# 对偶四元数朱利亚分形

Ben Kenwright

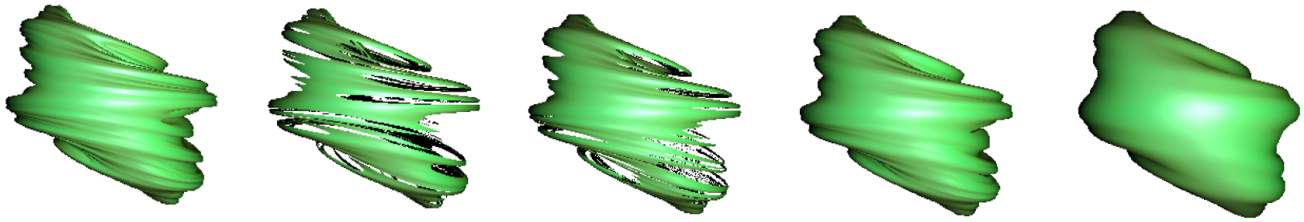


**图 1：** 使用双四元数对二次分形早期原型朱利亚集进行的探索性参数分析。

arXiv:2303.14827v1 [cs.CG] 26 Mar 2023


## 摘要

分形能够生成引人入胜、各具特色的几何形状（例如，分形几何为模拟自然界中的无限细节提供了基础）。分形是具有多种性质（如吸引性和对称性）的非欧几里得数学对象，同时也能够在嵌入式环境中缩放、旋转、倾斜和复制。因此，自分形最初被发现以来，许多不同类型的分形逐渐受到关注。生成分形几何的一种尤为流行的方法是使用朱利亚集。朱利亚集提供了一种直接而新颖的方法，通过迭代计算建模算法生成分形几何。本文提出一种将朱利亚集与双四元数代数相结合的方法。双四元数这一数学概念引人入胜，蕴含着丰富的数学可能性。将分形朱利亚集扩展到涵盖双四元数代数，为我们提供了一种新颖的可视化方案。我们阐述了如何将双四元数与朱利亚集结合来构造分形。我们的原型实现表明，基于简单的光线追踪算法，使用双四元数朱利亚集渲染分形几何十分高效。我们展示了多个不同的实验性等值面示例，以证明该方法的可行性。


**关键词：** 图形学、对偶四元数、可视化、分形、几何、图形处理单元

## 1 引言

**复数与对偶数的几何性质** 复数具有非常优雅的几何解释。具体而言，我们可以将复数视为复平面上的向量。加法和减法遵循常规的向量算术规则，而复数乘法则可理解为：分别按另一个向量的模和辐角，对一个向量进行缩放和旋转。对偶数是形如 $a + b\varepsilon$ 的数，其中 $a$，$b \in \mathbb{R}$ 且$\varepsilon$ 是一个具有如下性质的非实数 $\varepsilon^2 = 0$。对偶数在某种程度上类似于复数 $a + bi$，其中 $i^2 = -1$。可以将对偶数想象为位于对偶数平面上以“a”为中心的正交直线上。将对偶数视为向量时，其模计算为 $||a + b\varepsilon|| = abs(a)$，这是因为其具有幂零性 [Wang 和 Yu 2010]。

**双四元数** 对偶四元数自1882 [Clifford 1882]以来便已存在，但与单独使用四元数相比，受到的关注较少。与四元数类似，对偶四元数也一直带有某种禁忌，人们会避开四元数，也因此避开对偶四元数 [Kenwright 2012]。近年来，机器人学界开始采用对偶四元数，而计算机图形学界却尚未全心接纳它们。近期的一些研究已取得进展，并证明了对偶四元数在计算机图形学中的实用性，示例包括：Kavan 等人 [Kavan 等人 2008]展示了对偶四元数在角色蒙皮和混合方面的优势。Frey 和 Herzeg [Frey 和 Herzeg2011] 扩展了 Kavan 等人 [Kavan 等人 2008] 关于对偶四元数的工作，并将 q切线作为一种替代方法，用于表示刚性变换，而非使用矩阵。这些研究表明，当每个顶点的推断数量足够大时，累积关节变换能够带来更快的结果。Selig [Selig 2010] 通过探讨如何实时求解运动方程，提出对偶四元数能以简洁明了的方式表示刚体变换，从而解决电脑游戏中的关键限制。Vasilakis 和 Fudos [Vasilakis 和 Fu-dos 2009]讨论了用于角色动画的基于骨架的刚性蒙皮。Kuang 等人 [Kuang 等人 2011] 提出了一种实现着衣人体运动实时动画的策略。

**代数与分形** 函数的朱利亚集 $f$ 是所有满足以下条件的点的集合：$z$ 在 $C$ 中，使得 $f$ 在 $z$处呈现敏感依赖性。本文使用一种常见的朱利亚集函数，其形式为 $f(z) = z^2 + c$ 其中 $c$ 为某个常数。在二维复数域中，对于给定的复数 $z_{n+1} = z_n^2 + c$，可使用迭代算法 $c$生成朱利亚集。朱利亚集包含系统收敛时所对应的初始复数值 $z_0$ 。已有许多文献研究朱利亚集的计算与图形方面[Ke 和 Panduranga，1990年；Crane 2005]，其中包括用于生成分形几何的三维和四维情形。尤其是，在计算机图形学中，使用四元数对分形几何进行建模已得到充分研究 [Pickover 1998]。类似地，文献也从计算和图形角度研究了八元数朱利亚集。对偶四元数遵循一套规则明确的运算法则（用于执行乘法、加法和求模等常见运算——见附录 A和图2）。在为构造四元数朱利亚集而发展的概念基础上，我们可以提出类似的方法来构造双四元数 Julia 集。

**贡献** 本文的主要贡献包括：(1) 构建双四元数朱利亚集的方法；(2) 图形示例与实验。

## 2 相关工作

**从四元数到双四元数** Clifford [Clifford 1882]于1873年发表了关于对偶数的精彩研究，为我们提供了一种强大的工具，便于分析复杂系统（例如机械系统、几何系统） [Kenwright 2012]。对偶四元数是对偶数理论的扩展，其中对偶数方程中的数由四元数表示。

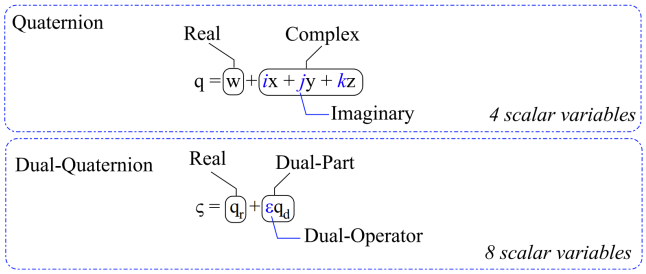



**图2：** 对偶四元数 - 四元数与对偶四元数组分的视觉概览。为避免混淆，并使读者能够轻松区分四元数和对偶四元数，我们使用两种易于辨认的符号来标识它们。

由此得到的双四元数代数十分简洁优雅，为我们提供了一种代数形式紧凑、高效的系统，用于解决原本复杂的问题[Kenwright 2012]。对偶数与虚数十分相似，但有一个重要区别。对于虚数，$i^2 = -1$；而对于对偶数，$\varepsilon^2 = 0$（且 $\varepsilon$ 不为0）。这看似微小的差异，却开启了一个充满数学可能性的精彩世界。

**分形几何** 生成四维版本复数的数称为四元数。Alan Norton [Norton 1982]首次展示了四元数朱利亚集的应用，他通过显示四维空间的三维“切片”来呈现它。四元数朱利亚集实际上是从四维到三维的投影，类似于二维正方形是三维立方体的二维投影呈现。尽管复杂度有所增加，它仍具有平滑的整体外观（也就是说，与二维版本相比，放大时呈现的分形细节较少）。正如本文所示，更高维度的数学可用于创建三维朱利亚集分形。Daniel White 的“曼德布尔球” [White 2007] 采用了不同的方法。他利用“复平面”的几何性质——在复平面上，乘法表现为旋转，加法表现为沿特定方向移动平面——并将其应用于三维空间。这一概念能够生成引人注目的视觉效果，展现一个看似简单的过程如何产生高度精细复杂的集合。此前已有许多研究探讨分形和朱利亚集的性质，以及如何将其应用于计算机图形学 [Crane 2005]。与现有成果类似，我们对插值朱利亚集进行光线追踪，以产生类似明暗反射的效果，并将所提出的方法应用于复空间的立体投影，从而获得一种不平整且带有明暗效果的表面。

## 3 方法

**朱利亚集分形** Gaston Julia 是法国数学家，他在 1918 [Mandelbrot 1983]发现了令人振奋的朱利亚集。朱利亚集是通过公式迭代形成的一种常见分形。朱利亚集是所有点的并集 $z$ 这些点在经过迭代的复函数中 $f(z)$ 当函数的迭代次数趋于无穷时，这些点不会发散（这些点构成一个“填充”的体积区域）。朱利亚集以填充集为基础形成边界。涵盖朱利亚集的函数种类繁多，但本文采用方程 1中给出的常见二次表达式

$$f(z) = z_{n+1} = z_n^2 + c \tag{1}$$

其中，$z$ 表示形式为 $a + bi$ 的变量（$a$ 和 $b$ 是实数，$i$ 是虚数）。$c$ 的值也是一个复数，并且对于每个不同的 Julia 都是常数。因此，每当

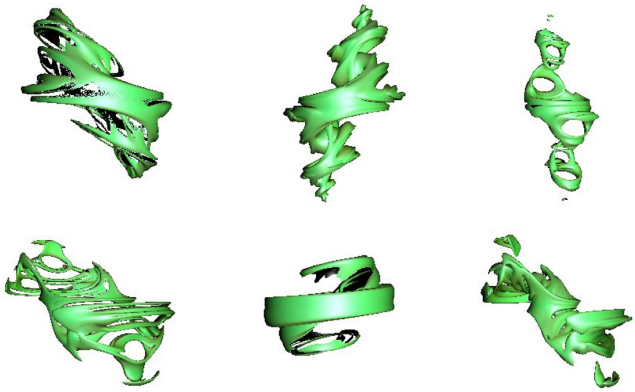


**图 3：** 实验 - 通过随机参数探索视觉可能性。参数常数值按从左上到右下的顺序排列：*c=(-0.10,0.8,-0.26,0.15)(-023,-0.38,- 0.86,0.64); c=(-0.67,-0.54,-0.07,0.02)(0.06,-0.53,0.15,-0.27); c=(-0.98,0.27,0.40,0.20)(-0.37,-0.41,-0.24,-0.34); c=(0.35,0.78,0.85,-0.57)(-0.22,0.06,-0.46,0.05); c=(-0.17,-0.28,0.11,0.8)(0.06,0.44,-0.66,0.06);c=(-0.04,0.95,0.4,-0.43)(0.09,-0.45,-0.27,-0.31)*。

$c$ 的取值不同时，就会得到不同的朱利亚集，因此存在无限多个集合。在简单的情况下，$a + bi$ 表示一个二维复平面，其中迭代序列中的每个起始点 $z_0$ 都对应一个像素位置。对于每个起始值，可能出现两种情况：（1）随着 $n$ 增大，$f(z)$ 趋向无穷大；或（2）该值始终保持在有界范围内。不在有界范围内的点称为逃逸集中的点，而其他所有点则称为囚禁点（囚禁集）。逃逸集与囚禁集的共同边界称为朱利亚集（针对特定的 $c$ 定义）。

**四元数朱利亚集** 四元数由两个基本部分表示：标量实部（$w$）和虚向量部（$x$、$y$、$z$）。我们可以扩展公式 1 中给出的递归 Julia 集算法，使用公式 3 中的四元数。不过，我们需要用到公式 2给出的四元数平方。四元数的平方取自先前的四元数几何研究 [Hart 等人，1989 年；Hart 等人 1990] 在研究 Julia 集时采用了这一方法（利用了四元数的三角性质）。

$$\begin{aligned} q &= (\vec{v}, w) = (x, y, z, w) \\ q^2 &= (x^2 - y^2 - z^2 - w^2,\ 2xy,\ 2xz,\ 2xw) \end{aligned} \tag{2}$$

正向 Julia 迭代由公式 3给出：

$$q_{n+1} = q_n^2 + c \tag{3}$$

**双四元数朱利亚集** 双四元数 Julia 集与普通 Julia 集相同，只是使用双四元数而非复数。生成分形时，会对函数 $\zeta_{n+1} = f(\zeta)$ 进行迭代；如果当 $n$ 趋于无穷时，该级数有界，那么双四元数 $\zeta_0$ 就属于 Julia 集。

$$\begin{aligned} \zeta &= q_a + q_b\varepsilon \\ \zeta^2 &= q_a^2 + q_b^2\varepsilon \end{aligned} \tag{4}$$

其中 $a$ 和 $b$ 是双四元数参数（即分别由两个四元数表示的实数和对偶数）。我们应用方程2计算每个分量的“平方”，并加上对偶四元数常数，从而得到由以下公式给出的每次前向迭代

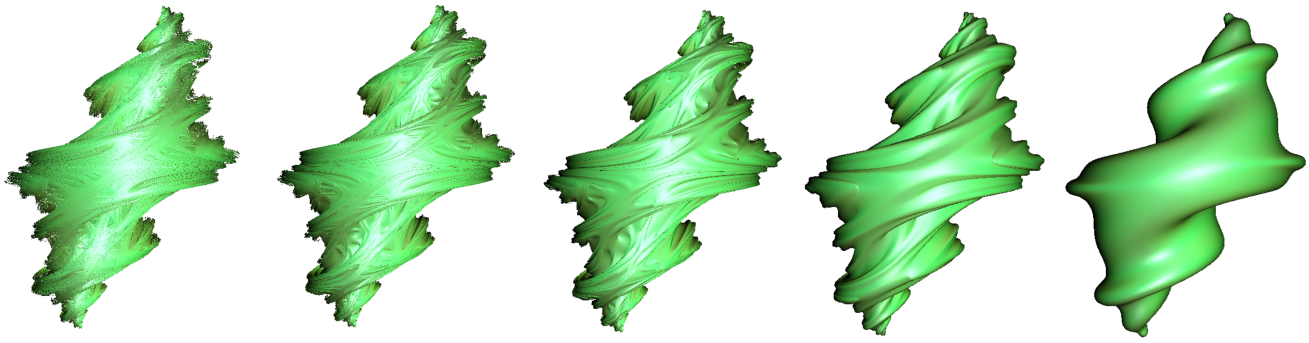


**图 4：** 细节层次 - 增加细节与计算成本之间的权衡 - 用于判断一个点位于朱利亚集内部还是外部的采样点数量和最大迭代次数 *(n)*。

朱利亚方程 5。

$$\zeta_{n+1} = \zeta_n^2 + c \tag{5}$$

其中 $\zeta$ 和 $c$ 是双四元数； $n$ 是迭代次数，通常介于 6 和 15（图 1展示了改变迭代次数对细节层次的影响）。

**实际考量** 不可能将每个采样点迭代至无穷大，因此需要定义合适的最大迭代范围。如果级数在此范围内尚未达到逃逸极限，则认为该点位于集合内部。实践中，较少的迭代次数也能获得相当不错的结果 [Hart 等人 1989]（例如，少于 10 次）。不过，迭代次数越多，分形表面就越准确。如图 4所示，增加迭代次数并提高分辨率可以为几何模型增添更多细节，但也会增加计算成本。

**降维** 双四元数 Julia 集位于 8 维空间中。如何在三维空间中可视化双四元数朱利亚集？我们需要将 8 个维度映射到三维空间中。有两种方法：（1）渲染双四元数空间的单个三维切片，或（2）将双四元数空间投影到三维空间中，以观察朱利亚集的三维投影。本论文采用的方法是用一个平面与双四元数空间相交；本质上是将五个双四元数组分（维度）设为常数。

**光线步进** 降低对三维空间中每个点进行测试的计算成本。在合理的时间内生成具有可接受粒度的模型速度很慢。例如，请参见图中所示的基于体素的渲染对比 6。为了提供快速高效的可视化方案，我们采用了光线步进算法。该算法能够以不同细节层级生成高质量的双四元数朱利亚集渲染图。我们采用朱利亚集的距离估计方法 [Dang 等人 2002]。该距离估计器可利用无界体积加速光线追踪过程。对于任意一个位于分形外部的给定点，该估计值给出了该点到分形距离的下界。虽然公式 6 提供了初始近似值，但我们发现 Hart 等人 [Hart 等人1989] 在公式 7 中给出的方法能产生更美观的结果（且所需的调整更少）。

$$d \geq \alpha \left( \frac{|\zeta_n|}{|\zeta_n'|} \right) \tag{6}$$
$$|\zeta_n'| = 2|\zeta_{n+1}||\zeta_{n+1}'|$$

其中 $\alpha$ 与网格尺寸类似，通常取值较小（例如 $\leq 0.1$）。一旦确定距离下界，我们就可以沿着光线步进，直到击中分形，或者步进多次仍未击中分形。另一种优化方法是先让光线与包围分形的球体相交。这样可以加快光线追踪算法的速度，迅速排除那些不会与分形相交的光线。

$$d \geq 0.5 \frac{|\zeta_n|}{|\zeta_n'|} log(|\zeta_n|) \tag{7}$$

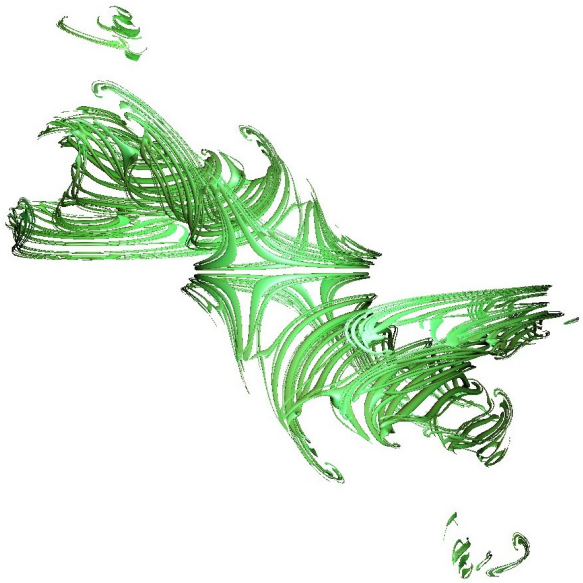


**图5：** 高细节 - 通过使用更多的迭代次数，突出双四元数朱利亚集几何中的细节。*c =(-0.04,0.95,0.4,-0.43)(0.09,-0.35,-0.27,-0.31)，n=15*。

**光照**着色是创建逼真且美观图像时极其重要的过程。由于我们正在渲染三维模型，光照对于通过视觉辨别几何细节的差异至关重要。我们采用 Phong 模型，因为它简单直接，易于融入光线追踪算法。然而，Phong 模型需要表面法线。我们通过对周围点进行采样（即上方、下方、左侧和右侧）来计算近似的表面法线。当找到朱利亚集中的一个点时，还会检测其四个相邻点，以确定它们位于集合内部还是外部。这些点是必需的

用于生成相邻表面，从而计算表面法线（$\vec{n}$）。我们通过对若干点进行迭代，并使用一个较小的 delta，来计算每个像素处的发散度。虽然朱利亚集表面上任意一点的斜率都未定义，但进一步考察该点周围的环境，例如以更高的迭代次数和 z 分辨率进行扫描，就会发现不同的环境。无法计算精确的法线，不过，对周围点进行采样得到的近似值，在我们的实验中可以提供可接受的表面法线。单个光源的 Phong 光照计算如下式所示：8。

$$I = k_a I_a + k_d I_d(\vec{n} \cdot \vec{l}) + k_s(\vec{r} \cdot \vec{v})^s \tag{8}$$

其中，$I$ 是总光照，$k$ 是物体材质（环境光、镜面反射和漫反射）的衰减系数，$l$ 是指向光源的向量，$\vec{r}$ 是反射光向量（即 $R = 2(\vec{n}(\vec{n} \cdot \vec{l})) - \vec{l}$），$\vec{n}$ 是物体表面处的法向量，$\vec{v}$ 是从视点指向该处的向量，$s$ 是镜面反射指数，$I_a$，$I_s$，$I_d$ 是光强。

**图形处理器（GPU）**分形渲染的计算成本很高，但很容易并行化。例如，Keenan [Crane 2005] 展示了使用图形处理器通过光线追踪算法（片段着色器）生成四元数朱利亚集。我们采用类似的方法，在图形处理器上使用光线步进算法，渲染对偶四元数朱利亚集的几何表面。

## 4 实验结果

**参数调优/实验** 图1和图7 展示了一些通过调优/更改参数获得的初步实验结果。如图 4所示，调整最大迭代次数以及朱利亚集的内外范围，可以捕捉到更多低级细节（即，计算成本也会增加，而且图像/模型中的细节过多会产生莫尔效应）。这些模拟在风格和外观上与四元数朱利亚集（旋转的棉花糖外观）高度相关。尽管双四元数空间具有明确定义的和、积与范数，但本文中的距离平方估计器对实部和对偶部分采用四元数计算，这意味着它仍与四维空间紧密耦合。当然，这也为进一步探索采用双四元数代数的其他距离平方估计器变体敞开了大门。

## 5 结论/讨论

总之，分形是生成有趣图案的有力工具。将朱利亚集扩展到更高维度，可以形成具有强烈立体视觉特性的形状，并呈现相互交织、不规则的边缘和变换。本文采用双四元数代数，利用朱利亚集实现了三维分形。为了利用朱利亚集捕捉双四元数的维度特性，我们从不同采样点渲染了快照。关于双四元数空间中的朱利亚集，仍有许多有待发现之处。双四元数代数定义的规则适用于朱利亚集，可将这一概念扩展到多维。将高维朱利亚集可视化为三维对象的技术。我们介绍了将既有的朱利亚集光线追踪技术扩展至纳入双四元数特性的尝试。我们还讨论了利用图形处理器来减少计算时间的优化算法（从而支持探索性原型设计）。这些图像的科学价值或许值得商榷，但它们具有令人着迷的美学特质。

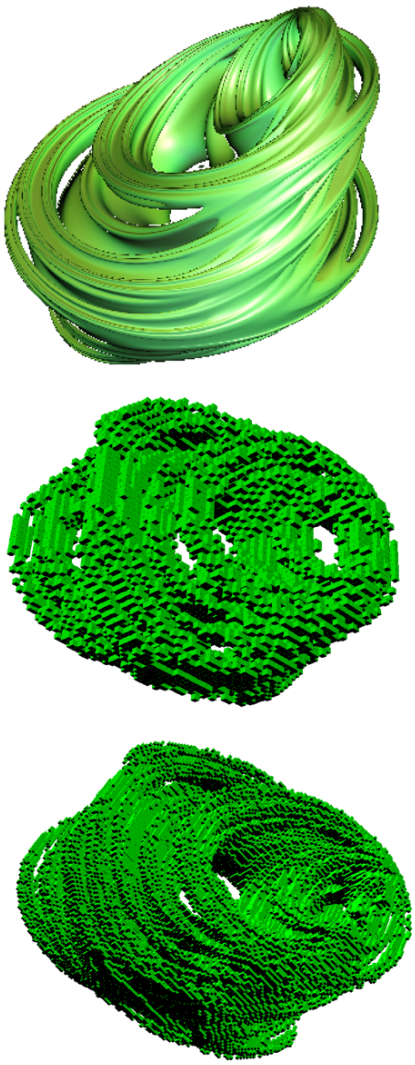


**图6：** 体素与光线追踪 - 初期原型尝试在图形处理器上使用体素 *[Zadick等人 2016]*（即几何着色器）——以创建用于实验的实时原型工具。然而，由于计算能力与分辨率限制，光线追踪算法更适合突出细节。上方为光线追踪图像，中间为 $50 \times 50 \times 50$ 体素网格，下方为 $100 \times 100 \times 100$ 体素网格。

## 参考文献


CLIFFORD, W. 1882年。《数学论文集》。麦克米伦公司。 1

K. Crane，2005年。使用图形处理器对四元数朱利亚集进行光线追踪。1，2，4 Y. Dang、L. H. Kauffman 和 D. J. Sandin，2002年。超复数迭代：距离估计与高维分形，第1卷。世界科学出版社。3 I. Z. Frey 和 I. Herzeg，2011年。使用四元数和q切线进行球面蒙皮。载于《SIGGRAPH 演讲》，11。1 J. C. Hart、D. J. Sandin 和 L. H. Kauffman，1989年。确定性三维分形的光线追踪。载于《ACM SIGGRAPH 计算机图形学》，第23卷，ACM，289–296。2，3 J. C. Hart、L. H. Kauffman 和 D. J. Sandin，1990年。四元数朱利亚集的交互式可视化。载于《第一届 Visualization'90 会议论文集》，IEEE 计算机学会出版社，209–218。2 L. Kavan、S. Collins、˘Z ´ARA 和 C. O' Sullivan，2008年。采用近似对偶四元数混合的几何蒙皮。《ACM 图形学汇刊》（TOG）27卷，4期，105。1

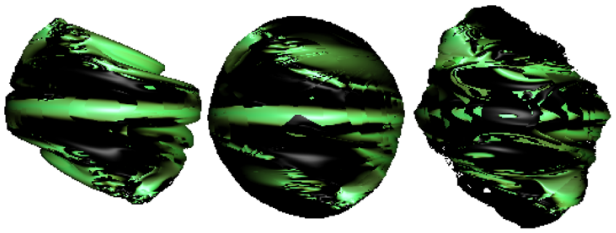


**图 7：** 早期实验 - 距离参数定义不佳——通过随机值探索视觉效果。

Y. Ke 和 E. S. Panduranga，1990年。四维之旅。载于《第一届 Visualization'90 会议论文集》，IEEE 计算机学会出版社，219–229。1 B. Kenwright，2012年。对偶四元数入门指南：它们是什么、如何运作，以及如何将其用于三维角色层级结构。载于《第20届计算机图形学、可视化与计算机视觉国际会议》，WSCG 2012通讯论文集，1–13。1，2，5 Y. Kuang、A. Mao、G. Li 和 Y. Xiong，2011年。着衣人体运动实时动画策略。载于《多媒体技术（ICMT）》，2011年国际会议，IEEE，4793–4797。1 B. B. Mandelbrot，1983年。《自然界的分形几何》，第173卷。WH Freeman，纽约。2 A. Norton，1982年。三维几何分形的生成与显示。载于《ACM SIGGRAPH 计算机图形学》，第16卷，ACM，61–67。2 C. A. Pickover，1998年。《混沌与分形：计算机图形学之旅》。爱思唯尔。1 J. Selig，2010年。刚体运动的有理插值。载于《控制、信号与系统理论及物理建模进展》。施普林格，213–224。1 A. Vasilakis 和 FUDOS，2009年。基于骨架的角色动画刚性蒙皮。载于 GRAPP，302–308。1 X. Wang 和 C. Yu，2010年。基于单位双四元数的耦合姿态与平移动力学反馈线性化调节器。载于《智能控制（ISIC）》，2010年 IEEE 国际研讨会，IEEE，2380–2384。1 D. White，2007年。www.fractalforums.com - www.skytopia.com - 搜索 Mandelbulb 章节（http://www.skytopia.com/project/fractal/mandelbulb.html）。载于（在线访问日期：2018年6月2日）。2 J. Zadick、B. Kenwright 和 K. Mitchell，2016年。将实时流体仿真集成到体素引擎中。《计算机游戏期刊》5卷，1–2期，55–64。4

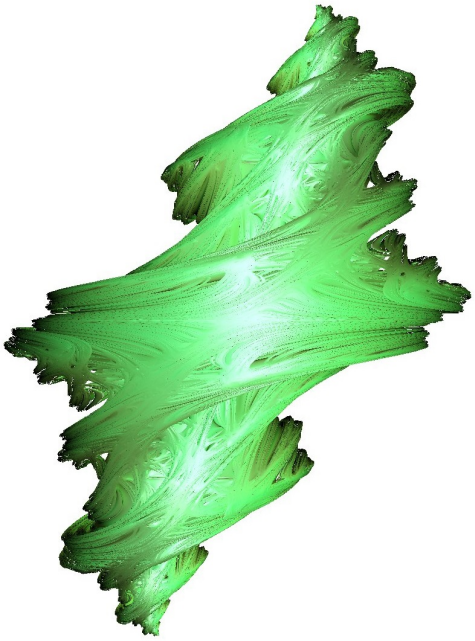


**图8：** 高细节 - 通过增加迭代次数，突出对偶四元数 *Julia* 集的几何细节。*c =(-0.39054,-0.58679,0,0)(0.0,0.5632,0,0.05), n=15*。

## A 附录

**四元数运算** Hamilton 于 1843年 发现了四元数，并将其作为进行三维乘法 [Kenwright 2012]的方法。一个四元数 $q$ 由公式 9 给出。由于我们将四元数与对偶数理论结合起来，因此给出基本四元数算术运算。

$$q = [s, \vec{v}], (q_w, q_x, q_y, q_z) \tag{9}$$

其中 $s$ 标量部分为 $s = q_w$，向量部分为 $\vec{v} = (q_x, q_y, q_z)$。由分别赋予一个实轴和三个正交归一虚轴的独立实数值组成的四元组：$i, j, k$。

- **addition**: $q_1 + q_2 = [s_1, \vec{v}_1] + [s_2, \vec{v}_2] = [s_1 + s_2, \vec{v}_1 + \vec{v}_2]$
- **additive identity**: $0 = [0, 0]$
- $kq = [ks, k\vec{v}]$ **标量乘法：** **乘法:**
- $q_1 q_2 = [s_1, \vec{v}_1][s_2, \vec{v}_2] = [s_1 s_2 - \vec{v}_1 \cdot \vec{v}_2, s_1 \vec{v}_2 + s_2 \vec{v}_1 + \vec{v}_1 \times \vec{v}_2]$
- $1 = [1, 0]$ **乘法单位元：** **点积：** $q_1 \cdot q_2 = ($
- $q_{1x} q_{2x} + q_{1y} q_{2y} + q_{1z} q_{2z} + q_{1w} q_{2w})$
- **模**: $||q|| = \sqrt{(s^2 + ||\vec{v}||^2)}$
- **共轭** $q^* = [s, -\vec{v}]$

**双四元数运算** 使用双四元数所需的基本算术运算。

- **对偶四元数**: $\zeta = q_r + q_d \varepsilon$
- **标量乘法**: $s\zeta = sq_r + sq_d \varepsilon$
- **加法**: $\zeta_1 + \zeta_2 = q_{r1} + q_{r1} + (q_{d1} + q_{d2})\varepsilon$
- **乘法**: $\zeta_1 \zeta_2 = q_{r1} + q_{r2} + (q_{r1} q_{d2} + q_{d1} q_{r2})\varepsilon$
- **共轭**: $\zeta^* = q_r^* + q_d^* \varepsilon$
- **模**: $||\zeta|| = \zeta\zeta*$

其中 $q_r$ 和 $q_d$ 分别表示对偶四元数的实部和对偶部。