

基于表演的2维角色动画技术研究

余春龙^{1),2)}, 王丹力¹⁾

¹⁾(中国科学院软件研究所, 北京 100190) ²⁾(中国科学院研究生院, 北京 100049)

摘要: 鉴于基于表演的动画技术非常直观自然, 因此特别适合新手以及非专业人员使用, 已在近些年受到了重要关注。但是已有的基于表演的动画系统, 或是需要特殊的交互设备, 或是只适合于低自由度的物体的动画, 比如刚体动画。针对这一问题, 提出了一种新的基于表演的动画技术, 并根据该技术开发了一个2维角色动画系统。它可使用笔作为道具, 实时地把笔的运动映射到动画角色上, 而角色的运动过程被录制下来即生成动画。评估结果表明, 该系统降低了动画制作的门槛, 让非专业人员也可以简单快速地创作角色动画。

关键词: 基于表演的动画; 新手或非专业人员; 逆向运动学; 角色(人, 动物)

中图法分类号: TP391.41 **文献标志码:** A **文章编号:** 1006-8961(2010)09-1285-06

Performance-based method for 2D character animation

YU Chunlong^{1),2)}, WANG Danli¹⁾

¹⁾(Institute of Software, Chinese Academy of Science, Beijing 100190)

²⁾(Graduate University of Chinese Academy of Science, Beijing 100049)

Abstract: Performance-based animation has received significant attention because of its naturalness, intuition, and its easy-to-use for novices and non-professional animators. However, most existing performance-based systems either require special and expensive device or can only animate low degree of freedom objects such as a rigid body. This paper presents a new performance-based animation technique for novice animators and develops a 2D character animation system. It uses a pen as the prop and maps its movement to the character in real-time, the character's motion is recorded to form an animation. The evaluation result shows that the system lowers the barrier of animation authoring and allows novices to create animation simply and quickly.

Keywords: performance-based animation; novices and non-professional animators; inverse kinematics; character(human, animal)

0 引言

近些年来,很多针对新手的、直观的动画技术和工具相继出现。Davis等人介绍了一个2维草图动画系统^[1],该系统允许用户随意地画出想要的物体,并能产生诸如平移、旋转、缩放、出现/消失等动

作,而且这些动作可以相互组合,从而生成更为复杂的运动。Igarashi提出的变形动画系统可以让用户直接拖拽草图,通过草图的变形过程被记录下来生成动画^[2]。Kato等人引入了effect-line用来描述物体的运动^[3],Popovic等人使用草图来描述刚体的运动轨迹与运动时间^[4]。文献[5-6]引入了邻域知识,用户只需要绘制物体,并简单地设置相关的参

基金项目: 国家自然科学基金项目(60970090);国家科技支撑计划课题基金项目(2008BAI50B02);国家高技术研究发展计划(863)项目(2008AA01Z303);中国科学院知识创新工程领域前沿项目(ISCAS2009-QY03)。

收稿日期: 2010-03-16; **改回日期:** 2010-04-23

第一作者简介: 余春龙(1984—),男,中国科学院软件研究所计算机应用技术专业硕士生。主要研究方向为人机交互与计算机图形学。E-mail: chunlongyu@gmail.com。

通讯作者: 王丹力, E-mail: danli_wang@163.com。

数,就可以自动生成物理或数学动画。尽管如此,角色(包括人,动物等)动画的制作仍然是一个复杂的问题,由于角色比一般的物体(上述系统所针对的)具有更高的自由度,因此研究如何容易地控制这种高自由度的运动,并使其适合新手,仍然是一个有挑战性的课题。

基于表演的(performance-based)动画技术是通过在用户的表演与角色的运动之间建立映射关系,使用户的表演实时地生成角色的运动,运动过程被录制下来即生成动画。这种映射可以是直接的,即用户可直接拖拽角色的某个部分或者整体^[1],也可以是间接的,即用户的表演与角色的运动可不在同一个坐标空间^[7]。同时,表演所用的道具也是多样化的,可以是专门设计的交互设备^[7],也可以是通用的设备^[8]。这种技术具有“所见即所得”的特点,因此非常直观自然,特别适合新手以及非专业人员使用,已在近些年来受到了广泛关注。

Dontcheva 等人以一种特殊的交互设备为道具,通过把对道具的操作实时映射到动画角色上来生成动画^[7]。映射可以是显式的定义,也可以是隐式地从道具运动与角色运动的相似关系中推导出来。该文首次提出了层次式技术(layering),即把多次表演生成的动作进行合成,或者是在已有动作上添加细节。Igarashi 等人首次提出了空间帧(spatial frame)的概念^[8],并在移动鼠标的过程中,根据鼠标的位置实时地插值来生成中间每一帧,其中角色的运动时间和速度取决于用户的表演(即用户对鼠标的划动)。这种方法操作简单,解决了如何用低自由度的交互设备实时控制高自由度的角色运动这个问题,但是它在鼠标的运动与角色的运动之间的映射关系不够直观。文献[9]扩展了空间帧技术,它把所有的空间帧分成多个集合(而不是一个集合),对于每一个集合,即用一个相应的最底层控制器来控制这个角色的某一个部分。所有的控制器组成一个层次式的结构,每一层的控制器被上一层的控件器所控制,最底层的每一个控制器直接控制角色的一个部分。最后用户只需移动最顶层的控制器,就可控制角色的整体运动。Neff 等人引入了 correlation map^[10],它首先通过参数化降低角色的自由度,然后把这些参数映射到鼠标的 x 坐标或 y 坐标,这种映射称之为 correlation map。通过这种方式,用户就可以通过鼠标控制整个角色或角色的一个部分。同样,它也使用了层次技术来合成多次不同的表演。

文献[11]利用基于表演的技术来降低关键帧技术的复杂性,它把关键帧的值的指定与时间的指定分离,值的指定与传统的方式一样,但是通过基于表演的技术来指定每一个关键帧对应的时间,表演所用的道具为笔或鼠标。

本文在分析现有的基于表演的动画技术的基础上,提出了一种新的基于表演的动画技术。该技术有以下几方面的特点:1)通过引入正向运动学(forward kinematics)与逆向运动学(inverse kinematics)技术来方便地设置角色的姿态;2)结合了空间帧技术与动作合成技术,从而能创作更为形象的动画;3)提供了一个包含有规则排列的空间帧标记的控制面板,提高了映射的直观性。

最后在此基础上开发了一个2维角色动画制作系统。它使用笔作为道具,把笔的运动映射到2维动画角色上,通过笔的运动实时生成角色的运动,这个过程被录制下来即生成动画,同时它也可以把整体的轨迹运动与局部的肢体运动进行合成。

1 基于表演的角色动画技术

1.1 角色建模

角色是由多个链杆组成,所有链杆被组织成一个树形结构,在今后的工作中还会考虑为角色蒙皮。在用户创作模型阶段,角色被组织成一个无环的图结构,其中,每一个关节作为图的结点,每一个链杆作为图的边。创作完成之后,指定树的根结点,系统就自动把这个图结构转换成一棵树:以链杆作为树的结点,以关节作为树的边,结点 W 为世界坐标系(如图1所示)。

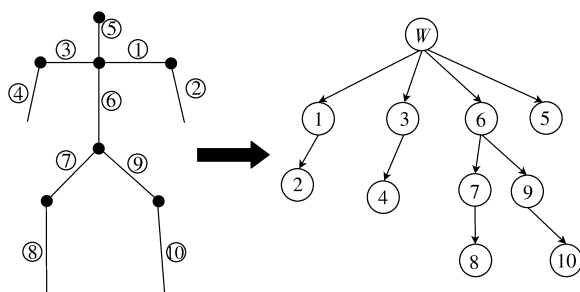


图1 左边为一个角色,右边为对应的树形结构

Fig.1 Left is a character, right is its bone tree

本文采用轴位置关节表示法(axis-position joint representation)^[12]来对整个树进行建模。在这个树

形结构上,每个关节的参数是相对于其父关节所在的局部坐标系测量的,而不是相对世界坐标系(只有根关节的参数是相对世界坐标系)。如图2所示,点 P_1 所在的关节为父关节,点 P_2 所在的关节为子关节,从局部坐标系 $X_1 Y_1$ 逆时针旋转到坐标系 $X_2 Y_2$ 所形成的角度 θ 为子关节的旋转角度。计算公式如下:

$$\cos \theta = \frac{(P_2 - P_1) \cdot (P_3 - P_2)}{\|P_2 - P_1\| \cdot \|P_3 - P_2\|}$$
$$\sin \theta = \frac{(P_2 - P_1) \times (P_3 - P_2)}{\|P_2 - P_1\| \cdot \|P_3 - P_2\|} \quad (1)$$

这种表达方式使人们可以独立地对待每一个关节,也就是说,当改变某一个关节的参数时,其所有子关节的参数保持不变。

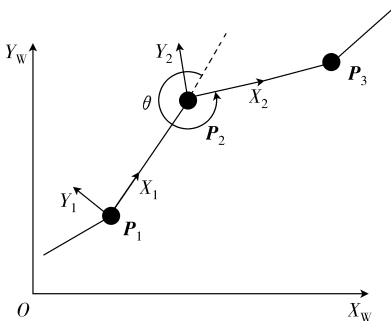


图2 旋转角的表达方式
Fig.2 Representation of the rotation angle

1.2 姿态调整

本文结合使用正向运动学与逆向运动学技术来设置角色的姿态。正向运动学技术是指当用户拖动中间关节时,只改变该中间关节的旋转角度,其所有子关节的角度保持不变,但由于子关节是相对父关节度量的,因此所有子关节会随着父关节一起旋转。这种方法虽比较简单,但需要逐个调整每一个关节,交互复杂。而应用逆向运动学技术,用户则只需要拖动最后一个关节的末端(称为末端影响器)来调整整个关节链,交互相对简单。本文所选择的逆向运动学算法是循环坐标下降法(CCD)^[13-15]。相对于其他方法^[16-18],它的运算速度是最快的。

本文的多约束CCD算法如下:

1) 算法输入

(1) 一个小人。如图3所示,其根关节在脖子位置,它有4个末端影响器 E_1, E_2, E_3, E_4 分别在四肢的末端;

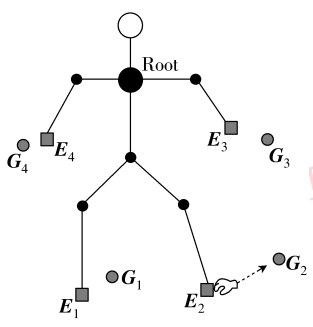


图3 小人的4个末端影响器
Fig.3 The 4 end-effectors of the character

(2) E_2 的目标位置 G_2 。

2) 算法输出

E_2 到达 G_2 时,小人的目标姿态。

3) 计算步骤

(1) 根据根节点的位置,判断受影响的关节链。

在图3中, E_3, E_4 所在的两条关节链与 E_2 所在的关节链是独立的,不受影响,而 E_1 所在的关节链则将受 E_2 的影响;

(2) 调整 E_2 所在的子关节链,使 E_2 到达目标位置 G_2 ,其他3个末端影响器保持不动,也就是说, $E_1 = G_1, E_3 = G_3, E_4 = G_4$ (初始时, G_1, G_3, G_4 是与 E_1, E_3, E_4 的位置重合的,为了方便显示,其在图中是分开的);

(3) 调整 E_1 所在的子关节链,使 E_1 到达目标位置 G_1 (在第1次调整时, G_1 与 E_1 的位置重合,因此该子链保持不动);

(4) 调整 E_1, E_2 所在的两个子链的父链,即与Root相连的最顶层链杆。假设从Root到两个末端影响器 E_1, E_2 的向量分别为 P_1, P_2 ,从Root到两个目标位置 G_1, G_2 的向量分别为 $\bar{P}_1, \bar{P}_2$,第1次, P_1 与 $\bar{P}_1$ 重合(如图4所示),则最顶层链杆的旋转角度 ϕ 的计算如下:

$$\phi = \tan^{-1} \frac{K_1}{K_2}$$
$$K_1 = \sum_{j=1}^n P_j \times \bar{P}_j (n = 2) \quad (2)$$
$$K_2 = \sum_{j=1}^n P_j \cdot \bar{P}_j (n = 2)$$

当调整完最顶层链杆之后,其所有子链杆都旋转相同的角度,则 E_1, E_2 的当前位置也会发生变化。之后,再重复第(2)-(4)步,直到满足如下终止条件: $E = \|E_i - G_i\|^2 < \varepsilon$ (ε 为自定义的精度)。

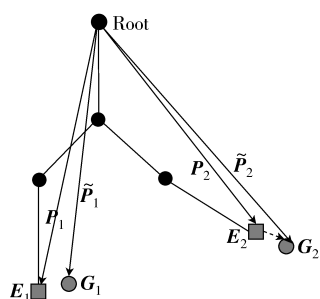


图 4 与 Root 相连的链杆的调整方式

Fig. 4 Adjusting the link connecting the root joint

1.3 基于表演的运动生成

在表演之前先建立映射规则。映射规则描述了道具(笔)的运动如何生成角色的运动。在本文中,一个映射是一个 (s, p) 对,其中 s 是一个状态向量 $(s_1, s_2, s_3, \dots)$,代表了角色的姿态; p 是屏幕上的一个点(笔的位置)。定义好多个映射之后,笔的运动轨迹上的每一个点就可以实时地反向映射到角色的一个状态。因此,一条运动轨迹就对应角色的一次运动过程,称之为一次表演 $P(t)$ 。在表演过程中,系统只需记录笔的运动轨迹,其中轨迹上的每一个点被打上一个时间戳。当播放时,先根据时间索引到轨迹上的某一个点,再根据该点的位置插值计算出角色的状态。

采用如下的线性函数进行插值:

$$s = ((p_{2,x} - p_x) \cdot s_1 + (p_x - p_{1,x}) \cdot s_2) / (p_{2,x} - p_{1,x}) \quad (3)$$

$$s = ((p_{2,y} - p_y) \cdot s_1 + (p_y - p_{1,y}) \cdot s_2) / (p_{2,y} - p_{1,y}) \quad (4)$$

其中 (s_1, p_1) 和 (s_2, p_2) 是两个映射向量对(角色状态,道具位置), $p_{i,x}, p_{i,y}$ 是点 p_i 的两个坐标分量。在鼠标移动过程中,系统会自动判断鼠标的位置是离水平,还是离垂直方向更近,从而决定采用式(3)或式(4)进行插值计算。如图 5 所示,有 4 个 (s, p) 对,

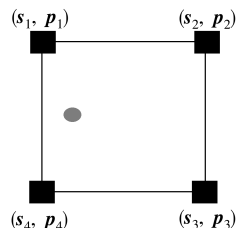
图 5 4 个 (s, p) 对对应角色的 4 个不同状态

Fig. 5 4 blocks corresponding to 4 character's states

鼠标(圆圈所在的点)离左边的竖直线最近,所以会用 (s_1, p_1) 和 (s_4, p_4) 与进行插值。实践证明这种简单的线性插值效果很好,并且速度非常快。

1.4 运动合成

运动合成时,可以把原地的四肢运动与整体的运动轨迹进行合成。用户既可以首先生成四肢的运动,也可以先产生整体的运动轨迹。如果这两个运动在同一个时间段内,则它们将被合成。

合成技术的一个很重要特点是当用户表演当前的动作时,之前所有该时刻的动作会合成,并播放。即一边表演,一边播放,因此,在表演过程中,用户能够看到最终的动画效果。这种特性大大提高了制作过程的直观性,真正达到“所见即所得”的效果。

2 基于表演的 2 维角色动画系统

2.1 动画创作流程

利用上述技术,开发了一个基于笔的 2 维角色动画系统。该系统的动画创作流程如图 6 所示。动画创作时,用户首先创建一个角色,然后拖动任意一个关节来调整其姿态,并建立每个姿态与空间点之间的映射关系,之后就可以通过笔的运动实时生成角色的运动。录制的运动可以直接播放,也可以与已有运动合成之后再播放。

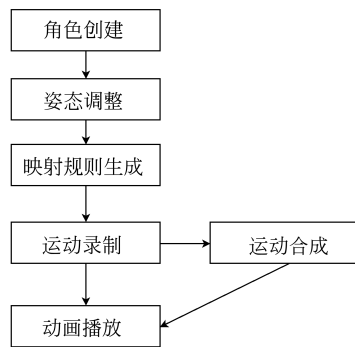


图 6 动画创作流程

Fig. 6 The animation process

2.2 交互过程

在此系统中,笔不仅用来创作角色,更作为一个表演的道具。如图 7 所示,界面分成 4 部分,其中最上面是工具条,中间白色部分是创作区,右边是控制面板(表演区),下面是时间控制区。在创作区,用户进行角色的创作,创作完成之后,可以通过直接拖

动任意一个关节来调整它的姿态;在表演区,每一个方块对应角色的一个姿态,笔在方块之间的划动(表演)产生角色的运动;时间控制区用来控制动画的录制、播放,并用来指导动作的合成。

具体交互过程如下:用户首先创作一个角色(也可以直接加载之前保存的角色),然后通过点击操作来指定整个关节树的根(如图7所示,蓝色结点为树的根结点)。不同于很多系统提供固定拓扑结构的角色,本系统允许用户自由地绘制各种拓扑结构的角色(比如人,动物等)。

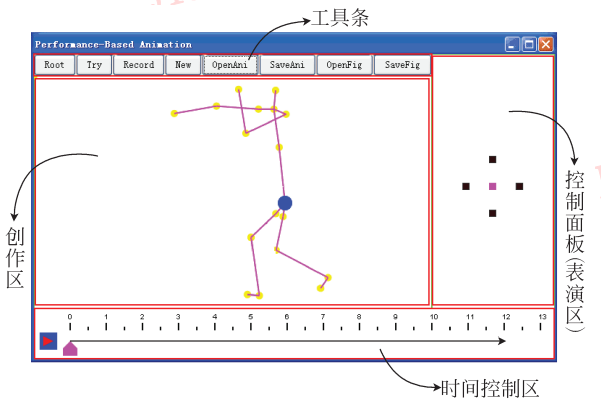


图7 系统主界面
Fig.7 The system interface

完成角色的创作之后,用户开始定义映射。其方法是先直接拖动任意一个关节,调整到满意位置后,点击右边的控制面板中的一个方块,就完成了角色姿态与该方块的绑定;然后重复这个过程,完成多个绑定。如图8所示,粉红色方块代表当前状态,蓝色方块代表之前设计的状态,黑色方块代表用户未来可选的位置。黑色方块总是围绕在粉红色方块周围,代表当前状态的下一个可选状态。

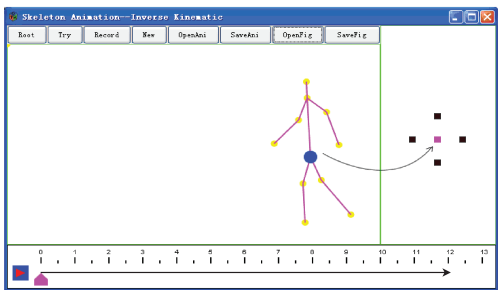
定义好这些映射之后,用户就可以在表演区任意划动鼠标,观看表演结果。当用户觉得满意之后,即可以开始正式地录制。在表演过程中,时间滑块会通过自动向前滑动来录制整个表演过程,从而生成动画。

一次表演完成之后,用户就可以把滑块拖回到某个时间点,再继续下一次的表演,这时第2次表演产生的运动将和上一次的表演进行合成;或者不拖回滑块,直接进行下一次表演,这意味着第2次表演产生的运动将和第1次表演顺序播放。在拖回滑块的过程中,动画会自动进行回放,使用户能够准确定

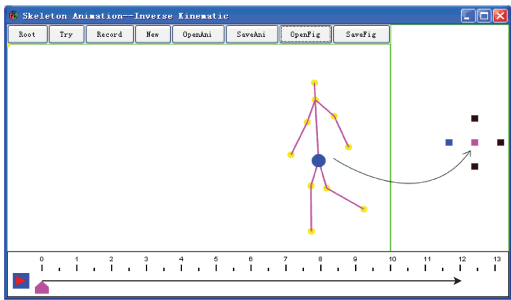
位到他所期望的那个时间点。

2.3 一个应用实例

实例1 通过该系统可以非常容易地针对同一角色,创作不同幅度和不同速度的走路、跑步的动画。为了达到这个目的,只需要定义两个映射对(s , p)(如图8所示)。定义好这两个映射之后,就可以开始表演。笔划动的速度越快,四肢摆动的越快。划动的幅度越大,四肢摆动的幅度越大。通过一次表演,就完成了四肢前后摆动的动作,之后,再把时间滑块拖回到原始位置,然后拖动蓝色的根关节,就实现了角色的整体平移。在拖动过程中,整体平移与四肢摆动会自动合成,用户也就直接看到了最终的动画效果。



(a) 第1个状态



(b) 第2个状态

图8 该角色的两个不同的状态对应两个不同位置的方块
Fig.8 The character's 2 states, associated with 2 blocks

2.4 简单评估

对于应用效果的评估,邀请了8位用户试用了笔者开发的工具,其中包括4位新手(从来没制作过动画)和4位有动画制作经验的用户(用过Flash软件)。在经过几分钟的学习和练习之后,让他们创作了一个“小人跳远”的动画。

在创作过程中,记录了每个用户的创作时间。实验结果表明,所有用户都在2~3 min内创作出了该动画。其中主要时间是用来调整角色的姿态。另外,有经验的用户的创作时间并不比新手更短,这是

由于他们对动画的形象性要求更高,因此在设置姿态的时候比新手更加仔细,甚至会反复斟酌,所以其花费的总时间可能比新手还要长。但是这并不影响每个用户对他自己所创作的作品的满意度,因为新手和专业人员对自身创作动画的水平和动画的形象性的要求不同。

同时,他们认为该工具的主要特点是简单和有趣,能够快速地创作人、动物的动画。另外,他们也对工具提出了一些改进意见:1) 为骨骼蒙皮,也就是在骨骼上添加图形或者图片,让角色更为形象;2) 为用户提供一个角色库,包含常用的各种角色,以方便用户创作。在将来的工作中,会考虑添加这些功能。

3 结 论

提出了一种针对新手的、基于表演的 2 维角色动画技术,该技术可利用笔这种低自由度的交互设备来实时地控制高自由度的角色运动。实验结果表明,该技术降低了角色动画制作过程的复杂性,能够让新手或非专业人员简单快速地创作角色动画。在以后的工作中,将会研究更为简洁而有效的动作合成技术和多个角色的同步机制,以及简单、直观的动作编辑技术。

参考文献 (References)

- [1] Davis R C, Colwell B, Landay J A. K-sketch: A "Kinetic" sketch pad for novice animators [C] // Proceedings of the Twenty-sixth Annual SIGCHI Conference on Human Factors in Computing Systems. New York, USA: ACM Press, 2008: 413-422.
- [2] Igarashi T. Moving Sketch [EB/OL]. (2007-4-1) [2009-7-27] <http://www-ui.is.s.u-tokyo.ac.jp/~takeo/research/rigid/movingsketch/index.html>.
- [3] Kato Y, Shibayama E, Takahashi S. Effect lines for specifying animation effects [C] // Proceedings of the IEEE Symposium on Visual Languages - Human Centric Computing. Washington, DC, USA: IEEE Computer Society, 2004: 27-34.
- [4] Popovic J, Seitz S M, Erdmann M. Motion sketching for control of rigid-body simulations [J]. ACM Transactions on Graphics, 2003, 22(4): 1034-1054.
- [5] Davis R. Sketch understanding in design: overview of work at the MIT AI lab [C] // Proceedings of AAAI Spring Symposium on Sketch Understanding. Menlo Park, California, USA: AAAI Press, 2002: 24-31.
- [6] LaViola J, Zeleznik R C. MathPad2: A system for the creation and exploration of mathematical sketches [J]. ACM Transactions on Graphics, 2004, 23(3): 432-440.
- [7] Dontcheva M, Yngve G, Popovic Z. Layered acting for character animation [J]. ACM Transactions on Graphics, 2003, 22(3): 409-416.
- [8] Igarashi T, Moscovich T, Hughes J F. Spatial keyframing for performance-driven animation [C] // Proceeding of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation. New York, USA: ACM Press, 2005: 107-116.
- [9] Choi B, You M, Noh J. Extended Spatial Keyframing [D]. Daejeon, Korea: Advanced Institute of Science and Technology, 2008.
- [10] Neff M, Albrecht I, Seidel H. Layered performance animation with correlation maps [J]. Computer Graphics Forum (EUROGRAPHICS), 2007, 26(3): 675-684.
- [11] Terra S C L, Metoyer R A. Performance timing for keyframe animation [C] // Proceeding of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation. New York, USA: ACM Press, 2004: 253-258.
- [12] Sims K, Zeltzer D. A figure editor and gait controller for task level animation [C] // Proceedings of the ACM SIGGRAPH Course Notes 4. New York, USA: ACM Press, 1988: 164-181.
- [13] Welman C. Inverse Kinematics and Geometric Constraints for Articulated Figure Manipulation [D]. Burnaby, BC, Canada: Simon Fraser University, 1993.
- [14] Wang L T, Chen C C. A combined optimization method for solving the inverse kinematics problem of mechanical manipulators [J]. IEEE Transactions on Robotics and Automation, 1991, 7(4): 489-499.
- [15] Fedor M. Applications of inverse kinematics for skeleton manipulation in real-time [C] // Proceedings of the 19th Spring Conference on Computer Graphics. New York, USA: ACM Press, 2003: 203-212.
- [16] Yamane K. Natural motion animation through constraining and deconstraining at will [J]. IEEE Transactions on Visualization and Computer Graphics, 2003, 16(2): 352-360.
- [17] Phillips C B, Zhao J, Badler N I. Interactive real-time articulated figure manipulation using multiple kinematic constraints [J]. ACM SIGGRAPH Computer Graphics, 1990, 24(2): 245-250.
- [18] Zhao J, Badler N I. Inverse kinematics positioning using nonlinear programming for highly articulated figures [J]. ACM Transactions on Graphics, 1994, 13(4): 313-336.