍历结束。

[0066] 自适应邻域颜色插值方法最后得到的结果是一张完整的建模物体表面彩色纹理图片。

[0067] 步骤六、计算机软件程序对经过纹理映射后的三维模型,进行模型渲染生成二维图像。

[0068] 本发明的优点在于:

[0069] (1) 本发明一种实体三维模型自动建模方法,基于的建模系统设备简单,价格廉价,主要利用微软的深度传感器Kinect来搭建,具有重要的实用价值,Kinect作为主动传感器,不受环境可见光谱的干扰。

[0070] (2) 本发明一种实体三维模型自动建模方法,三维建模过程实现自动化,从放置建模对象、启动系统开始,全过程通过软件控制,建模的操作步骤简单,无须像其他三维建模工具一样在建模过程中频繁地进行手动交互,或者手动调整参数;

[0071] (3) 本发明一种实体三维模型自动建模方法,利用微软Kinect Fusion提供的函数接口实现实时自动的点云配准,大大缩短建模时间,最终所得模型的精度可以满足一般需求;

[0072] (4) 本发明一种实体三维模型自动建模方法,经过基于彩色信息的纹理映射和模型渲染,表面重建后的物体三维模型在真实感上得到增强。

## 附图说明

[0073] 图1是本发明一种实体三维模型自动建模方法的流程图;

[0074] 图2是本发明一种实体三维模型自动建模方法中点云预处理的流程图;

[0075] 图3(a)是本发明没有放置建模物体时的转台点云效果图;

[0076] 图3(b)是本发明物体和转台的混合点云效果图;

[0077] 图4(a)是采用RANSAC方法处理后的物体和转台的混合点云效果图;

[0078] 图4(b)是滤除转台底座点云后得到的转台旋转平台部分的点云效果图;

[0079] 图5是物体和转台的混合点云与转台点云的特征点云进行配准之后的点云效果图;

[0080] 图6(a)是经过RANSAC方法处理后的物体混合点云效果图;

[0081] 图6(b)是RANSAC方法提取出的转台旋转平台点云效果图;

[0082] 图6(c)是欧式距离聚类方法处理物体点云后得到的没有底部的物体点云效果图;

[0083] 图7是带有底面点的建模物体的表面点云效果图;

- [0084] 图8(a)是水杯经过Poisson表面重建的效果图；
- [0085] 图8(b)是水杯经过表面重建后的三角形网格效果图；
- [0086] 图8(c)是水杯的三角形网格效果局部放大图；
- [0087] 图9(a)是纸盒模型简化前的效果对比图；
- [0088] 图9(b)是纸盒模型简化后的效果对比图；
- [0089] 图10是投影过程的示意图；
- [0090] 图11是本发明一种实体三维模型自动建模方法中投影的流程图；
- [0091] 图12是投影方法形成的彩色纹理点图；
- [0092] 图13是颜色插值得到的纸盒的纹理图片；
- [0093] 图14是本发明一种实体三维模型自动建模方法中邻域颜色插值的流程图；
- [0094] 图15是自适应邻域检索发生跨越的示意图；
- [0095] 图16是自适应邻域颜色插值方法形成的纸盒的纹理图片；
- [0096] 图17(a)是纹理映射之前的纸盒三维模型示意图；
- [0097] 图17(b)是纹理映射之后的纸盒三维模型示意图；
- [0098] 图18是三个典型物体的重建效果图；
- [0099] 图19(a)是未简化的物体模型在OGRE中的显示效果图；
- [0100] 图19(b)是简化后的物体模型在OGRE中的显示效果图。

### 具体实施方式

- [0101] 下面将结合附图和实施例对本发明作进一步的详细说明。
- [0102] 一种实体三维模型自动建模方法,基于Kinect传感器搭建实体三维模型建模系统,该系统包括Kinect传感器、转台和计算机。
- [0103] 计算机具有独立显卡配置,通过RS232串口连接步进电机,驱动转台做匀速360°旋转运动。同时通过信号线与Kinect传感器连接,控制Kinect传感器的采集过程。
- [0104] 转台选用单轴变速转台系统,包括旋转平台和底座;转台由步进电机驱动,步进电机控制转台的旋转速度和圈数,实现平台上物体的旋转运动。
- [0105] Kinect传感器则通过USB 2.0接口与计算机进行通讯,Kinect传感器是微软公司开发的Xbox-360游戏主机的周边设备,由彩色相机、红外发射装置和红外深度传感器构成,Kinect传感器定时地采集不同视角下物体的表面信息;表面信息包括深度信息和彩色信息,深度信息为物体表面点相对Kinect传感器坐标系的距离,彩色信息为物体表面点的颜色或纹理等。
- [0106] 一种实体三维模型自动建模方法,分为实时处理和离线处理两部分。
- [0107] 实时处理获取建模物体的点云数据,实时配准生成物体和转台的混合点云后,进行离线处理,实现自动化的处理,处理过程中无需进行参数调整等手动交互过程。
- [0108] 实时处理为:放置物体后启动转台和Kinect传感器,Kinect传感器以固定的时间间隔采集物体的表面彩色信息和深度信息,同时,计算机中将每一帧的深度信息转化成点云数据,并做预处理;采样时刻 $t_i$ 获得的点云数据将会与前一时刻 $t_{i-1}$ 已经配准的点云进行实时的匹配;实现对相邻帧的点云进行实时配准。当转台旋转一周后,实时处理结束,得到物体和转台的混合点云,然后对物体和转台的混合点云做离线处理。

[0109] 离线处理包括：点云处理、表面重建、模型简化以及纹理映射四个部分。

[0110] 点云处理对实时处理获得的物体和转台的混合点云进行过滤，并将其与未放置建模物体前采集到的转台点云配准，得到带有底面点的全封闭的物体表面点云。

[0111] 表面重建，连接物体表面点云上的点，使物体点云上的点与点之间形成三角形网格，生成物体表面点云的三维网格模型。

[0112] 模型简化，根据仿真系统的具体要求，采用边折叠模型简化方法，按照比例删去三维网格模型上的点和边，建立点数较少、三角形面片数较少的三维模型。

[0113] 纹理映射包括纹理图片生成和纹理贴图，经过映射后创建该物体的mesh文件，仿真系统直接调用mesh文件，进行模型渲染，在OGRE环境中显示、浏览。

[0114] 具体步骤，如图1所示：

[0115] 步骤一、利用Kinect传感器采集转台点云和建模物体点云，并通过实时配准获得物体和转台的混合点云；

[0116] 点云是指，通过测量仪器得到的建模物体外观表面的点数据集合，其中通常记录了空间点集的三维坐标信息、表面法向量、RGB值或者灰度值。

[0117] 步骤101、启动Kinect传感器，采集没有放置建模物体时的转台点云。

[0118] 如图3(a)所示，没有放置建模物体时的转台点云。

[0119] 步骤102、放置建模物体后，启动转台，Kinect传感器采集建模物体的表面信息。

[0120] 计算机驱动转台做匀速360°旋转运动，同时，Kinect传感器定时地采集物体表面信息的彩色信息和深度信息。

[0121] 步骤103、根据Kinect传感器采集到的表面信息生成建模物体点云，从而获得不同时刻所采集到的不同视角下的建模物体点云。

[0122] Kinect传感器采集到的物体表面信息输入到计算机的软件程序中，生成建模物体的点云。该点云记录的是空间点集的三维坐标和三原色光模式RGB值。

[0123] 步骤104、对不同时刻所采集到的不同视角下的建模物体点云进行实时配准，得到物体和转台的混合点云。

[0124] 点云配准，是将在不同时刻获取的不同视角下同一物体的点云统一到同一个三维坐标系下的过程。

[0125] 在步骤103中，Kinect传感器在每个采样时刻只能得到一个视角下的物体表面信息，采集到物体的一部分表面信息，不同视角下的信息只体现出物体的部分特征。当转台旋转一周后，不同视角下的物体表面信息经过点云配准整合成一个完整的模型来表示建模物体的外形特征。

[0126] 利用微软开发的实时场景三维重建项目Kinect Fusion对不同时刻采集的点云进行实时配准。Kinect Fusion Explorer程序利用高性能GPU处理Kinect传感器采集的数据，在转台转动过程中实时自动地配准点云，重建出包含三维建模场景全表面信息的点云，即物体和转台的混合点云，如图3(b)所示。

[0127] 步骤二、对物体和转台的混合点云做点云预处理，得到带有底面点的建模物体的表面点云。

[0128] 物体和转台的混合点云中包含大量噪声点，需过滤噪声和聚类分割。采用RANSAC方法、欧氏距离聚类分割方法、点云FPFH(快速点特征直方图, Fast Point Feature

Histograms) 三维特征估算方法以及点云配准方法等系列处理,过滤物体和转台混合点云中的属于转台旋转平台部分的点云以及其他噪点,填充物体底部点云空洞,得到全封闭的带有底面点的物体表面点云。

[0129] 具体步骤,如图2所示。

[0130] 步骤201、采用RANSAC平面提取方法,提取并删除物体和转台的混合点云中包含的转台的底座点云;

[0131] 物体和转台的混合点云中包括了建模物体的点云,转台底座点云和转台旋转平台部分的点云,采用RANSAC平面提取方法,提取出转台底座点云,并将该部分点云剔除,从而获得物体和转台旋转平台的混合点云,如图4(a)所示。

[0132] 步骤202、采用欧氏距离聚类分割方法,滤除步骤一得到的转台点云中所包含的转台底座点云;

[0133] 步骤一中没有放置建模物体的转台点云中包括了转台底座的点云和转台旋转平台点云。采用欧氏距离聚类分割方法,滤除转台底座点云,效果如图4(b)所示,得到聚类后的转台旋转平台部分的点云。

[0134] 欧氏距离聚类分割方法根据点与点之间的距离,将相近的点集划分为一个类,把点云分割成若干个小的点云,用于点云的过滤。

[0135] 步骤203、对步骤201中物体和转台旋转平台的混合点云以及步骤202中滤除转台底座点云后的转台点云,均采用体素栅格降采样方法处理,分别获得物体和转台混合点云的简化点云,以及转台点云的简化点云。

[0136] 步骤201和步骤202所获得的点云点数十分庞大,后续处理需要消耗很多时间。采用体素栅格降采样方法处理这两个点云,分别获得点数较少但是几何外观特征保留较完整的转台点云和物体和转台的混合点云,形成各自的简化点云。

[0137] 步骤204、计算步骤203所得到的物体和转台混合点云的简化点云的FPFH三维特征,以及转台点云的简化点云的FPFH三维特征,获得物体和转台混合点云的特征点云,以及转台点云的特征点云。

[0138] 步骤205、经过步骤204处理后,将物体和转台的混合点云的特征点云与转台点云的特征点云进行配准,获得配准点云。

[0139] 以FPFH特征作为配准依据,通过比较物体和转台的混合点云的特征点云的FPFH特征与转台点云特征点云的FPFH特征,将两个特征点云统一到同一个坐标系下,得到配准点云,如图5所示。

[0140] 步骤206、使用RANSAC方法分割配准点云,获得转台旋转平台点云和物体点云;

[0141] 物体点云是指没有底部点的建模物体点云。

[0142] 如图6(a)和6(b)所示,经过点云配准后,再次使用RANSAC方法提取平面,将步骤204中两个特征点云中重合的属于转台旋转平台部分的点云提取出来,形成转台旋转平台点云,其余部分形成物体点云;

[0143] 步骤207、用欧氏距离聚类分割方法滤除物体点云中的噪声,得到没有底部的物体点云。

[0144] 如图6(c)所示,对物体点云进行欧氏距离聚类分割处理,滤除噪声,得到没有底部的物体点云。

[0145] 步骤208、从步骤206中的转台旋转平台点云通过基于k-d树的点搜索方法提取构成物体点云底部的点云。

[0146] 从步骤206中提取出来的转台旋转平台点云中,获取部分属于建模物体的底面点,该部分点的密度明显低于其他部分的点,通过基于k-d树的点搜索方法去除点密度较大的点集,剩余的点构成物体点云底部的点云。

[0147] 步骤209、融合步骤208中的物体点云底部的点与步骤207中的没有底部的物体点云,得到带有底面点的建模物体的表面点云。

[0148] 如图7所示,融合步骤208得到的物体点云底部的点云与步骤207中没有底部的物体点云,再过滤噪点,就获得带有底面点的建模物体的三维点云。

[0149] 步骤三、将带有底面点的建模物体的表面点云上的点连接成三角形网格,表面重建后获得物体的三维表面模型;

[0150] 表面重建,将物体点云上的点连接成三角形网格,使物体表面被各边彼此相接的三角形面片覆盖。

[0151] 在获得带有底面点的建模物体的表面点云后,采用Poisson表面重建方法建立三维网格模型。如图8(a)所示,以水杯为建模物体经过Poisson表面重建的结果,同一个三维模型,如图8(b)和图8(c)所示,显示了三角面片的边缘。重建表面后获得的物体三维表面模型体现了物体的几何形状特征。

[0152] 步骤四、对物体的三维表面模型进行模型简化处理,建立出物体的三维简化表面模型。

[0153] 不同的仿真场景对三维模型的精细程度有不同的要求,表面重建得到的三维模型一般由数十万个点和几万个三角面片构成,其加载过程将占用大量的计算资源,影响三维场景的显示效率,本发明采用边折叠简化方法处理,按照比例删去三维网格模型上的点和边,通过去除相应顶点来简化三维模型,建立出点数较少、三角形面片数较少的三维模型。如图9(a)和图9(b)所示,纸盒模型简化前后效果。

[0154] 步骤五、对步骤四中物体的三维简化表面模型进行纹理映射。

[0155] 纹理映射是指,将物体表面的纹理图像覆盖到三维模型表面上,建立起纹理与物体三维模型的一一对应关系。具体过程分为两部分:首先是将带有彩色信息的点投影,得到平面点图,生成彩色纹理点图;然后是对彩色纹理点图进行邻域颜色插值,填充点图上的未着色区域,形成完整的纹理图片。

[0156] 将物体点云上的点投影到该点云外部的一个球体上,再经过等距圆柱投影,展开圆柱就获得带有物体表面点坐标和颜色特征的彩色点图,随后对该点图采用自适应邻域颜色插值方法做颜色插值处理,从彩色点图形成纹理图片。

[0157] 投影的流程如图10所示,首先以一个半径较大的球体将步骤207所得到的没有底部的物体点云包含在内,将点云上的点投影到该球体上,具体为球心C与点云上的每个点 $p_i$ 相连,并向外延伸形成射线 $l_i$ ,射线 $l_i$ 又与球体相交,交点为点 $p_{si}$ ,所有 $p_{si}$ 构成一个位于球面上的点集 $P_s$ 。将点集 $P_s$ 平移到以C为原点、三轴方向与物体点云坐标系一致的新坐标系下,形成直角坐标系下的点集 $P_s'$ 。

[0158] 点集 $P_s'$ 中的点直角坐标形式为 $(x_{si}', y_{si}', z_{si}')$ ,然后将其转换到以C为原点的球坐标系下,坐标形式为 $(r_i, \phi_i, \theta_i)$ ,得到点集 $P_s''$ 。对点集 $P_s''$ 进行等距圆柱投影,得到分布于

圆柱面上的点集 $P_c$ 。最后固定圆柱的一条母线,把圆柱面展开成二维平面,得到带有物体表面点坐标和颜色特征的二维点集 $P_c$ ,再根据 $P_c$ 形成二维纹理图片。

[0159] 投影方法的具体步骤,如图11所示:

[0160] 步骤501、计算输入点云 $X$ 的三维质心坐标,得到投射中心点 $C$ :

$$[0161] \begin{cases} x_c = \sum_{i=1}^n x_i / n \\ y_c = \sum_{i=1}^n y_i / n \\ z_c = \sum_{i=1}^n z_i / n \end{cases}$$

[0162] 其中,点云 $X$ 是步骤207中没有底部的物体点云,投射中心点 $C$ 坐标为:  $(x_c, y_c, z_c)$ ,  $(x_i, y_i, z_i)$  是点云 $X$ 上的点坐标, $n$ 是点云 $X$ 的点数, $n$ 为整数;

[0163] 步骤502、点云 $X$ 投影到以中心点 $C$ 为球心、半径 $R$ 的球体上,得到点集 $P_s$ 。计算公式如下:

$$[0164] \begin{cases} x_{si} = x_c + (x_i - x_c)t_i \\ y_{si} = y_c + (y_i - y_c)t_i \\ z_{si} = z_c + (z_i - z_c)t_i \end{cases}$$

[0165] 其中,  $(x_{si}, y_{si}, z_{si})$  是 $P_s$ 上的点,表示点云 $X$ 上的点  $(x_i, y_i, z_i)$  在球体上的投影点;

[0166] 半径 $R$ 需要满足条件  $R > \text{MAX}\{x_i^2 + y_i^2 + z_i^2\}$ ,  $t_i$  是点云 $X$ 上的点  $(x_i, y_i, z_i)$  在球体上的距离系数,该系数的数值不同,则点  $(x_i, y_i, z_i)$  在球体上投影点  $(x_{si}, y_{si}, z_{si})$  位置不同;

$$[0167] t_i = \frac{R}{\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2}}$$

[0168] 步骤503、把点集 $P_s$ 平移到以 $C$ 为原点的直角坐标系下,得到点集 $P_s'$ ,再转换成球坐标,得到点集 $P_s''$ :

$$[0169] \begin{cases} x_{si}' = x_{si} - x_c \\ y_{si}' = y_{si} - y_c \\ z_{si}' = z_{si} - z_c \end{cases}$$

$$[0170] \begin{cases} r_i = R \\ \phi_i = \arctan\left(\frac{y_{si}'}{x_{si}'}\right) \\ \theta_i = \arccos\left(\frac{z_{si}'}{r_i}\right) \end{cases}$$

[0171] 其中,  $(x_{si}', y_{si}', z_{si}')$  是点集 $P_s'$ 中点 $p_{si}'$ 的直角坐标系坐标,  $(r_i, \phi_i, \theta_i)$  是点 $p_{si}'$ 的球坐标系坐标,  $r_i$ 表示球坐标系原点 $C$ 与点 $p_{si}'$ 之间的距离,  $\theta_i$ 表示有向线段 $Cp_{si}'$ 与 $z$ 轴正

向的夹角,  $\phi_i$ 表示从z轴正方向看自x轴按逆时针方向转到有向线段 $C_{p_{si}}$ 在xCy平面的投影所转过的角度;

[0172] 步骤504、将点集 $P_s$ 进行等距圆柱投影,得到纹理坐标点集 $P_c$ ,公式如下:

$$[0173] \quad \begin{cases} u_i = r_i \phi_i \\ v_i = r_i \left( \frac{\pi}{2} - \theta_i \right) \end{cases}$$

[0174] 其中,  $(u_i, v_i)$ 是纹理坐标点集 $P_c$ 内的点二维坐标系内的坐标,  $u_i$ 为水平方向的像素坐标,  $v_i$ 为垂直方向的像素坐标;

[0175] 点云X中的每个点都对应一个纹理图片上二维坐标系内的坐标  $(u_i, v_i)$ 。

[0176] 步骤505、新建一张黑色图片,将纹理坐标点集 $P_c$ 中点  $(u_i, v_i)$ 位置的RGB值填充为点云X的点  $(x_i, y_i, z_i)$ 的RGB值  $(red_i, green_i, blue_i)$ ,形成彩色纹理点图。

[0177] 如图12所示,上万个彩色点投影到平面上仍然不够密集,几乎无法显示出颜色或者纹理,在某些区域内,未着色的点的邻域有可能没有任何彩色点存在,导致最后形成的纹理点图仍然有很多空缺。如图13所示,以底部有空洞的彩色纸盒为建模对象,经过颜色插值后,由于边缘点密度较低,纹理图片中出现了较多的黑色空缺区域,为了得到完整的物体纹理图片,需要进行邻域颜色插值,填充点图上的未着色区域,形成完整的纹理图片。

[0178] 颜色插值方法,根据未着色点邻域内的彩色点数量和权重,决定该点的RGB值。邻域的形状为正方形、矩形或者圆形。采用自适应的邻域检索,当某个点的邻域内没有彩色点存在时,方法按一定的步长扩大邻域尺寸,继续搜索彩色点,如此迭代检索,直到有彩色点出现或者满足迭代计算结束的条件。

[0179] 自适应邻域颜色插值方法的步骤,如图14所示,具体如下:

[0180] 步骤I、检索彩色纹理点图的每一个像素点,如果当前检索的像素点 $p_i$ 的RGB值是  $(0, 0, 0)$ ,该点为非彩色点,进入步骤II,否则,像素点 $p_i$ 的RGB值不是  $(0, 0, 0)$ ,为彩色点,跳到步骤V。

[0181] 步骤II、在以非彩色点 $p_i$ 为中心、边长为 $l_{rect}$ 像素的正方形邻域中进行遍历,判断遍历结束后邻域内是否存在彩色点,如果至少存在一个彩色点,进行步骤III,否则,该邻域内不存在彩色点,进入步骤IV。

[0182] 如图15所示,当邻域跨越彩色纹理点图的边界时,超出边界的部分从当前边界相对的另一条边界处起继续遍历像素。

[0183] 步骤III、计算邻域中每个彩色点的权重,进入步骤V;

[0184] 公式如下:

$$[0185] \quad \omega_k = \frac{1/d_k}{\sum_{j=1}^m (1/d_j)}$$

[0186]  $\omega_k$ 为当前彩色点 $p_k$ 的权重,  $k$ 为整数;  $d_k = \|p_k - p_i\|$ ,  $d_k$ 为当前彩色点 $p_k$ 到中心点 $p_i$ 的距离;  $d_j$ 是邻域中每个彩色点 $p_j$ 距中心点 $p_i$ 的距离,  $j = 1 \dots k \dots m$ ;  $m$ 是邻域内彩色点的数量,  $m$ 为整数;

[0187] 步骤IV、改变邻域大小为 $l_{rect} = l_{rect} + \Delta l$ ,遍历次数加1,重新进行遍历。如果存在



至少一个彩色点,则进入步骤III;如果遍历次数N达到预设的上限后,没有任何彩色点,则进入步骤VI;

[0188] 设置遍历次数为N,遍历完整个边长为 $l_{\text{rect}}$ 的正方形邻域后,判断该邻域内是否存在彩色点,如果邻域中不存在彩色点,则改变邻域大小为 $l_{\text{rect}}=l_{\text{rect}}+\Delta l$ ,遍历次数N加1,回到步骤II重新进行遍历。如果存在至少一个彩色点,或者N已达到预设的上限 $N_{\text{max}}$ ,则进入步骤V;

[0189] 步骤V、如果 $p_i$ 是步骤I传输过来的点,则直接将 $p_i$ 的RGB值添加到新的点集 $P_t$ 中;如果 $p_i$ 是步骤IV传输过来的点,则利用权重计算点 $p_i$ 的RGB值,并添加到新的点集 $P_t$ ,完成邻域颜色的插值;

[0190] 计算点 $p_i$ 的RGB值,并将这个点的信息添加到新的点集 $P_t$ ,公式如下:

$$[0191] \quad \begin{cases} red_i = \sum_{k=1}^n red_k \omega_k \\ green_i = \sum_{k=1}^n green_k \omega_k \\ blue_i = \sum_{k=1}^n blue_k \omega_k \end{cases}$$

[0192] 点 $p_i$ 的RGB值为 $(red_i, green_i, blue_i)$ ,点 $p_k$ 的RGB值为 $(red_k, green_k, blue_k)$ 。

[0193] 步骤VI、对下一个像素点 $p_{i+1}$ ,执行步骤I到步骤V,直到彩色纹理点图遍历结束。

[0194] 如图16所示,自适应邻域颜色插值方法最后得到的结果是一张完整的建模物体表面彩色纹理图片。

[0195] 步骤六、计算机软件程序对经过纹理映射后的三维模型,进行模型渲染生成二维图像。

[0196] 模型渲染,是通过计算机程序将三维模型生成二维图像的过程。

[0197] 如图17(a)和17(b)所示,采用开源3D图形引擎OGRE(Object-Oriented Graphics Rendering Engine)环境中完成物体的三维简化表面模型的纹理映射,并渲染模型,用户可以自由地浏览、调用实体三维模型。

[0198] OGRE是基于C++开发的面向场景的3D图像引擎。在OGRE环境中,三维模型的显示通过mesh格式来进行。根据实体三维表面模型和纹理信息,创建对应的mesh文件,在OGRE环境中进行浏览。

[0199] 本发明基于图像的建模与渲染技术,提出一套低成本全自动的三维模型自动建模方法,获得带有纹理的实体三维表面模型。

[0200] 实施例:

[0201] 以三类典型物体为例,包括:1、表面没有遮挡、各视角下形状变化很大的长方体纸盒;2、表面没有遮挡、各视角下形状变化很小的圆柱状水杯;3、有大量遮挡的椒草盆栽。如图18所示,这些物体的图片、点云、重建后的三维模型、简化网格模型和纹理映射后在OGRE中的显示效果。从重建效果上看,各类型的物体都能建立出效果良好的完整的三维模型,包括有遮挡的盆栽,其最后的显示效果也和实物比较接近。

[0202] 针对仿真场景不同精细程度的需求,对物体三维模型进行不同程度的简化。如图19(a)和19(b)所示,表明了由精细到粗糙的不同等级的模型,点数、面片数依次减少,纹理

映射时针对的都是同一张纹理图片。可以看到,三维模型即使在点数、面片数上大大减少,贴上纹理之后的显示效果并不比原始的点分布密集的三维模型差,足以满足仿真系统的不同需求。

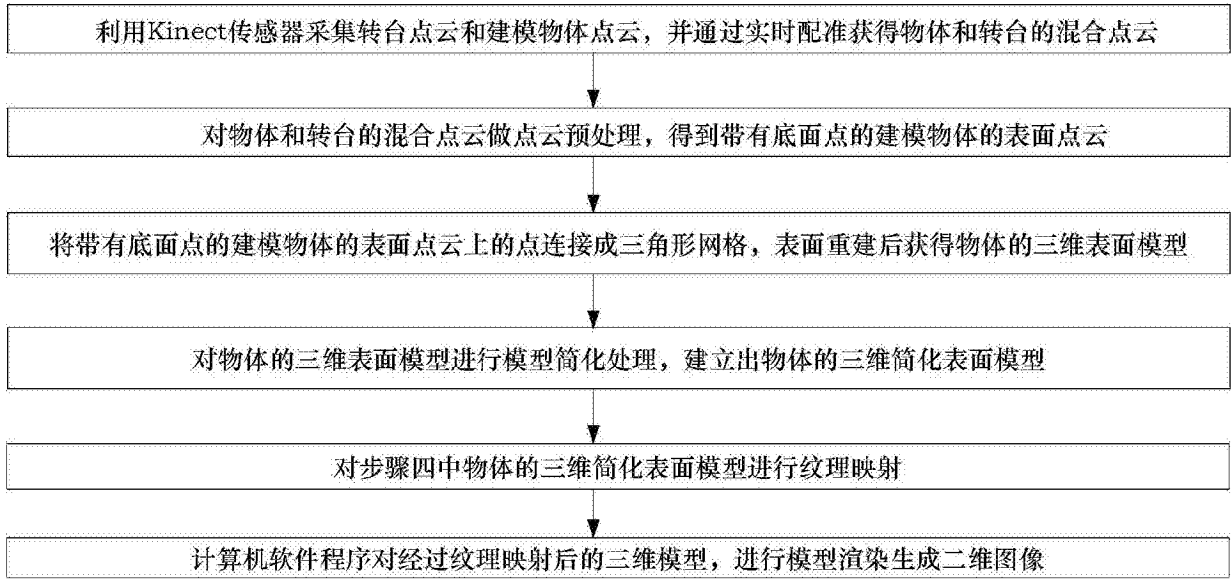


图1

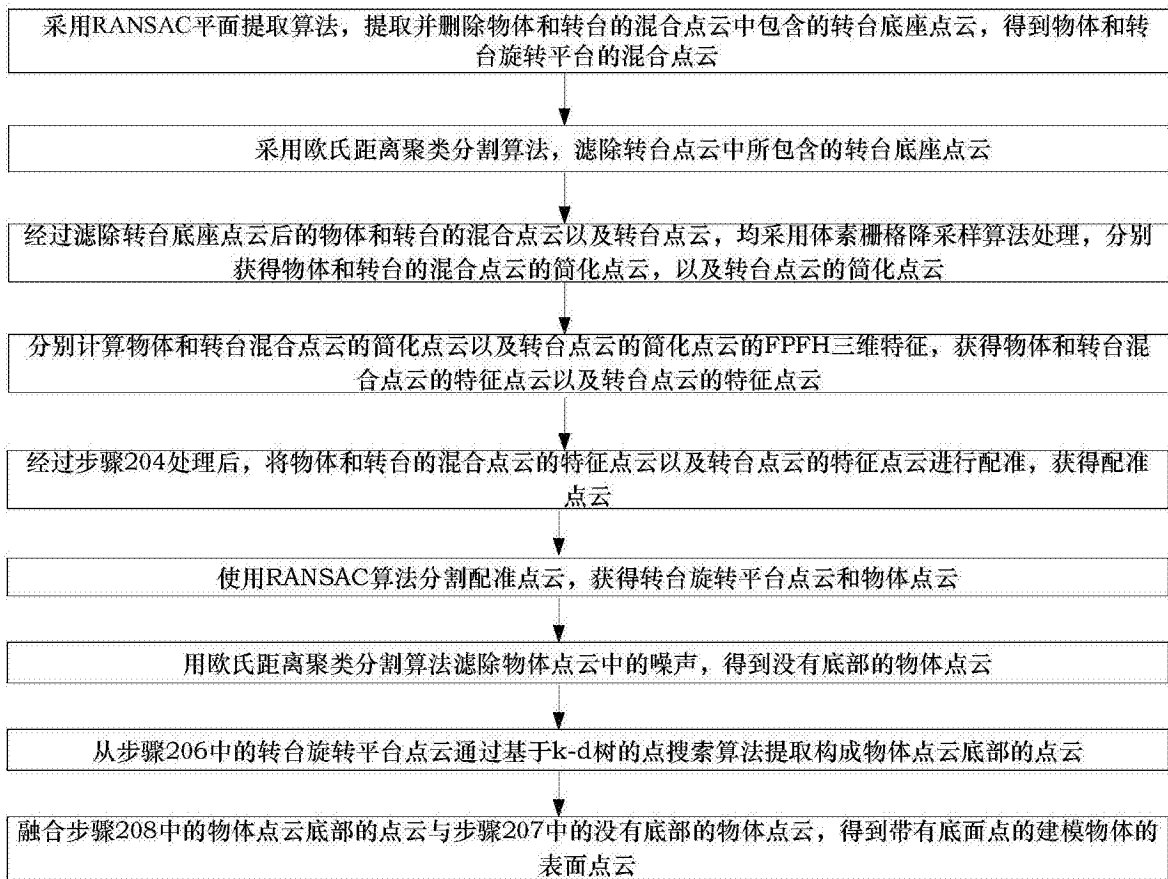


图2



图3 (a)

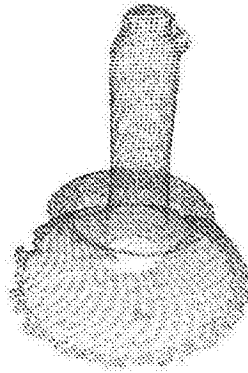


图3 (b)

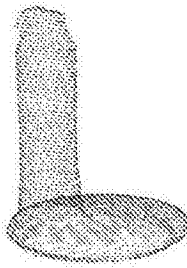


图4 (a)

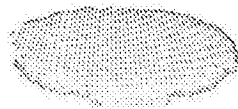


图4 (b)

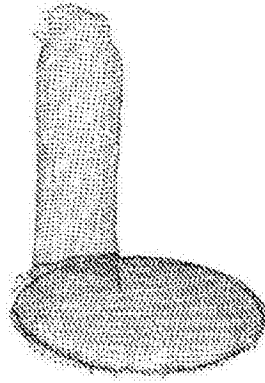


图5

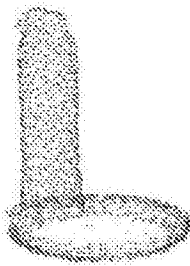


图6 (a)

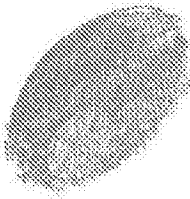


图6 (b)

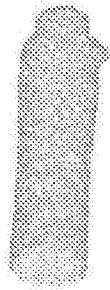


图6 (c)

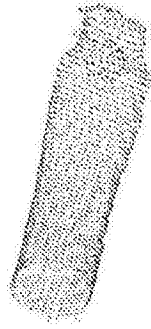


图7

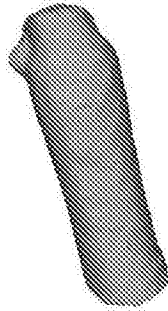


图8 (a)

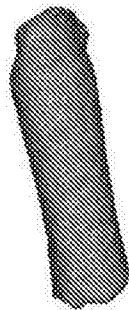


图8 (b)

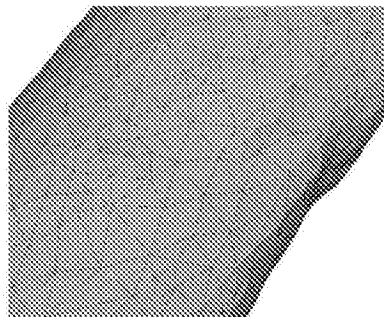


图8 (c)

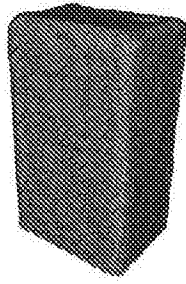


图9 (a)

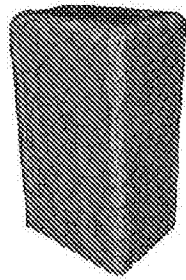


图9 (b)

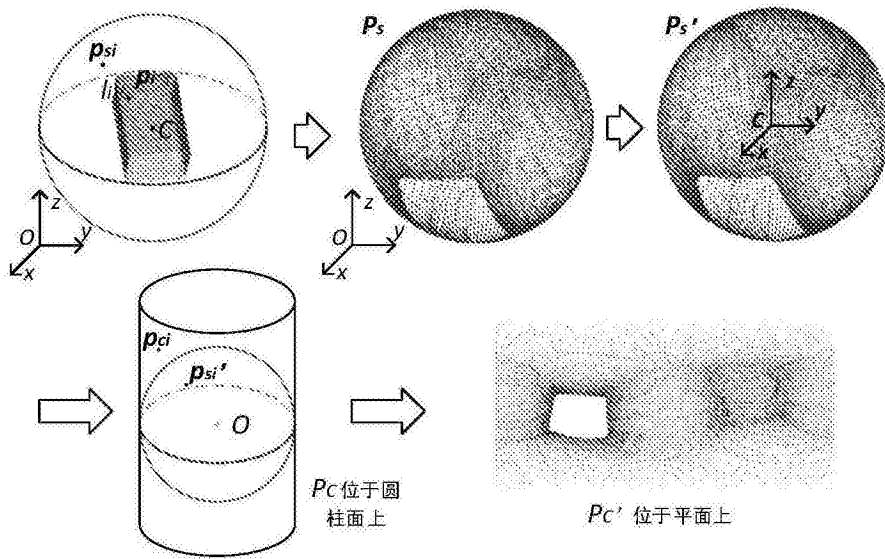


图10

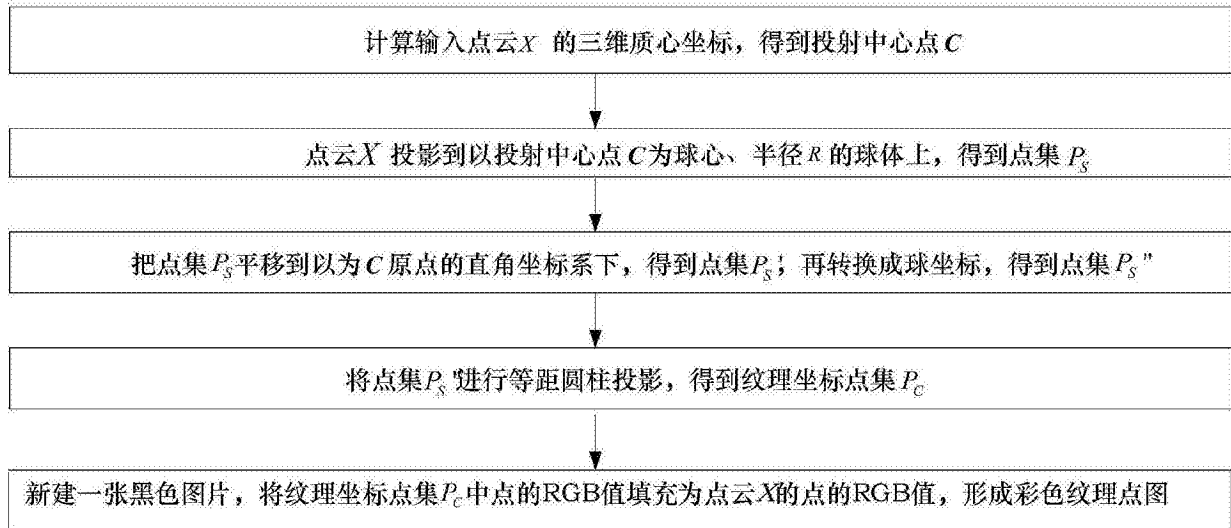


图11

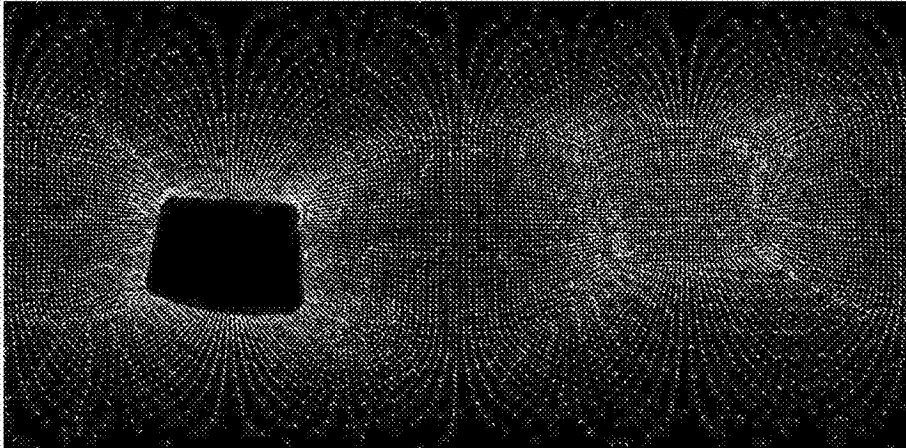


图12



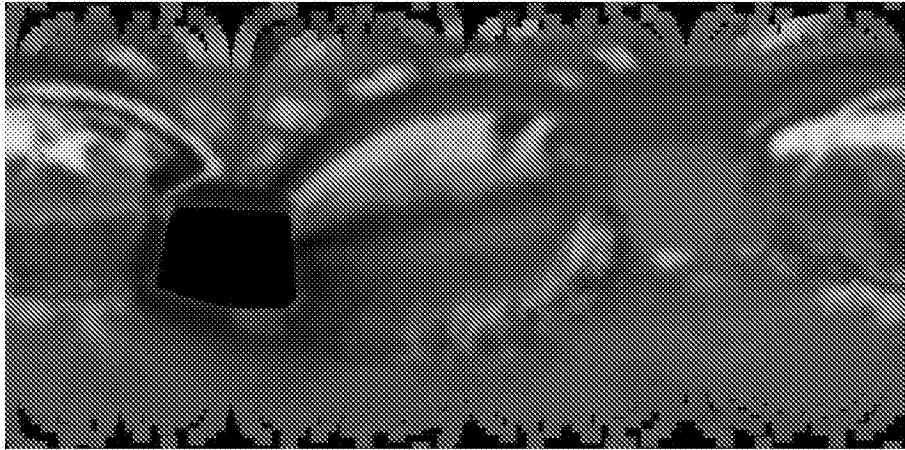


图13

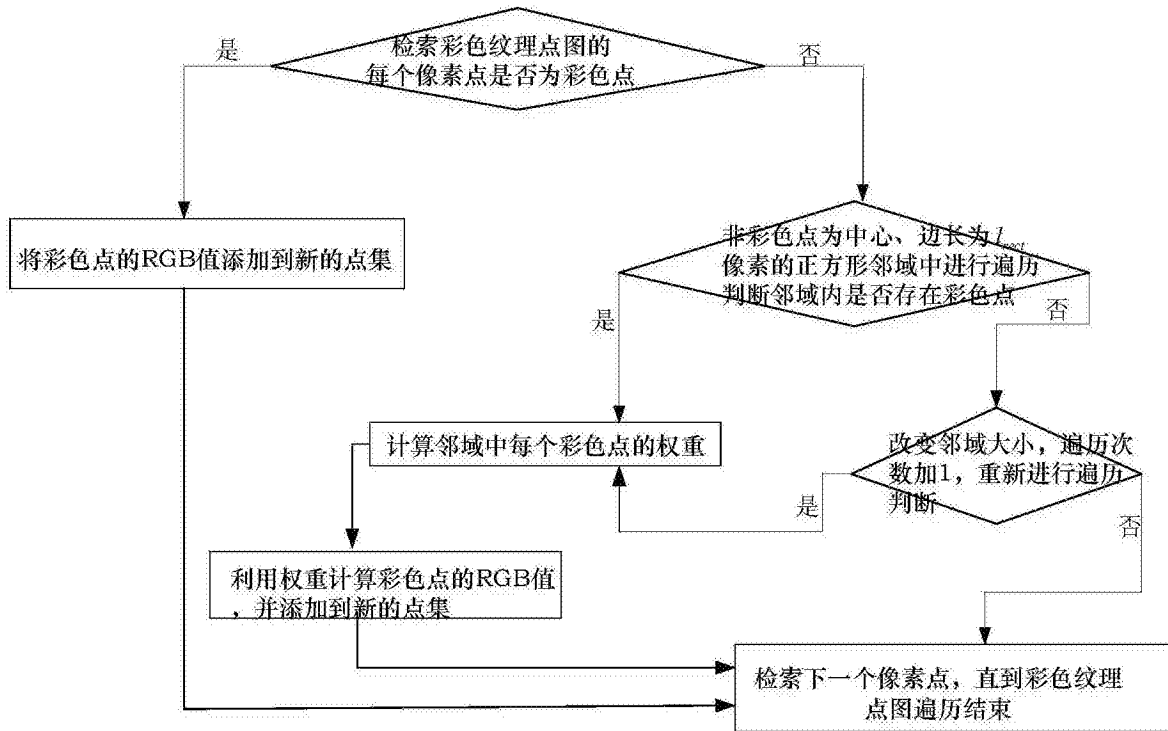


图14

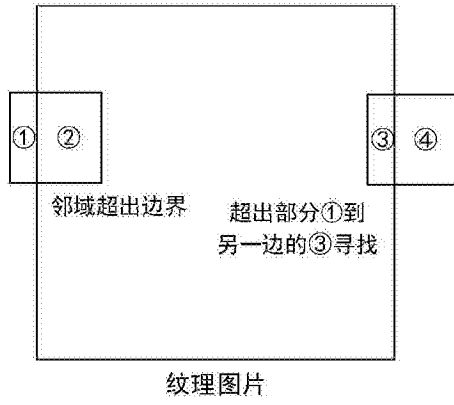


图15

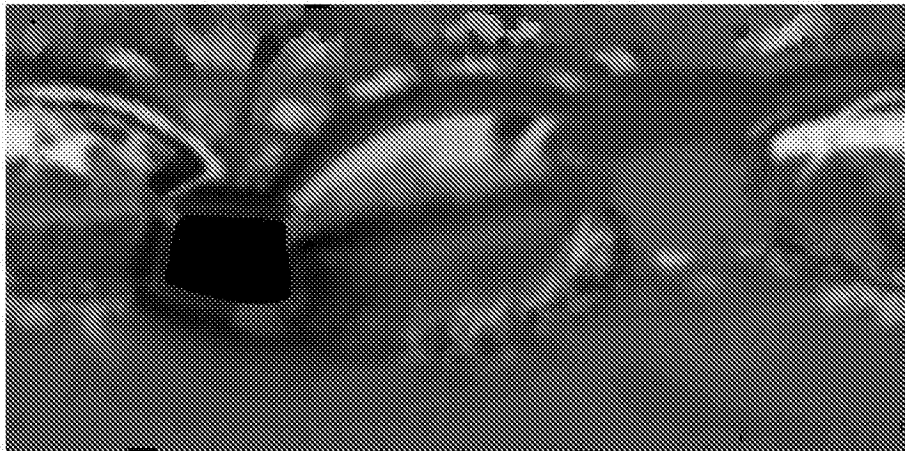


图16

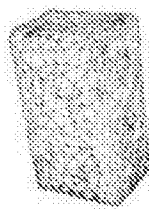


图17 (a)



图17 (b)

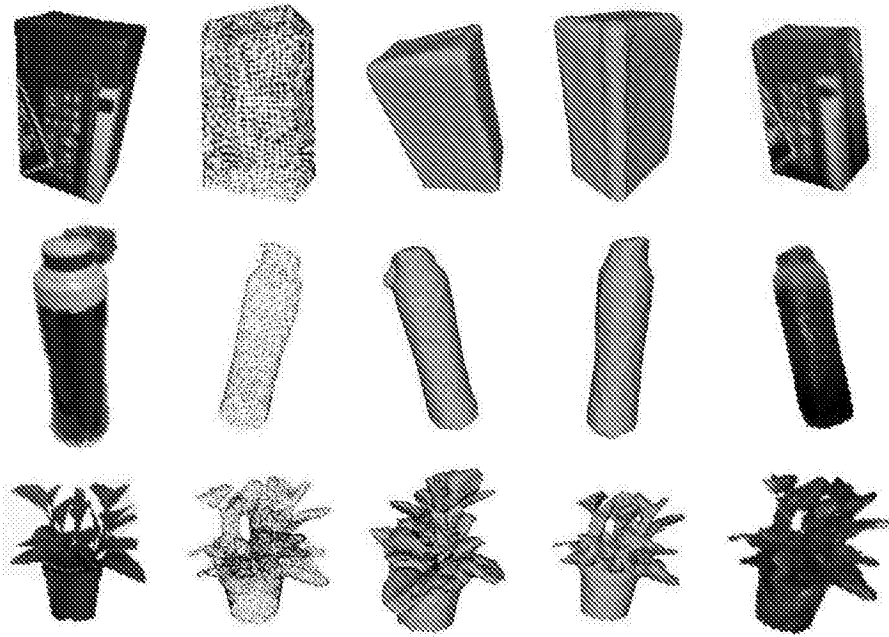


图18

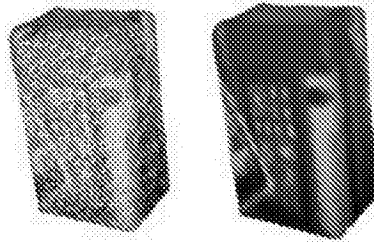


图19 (a)

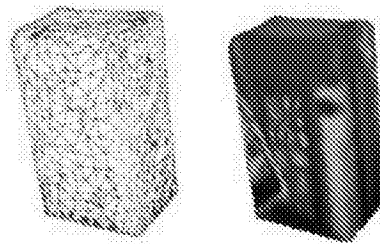


图19 (b)