

硕 士 研 究 生 读 书 报 告



题目 参数形状集合的检索

作者姓名 洪静

作者学号 21651004

指导教师 李启雷

学科专业 软件工程

所在学院 软件学院

提交日期 二○一七年四月

Retrieval on Parametric Shape Collections

A Dissertation Submitted to

Zhejiang University

in partial fulfillment of the requirements for

the degree of

Master of Engineering

Major Subject: Software Engineering

Advisor: Li QiLei

By

Hong Jing

Zhejiang University, P.R. China

2017

摘要

虽然参数形状的集合在规模和使用范围上不断增加，但是在集合中参数形状基于形状匹配和检索的基本问题几乎没有进展。这种集合的搜索空间是离散的（形状数量）和连续的（参数值）。在这项工作中，本文建议使用已显示对单一形状检索有效的描述符来表示该空间。虽然单个形状可以表示为描述符空间中的点，但参数形状可映射到更大的连续区域。对于平滑描述符，假设这些区域是有界的低维流形，其中维度由形状参数的数量给出。本文建议用一组基元表示这些流形，即点和有界切线空间。文中的算法描述了如何定义这些基元，以及如何使用它们来构造允许准确和快速检索的多边形近似。本文根据曲率，边界评估和允许的近似误差进行分析，以在基元类型之间进行选择。

**关键词**： 参数化形状、描述符、流形、基元

Abstract

While collections of parametric shapes are growing in size and use, little progress has been made on the fundamental problem of shape-based matching and retrieval for parametric shapes in a collection. The search space for such collections is both discrete (number of shapes) and continuous(parameter values). In this work, the article propose representing this space using descriptors that have shown to be effective for single shape retrieval. While single shapes can be represented as points in a descriptor space, parametric shapes are mapped into larger continuous regions. For smooth descriptors,we can assume that these regions are bounded low-dimensional manifolds where the dimensionality is given by the number of shape parameters. The article proposes representing these manifolds with a set of primitives, namely, points and bounded tangent spaces. Our algorithm describes how to define these primitives and how to use them to construct a manifold approximation that allows accurate and fast retrieval. The article performs an analysis based on curvature, boundary evaluation, and the allowed approximation error to select between primitive types.

**Keywords：**parametric shapes、descriptors、manifolds、primitives

1引言

在图形和模型的许多应用中的一个根本问题是从大型集合中检索形状。虽然基于形状的匹配和检索已被广泛应用于简单（非参数）形状数据库，但是在有效检索参数形状集合方面几乎没有进展。在此工作中，我们提出了一种通过参数数据库进行搜索的策略将输入查询表示为单个3D形状的模型。

参数化形状，为不同参数设置返回不同形状的通用模型，是图形和建模中的重要工具。实质上，可以将单个参数设计视为表示整个3D形状系列（见图1）。使用参数设计可以节省存储，但更重要的是，它们可以支持用户定制。

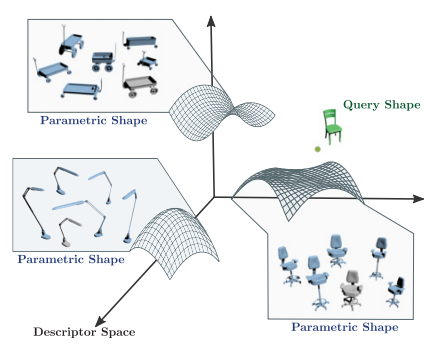


图1

我们提出了一种从参数形状集合中进行匹配和检索的方法。关键思想是在描述符空间中表示完整的参数形状，包括连续变化。参数形状占据较大的“区域”，为了找到给定查询模型（描述符空间中的单点）的最接近的参数形状，需要有效地计算从该点开始的距离到每个形状“区域”并检索最近的一个。我们通过为这些区域创建一个紧凑的表示来解决这个问题，从而在保证存储小和评估快的同时确保精确的距离测量。对于平滑描述符，这些区域是嵌入在高维空间中的有界低维流形。流形的维数由参数的数量给出，边界由可行的参数值集合给出。我们还可以访问定义流形的实际功能，由参数化形状函数的组合和描述符的签名函数给出（见图2）。

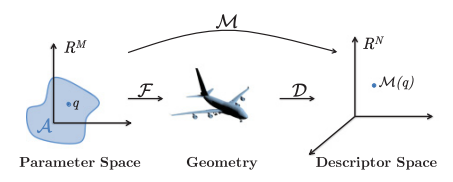


图2

我们提出一种算法，用一组可以有效地用于检索的基元来覆盖每个流形。我们使用两种类型的基元：点和有界切线空间。我们讨论创建这些基元的方法（具体来说，定义切线空间的界限），并在它们之间选择以保证有效的存储和检索。一般的想法是，扁平区域应该被切面覆盖，而更弯曲的区域应该被点覆盖。然而，由于不同的原理具有不同的存储和检索成本，所以最佳覆盖不仅取决于流形的几何形状，还取决于所需的准确度。因此，我们定义了分类应用的近似误差，并提出了一种基于曲率，边界评估和允许近似误差的基元选择方法。我们的理论分析允许我们计算流形，不需要经验参数调整，并提供检索准确性的保证。

除了提出利用描述符空间表示的参数形状的第一个检索算法，我们还提供以下技术贡献：将流形表示为点和切线基元的混合，以及用于最佳选择原始类型以实现有效覆盖的策略；基于目标拟合误差，曲率和到边界的距离来确定切线图元的边界的方法等。

2 相关工作

我们的工作从数据驱动建模，形状检索，形状集合的模板驱动探索，高纬度点云表示以及流形的距离等方面得出。

数据驱动建模与参数形状。数据驱动的建模利用形状或形状部分的集合来构建新设计。然而，这些形状操作需要保留结构和各种可行性约束。通过在参数空间中明确定义一个可行的集合，参数化表示允许大的可变性，同时保证有效性。

3D形状的有效检索已经引起了图形界的关注多年。用于快速检索的最常用的方法之一是使用将几何模型表示为高维特征空间中的点的描述符。

我们的问题也与使用参数化3D模板的分立设计类别的作品有关。在我们的工作中，每个流形被定义通过单个参数形状，而不是一组非参数形状。我们的目标是代表一组这样的歧管，由参数形状的集合定义，并支持所有这些流形的距离查询以允许有效的检索。

由于我们将参数形状表示为描述符空间中的低维流形，因此我们的工作与低维数据在高维中的紧凑表示相关。

我们的方法依赖于使用已知参数化图来估计点到流形的距离，这也是几个研究领域中已经解决的问题。在我们的方法中，我们解决了如何在基元类型之间进行选择以优化资源分配并讨论检索精度的理论和经验边界的问题。

3 参数化形状的表示

我们将参数形状定义为，其中是约束参数值的可行集合，是从参数值到几何（例如网格）的函数。

给定一个查询形状s，我们想计算从s到T的距离。这个距离是由如下公式定义的：



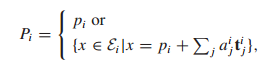
其中两个固定形状之间的距离可以由给定的形状描述符定义。然而，我们将通过在整个参数形状的描述符空间中定义一个表达式而不是找到q的最优值并计算该参数的距离（即首次拟合）。与以前的工作类似，我们使用描述符来表示几何，该描述符采用3D网格并计算签名向量（通常签名向量是高维度的）。该签名向量紧凑地表示作为描述符空间中的高维点的单个几何。然而，这种方法并不明显地适用于参数形状，因为参数形状跨越一大组可能的几何形状，因此占据描述符空间的较大区域。

如图2所示，我们定义，其中是生成给定几何的描述符的签名函数。我们可以将解释为到的参数化，其中形状参数K的数量远小于描述符空间N的维度。我们的方法假定F是光滑的。 这适用于从单一几何的自动转换和大多数CAD模型转换，这些形状通常被设计成使得参数变化平滑变形几何。 因此，对于平滑的描述符，我们可以假设图像位于流形上。 因此，给定一个查询形式，我们可以应用签名函数来计算其在描述符空间中的值，并定义，其中d2是在欧氏距离。

我们的目标是有效地评估从x到一个流形集合的距离，该集合表示了在数据库中的每个参数形状，以便检索最接近的数据库（见图1）。 我们的方法是构建每个流形的紧凑表示。这是允许有一点误差的近似值。我们的目标是找到具有最小存储要求的近似值，并且要求是快速准确的距离评估。

4 算法

近似中的每个基元Pi被定义为点或有界正切空间，其由给定点处的切线空间与椭圆以此为中心。我们可写成如下：

 （1）

其中pi是M上的点，是在pi处形成M的切线空间的方向导数，并且是权重。

为了定义，我们提出一种算法，随机对M上的点y进行采样，如果，则向M随机添加一个基元。通过随机选择点并计算来完成M点上的随机抽样。添加的基元可以是等式(1)中定义的单点或有界切线空间。我们认为，在有限制的情况下，这种采样算法可以确保我们得到一个完整的流形覆盖。实验中，我们在2000次拒绝之后终止采样。这不能提供完全覆盖的技术保证，但却是一个很好的近似。这是因为我们使用的拒收采样方案将保留所有不被近似值覆盖的点。当Pi是一个点时，满足紧密度的条件，但是Pi是切线空间时则不满足。在这种情况下，我们使用紧密引理来定义关于如何确定Pi的椭圆边界的规则，如下面将讨论的。

当我们的拒绝采样方案选择向添加一个基元时，主要决定是确定它是否应该被表示为单点基元或是以点为中心的有界切线空间。这样做是为了最大限度地提高效率。除了中心点之外，有界切线基元还需要存储K个切向量；因此，我们认为其成本是点基元成本的K+1倍。这也大致对应于查询计算时间的增加。因此，如果切线基元的边界足够紧，使得其覆盖的区域小于可以被K+1个点覆盖的区域，则不值得使用该基元。为了做出决定，我们需要定义和测量点和切线基元的覆盖范围。首先，我们将考虑多边形是无界的（即）的情况，然后我们将考虑可行集合A施加的附加边界。

## 4.1 无界流形

如果一个流形没有界限，决定如何由切线空间表示的唯一方面是它偏离平坦度的程度。我们将根据覆盖引理来衡量切线空间在多大程度上局部近似的程度。然后，我们将讨论如何基于紧密引理定义边界超椭圆Ei。

*覆盖范围*。首先，让我们考虑一维情况，其中是中的曲线，并假定采样点为而不失一般性。在这种情况下，切线近似由下行给出：



由于我们允许大小为的误差，所以一旦采样了一个点，则以该点为中心的半径为的圆上的任何点都可以根据覆盖率引理被采样点很好地表示。另一方面，如果我们在该点上采用切线，则曲线上任何位于该线的距离内的点都被线表示覆盖。因此，当点的覆盖率与成比例时，切线的覆盖率与d成比例，其中d是曲线上从点p到最远点的距离，其足够接近于切线（见图3）。

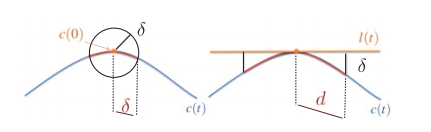


图3

我们可以使用泰勒展开来近似。如果，那么从一个点到线的距离通过如下式子给出：



我们可以使这个距离小于通过限定t如下：



然后，距离d可以通过得到，如下

 （2）

我们观察表达式中这个的平方根内分母正是曲线在时的曲率K的定义。从这里我们可以写成。

覆盖与切线相同的区域所需的点数的下限由两个覆盖率给出。因此，如果这个比例大于额外的K + 1存储要求，我们应该存储切线基元。也就是说，我们应该存储切线基元如果

 （3）

在我们的算法中，我们测量点处的曲率，如果曲率小，那么我们存储有界的切线基元；如果它太大，我们存储点基元。上述方程定义了我们如何基于原始参数和参数形状的维数来确定该阈值，因此我们不需要额外的经验参数估计。

这种曲率解释可以很容易地扩展到：。在多维情况下，我们使用主曲率的最大值，测量在某个方向的曲率的最大值。 由于覆盖率现在是由给出，我们可以得到：

 （4）

在我们的实现中，我们由K导数方向上的最大曲率来近似最大主体曲率。

使用表达式计算每个方向上的曲率，对于等式（2）中的k，用流形的偏导数去替换曲线的导数。

*紧密度*。为了限制切线空间，我们必须确保满足紧密引理。正如我们在上述介绍中所做的那样，我们将首先查看一维的案例，并使用泰勒近似。然后，从点到曲线c的距离可以通过从点到点的距离来界定，（见图4（a））:



为了确保小于，我们限定t, 如下



从中我们可以得到切线空间应该由一个半径圆来限定：



另外，在多维情况下，我们使用超球，并在最大主曲率的方向上取第一和第二导数。

## 4.2 有界流形

接下来，我们讨论如何将可行集合纳入我们的表示。因为可行的集合在描述符空间中的歧管上引入边界，所以需要将这个效应结合到中以保证紧密性。这反过来影响切线基元的覆盖，并且在选择要存储的原始图形时也应该考虑它们。

*紧密约束*。再次从一维情况开始。如前所述，流形的曲率定义了一个绑定的到切线基元，如图4(a)所示。将边界约束定义为最大半径, 这保证了有界切线的投影落在点上，使得（见图4（b））。

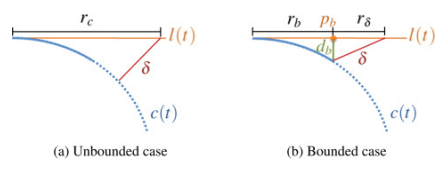


图4

如果点接近边界，那么可能大于。为了保证在这种情况下的紧密性，切线必须由限定，其中是可以扩展曲线的量，以保证从它到有界流形的距离小于允许的近似误差。

在多维情况下，考虑在描述符空间中的方向来计算一个样本点到边界的距离。 假设有一组表示边界约束的分析表达式参数空间，然后使用采样点q处的雅各布将其映射到描述符空间。如果参数空间中的边界约束被写为函数，则其在描述符空间中将变为。可以通过求解下行表达式找到方向到边界的距离：

 （5）

如果沿着方向的射线与边界约束相交，则该最小化的值将为零，并且所得到的将返回从到该边界约束的距离。通过对每个约束计算这个距离，并取最小值作为沿着方向到边界的距离。

为了计算，首先需要评估点到流形的距离。 然后，如图4(b)所示，可以计算，使得。以类似的方式使用二阶泰勒近似去计算。

计算出的边界距离取决于方向。在多个方向拍摄射线取最小半径将确定一个边界超球体。 但是，这非常有限制，因为一个点可以接近一个方向的边界，而不是在其他方向。 所以选择使用椭圆体代替超球体来绑定空间。

很显然，椭球覆盖的区域取决于其方向。为椭球选择最佳取向可以减少代表流形所需的基元数目（见图5）。为了确定椭圆方向的良好基础，目标是将其与流形的最小约束方向对齐。我们选择在切线空间上的一组方向。首先，计算从他们各自到边界的距离（使用前面描述的方法）。 其次，采取与边界最小距离的方向，并将其设置为基向量。然后，将搜索限制在当前基础的正交空间上，并重复第一步。当得到一个完整的基础后，算法将结束。

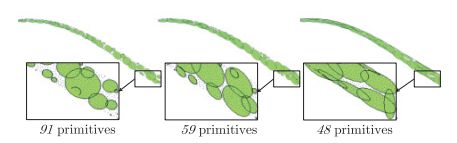


图5

*覆盖范围*。为了在点和切线基元中选择，首先验证等式（4），然后考虑到边界施加的约束来比较覆盖。 对于每个方向，我们将覆盖半径设置为。 然后，按照等式（3），选择添加有界切线，而不是一个点如果满足。

5 检索方法

我们的检索方法通过找到查询形状的最接近的基元来确定最接近的参数形状。使用标准欧几里得范数测量距离点，并且通过首先将查询点x投影到切线空间上，然后计算距离。使用将椭圆体映射到以原点为中心的单位超球体的缩放函数S近似该距离。如果，则距离由投影误差给出。 否则，我们将从p到椭球的距离近似为。 因此最终误差由给出（见图6）。

当计算从查询点到中的一个原始点的距离为时，计算到切线空间原语的距离涉及额外的计算，用于评估在切线空间p上的投影及其与椭球体的距离。我们可以为每个切线基元预先计算N×K投影矩阵，并将其作为数据结构的一部分进行存储。这样，计算查询点到切线空间的投影是。使用前面讨论的进行简化，的计算是。所以，当到一个点的原始距离为时，到切线的原始距离是。由此，得出结论，与存储要求类似，切线基元的额外检索时间与参数的点数成正比。

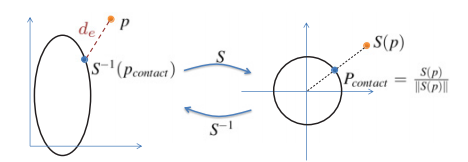


图6

虽然我们的方法找到最接近的参数模型查询，但是找到最接近的匹配仍然涉及拟合参数的最后一步。由于我们找到最接近的基元，我们可以使用该基元的参数作为初始猜测并使用现有的搜索方法进行细化。这个问题在以前的工作中用迭代最近点（ICP）方法解决了。

6 小结

本文主要介绍了一种通过参数数据库进行搜索的策略将输入查询表示为单个3D形状的模型。虽然基于形状的匹配和检索已被广泛应用于简单（非参数）形状数据库，但是在有效检索参数形状集合方面几乎没有进展。为了解决这个问题，本文展开了一系列的工作。主要提到了一种从参数形状集合中进行匹配和检索的方法，以及用一组可以有效地用于检索的基元来覆盖每个流形的一种算法，最后介绍了一种通过找到查询形状的最接近的基元来确定最接近的参数形状的检索方法。

参考文献

[1] [Adriana Schulz](http://dl.acm.org/author_page.cfm?id=99658628097&CFID=928263538&CFTOKEN=66262185), [Ariel Shamir](http://dl.acm.org/author_page.cfm?id=81100081895&CFID=928263538&CFTOKEN=66262185), [Ilya Baran](http://dl.acm.org/author_page.cfm?id=81100129752&CFID=928263538&CFTOKEN=66262185),et al.2017. Retrieval on Parametric Shape Collections. ACM Transactions on Graphics (TOG): Volume 36 Issue 1( February 2017).

[2] Melinos Averkiou, Vladimir Kim, Youyi Zheng,et al.2014.ShapeSynth: Parameterizing model collections for coupled shape exploration and synthesis. Computer Graphics Forum (Special Issue of Eurographics 2014) (2014), 10.

[3] SHREC. 2014. 3D Shape Retrieval Contest at EUROGRAPHICS. Retrieved June 2, 2015 from http://3dor2014.ensea.fr/SHREC2014.html.