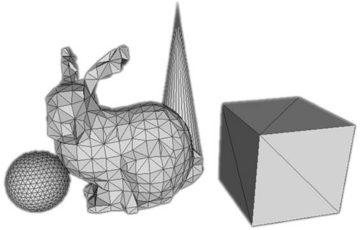
Introduction to Computer Graphics

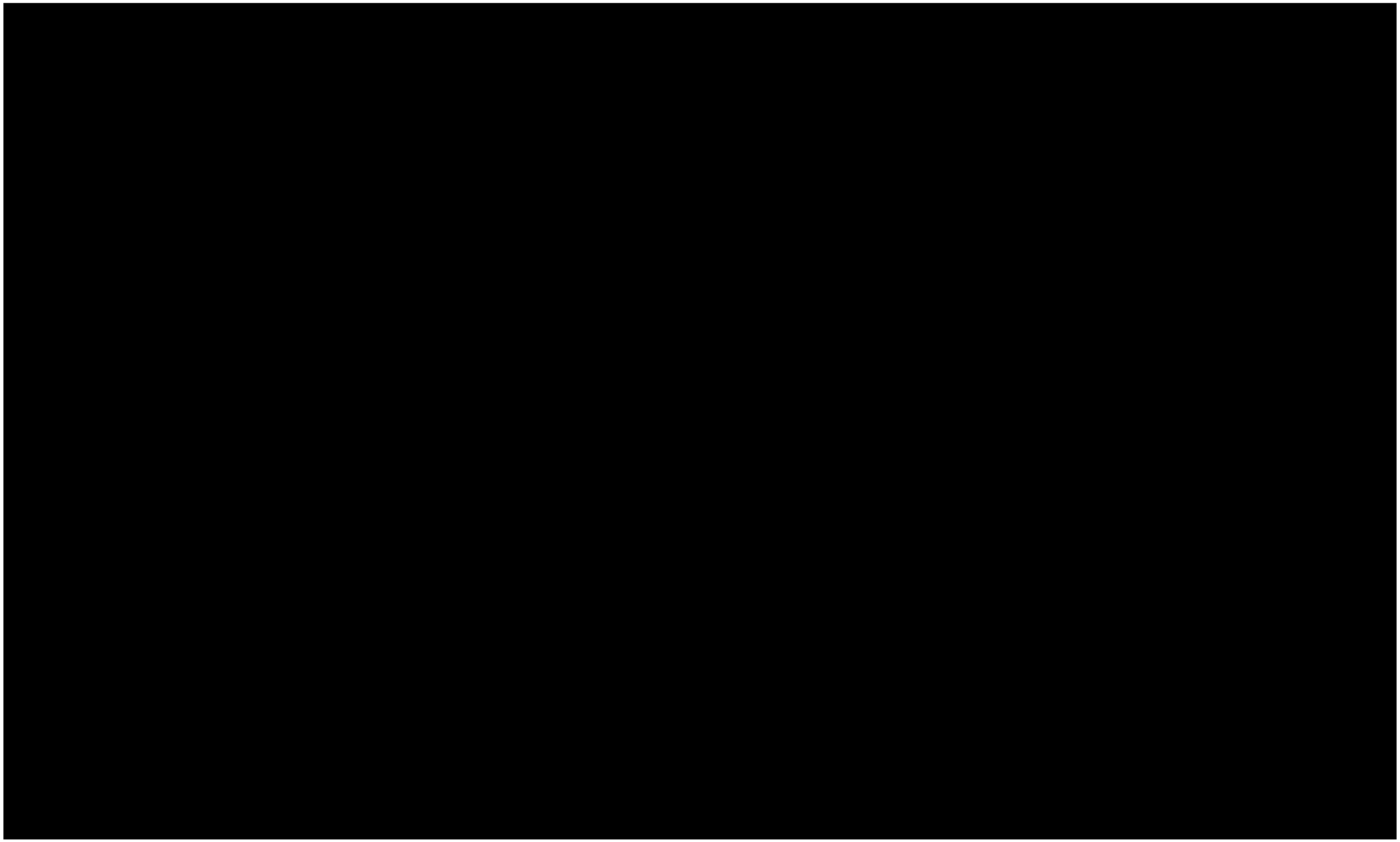


Animation
动画

第九章 动画

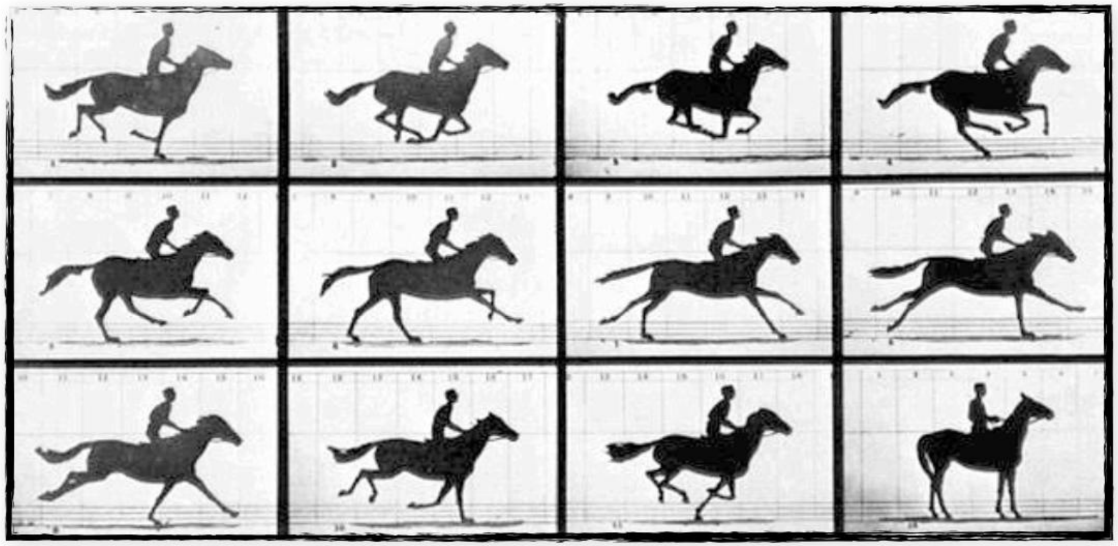
- 动画的概念
- 关键帧插值
- Blend Shape
- LBS (Linear Blend Skinning)
  - Rigging
  - Skinning
- 动作捕捉（Motion Capture）
  - 前向运动学/反向运动学
- 仿真与模拟
  - 质点-弹簧系统

Animation



动画的概念

- 随时间改变的形状->关于时间的形状函数
- 一组快速播放的图像，使人感觉图像里的物体是运动的
- 帧速要求
  - 电影24fps
  - 视频30fps
  - VR眼镜 90fps

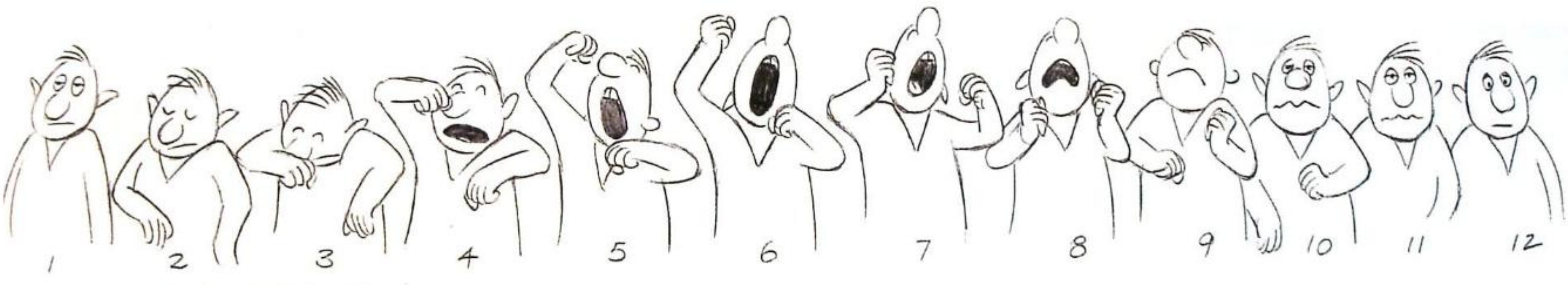


动画的概念

https://www.youtube.com/user/etoilec1

关键帧插值

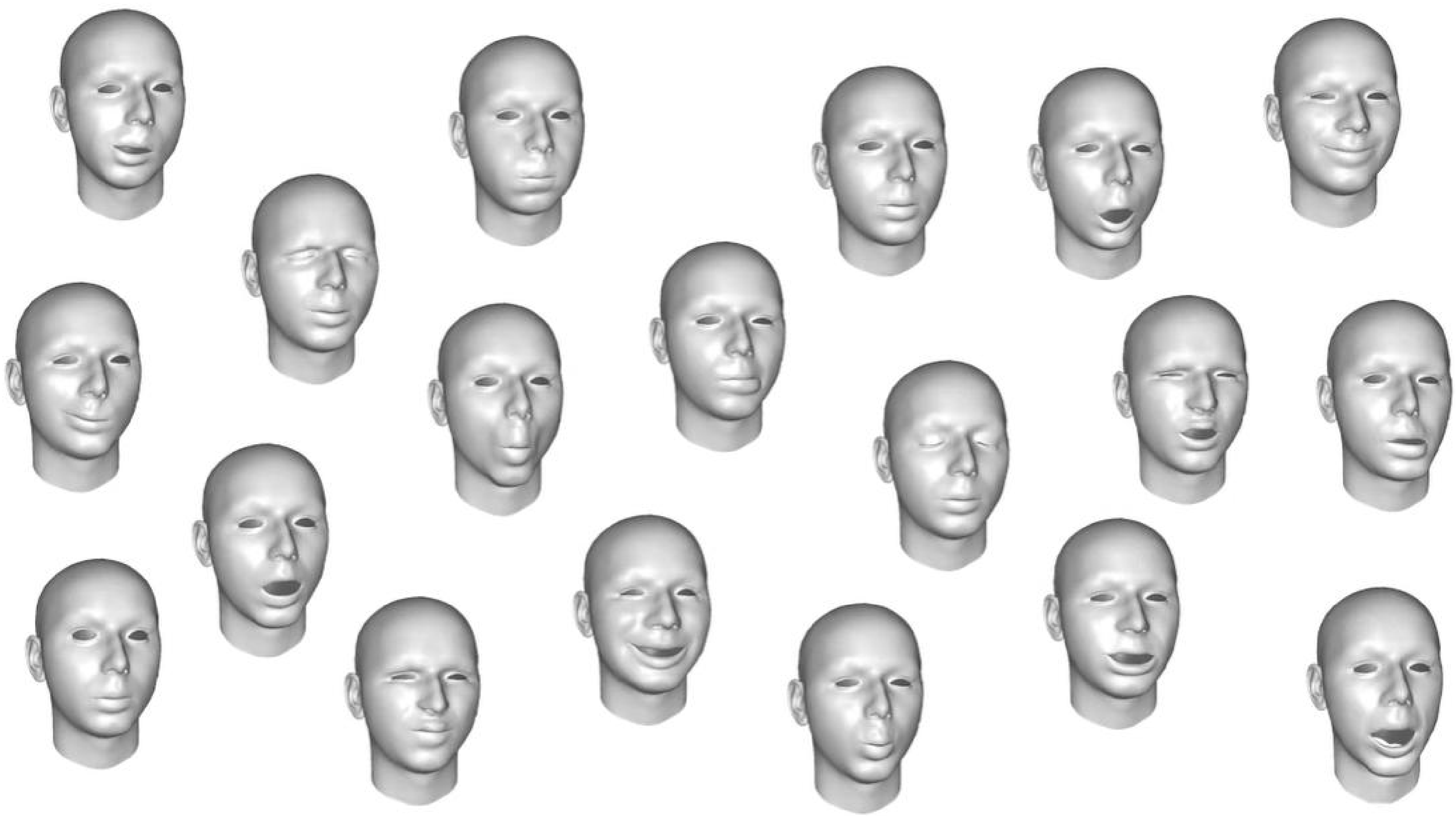
- 关键帧制作
  - 关键帧的制作耗时费力



- 关键帧插值
  - 线性插值
  - 非线性插值

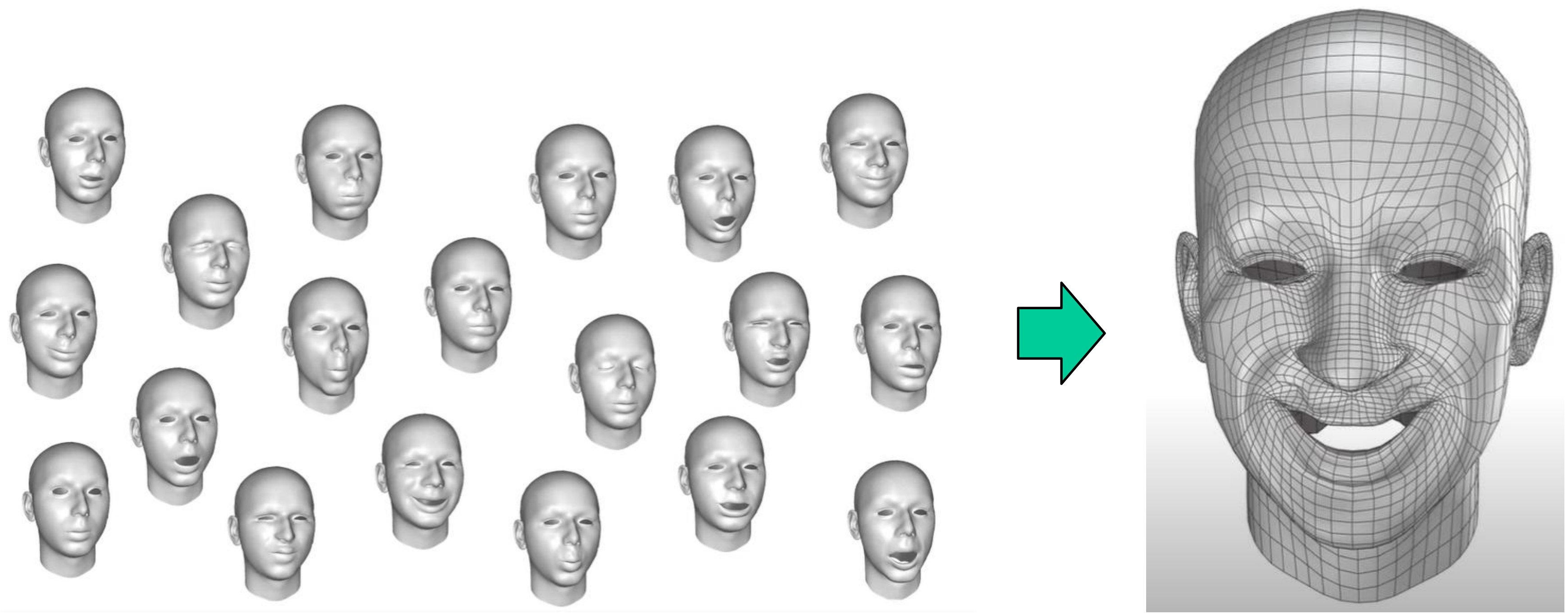
BlendShape

- 定义一组关键形状



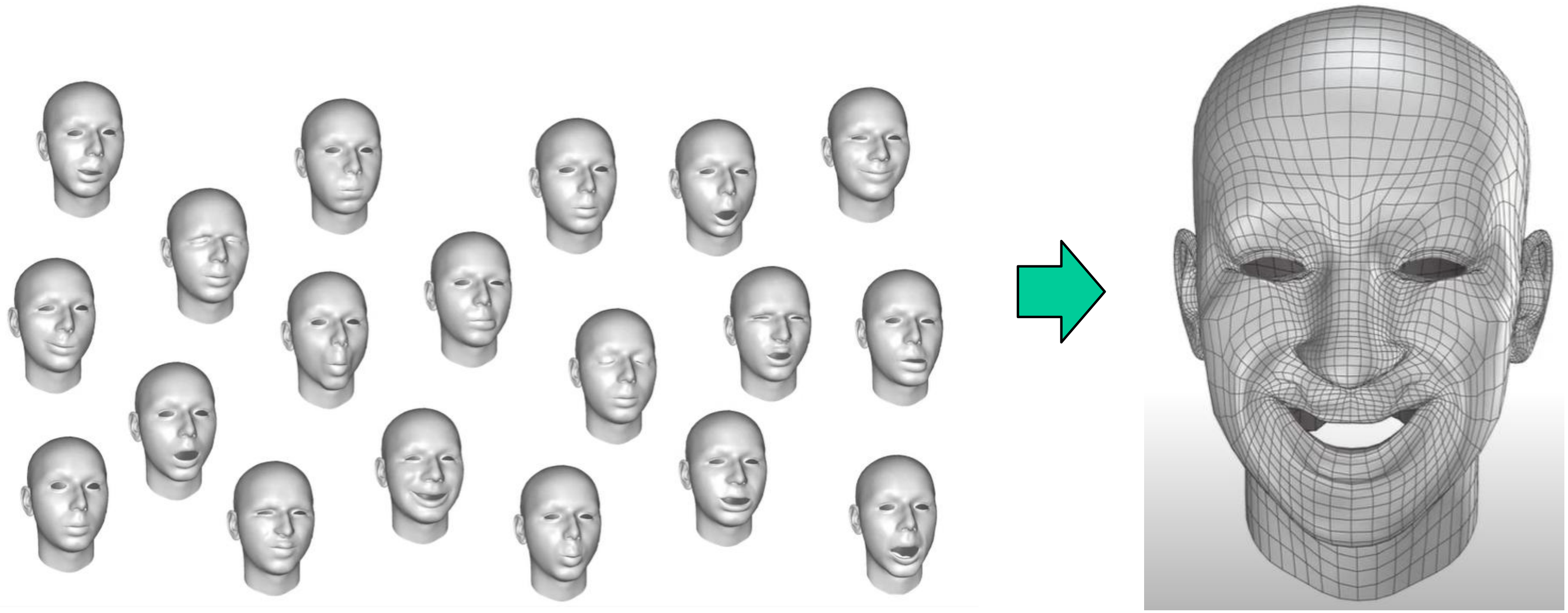
BlendShape

- 通过线性组合得到新的人脸表情



BlendShape

- 通过线性组合得到新的人脸表情
- F=F0+w1*(F1-F0)+w2*(F2-F0)+…

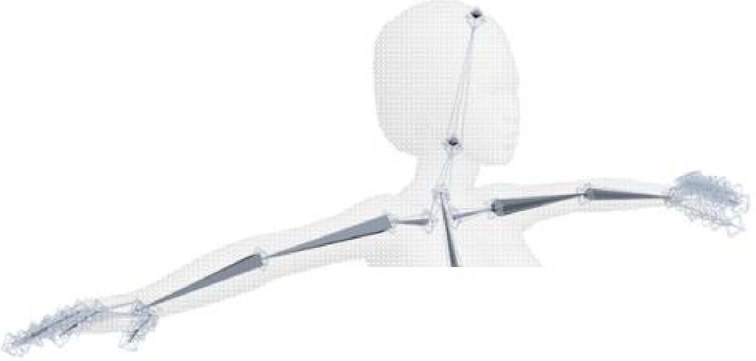


LBS (Linear Blend Skinning)

- 某一个点受不同变换的影响，用线性加权组合而成

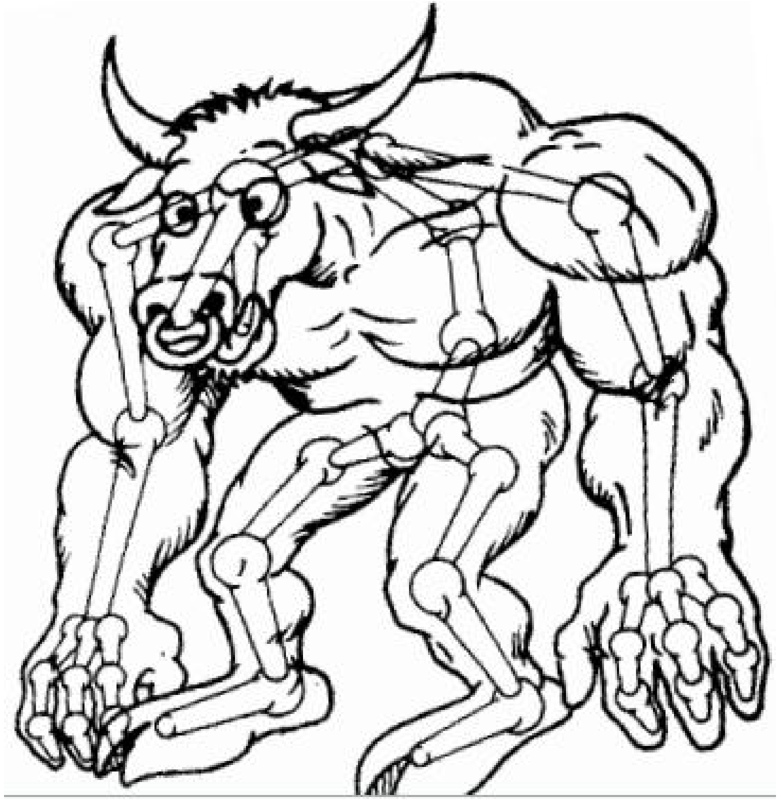
$$\mathbf{q}_i = \sum_j w_{ij} M_j \mathbf{p}_i$$

- 骨架控制形体变换



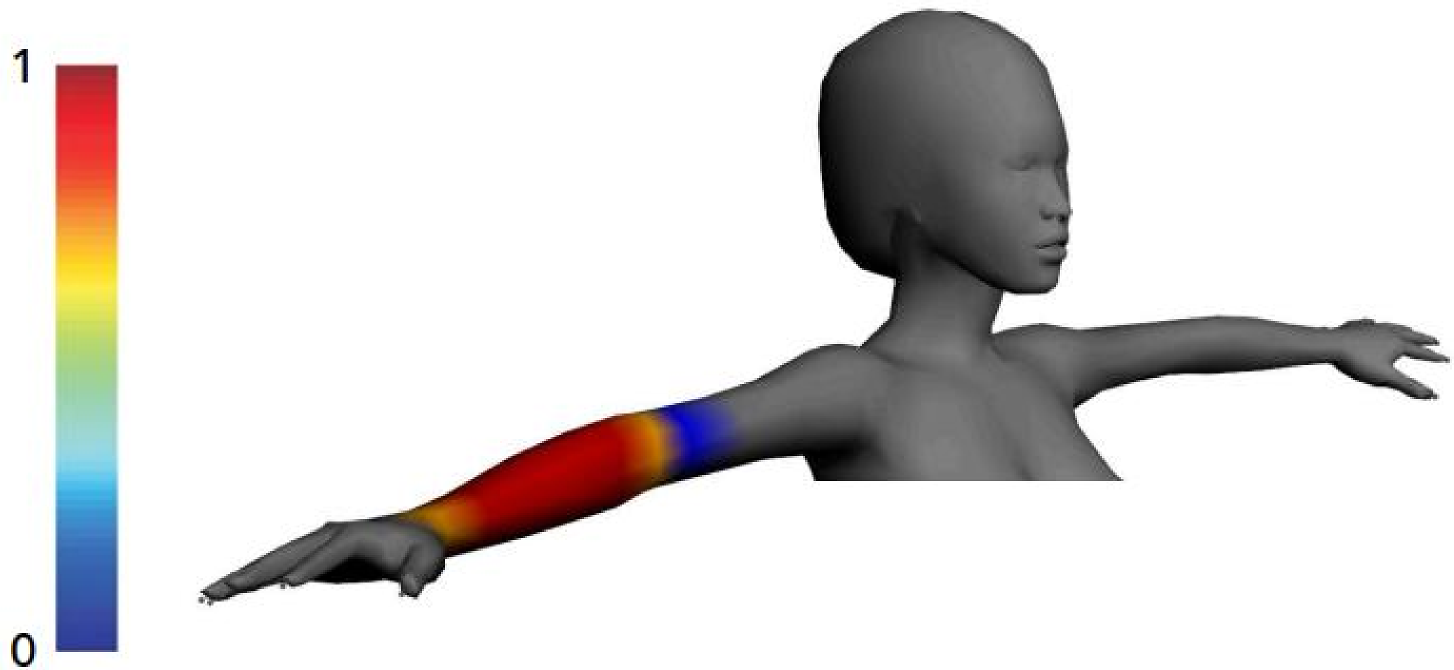
LBS (Linear Blend Skinning)

- Rigging：为动画设置必要控制元素的总称
  - 绑定骨架
  - 设置关键帧
  - 设置blendshape



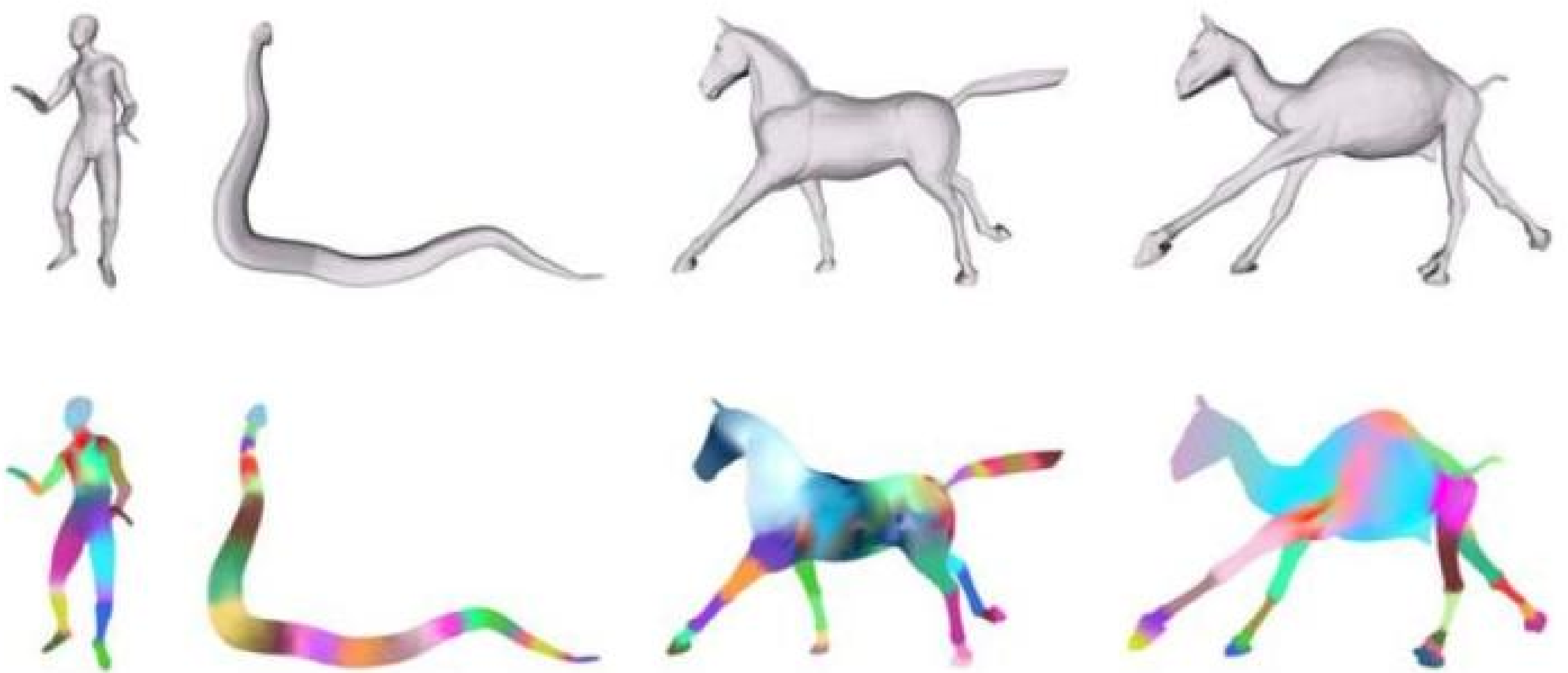
LBS (Linear Blend Skinning)

- Skinning（蒙皮）



LBS (Linear Blend Skinning)

- Skinning（蒙皮）



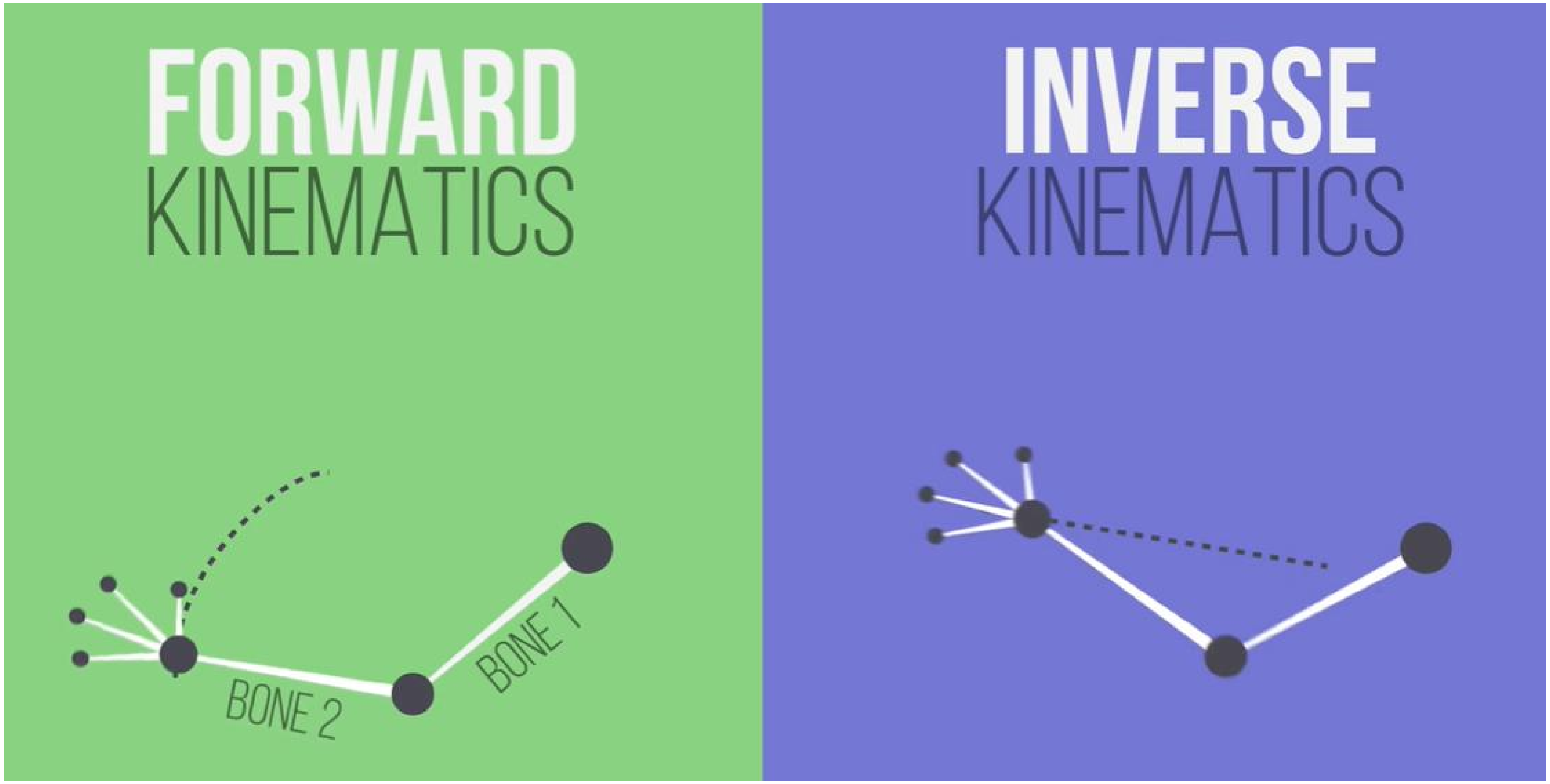
动作捕捉（Motion Capture）

- 用于获取人体运动姿态
  - 光学系统
  - 惯性系统
  - 机械系统
  - 磁系统

动作捕捉（Motion Capture）

动作捕捉（Motion Capture）

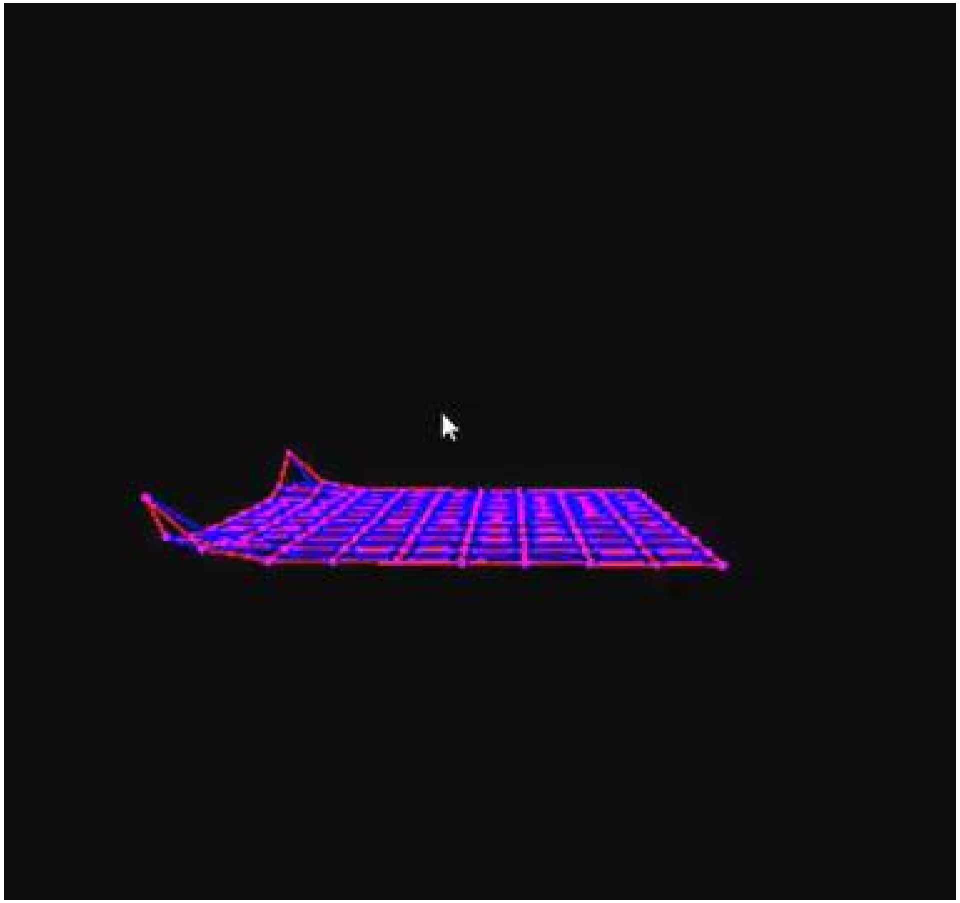
- 前向运动学(FK)/反向运动学(IK)



FK：已知姿态，求端点位置　　　　IK：已知端点位置，求姿态

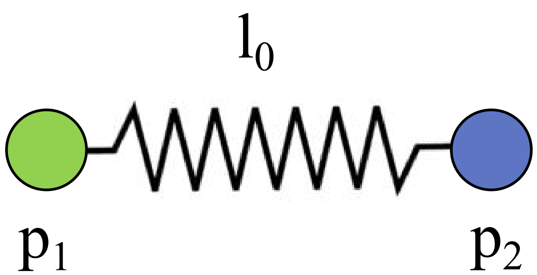
仿真与模拟

- 质点弹簧系统-柔性物质的模拟



仿真与模拟

- 质点弹簧系统-柔性物质的模拟

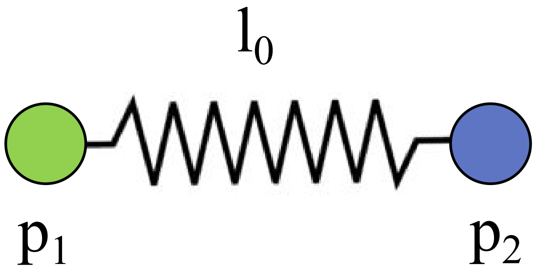


$$f_1 = -\left[k_s(|p_1 - p_2| - l_0) + k_d\left(\frac{(v_1 - v_2)\cdot(p_1 - p_2)}{|p_1 - p_2|}\right)\right]\cdot\frac{(p_1 - p_2)}{|p_1 - p_2|}$$

$$f_2 = -f_1$$

仿真与模拟

- 质点弹簧系统-柔性物质的模拟

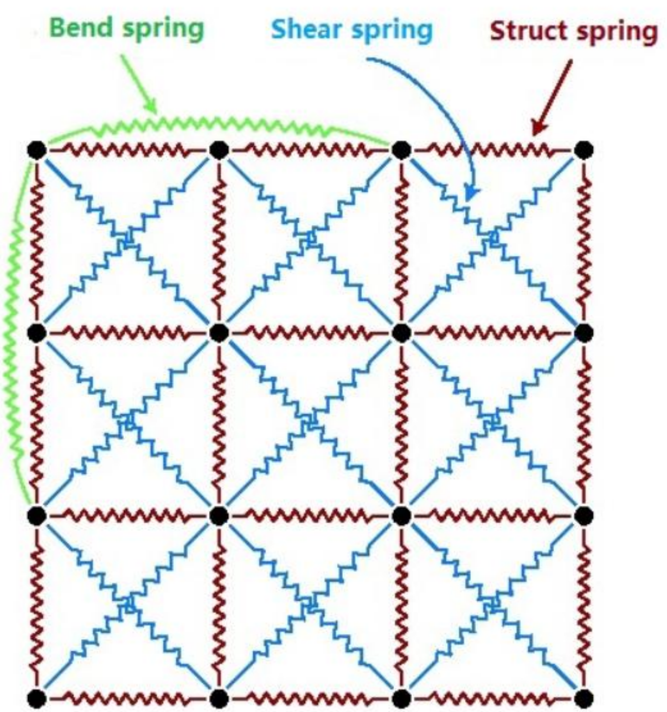


$$a = \frac{f}{m} \qquad \Delta v = at \qquad \Delta x = vt$$

- 已知物体1质量为m1，物体2质量为m2，给定弹性系数和阻尼系数，以及两个物体各自的位置，求其运动情况？

仿真与模拟

- 构建布料弹簧系统



Types of springs

Structural

[*i*, *j*]—[*i*, *j* + 1]; [*i*, *j*]—[*i* + 1, *j*]

Shear

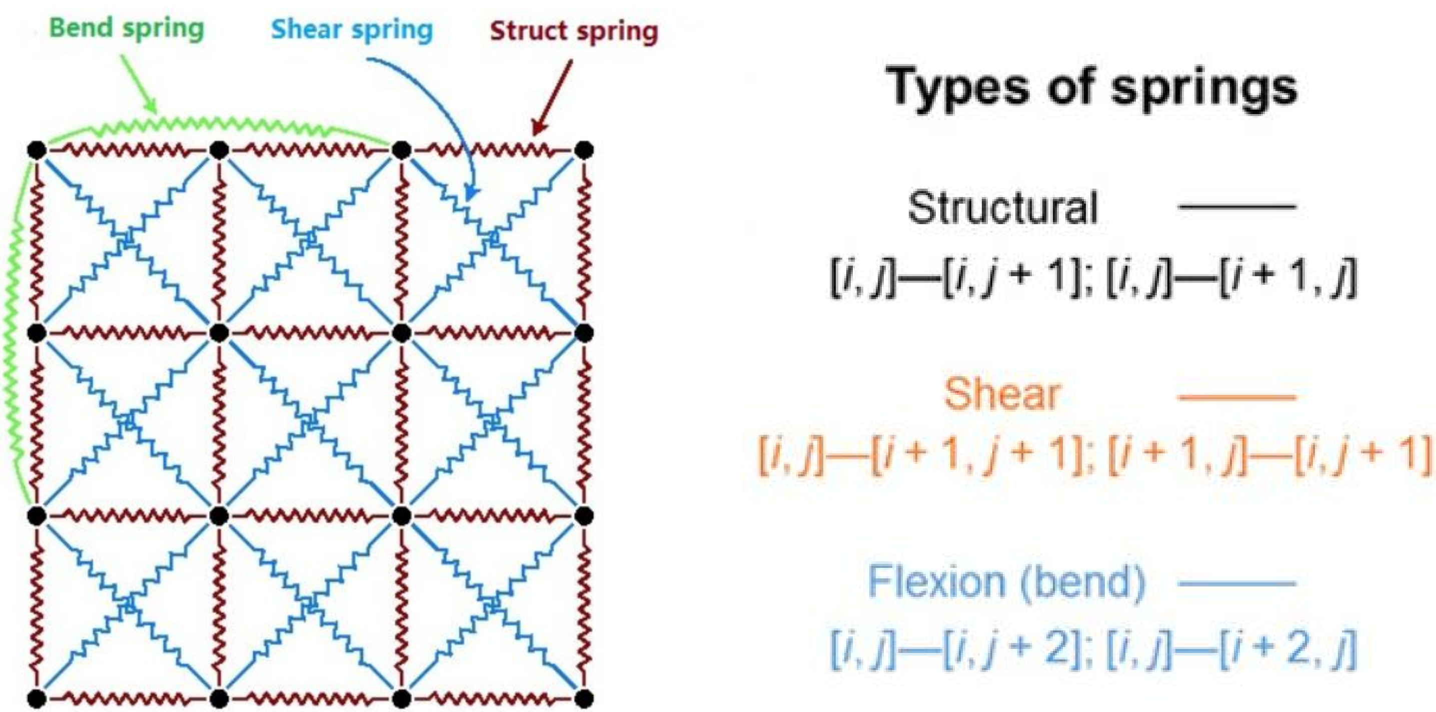
[*i*, *j*]—[*i* + 1, *j* + 1]; [*i* + 1, *j*]—[*i*, *j* + 1]

Flexion (bend)

[*i*, *j*]—[*i*, *j* + 2]; [*i*, *j*]—[*i* + 2, *j*]

仿真与模拟

- 根据各自连接关系，写出各个顶点的受力情况



仿真与模拟

- 在时间上进行位置、受力、速度、加速度更新。

$$v_{n+1} = v_n + a_n \cdot \Delta t$$
$$p_{n+1} = p_n + v_n \cdot \Delta t$$

- 显式迭代。当步长较大时，容易造成数值不稳定。

仿真与模拟

- Runge-Kutta
- 对于时间步长h，其算法如下：

$$\mathbf{v}_1 = \mathbf{v}^t \qquad \mathbf{a}_1 = \mathbf{f}\left(\mathbf{x}^t, \mathbf{v}^t\right)/m$$

$$\mathbf{v}_2 = \mathbf{v}^t + \frac{h}{2}\mathbf{a}_1 \qquad \mathbf{a}_2 = \mathbf{f}\left(\mathbf{x}^t + \frac{h}{2}\mathbf{v}_1, \mathbf{v}^t + \frac{h}{2}\mathbf{a}_1\right)/m$$

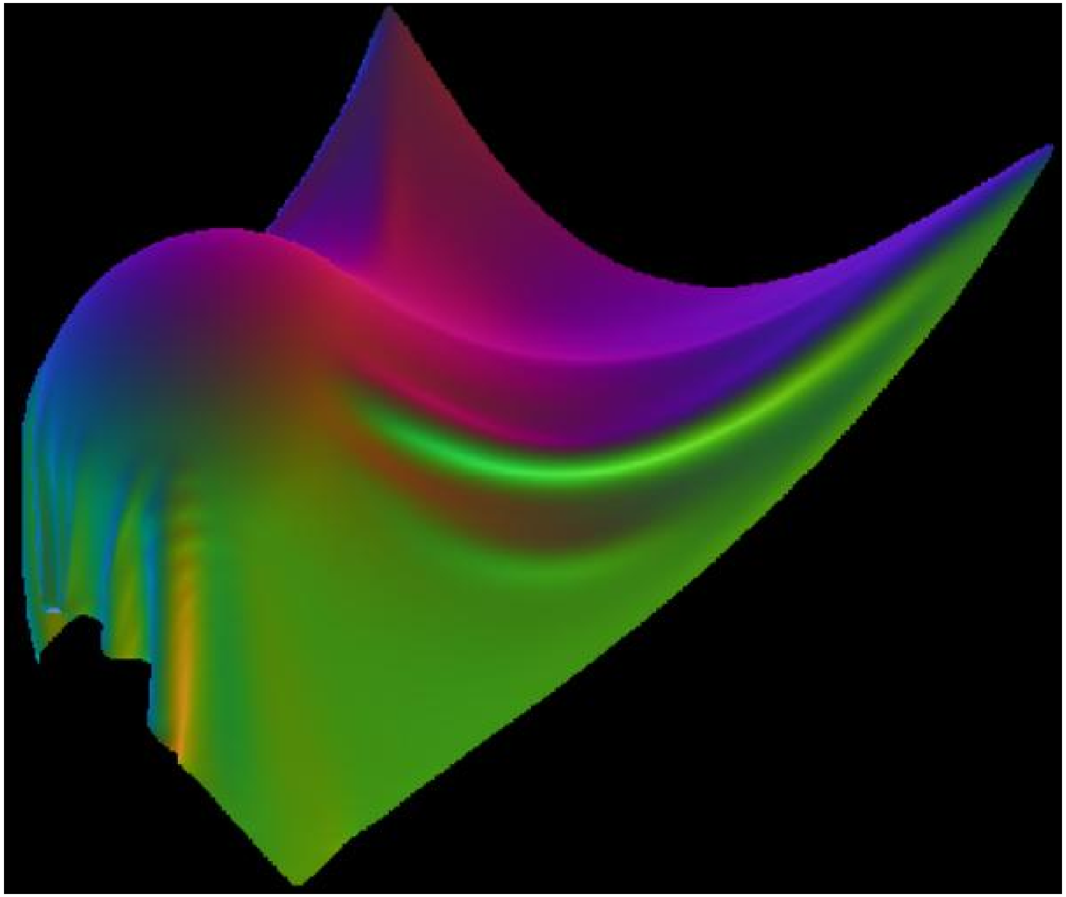
$$\mathbf{v}_3 = \mathbf{v}^t + \frac{h}{2}\mathbf{a}_2 \qquad \mathbf{a}_3 = \mathbf{f}\left(\mathbf{x}^t + \frac{h}{2}\mathbf{v}_2, \mathbf{v}^t + \frac{h}{2}\mathbf{a}_2\right)/m$$

$$\mathbf{v}_4 = \mathbf{v}^t + h\mathbf{a}_3 \qquad \mathbf{a}_4 = \mathbf{f}\left(\mathbf{x}^t + h\mathbf{v}_3, \mathbf{v}^t + h\mathbf{a}_3\right)/m$$

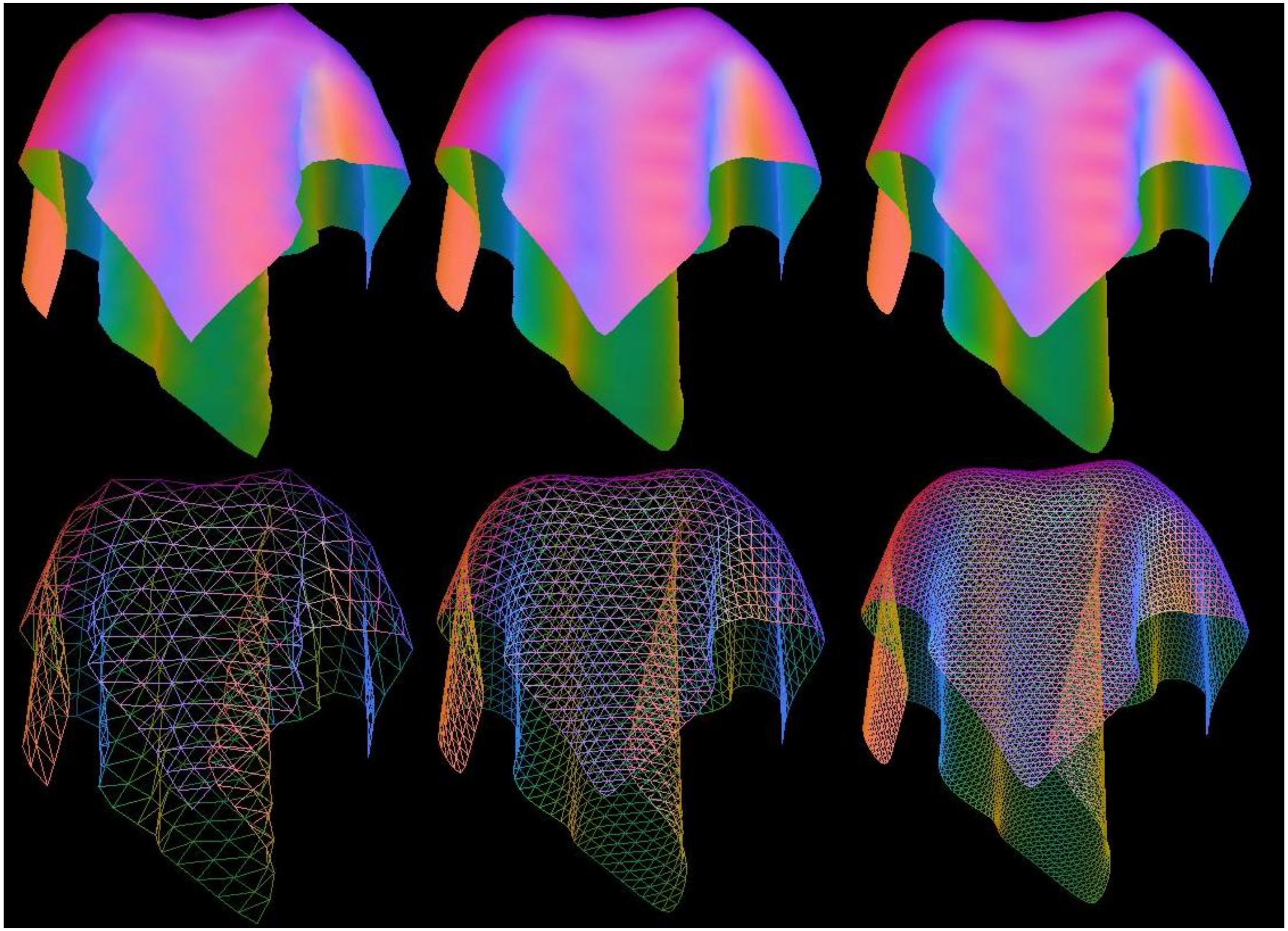
$$\mathbf{x}^{t+1} = \mathbf{x}^t + \frac{h}{6}\left(\mathbf{v}_1 + 2\mathbf{v}_2 + 2\mathbf{v}_3 + \mathbf{v}_4\right) \qquad \mathbf{v}^{t+1} = \mathbf{v}^t + \frac{h}{6}\left(\mathbf{a}_1 + 2\mathbf{a}_2 + 2\mathbf{a}_3 + \mathbf{a}_4\right)$$

引入外力

- 重力、风力
- 碰撞



引入外力



布料模拟效果

Fast Simulation of Mass-Spring Systems

Tiantian Liu[1]
Adam W. Bargteil[2]
James F. O'Brien[3]
Ladislav Kavan[1]

[1]University of Pennsylvania
[2]University of Utah
[3]University of California, Berkeley