

# 欧拉滤波在摄影机运动控制系统 轨迹规划中的应用研究

北京电影学院中国电影高新技术研究院 贺一慧 刘戈三 顾晓娟

**【摘要】** 摄影机运动控制系统在电影特效制作中具有重要作用，摄影机运动控制系统能够精确地在拍摄现场根据需求完成轨迹，灵活运用可以实现丰富的视觉特效。本文介绍了在为摄影机运动控制系统进行摄影机轨迹规划时，从旋转矩阵转换得到的用于表示摄影机姿态的欧拉角数值不连续的问题及其产生原因，并对如何使用欧拉滤波解决这一问题进行了详述。

**【关键词】** 欧拉滤波 摄影机运动控制系统 轨迹规划

## 1 引言

运动控制系统 (motion control system) 是指对运动机构的位置和速度进行快速、准确控制的系统，由系统软件发出指令，驱动电机完成运动。

在影视拍摄中使用的摄影机运动控制系统，是为了将运动控制系统应用于精确控制摄影机运动轨迹，根据影视拍摄需求对其进行专业改造的产物。摄影机运动控制系统广泛应用于影像合成、高精度摄影以及逐格拍摄等工作<sup>[4]</sup>。在影视应用中，常常以运动控制系统代称摄影机运动控制系统，本文遵循惯例，文中出现的运动控制系统均意指摄影机运动控制系统。

本文以英国 MRMC (Mark Roberts Motion Control) 公司研发的运动控制设备 Milo<sup>[7]</sup> 为例，探讨了在三维软件中进行摄影机运动轨迹规划时遇到的，表示摄影机姿态的欧拉角数值不连续问题及其解决方法。

## 2 运动控制系统介绍

### 2.1 运动控制系统在影视中的应用

运动控制系统是目前视觉特效领域必不可少的工具，使用运动控制系统，可以达到多次重复拍摄得到的画面像素级别的吻合，从而进行实拍与三维

元素合成的精确运动匹配，灵活运用可以达到多人复制、超长距离运动、前(后)景不变而后(前)景替换、不同大小主体合成等等各类精彩的视觉效果<sup>[1]</sup>。

近年来，运动控制系统与计算机三维动画系统在实际应用中结合得非常紧密，三维虚拟世界中摄影机的外部参数(位置以及旋转姿态信息)和内部参数(焦距、对焦距离以及光圈信息)都可以和真实世界中摄影机的对应参数相同步，从而得以实现更精确的拍摄要求。影视工作中的流程通常可分为以下步骤：

在三维动画软件中对影片预期的效果进行设计和预览，包括对场景、角色动画进行精确的设计预览，利用三维软件快速找出满意的效果。

利用三维软件的运动轨迹编辑功能对摄影机运动轨迹进行规划。

将摄影机运动轨迹导出到运动控制系统的控制软件，驱动运动控制系统在拍摄现场完成规划好的动作。

### 2.2 运动控制系统 Milo 的机械结构

Milo 是一个在国内外影视制作中应用广泛的运动控制系统，它在机械运动上提供了 8 个自由度的控制，前 5 个自由度用于控制摄影机的位置，如图

1 所示, 分别是 Track (轨道上的前后移动)、Rotate (Milo 自身的旋转)、Lift (Milo 大臂的上下转动)、Extend (Milo 大臂的前后移动)、Angle (Milo 前臂的转动)。后 3 个自由度用于控制摄影机的姿态, 如图 2 所示, 分别是 Pan (摄影机摇动)、Tilt (摄影机俯仰)、Roll (摄影机转动)。此外, 它还有控制摄影机内部参数的 Zoom (镜头变焦)、Focus (镜头对焦距离)、Iris (镜头光圈大小) 这 3 个自由度。通过 Flair 软件可以操作运动控制系统的这些机械自由度<sup>[5]</sup>。

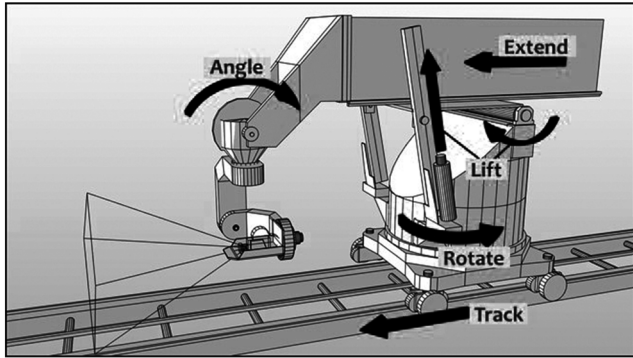


图 1 Milo 机械各自由度

(Track、Rotate、Lift、Extend、Angle) 及运动方向

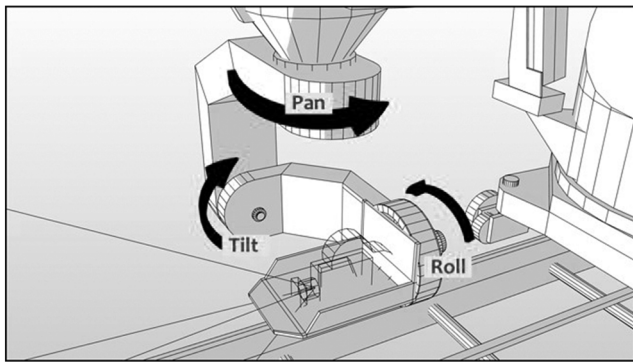


图 2 Milo 机械各自由度 (Pan、Tilt、Roll) 及运动方向

### 3 摄影机姿态的数学表示

#### 3.1 概述

本文所探讨的主题是, 使用欧拉滤波修正用于表达摄影机姿态的欧拉角数值不连续的问题。因此有必要对其中使用的数学原理进行简要说明。

在使用中, 摄影机实际上作为刚体参与计算, 下文描述的与刚体相关的内容都可以应用在摄影机上。

描述刚体在空间中的运动情况, 我们需要两个参数, 即刚体的位移与姿态。刚体位移用于表示刚体在空间中的位置, 我们可以在空间中定义原点, 建立一个正交坐标系, 用从原点指向刚体位置的三维向量来描述刚体的位移, 而位于原点的这个坐标系称作世界坐标系。刚体姿态用于表示刚体在空间中的朝向, 我们可以在刚体上选择一点作为它的旋转中心 (例如选择刚体的几何中心点), 在这一点建立新的坐标系来描述它的姿态。刚体姿态常用的数学表示方法有三种: 欧拉角, 矩阵和四元数。本文只涉及到欧拉角和矩阵, 因此只对这两种表达式进行讨论。

#### 3.2 欧拉角

欧拉角的基本思想是, 将旋转姿态分解为围绕着三个相互垂直轴的三次旋转, 即是说, 给出三个相互垂直的轴向, 欧拉角能够用来描述任意旋转。具体应该

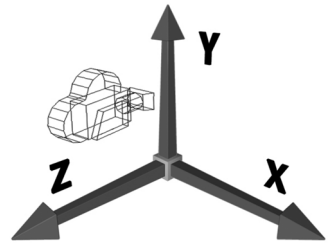


图 3 虚拟摄影机的轴向

采用哪三个轴向以及围绕这三个轴向的旋转顺序如何, 根据应用情况会有所不同。

通常在影视拍摄时, 我们将摄影机的三个轴向定义为 Pan, Tilt, Roll, 对应在 Maya 中分别是 y, x, z 三个轴向, 摄影机朝向 z 轴负方向, 如图 3 所示。

在这里我们使用的是右手坐标系, 如图 4 所示, 将右手的大拇指、食指和中指相互垂直起来, 分别指向 X、Y、Z 三轴, 满足此种条件的坐标系即为右手坐标系, 同时每个轴向旋转的正负方向也能够随之确定下来。

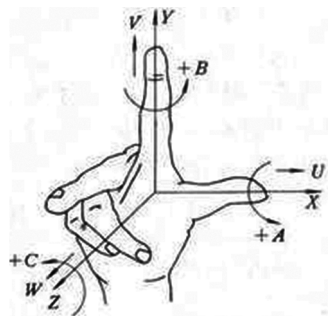


图 4 用右手指示坐标系的三个方向

根据不同的旋转顺序对刚体进行旋转, 得到的

刚体姿态会有所不同。例如，如果要围绕着  $x$  轴和  $y$  轴各旋转  $90^\circ$ ，先围绕  $y$  轴旋转再围绕  $x$  轴旋转，与先围绕  $x$  轴再围绕  $y$  轴旋转，得到的结果分别如图 5 与图 6 所示。

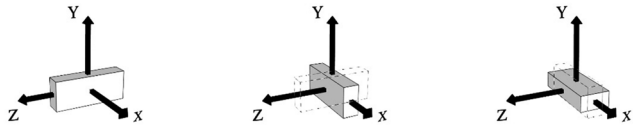


图 5 刚体先围绕  $y$  轴旋转  $90^\circ$ ，再围绕  $x$  轴旋转  $90^\circ$

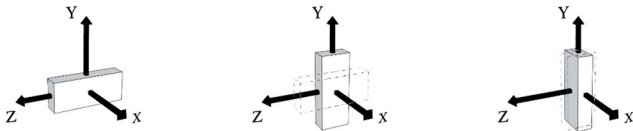


图 6 刚体先围绕  $x$  轴旋转  $90^\circ$ ，再围绕  $y$  轴旋转  $90^\circ$

在 Maya 以及大部分三维软件中，默认的旋转顺序是  $xyz$ ，即先绕  $x$  轴旋转，其次是  $y$  轴，最后是  $z$  轴，本文也默认使用这一旋转序列。

### 3.3 使用欧拉角的优点及缺陷

欧拉角比较直观，符合人们思考刚体姿态的方式，因而易于使用。在人机交互中，无论是显示还是要求用户输入，使用欧拉角是唯一的选择。

但是，使用欧拉角会带来一些问题，我们主要讨论其中的两个问题，即给定同一姿态得到的欧拉角表示并非唯一问题，以及万向节锁定问题。

同一姿态的欧拉角表示并非唯一的问题，又分为两种情况。

第一种情况，即简单的给某个轴向的角度值加上  $360^\circ$ ，这样做得到的数值变化了，但是还并不会影响到物体的姿态，如图 7 所示， $\theta$  与  $\theta'$  代表的是同一姿态，但是后者多转了一整圈。

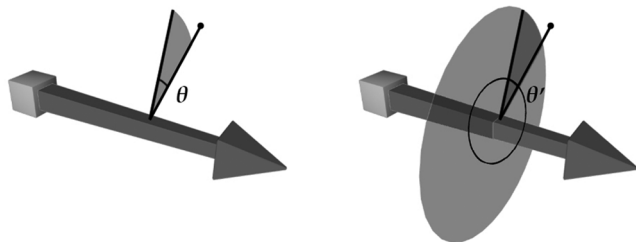


图 7 加上  $360^\circ$  的欧拉角与原始角度表达姿态相同

第二种情况稍稍复杂一些，是由于三个角度不

相互独立导致的。例如，右手坐标系中，旋转顺序为  $xyz$  的情况下，使用  $(0, 25, 0)$  与  $(180, 155, 180)$  这两组欧拉角分别旋转同一刚体得到的最终姿态相同。稍后我们会讨论这三个角度是以怎样的形式相关，以及如何找出这些情况。

在一般的使用中，同一姿态的欧拉角表示并非唯一的问题并不会给我们带来很大的麻烦，但是在某些特别的应用场合，我们需要保证任意姿态都能用独一无二的欧拉角表示（例如在使用动作捕捉设备采集刚体数据时，需要将刚体的姿态数据转换为欧拉角数据）。此时，必须对欧拉角的数值范围加以限制，一种常用的方法是将  $x$  和  $z$  限制在  $\pm 180^\circ$  之间，而将  $y$  限制在  $\pm 90^\circ$  之间。由此对于给定的任意角位移，都能且只能在此范围内得到一组对应的欧拉角数值（我们随后讨论的万向节锁定是一个特例）。

在旋转顺序为  $xyz$  的右手坐标系中，如果将刚体先沿  $y$  轴旋转  $\pm 90^\circ$ ，则物体的  $x$  轴和  $z$  轴相互重合，此时，我们会丧失一个旋转自由度。这一自由度的丧失，就是三维中的万向节锁定现象（Gimbal Lock）。万向节锁定会导致在某些情况下，空间中连续的姿态，不能用连续的坐标值来表示<sup>[8]</sup>。

### 3.4 旋转矩阵

矩阵是除欧拉角之外，在三维中表示刚体姿态的另一种方式，与欧拉角的直观易理解相比，矩阵有易于计算的优点，因而在计算机内部使用矩阵来表示刚体姿态。

矩阵可以由三个相互垂直的单位向量构建，这三个向量即为基向量，这样构建出来的矩阵为  $3 \times 3$  旋转矩阵。这三个基向量由前面所建立的世界坐标系来描述，也就是说，这个矩阵可以用来描述刚体在世界空间坐标系下的姿态。

想要从欧拉角构建矩阵，可以先分别构建各轴向的旋转矩阵，再组合得到最终的旋转矩阵。用  $\Phi$ ， $\theta$  和  $\psi$  分别代表刚体围绕着  $x$  轴、 $y$  轴、 $z$  轴旋转的角度，我们可以构建各轴向旋转矩阵  $R_x$ 、 $R_y$ 、 $R_z$  如下<sup>[2]</sup>：

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & \sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{bmatrix} \quad (1-1)$$

$$R_y = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \quad (1-2)$$

$$R = R_x R_y R_z = \begin{bmatrix} \cos\theta\cos\psi & \cos\Phi\sin\psi + \sin\Phi\sin\theta\cos\psi & \sin\Phi\sin\psi - \cos\Phi\sin\theta\cos\psi \\ -\cos\theta\sin\psi & \cos\Phi\cos\psi - \sin\Phi\sin\theta\sin\psi & \sin\Phi\cos\psi + \cos\Phi\sin\theta\sin\psi \\ \sin\theta & -\sin\Phi\cos\theta & \cos\Phi\cos\theta \end{bmatrix} \quad (1-4)$$

将矩阵转换为欧拉角的方法由于篇幅限制不在此详述，基本思想是分情况讨论解三角函数<sup>[6]</sup>。

## 4 在三维软件中进行运动控制系统的轨迹规划

### 4.1 在 Maya 中规划摄影机运动轨迹

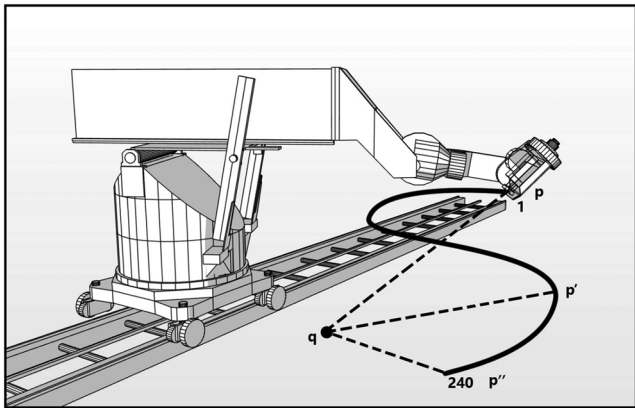


图 8 规划摄影机的运动轨迹

我们先在 Maya 中，利用绘制曲线的方式，结合虚拟的场景，设计出符合拍摄意图的摄影机运动轨迹。如图 8 中的 S 形曲线即为设计好的从第 1 帧到第 240 帧摄影机运动位移轨迹。

设计位移轨迹时，我们还需要设计摄影机运动的姿态轨迹，一个常见的做法是在空间中找到一个拍摄点，让摄影机的正前方始终对准这一点（相当于摄影机正中心始终对准拍摄物），当然，拍摄物本身也可以是运动的。我们现在可以在每一帧构建摄影机的姿态矩阵，设摄影机的位置点为 p，目标点为 q，如图 8 所示，q 点为目标点，p 点 p' 和 p'' 点分别为摄影机运动起点、过程中和终点三点。

为了构建摄影机的姿态矩阵，我们还需要一个

$$R_z = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1-3)$$

根据前文提到的，我们采用的旋转序列是 xyz，因此得到最终旋转矩阵 R 的方式为：

摄影机的正上方的朝向向量，通常我们使用 y 轴正方向。如果有特殊要求，也可以用一个点来表示，使用该点到摄影机位置点的向量作为摄影机正上方向量，我们将这个正上方向记作 u。

现在我们可以通过三个基向量来构建摄影机的姿态矩阵<sup>[3]</sup>：

$$\begin{aligned} z &= \text{normalize}(q-p) \\ y &= \text{normalize}(u) \end{aligned} \quad (2-1)$$

$$x = y \times z$$

z 是摄影机正前方的轴向，y 是摄影机正上方的轴向，x 需要与 z 和 y 垂直，因此求 z 与 y 的外积得到。

这里的 normalize() 是将向量归一化，计算方法为：

$$\text{normalize}(a) = \frac{a}{\sqrt{a_1^2 + a_2^2 + a_3^2}} \quad (2-2)$$

最终得到的旋转矩阵 R 为：

$$R = \begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \end{bmatrix} \quad (2-3)$$

### 4.2 摄影机姿态旋转矩阵转换为欧拉角

根据 4.1 小节中的方法，将规划好的姿态矩阵转换为欧拉角，得到曲线如图 9，我们可以很明显地看出在曲线的后半段 187—195 帧处，曲线的值发生了大幅度的跳跃，但实际上摄影机的运动轨迹曲线在三维空间中的旋转姿态是连续的，这就是我们在 3.3 小节中提到的用欧拉角表示旋转的缺点导致的问题。在通过旋转矩阵计算欧拉角数值时，为了得到唯一解，需要对欧拉角进行特定的限制，导致

转换出的动画曲线数值不连续。

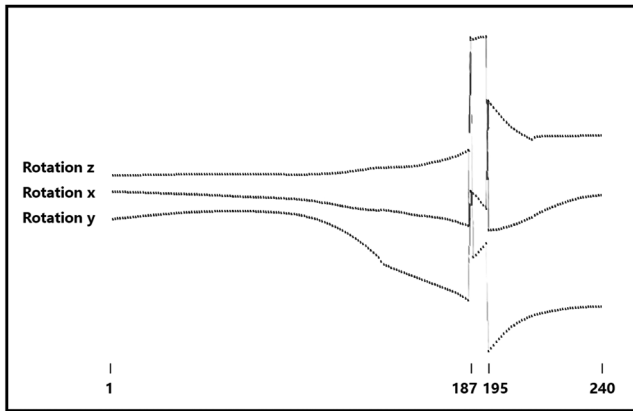


图 9 欧拉角数值不连续的摄影机姿态轨迹曲线

根据第 2 节介绍过的 Milo 机械结构，x、y、z

$$R = R_x R_y R_z = \begin{bmatrix} \cos\theta\cos\phi & \cos\Phi\sin\phi + \sin\Phi\sin\theta\cos\phi & \sin\Phi\sin\phi - \cos\Phi\sin\theta\cos\phi \\ -\cos\theta\sin\phi & \cos\Phi\cos\phi - \sin\Phi\sin\theta\sin\phi & \sin\Phi\cos\phi + \cos\Phi\sin\theta\sin\phi \\ \sin\theta & -\sin\Phi\cos\theta & \cos\Phi\cos\theta \end{bmatrix} \quad (3-1)$$

以下我们用 (3-2) 所示的下标参数形式来指代矩阵中的各个元素：

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \quad (3-2)$$

由于  $R_{31} = \sin\theta$ ，并且  $\sin(\pi - \theta) = \sin(\theta)$ ，在  $R_{31}$  与  $\theta$  不变的情况下， $R_{31}$  可能的值有  $R_{31} = \sin(\theta + 2b\pi)$ ，或  $R_{31} = \sin(-\theta + (2b-1)\pi)$  两种情况，其中 b 为任意整数。

第一种情况：

当  $R_{31}$  取  $\sin(\theta + 2b\pi)$  时，含有  $\cos\theta$  的项应该被替换为  $\cos(\theta + 2b\pi)$ ，并且其他各项也在各角度任意增减  $2\pi$  的情况下保持矩阵本身元素不变，由此可以得到矩阵 R 的第一种情况  $R_1$ ，即将 R 中  $\theta$  替换为  $(\theta + 2b\pi)$ ， $\phi$  替换为  $(\phi + 2c\pi)$ ， $\Phi$  替换为  $(\Phi + 2a\pi)$ ，其中 a、b、c 均为任意整数。

第二种情况：

当  $R_{31}$  取  $\sin(-\theta + (2b-1)\pi)$  时，由于  $\cos(-\theta + (2b-1)\pi) = -\cos\theta$ ， $R_{11}$ ， $R_{21}$ ， $R_{32}$ ， $R_{33}$ ，这四项含有  $\cos\theta$  的项需要乘以  $-1$  以平衡正负号，因此  $\phi$  需要被替换为  $(\phi + (2c-1)\pi)$ ， $\Phi$  需要被替换为  $(\Phi + (2a-1)\pi)$ ，其中 a、b、c 均为任意

三个轴向上的旋转（即 pan、tilt、roll）是由相互独立的机械转轴实现的，为了让机械能够运转我们必须为每个旋转轴提供连续的数值。因此，我们需要在不改变设计好的摄影机姿态轨迹的前提下，将这些数值变换为连续数值，这就是欧拉滤波的作用。

### 4.3 欧拉滤波

欧拉滤波逐样本检查 xyz 三轴动画曲线不连续的地方，并在保证刚体姿态不变的情况下在时间序列上使得动画曲线尽量连续起来<sup>[9]</sup>。

当刚体姿态不变时，它所对应的旋转矩阵是唯一的（尽管用欧拉角表示时数值会有所不同），我们从矩阵转换为欧拉角入手，研究同一矩阵求解出的欧拉角的变化规律。

整数。此时，由于  $\sin(\phi + (2c-1)\pi) = -\sin\phi$  并且  $\cos(\phi + (2c-1)\pi) = -\cos\phi$ ，对于  $\Phi$  也可以应用这一情况，因此  $R_{11}$ 、 $R_{21}$ 、 $R_{32}$ 、 $R_{33}$  这四项含有  $\cos\theta$  带来的负号被平衡， $R_{11}$ 、 $R_{21}$ 、 $R_{32}$ 、 $R_{33}$  保持不变，而在此种情况下可以计算得出  $R_{12}$ 、 $R_{13}$ 、 $R_{22}$ 、 $R_{23}$  的正负号也可以平衡，即这四项的值也保持不变，由此我们得到了矩阵 R 的第二种情况  $R_2$ 。

综合以上两种情况，我们可以整理出伪代码的逻辑为：

```

if ( abs ( 前后两帧 x 值之差 ) > π && abs ( 前后两帧 z 值之差 ) > π )
{
    x += ( 2a - 1 ) π ;
    y += ( 2b - 1 ) π ;
    z += ( 2c - 1 ) π ;
    y * = - 1 ;
}
if ( abs ( 前后两帧 x 值之差 ) > 2π )
    x += 2aπ
if ( abs ( 前后两帧 y 值之差 ) > 2π )
    y += 2bπ
    
```

if (abs (前后两帧 z 值之差)  $> 2\pi$  )

z +=  $2c\pi$

在 Maya 中根据伪代码编写 MEL 脚本, 读取 xyz 三轴动画曲线数据, 在每一帧检查动画曲线的连续性, 给不连续的部分加上适当的偏移值, 最终得到的结果如图 10。

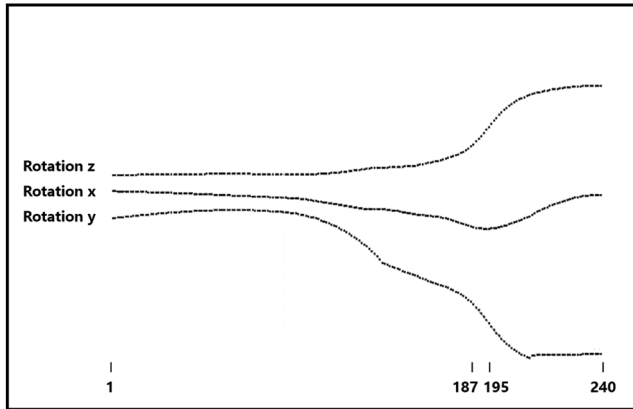


图 10 欧拉滤波后数值连续的摄影机姿态轨迹曲线

如图 10 所示, 进行欧拉滤波后的欧拉角数值曲线变为了连续的平滑曲线, 我们可以将这些数值进一步处理后用于驱动 Milo 运动。

## 5 结语

在摄影机运动轨迹规划中, 将摄影机姿态从旋转矩阵转换为欧拉角时, 需要对欧拉角的取值范围

进行限制以保证可以求出欧拉角的唯一解, 这会导致欧拉角数值不连续的情况发生。应用欧拉滤波, 在保持摄影机姿态不变的情况下, 对不连续的数值进行变换, 得到连续的欧拉角数值, 从而让运动控制系统的机械结构可以连续地运转。❖

## 参考文献

- [1] 李念芦, 李铭, 王春水, 朱梁. 影视技术基础 [M]. 北京: 世界图书出版公司, 2012: 295
- [2] Fletcher, Lan. 3D 数学基础: 图形与游戏开发 [M]. 北京: 清华大学出版社, 2005: 85-94
- [3] Gortler SJ. Foundations of 3D Computer Graphics [M]. Cambridge: The MIT Press, 2012: 39
- [4] 刘进. 先进的摄影机运动控制系统 [J]. 影视技术, 2004 (5): 7-11
- [5] 董屹. 关于摄影机运动控制系统的应用研究 [D]. 北京: 北京电影学院, 2013: 10-12
- [6] Slabaugh G G. Computing Euler angles from a rotation matrix [J]. 1999
- [7] Milo 官方网站 [EB/OL]. <http://www.mrmoco.com/products/rigs/milo/>
- [8] Wiki 词条: Gimbal Lock [EB/OL]. [http://en.wikipedia.org/wiki/Gimbal\\_lock](http://en.wikipedia.org/wiki/Gimbal_lock)
- [9] Wiki 词条: Euler Filter [EB/OL]. [https://en.wikipedia.org/wiki/Euler\\_filter](https://en.wikipedia.org/wiki/Euler_filter)