

基于StyleGan的肖像Morphing 和二维多边形形状渐变

金小剛

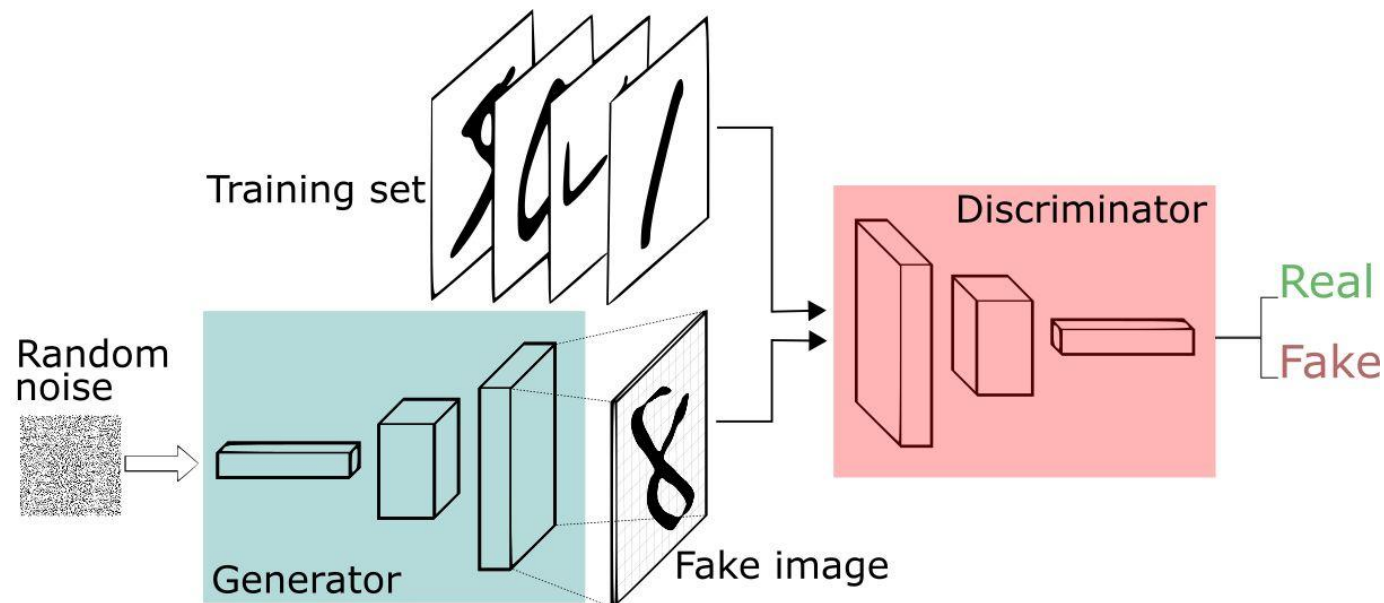
Email: jin@cad.zju.edu.cn

浙江大学CAD&CG国家重点实验室

紫金港校区蒙民伟楼512

基于StyleGan的肖像Morphing

- 2014年，Goodfellow提出了GAN(Generative Adversarial Network)



- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. Generative adversarial networks. *Communications of the ACM*, 2020, 63(11), pp.139-144. (The original version of this paper is entitled “Generative Adversarial Networks” and was published in *Advances in Neural Information Processing Systems* 27 (NIPS 2014).) (截止2026年3月23日, google scholar引用92,614次)

Generative Adversarial Networks

- 2019年3月，ACM宣布将图灵奖授予深度学习三巨头 Geoffrey Hinton、Yoshua Bengio 和 Yann LeCun，以表彰三人「在深度神经网络概念和工程上的突破」。
- Yann LeCun评价GAN为“adversarial training is the coolest thing since sliced bread”。



2014



2015



2016



2017

An illustration of progress in GAN capabilities over the course of approximately three years following the introduction of GANs.

Generative Adversarial Networks

- 以下是使用 GANs 在 MNIST 和 CELEBA-HQ (https://github.com/tkarras/progressive_growing_of_gans) 两个数据集上训练后获得的结果示例。



GAN基本思想

- GAN全称是**生成对抗网络**，顾名思义是一种生成模型。它是一种**无监督学习方法**，通过让**两个神经网络相互博弈**的方式进行学习。
- 生成式对抗网络由**生成器(Generator, G)**和**判别器(Discriminator, D)**构成。生成对抗网络的核心目的是**训练生成器**。生成器的目的是生成与真实样本尽可能相似的“假样本”，判别器的目的是尽可能区分出给定样本是真实样本还是生成的“假样本”。二者目的相悖，在**不断博弈**的过程中相互提高，最终在判别器判别能力足够可靠的前提下仍无法区分给定样本是真实样本还是生成样本，从而我们说生成器能够生成**“以假乱真”**的样本。

GAN 结构概览

- 在开始训练时，生成器会生成明显的虚假数据，而判别器会快速学会辨别数据是虚假的：

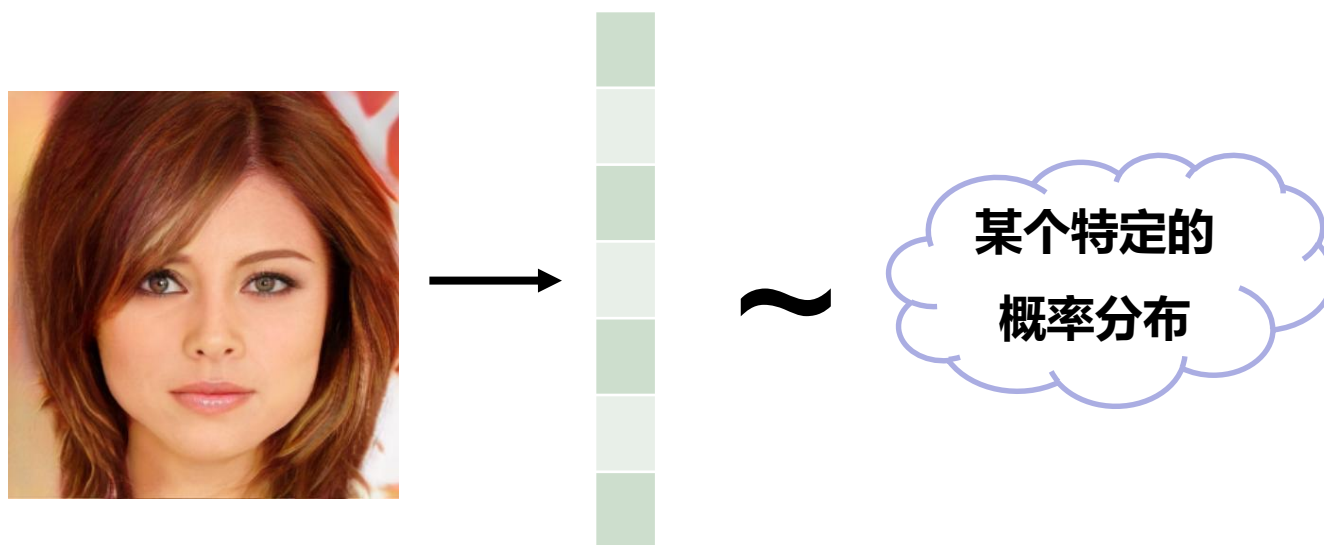


- 随着训练的进行，生成器越来越接近能够分辨判别器的输出：



Generative Adversarial Networks for Faces

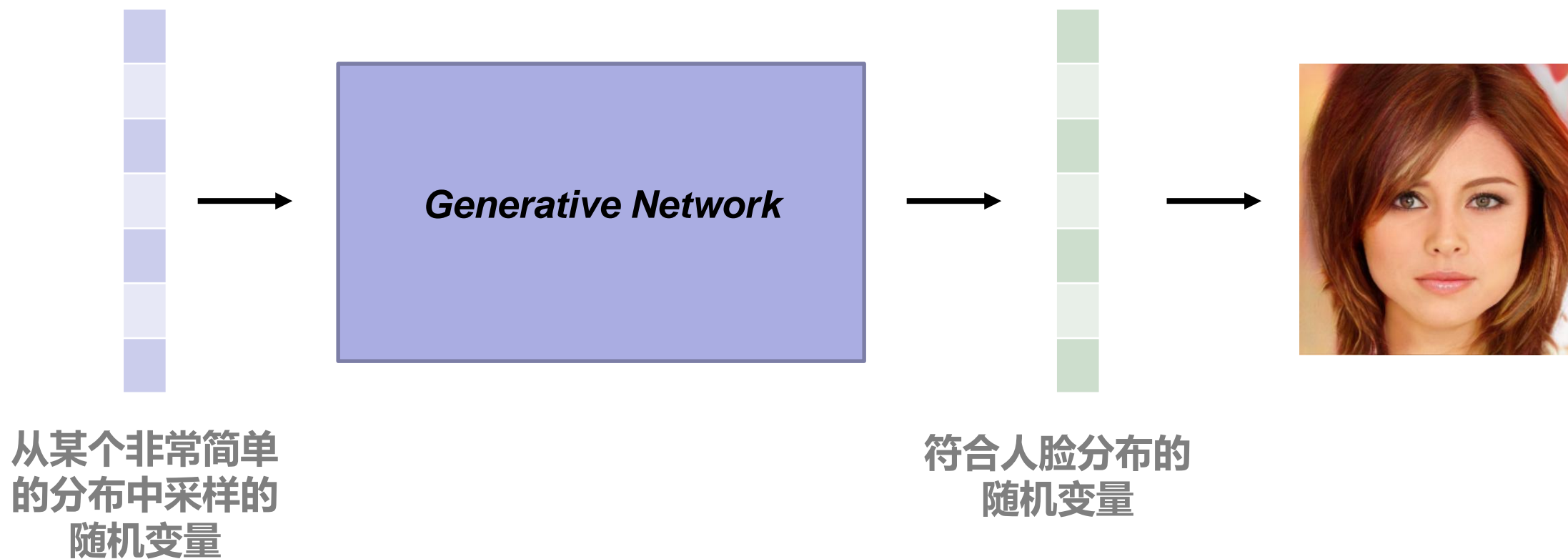
- **目标:** 针对特定的某个复杂概率分布, 生成随机变量。
- 例如, 我们希望能够生成分辨率为 $n \times n$ 的人脸图像, 每张图像可以视为一个 $N = n \times n \times 3$ 维向量 (r 、 g 、 b 三个分量)。
- 但是, 不是每个 N 维随机向量都能表示一张人脸。能表示合理人脸的 N 维向量符合 N 维向量空间上某个特定的概率分布。



Generative Adversarial Networks

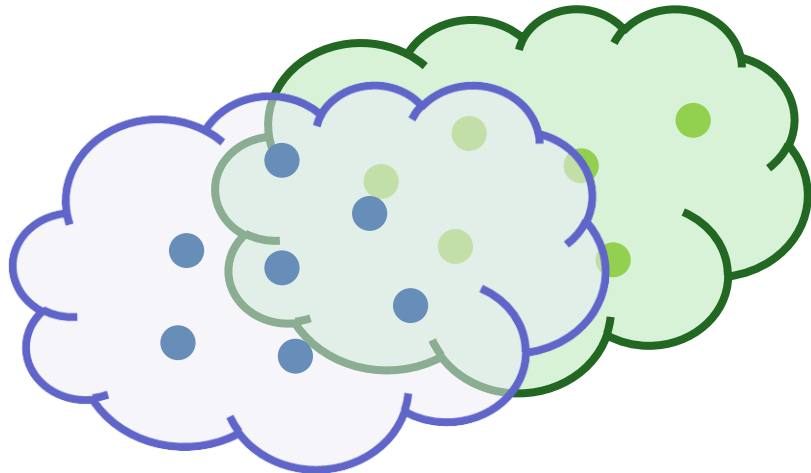
- **How to generate complex random variables?**
- 向量空间上的“人脸概率分布”非常复杂，无法进行显式地描述。
- 可以将符合人脸概率分布的复杂随机变量表示为一个**复杂的函数**应用于**简单的随机变量**的结果。
 - **complex random face variables = function(simple random variables)**
- 通过一个神经网络对这个复杂的函数进行建模，神经网络将一个**简单的随机变量**作为输入，并输出遵循“人脸概率分布”的随机变量。这个**简单的随机变量**被称为**隐码(latent code)**，隐码所在的空间称为**隐空间(latent space)**。

Generative Adversarial Networks



Generative Adversarial Networks

- **How to train the Generative Network?**
- 将**生成的分布与目标分布进行比较**，来训练生成网络。
- 但是，我们不知道如何显式地表达真正世界中的“人脸概率分布”，因为这个分布过于复杂。

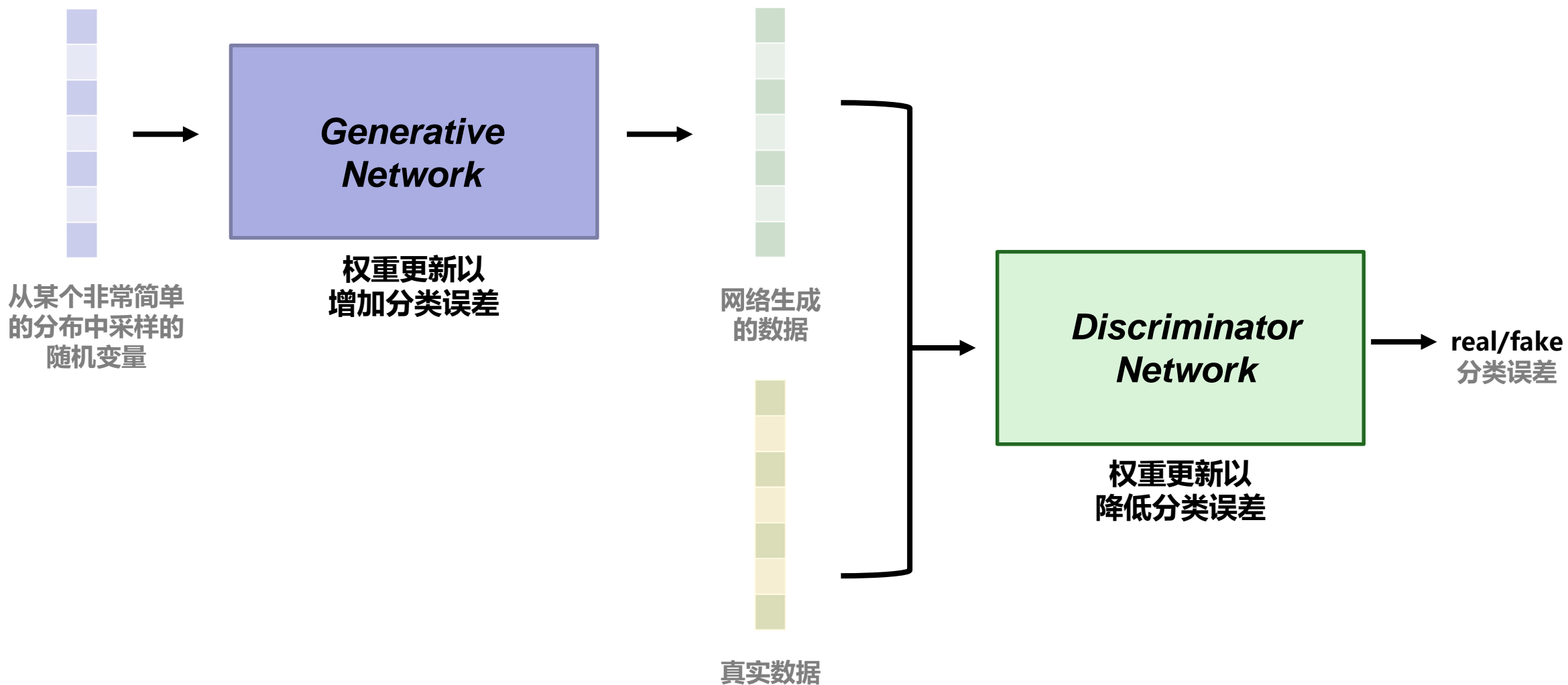


怎么比较**生成的概率分布**和**真实的“人脸概率分布”**？

Generative Adversarial Networks

- **Adversarial Training !**
- 用**间接比较**的方法训练生成模型。
- 引入一个**鉴别器**，用于执行真实样本和生成样本之间的区分任务。
- **鉴别器**：对真实和生成的数据进行分类，目标是**最小化**真实数据和生成数据之间的分类误差。
- **生成器**：生成尽可能真实的数据，用来欺骗鉴别器，目标是**最大化**真实数据和生成数据之间的分类误差。
- 在生成对抗网络中，生成器和鉴别器两个网络都试图击败对方，他们之间的竞争使这两个网络在各自的目标上“进步”。

Generative Adversarial Networks

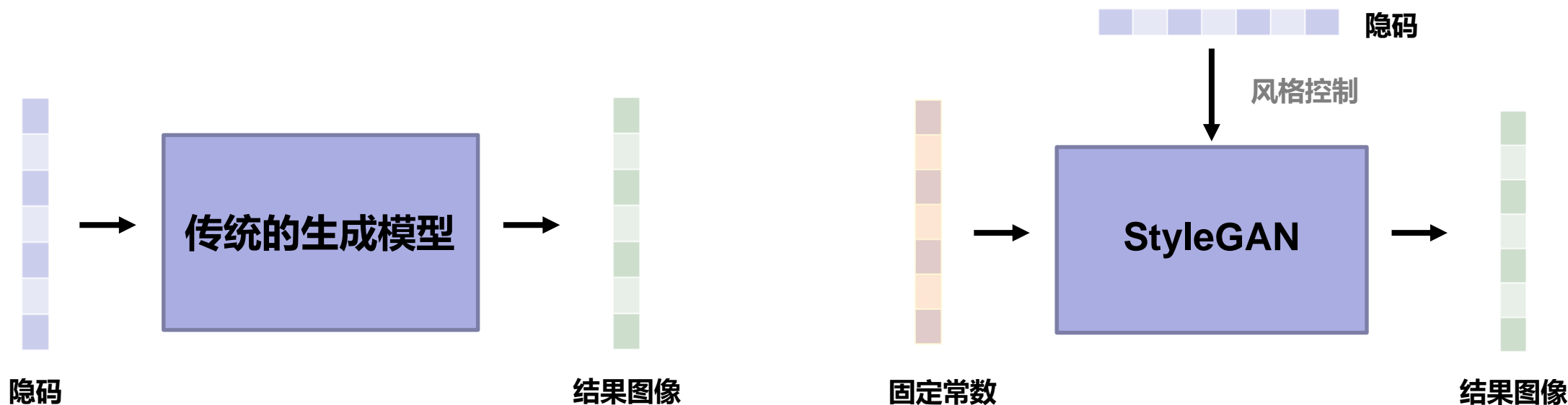


StyleGAN

- Tero Karras, Samuli Laine, and Timo Aila. 2019. **A style-based generator architecture for generative adversarial networks.** In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). 4401–4410.
- 一种引入了风格迁移的生成对抗网络变体。
- **StyleGAN的两大优点决定了其非常优越的图像编辑能力:**
 - 生成图像的质量非常高。
 - 隐空间性质优良，有良好的语义**解耦性质**，将不同level的语义属性进行**分离**，能生成更好的**插值结果**。

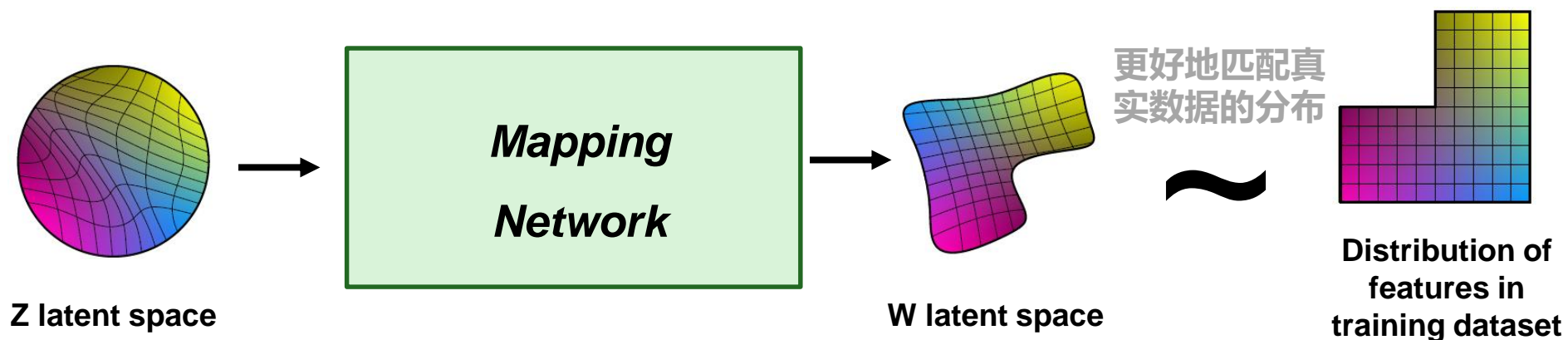
StyleGAN: 基于风格的生成器

- **传统生成模型**的生成器以隐码作为输入，经过卷积等操作后，输出结果图像。
- **StyleGAN**以一个固定常数作为输入，而隐码用于控制生成器中每一层的“风格”。



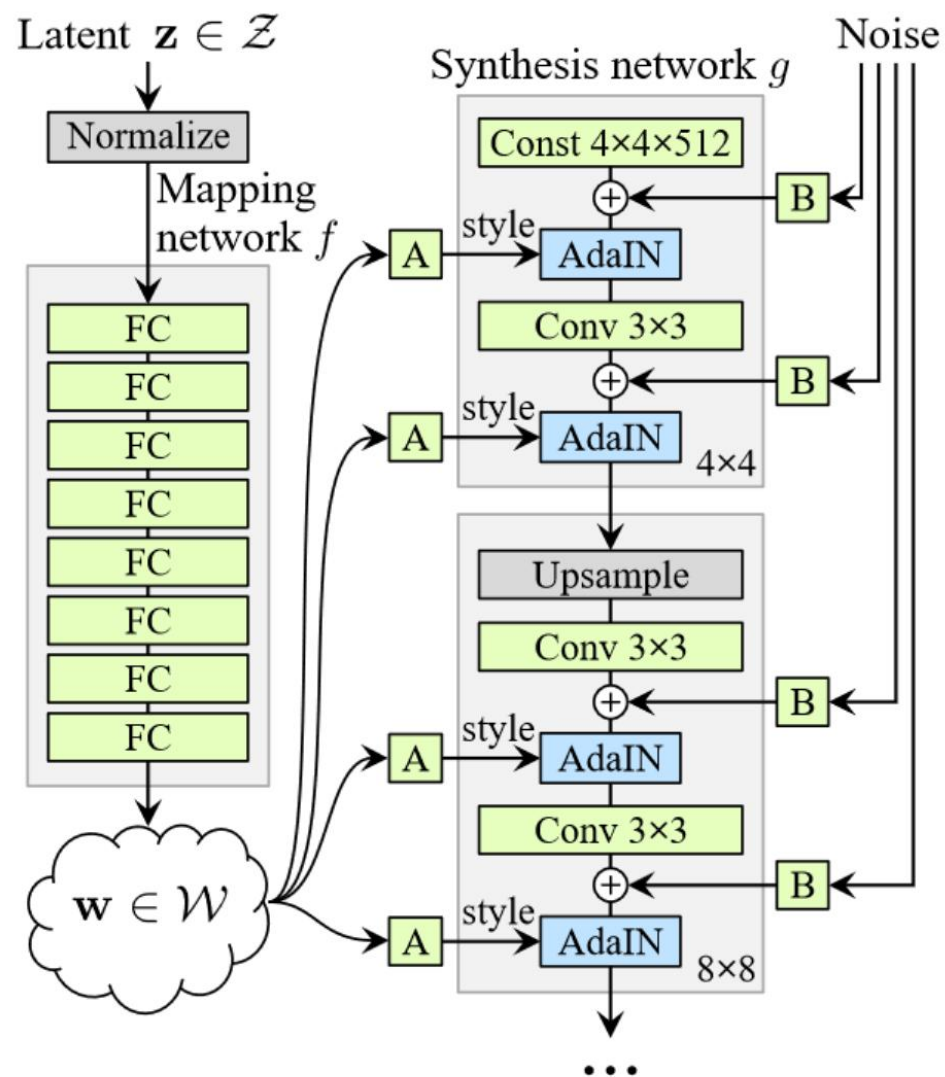
StyleGAN的隐空间

- StyleGAN有两种主要的隐空间：Z隐空间与W隐空间
- **Z隐空间**：一个符合标准正态分布的隐空间。
- **W隐空间**：从Z隐空间中随机采样的z隐码被输入映射网络(mapping network)，得到的w隐码分布于W隐空间中。W隐空间的分布是经过映射网络学习得到的，因此与Z隐空间相比，可以更好地匹配真实数据的分布，并且有更好的**解纠缠** (Disentanglement) 性质。



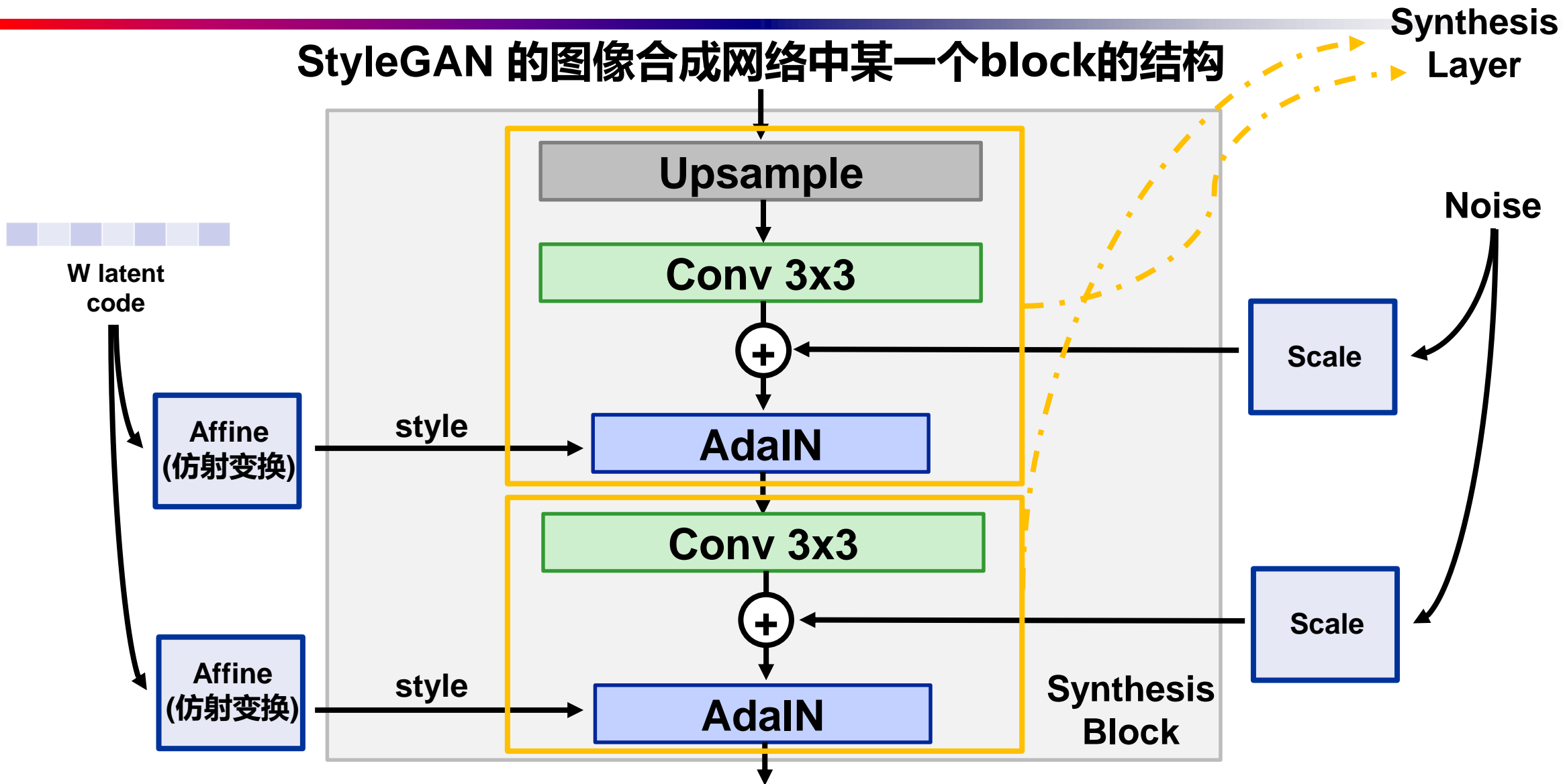
StyleGAN生成器中的风格控制

- 那么，StyleGAN是如何**生成图像**的？
- **StyleGAN利用隐码进行风格控制，隐码通过控制图像的风格决定最终的生成结果。**
- 从 Z 隐空间中随机采样的 z 隐码，通过映射网络后转化为 w 隐码。
- w 隐码通过**仿射变换**后再输入每一层图像合成网络进行风格控制。这种仿射变换是在图像合成网络的训练过程中学习的。



StyleGAN生成器中的风格控制

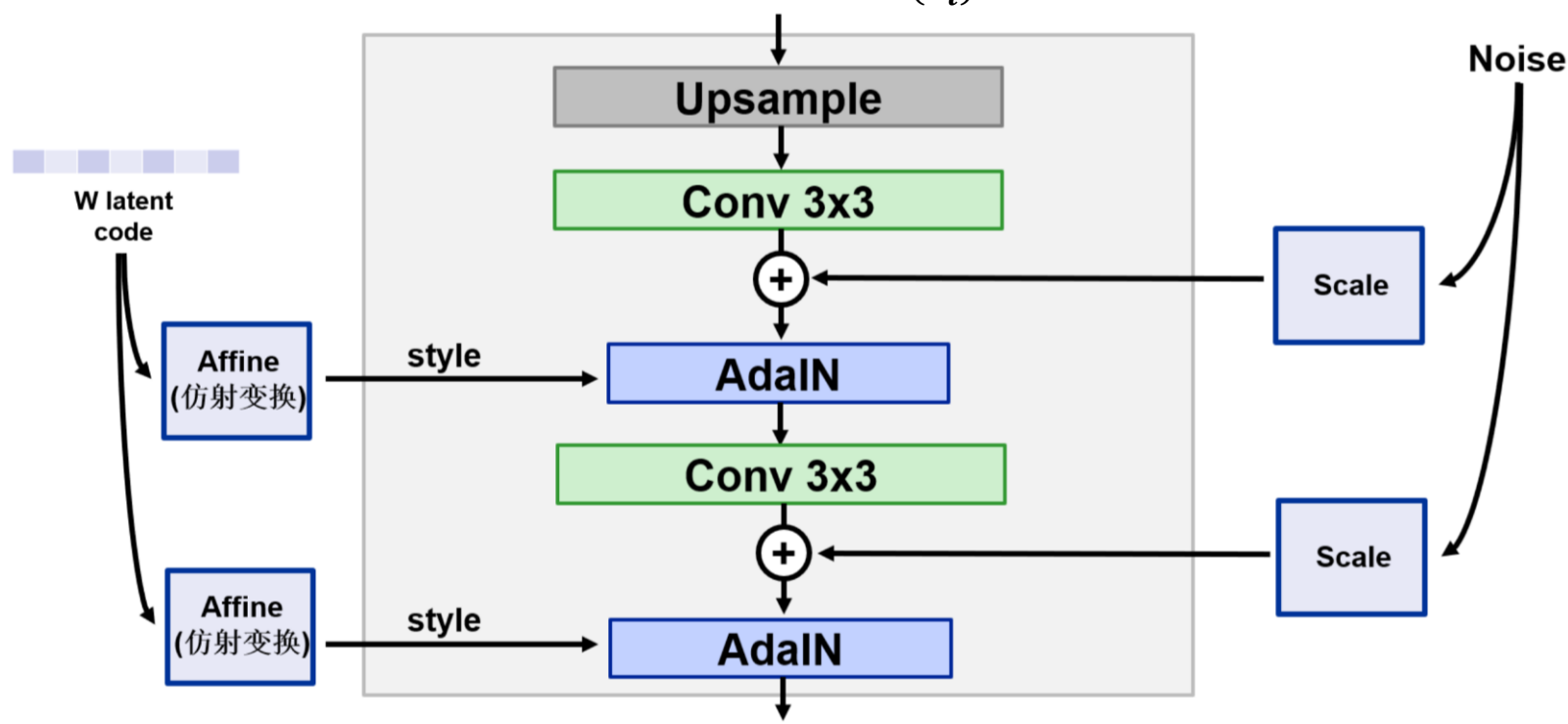
StyleGAN 的图像合成网络中某一个block的结构



StyleGAN生成器中的风格控制

- w 隐码经过仿射变换(Affine)以后生成风格(style), 在AdaIN模块 (实现风格迁移和风格混合) 中, 输入的特征图 x_i 在归一化后, 用style进行缩放和偏置, 从而实现特征图的风格控制。 $Styles = (y_s, y_b) = \text{Affine}(w)$

$$\text{AdaIN}(x_i, Styles) = y_{s,i} \frac{x_i - \mu(x_i)}{\sigma(x_i)} + y_{b,i}$$



StyleGAN生成器中的风格控制

- 而在**StyleGAN**的图像合成网络中，输入不同**Synthesis Layer**的style控制不同粒度的语义特征，即不同粒度的风格。
- **Coarse styles**: 控制人脸姿态、人脸几何形状等。大约对应网络的1-4层。
- **Middle styles**: 发型、部分人脸形状等。大约对应网络的5-8层。
- **Fine styles**: 肤色、发色、光影、背景等。大约对应网络剩下的层数。
- 因此，通过**混合不同粒度**的style，能产生不同层级的**风格混合**效果。

StyleGAN生成器中的风格混合

- **Coarse styles:** 控制人脸姿态、人脸几何形状等
- 下图为将Source A的coarse styles替换为Source B之后的结果



StyleGAN生成器中的风格控制

- **Middle styles:** 发型、部分人脸形状等
- 下图为将输入Source A的middle styles替换为输入Source B之后的结果



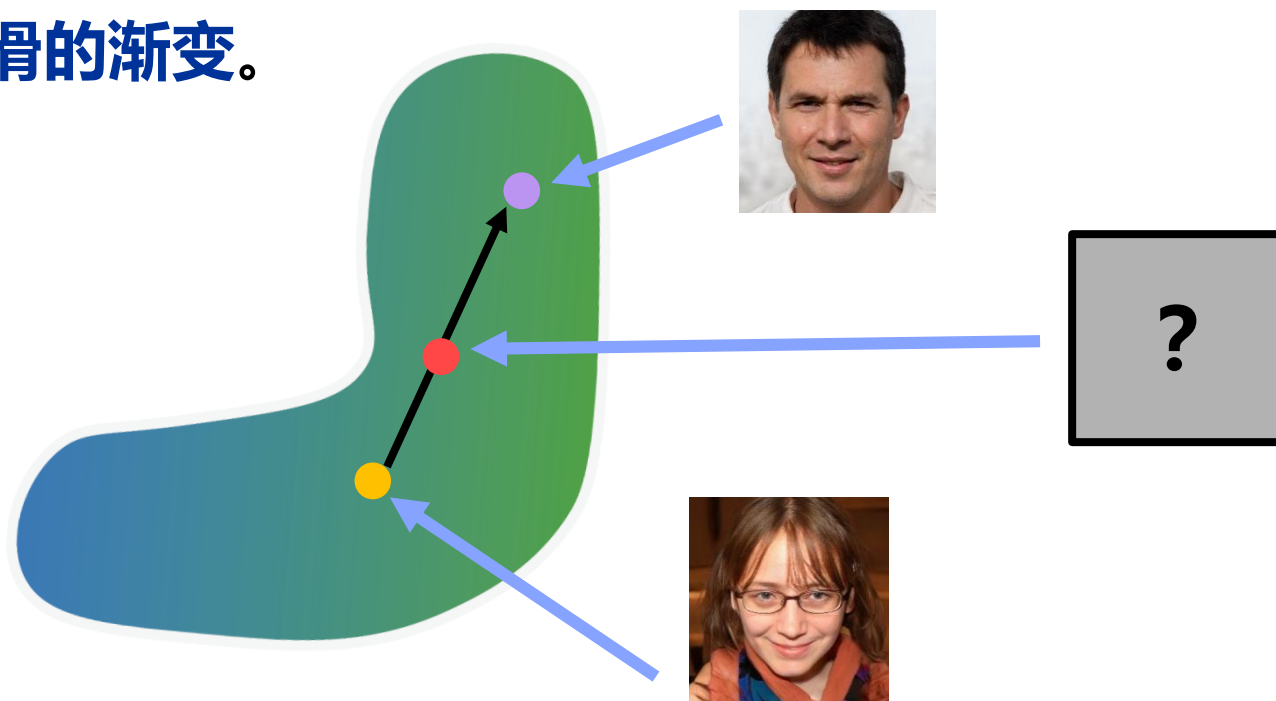
StyleGAN生成器中的风格控制

- **Fine styles:** 肤色、发色、光影、背景等
- 下图为将Source A的fine styles替换为Source B之后的结果



StyleGAN中的图像Morphing

- StyleGAN倾向于学习一个**平滑**的隐空间，即**隐空间的接近区域中采样的隐码能生成相似的图像**。这表明，在隐空间中的连续移动会产生一条**平滑变化的图像路径**，并且每个图像都接近目标域（真实的人脸）。
- 因此，对StyleGAN的两个隐码进行插值，可以在两个人脸图像之间产生非常**自然平滑的渐变**。



StyleGAN中的图像Morphing



StyleGAN中的图像Morphing

- 直接对两个 w 隐码进行混合:



linear interpolation



StyleGAN中的图像Morphing

- 对两个w隐码的coarse styles进行混合:



linear interpolation

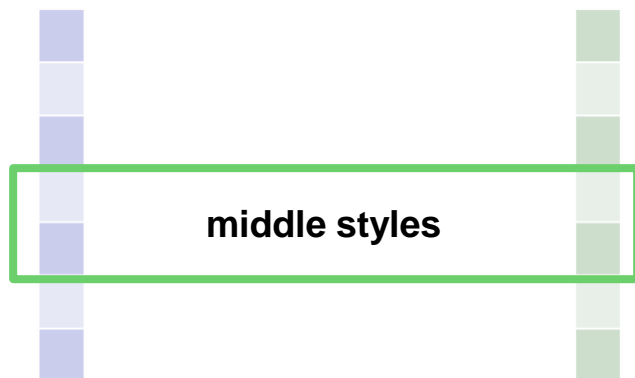


StyleGAN中的图像Morphing

- 对两个w隐码的middle styles进行混合:



linear interpolation

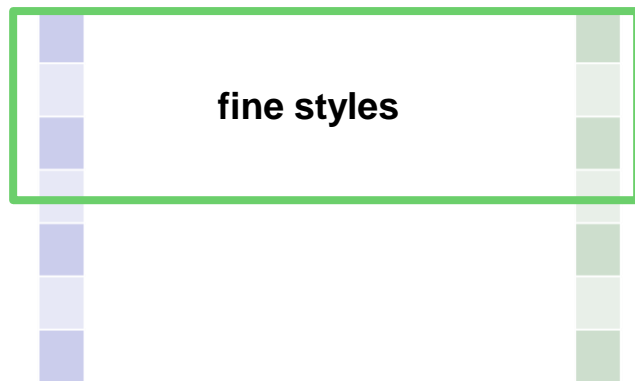


StyleGAN中的图像Morphing

- 对两个w隐码的fine styles进行混合:



linear interpolation



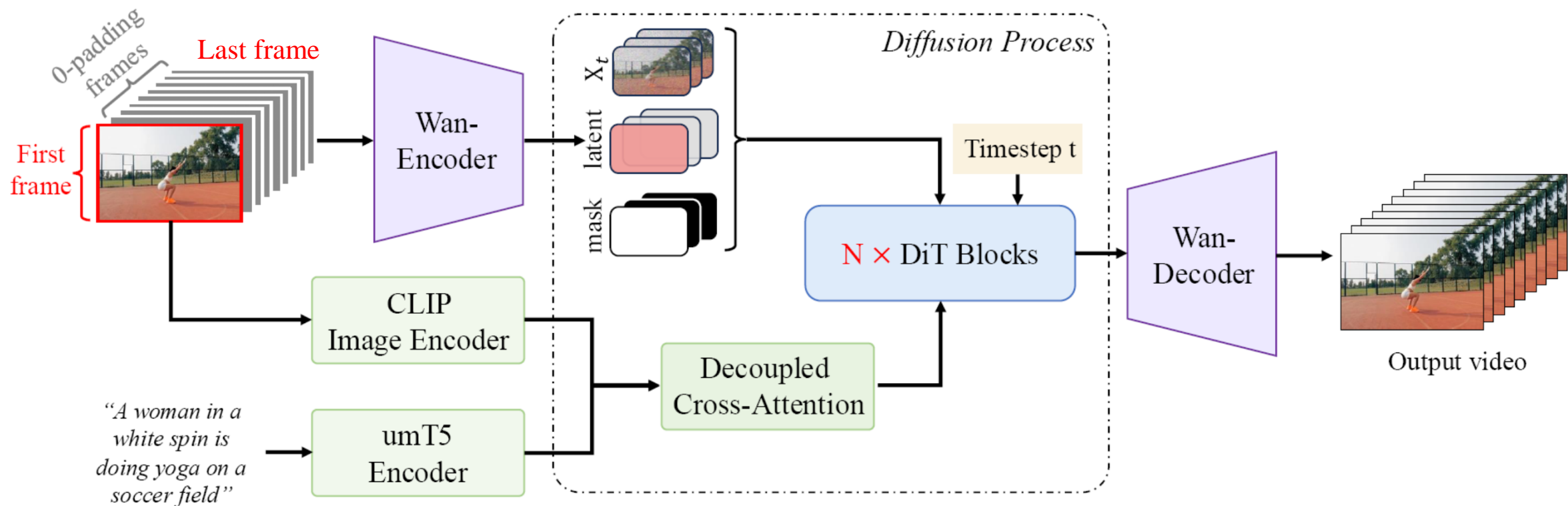
基于Diffusion模型的Morphing



基于视频大模型的Morphing方法——首尾帧生视频模型

- **首尾帧技术**在AI视频生成中扮演着核心控制角色，通过指定**起始帧（首帧）**和**结束帧（尾帧）**，AI自动生成中间过渡帧，实现画面的连贯变化。它可以精准控制视频内容、提升叙事流畅性、高效生成复杂视频。
- 通义万相的视频模型：首尾帧生视频模型是阿里通义万相于2025年4月17日开源的高清视频生成模型，参数量达14B，为业界首个百亿级参数规模的开源首尾帧视频模型。该模型基于Wan2.1文生视频架构，可根据用户指定的首帧与尾帧图像生成5秒时长的720p高清视频，支持写实、卡通、漫画等多种风格，应用于延时摄影、主体特效变化等场景。用户可通过通义万相官网免费体验，或在GitHub、Hugging Face等平台下载模型进行二次开发。
- 首尾帧生视频比文生视频、单图生视频的可控性更高。首尾帧视频生成需同时满足以下几点：
 1. 生成的视频内容要保证和用户输入的两张图像一致；
 2. 能够遵循用户的提示词指令；
 3. 能从给定的首帧自然、流畅过渡到尾帧；
 4. 视频本身动作协调、自然。

首尾帧生视频模型



Github: <https://github.com/Wan-Video/Wan2.1>

HuggingFace: <https://huggingface.co/Wan-AI/Wan2.1-FLF2V-14B-720P>

魔搭社区: <https://www.modelscope.cn/models/Wan-AI/Wan2.1-FLF2V-14B-720P>

直接体验入口: <https://tongyi.aliyun.com/wanxiang/videoCreation>

基于视频大模型的Morphing——狐变

①首帧提示词:

白狐踏着星辰碎片遨游太虚，以深邃靛蓝与珍珠白交织的星尘为底，柔光如薄雾笼罩。融合浅灰蓝、藕荷色、奶油白等淡雅色调，光影过渡自然。飞溅的星芒拖着螺旋状轨迹，仿若深海与宇宙共鸣的神秘韵律，C4D建模，OC渲染。

②尾帧图片:

为了保持背景一致性，可通过即梦图片3.0的智能参考功能生成。

③输入提示词:

"狐狸逐渐变成人"，生成视频。



基于视频大模型的Morphing——烟雾化仙

①首帧提示词:

黑色渐变背景里，一缕白烟向上飘



②尾帧提示词:

通过参考图功能写提示词：把烟换成烟雾构成的汉服女子



③输入提示词:

“烟雾逐渐变成人”，生成视频。



基于视频大模型的Morphing——机甲变摩托

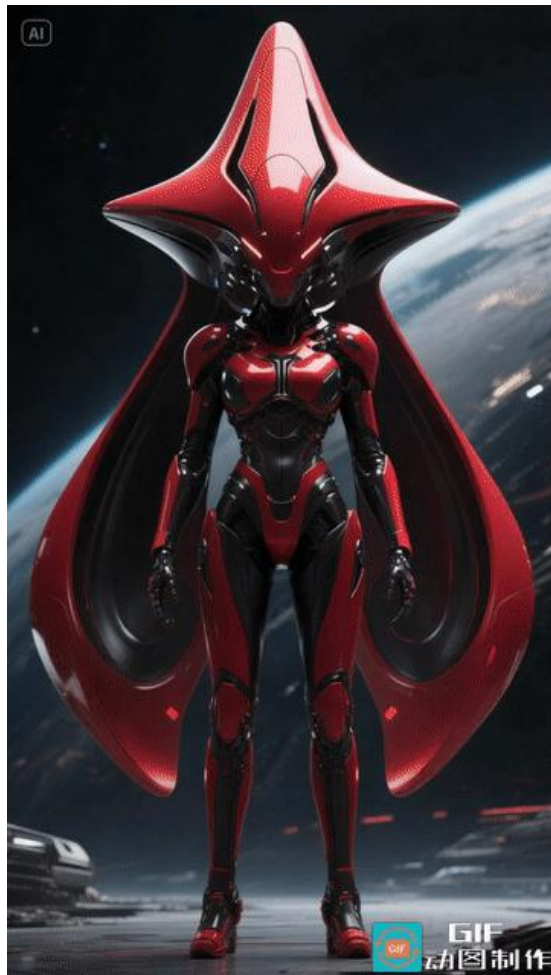
①首帧提示词:

全覆盖无缝太空机甲，奇异头甲造型，双曲面大弧形设计风格，超时空科技风格，身体极精风格，极轻盈风格，锥体造型，极圆润风格，特大奇形风格，设计独特，风格独特，外形优美，红黑色彩，体形修长，细节精致，布局完美，大气磅礴，层次分明，神秘空灵，色调暗黑，色彩明亮，光影分明，明暗对比强烈，大师摄影，大师补光，3d写实，OC渲染，超高清36k，电影胶片级质感和光影，超广角镜头，地平面超仰视镜头，超极端角度，全身特写，正面特写，头脚展示。

②尾帧图片:

经测试这张图片直接用参考图功能变摩托无法得到结果。通过参考图功能先把红色机甲消除，得到一张背景图，再通过参考图功能在背景图片中添加一辆摩托车，以下是提示词：“在图片里添加一辆红色巨大摩托车，车头正对镜头”。

③生成视频



基于视频大模型的Morphing——青蛇化龙

①首帧提示词：

一条巨大青蛇在天上飞，天上乌云密布。

②尾帧图片：通过参考图功能，输入提示词：“把青蛇换成有四只脚的青龙，天空出现一道闪电劈在龙身上”。

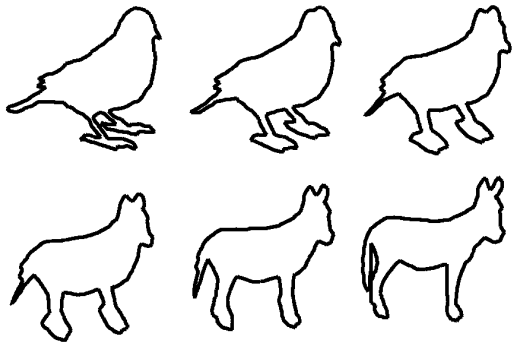
③生成视频

视频提示词：

蛇扭动身体飞行在空中，一道闪电从上方到下方快速劈下，天空下雨，蛇逐渐变成龙，上面的脚从身体里生长出来，下面的脚从身体里长出来，四只脚从身体里长出来



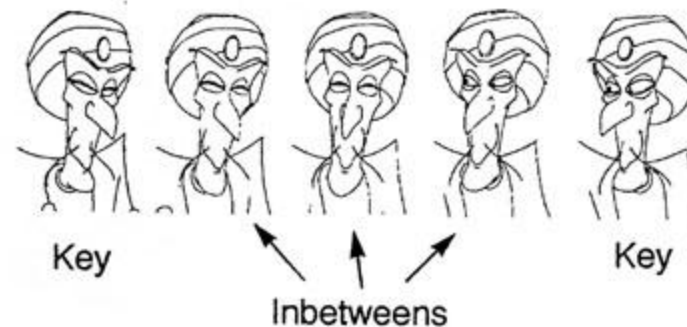
二维多边形形状渐变(2D Shape Blending)



Morphing Sequences

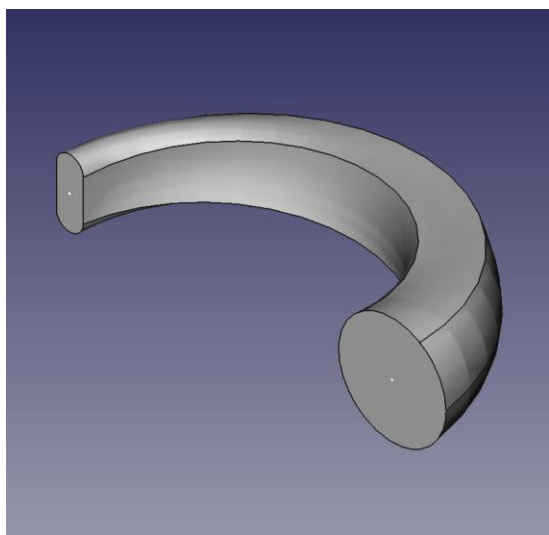
2D Shape Blending问题的引出

- 在二维角色动画(Character Animation)中，经常会碰到这样的问题：给定一个初始和最终的形状(shape)，我们称它们为**关键帧形状**，求从初始形状**光滑过渡**到最终形状的**中间形状**。
- 这个问题称为二维形状的自然渐变 (Shape Blending或Shape Morphing) 。
- 该问题实际上分为两个子问题: (1). **顶点的对应关系问题**; (2). **顶点的插值问题**。

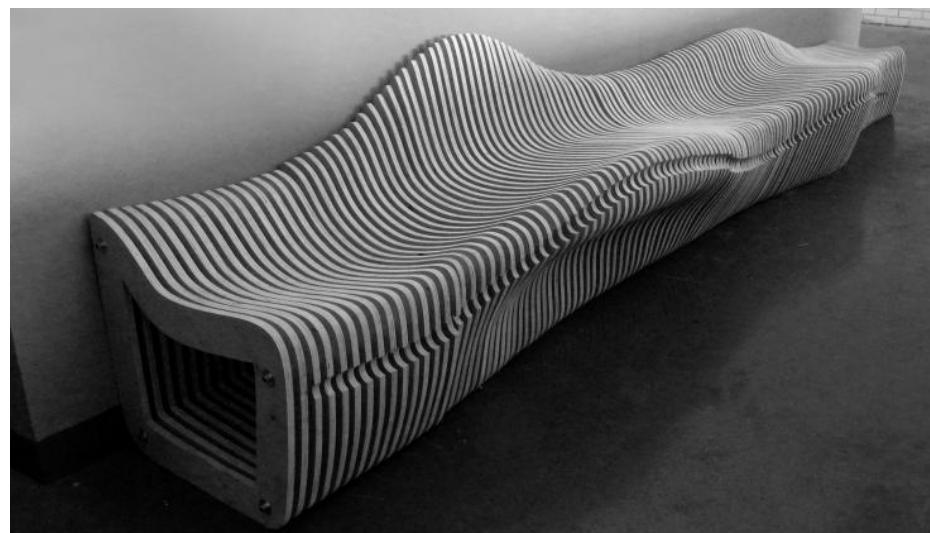
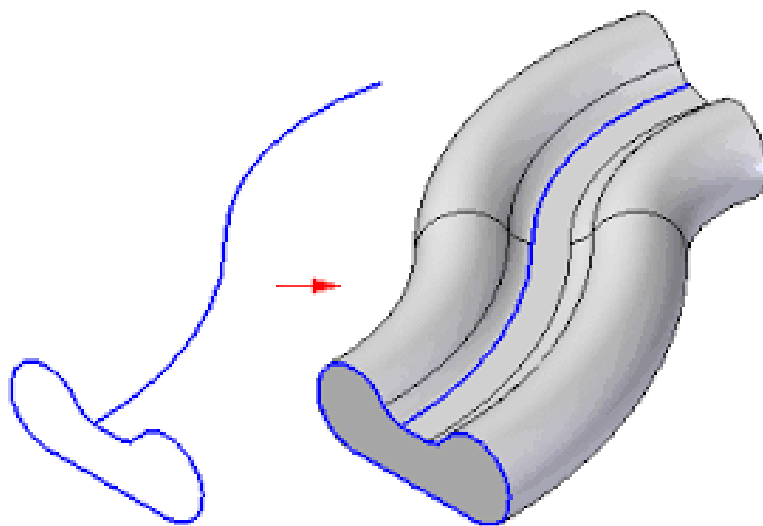


问题的引出

- 二维形状的自然渐变不仅在计算机动画，而且在模式识别、曲面重建和三维造型中也具有重要意义。

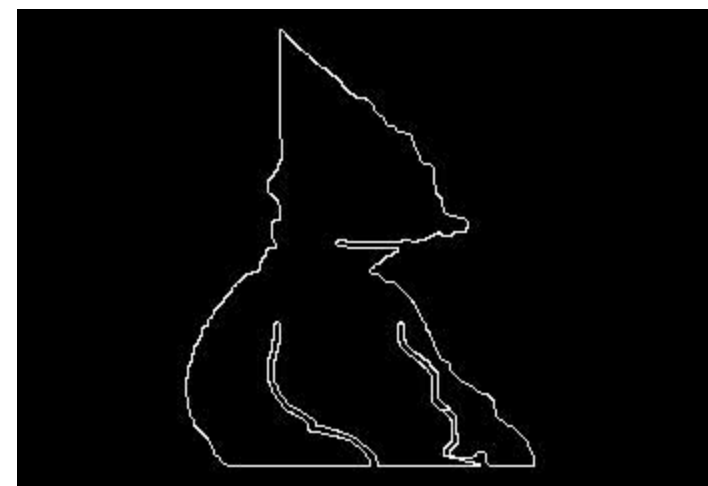


Sweeping曲面



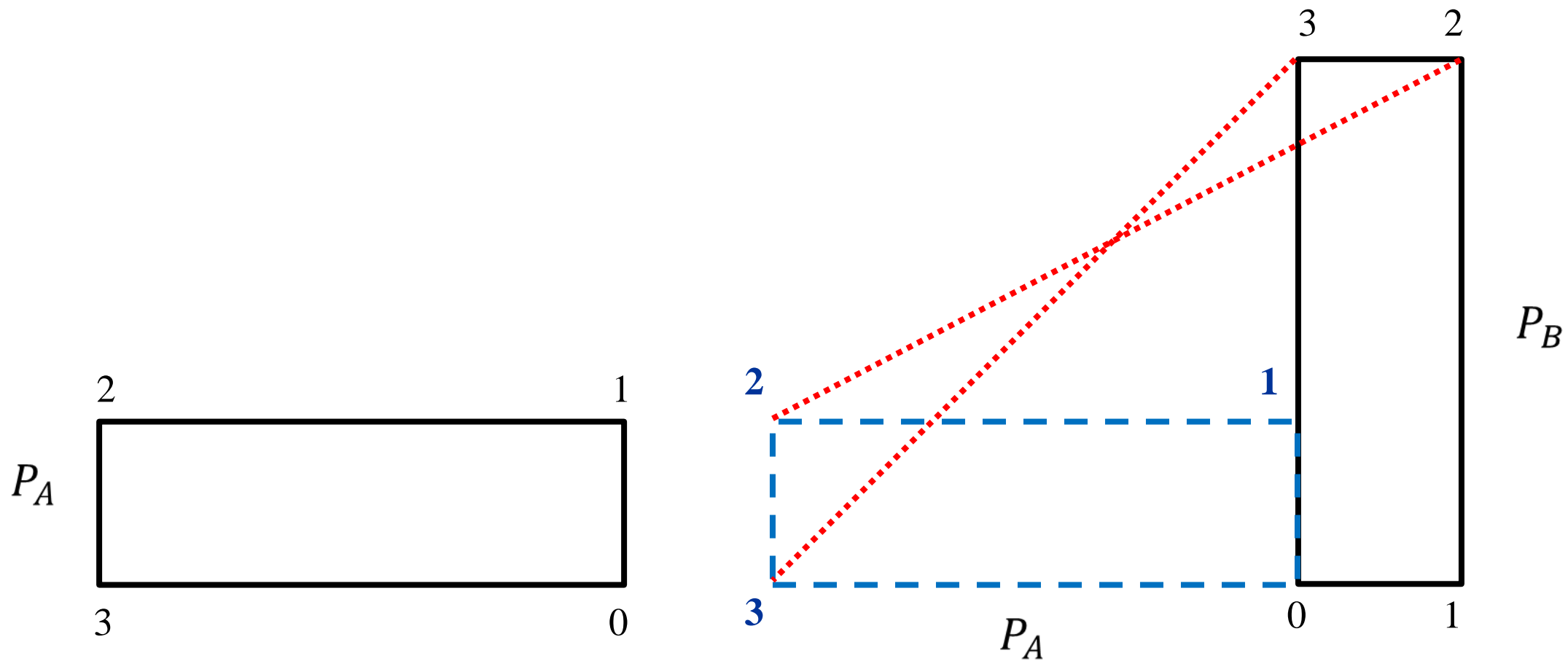
Skin曲面

问题的引出



轮廓

问题的引出



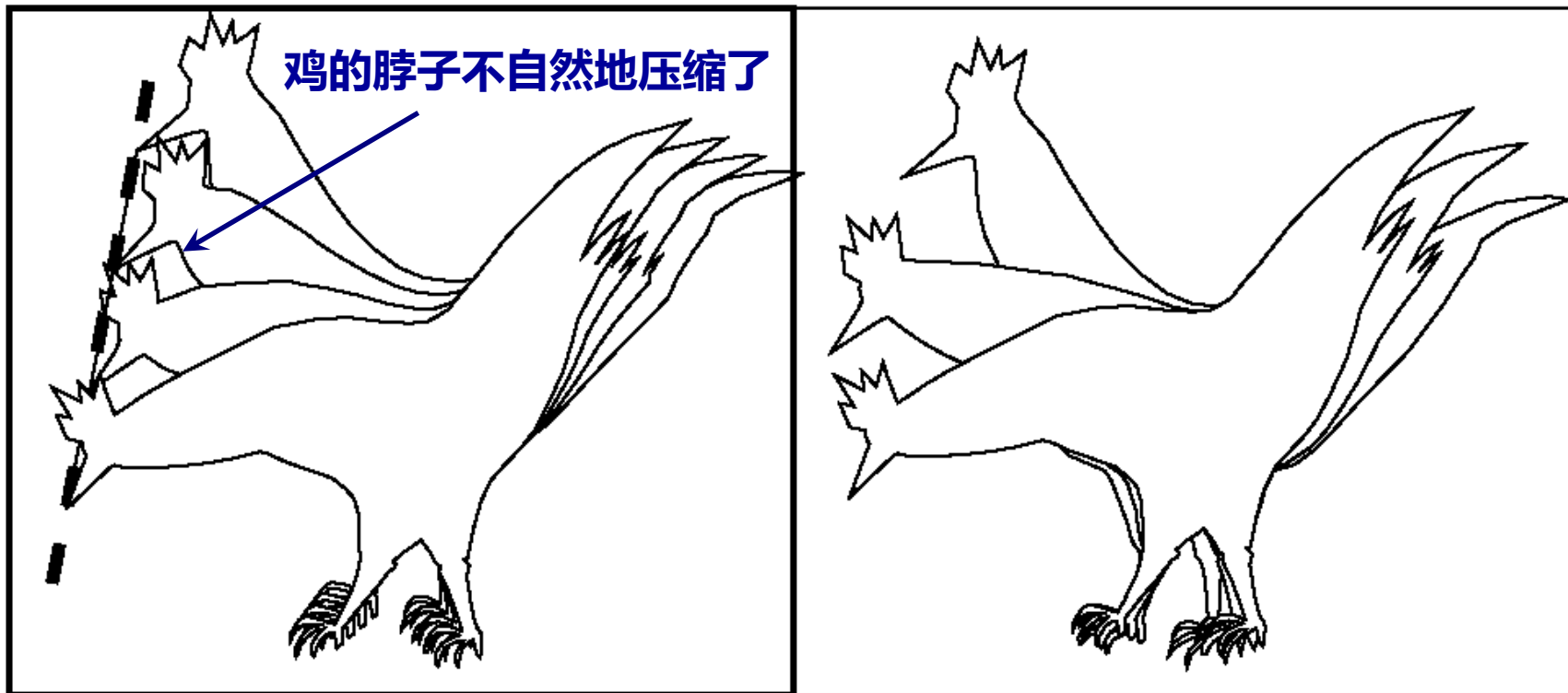
线性插值法

- 假设两个关键帧多边形的顶点为 P_{A_i} 和 P_{B_i} ($i=0, 1, 2, 3, \dots, n$), 顶点的数目都为 n 个, 则我们要解决的是:
 - 顶点的对应关系问题, 即多边形 P_A 中的一个顶点与 P_B 中的哪个顶点对应?
 - 顶点的路径问题, 即 P_A 以何种方式运动到 P_B ?
- 顶点路径问题一个简单的方法为采用**顶点线性插值**:

$$P_i = (1-t)P_{A_i} + tP_{B_i}, \quad (i = 0, 1, 2, \dots, n-1)$$

- 但线性插值会带来**收缩**(shrinkage)和**纽结**现象(Kink), 这在刚体旋转时表现得尤为明显。

线性插值法的缺点



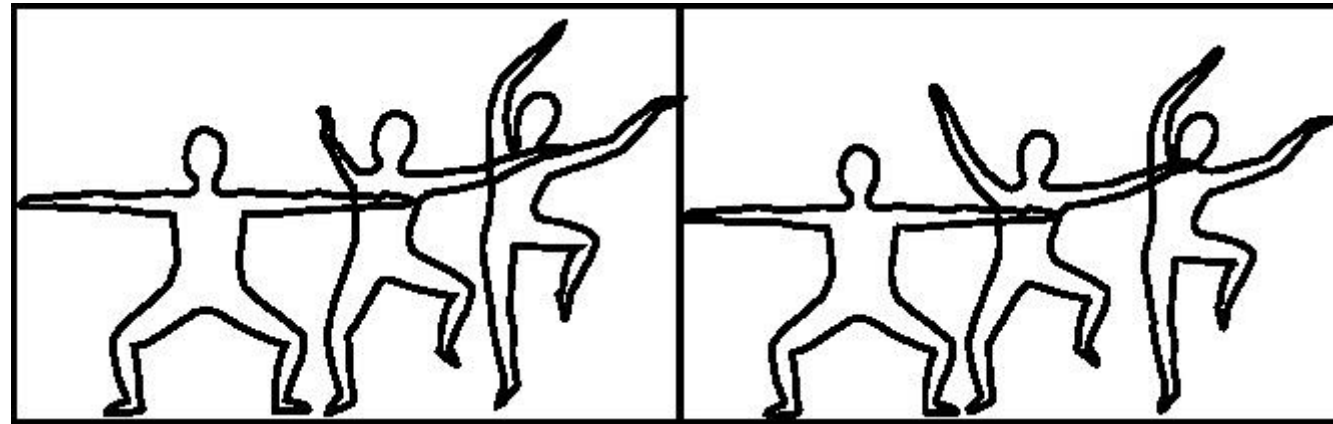
(a) 鸡的脖子压缩了

(b) 更自然的结果

鸡的形状渐变

基于内在形状插值的多边形渐变方法

- Intrinsic Shape Interpolation的数学原理: 乌龟几何+ **Lagrange乘数法优化**



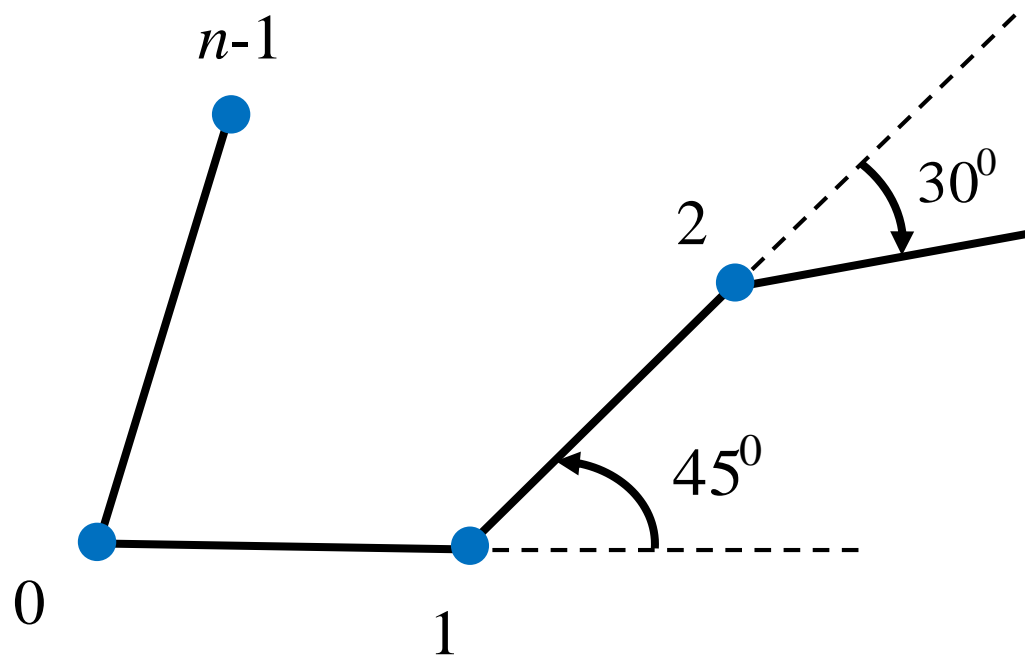
Withered arm

Normal arm

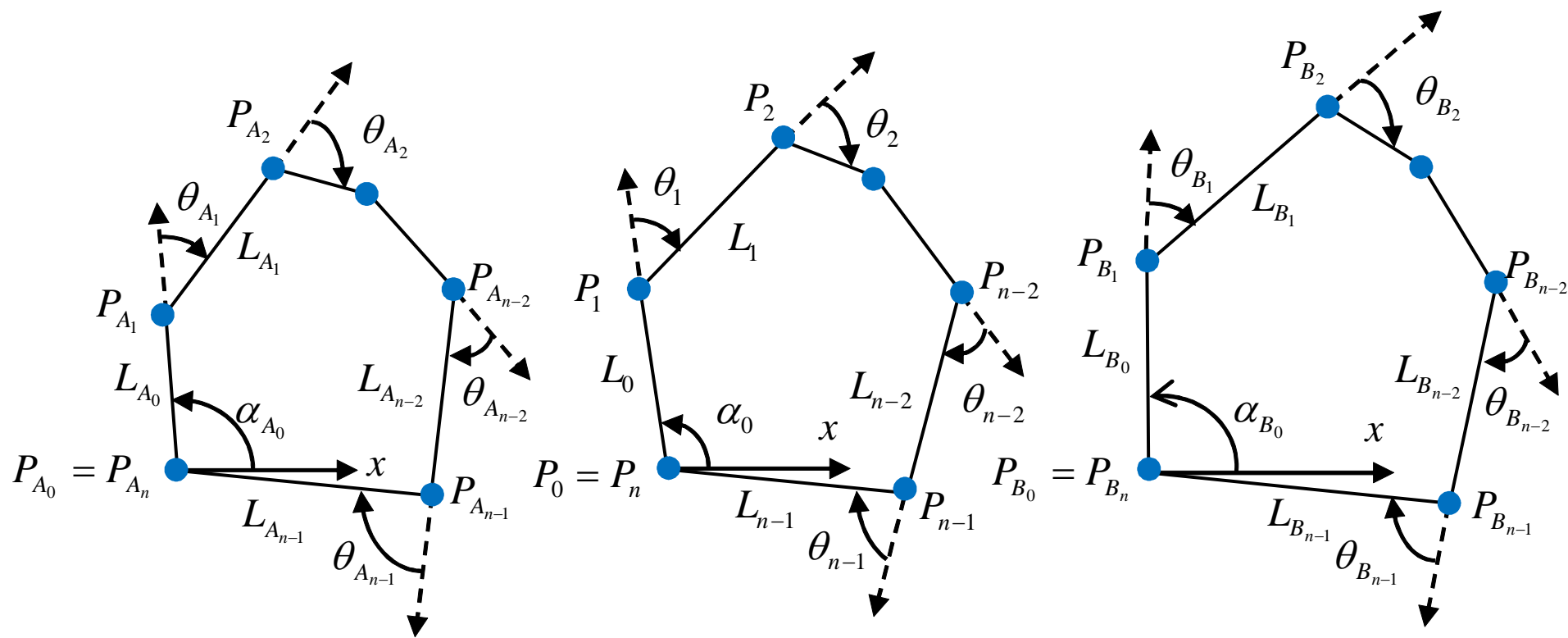
- 参考文献: Sederberg T W, Gao P S, Wang G J, Mu H. 2D shape blending: an intrinsic solution to the vertex path problem. Computer Graphics, 1993, 27(3):15~18

乌龟几何

- 在笛卡尔坐标系中，多边形是通过顶点的坐标显式给出的。
- 但多边形也可以通过乌龟几何来定义，即通过顶点处的边长和有向角来定义。例如，以某一点为起点，向东往前走10米，往左拐 45° ，继续往前走6米，往右拐 30° ，向前走5米，...，最后得到一多边形。
- 因而一个自然的想法是，能否对关键帧多边形的边长和顶点角进行插值来产生比线性插值更好的效果？
- 回答是肯定的，因为这种插值具有更好的几何意义。



多边形的内在定义



- 设 $m=n-1$ ，逆时针方向的角为正的， P_0 为形状的平移锚点(anchor point)，我们的目标是计算中间多边形的顶点 $P_i (i=1, 2, \dots, m)$ ， $0 \leq t \leq 1$ 。
- 通过计算多边形 P_A 和 P_B 的边长和有向顶点角，得到多边形的内在定义 $\{\alpha_0, L_0, (\theta_i, L_i)_{i=1}^m\}$ ：

$$\begin{aligned}
 \alpha_{A_0} &= \theta(x, P_{A_0}, P_{A_1}) & \theta_{A_i} &= \theta(P_{A_{i-1}}, P_{A_i}, P_{A_{i+1}}) & L_{A_i} &= |P_{A_{i+1}} - P_{A_i}| & \text{其中: } \theta(P_{A_{i-1}}, P_{A_i}, P_{A_{i+1}}) \\
 \alpha_{B_0} &= \theta(x, P_{B_0}, P_{B_1}) & \theta_{B_i} &= \theta(P_{B_{i-1}}, P_{B_i}, P_{B_{i+1}}) & L_{B_i} &= |P_{B_{i+1}} - P_{B_i}| & \text{表示边 } P_{A_{i-1}}P_{A_i} \text{ 和边} \\
 & & & & & & P_{A_i}P_{A_{i+1}} \text{ 之间的有向夹角}
 \end{aligned}$$

中间多边形生成

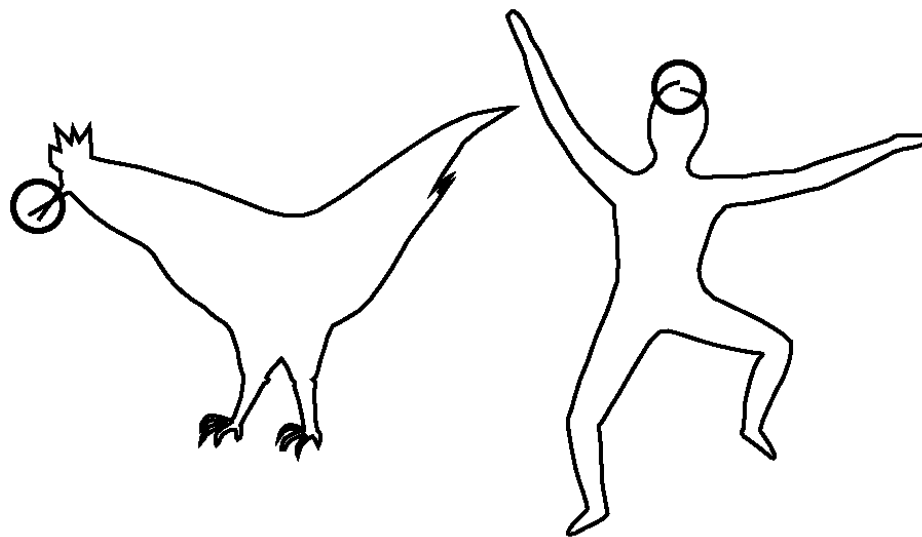
- 那么中间多边形可由插值相应的边长和顶点角得到:

$$\alpha_0 = (1-t)\alpha_{A_0} + t\alpha_{B_0}$$

$$\theta_i = (1-t)\theta_{A_i} + t\theta_{B_i} \quad (i = 1, 2, \dots, m)$$

$$L_i = (1-t)L_{A_i} + tL_{B_i} \quad (i = 1, 2, \dots, m)$$

- 遗憾的是, 这样得到的多边形通常是不封闭的。但实验发现, 得到多边形的起始点和终止点非常接近。



不封闭的多边形

中间多边形生成

- 一个解决方法为保持插值的顶点角不变，适当调整插值得到的边长(Edge Tweaking):

$$L_i = (1-t)L_{A_i} + tL_{B_i} + S_i, \quad (i = 0, 1, 2, \dots, m)$$

- 如果关键多边形的某一条对应的边具有相同的长度 L ，那么我们认为中间多边形的边长也为 L ，因此可认为 $|S_i| \propto |L_{A_i} - L_{B_i}|$ 。为了防止除以零，我们定义：

$$L_{AB_i} = \max \left\{ |L_{A_i} - L_{B_i}|, L_{tol} \right\}, \quad (i = 0, 1, 2, \dots, m)$$

其中： $L_{tol} = 0.0001 \times (\max_{i \in [0, m]} |L_{A_i} - L_{B_i}|)$

中间多边形生成

- 我们的目标是为了求得 S_0, S_1, \dots, S_m , 使目标函数:

$$f(S_0, S_1, \dots, S_m) = \sum_{i=0}^m \frac{S_i^2}{L_{AB_i}^2}$$

最小, 并且 S_0, S_1, \dots, S_m 应满足约束条件(强迫多边形封闭):

$$\varphi_1(S_0, S_1, \dots, S_m) = \sum_{i=0}^m [(1-t)L_{A_i} + tL_{B_i} + S_i] \cos \alpha_i = 0$$

$$\varphi_2(S_0, S_1, \dots, S_m) = \sum_{i=0}^m [(1-t)L_{A_i} + tL_{B_i} + S_i] \sin \alpha_i = 0$$

其中 α_i 是由矢量 $P_i P_{i+1}$ 和 x 轴构成的有向角: $\alpha_i = \alpha_{i-1} + \theta_i, (i = 1, 2, \dots, m)$

中间多边形生成

- 这是个具有约束条件的极值问题，可用Lagrange乘数法来求解。引进Lagrange函数：

$$\Phi(\lambda_1, \lambda_2, S_1, S_2, \dots, S_m) = f + \lambda_1 \varphi_1 + \lambda_2 \varphi_2$$

其中 λ_1, λ_2 为Lagrange乘数。对 Φ 求偏导得：

$$\frac{\partial \Phi}{\partial S_i} = \frac{2S_i}{L_{AB_i}^2} + \lambda_1 \cos \alpha_i + \lambda_2 \sin \alpha_i = 0 \quad (i = 0, 1, \dots, m)$$

$$\sum_{i=0}^m [(1-t)L_{A_i} + tL_{B_i} + S_i] \cos \alpha_i = 0$$

$$\sum_{i=0}^m [(1-t)L_{A_i} + tL_{B_i} + S_i] \sin \alpha_i = 0$$

中间多边形生成

- 由第一式求得 S_i 代入后二式得:

$$\begin{cases} E\lambda_1 + F\lambda_2 = U \\ F\lambda_1 + G\lambda_2 = V \end{cases}$$

其中:

$$E = \sum_{i=0}^m L_{AB_i}^2 \cos^2 \alpha_i, F = \sum_{i=0}^m L_{AB_i}^2 \sin \alpha_i \cos \alpha_i, G = \sum_{i=0}^m L_{AB_i}^2 \sin^2 \alpha_i$$

$$U = 2 \left\{ \sum_{i=0}^m \left[(1-t)L_{A_i} + tL_{B_i} \right] \cos \alpha_i \right\}$$

$$V = 2 \left\{ \sum_{i=0}^m \left[(1-t)L_{A_i} + tL_{B_i} \right] \sin \alpha_i \right\}$$

中间多边形生成

- 如果 $EG - F^2 \neq 0$, 则

$$\lambda_1 = \frac{\begin{vmatrix} U & F \\ V & G \end{vmatrix}}{\begin{vmatrix} E & F \\ F & G \end{vmatrix}}$$

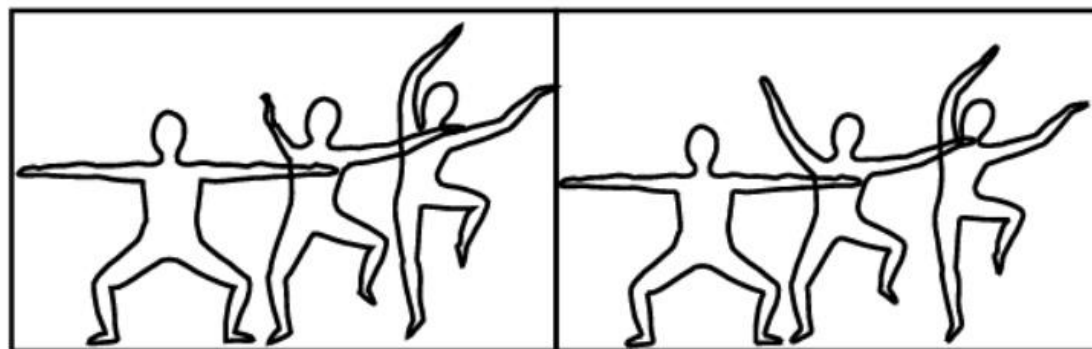
$$\lambda_2 = \frac{\begin{vmatrix} E & U \\ F & V \end{vmatrix}}{\begin{vmatrix} E & F \\ F & G \end{vmatrix}}$$

$$S_i = -\frac{1}{2} L_{AB_i}^2 (\lambda_1 \cos \alpha_i + \lambda_2 \sin \alpha_i), (i = 0, 1, 2, \dots, m)$$

因而可得到中间多边形顶点 (x_i, y_i) 的坐标:

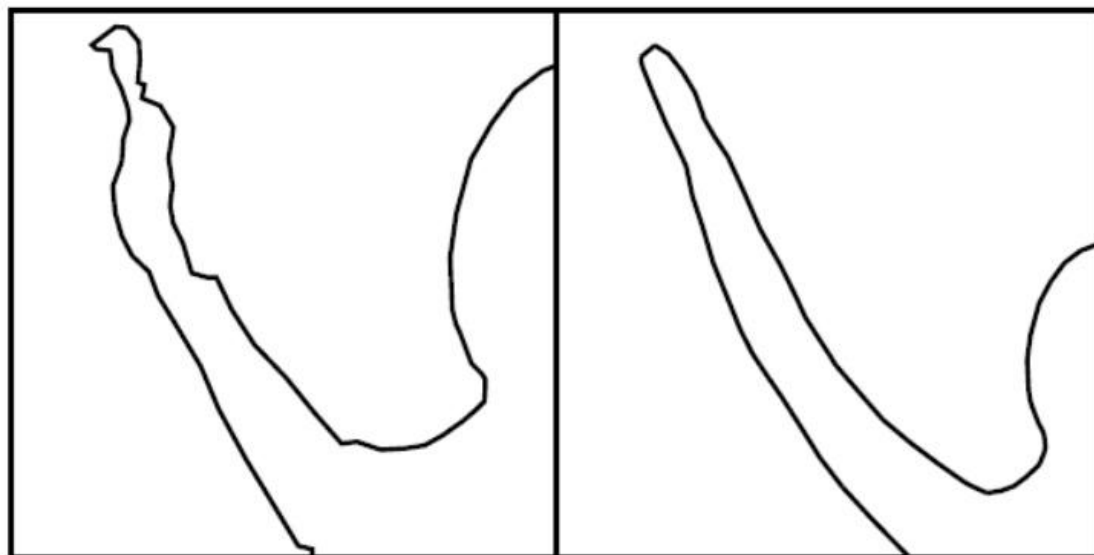
$$\begin{cases} x_i = x_{i-1} + L_{i-1} \cos \alpha_{i-1} \\ y_i = y_{i-1} + L_{i-1} \sin \alpha_{i-1} \end{cases}$$

与线性插值的结果对比



a. Withered Arm

b. Normal Arm

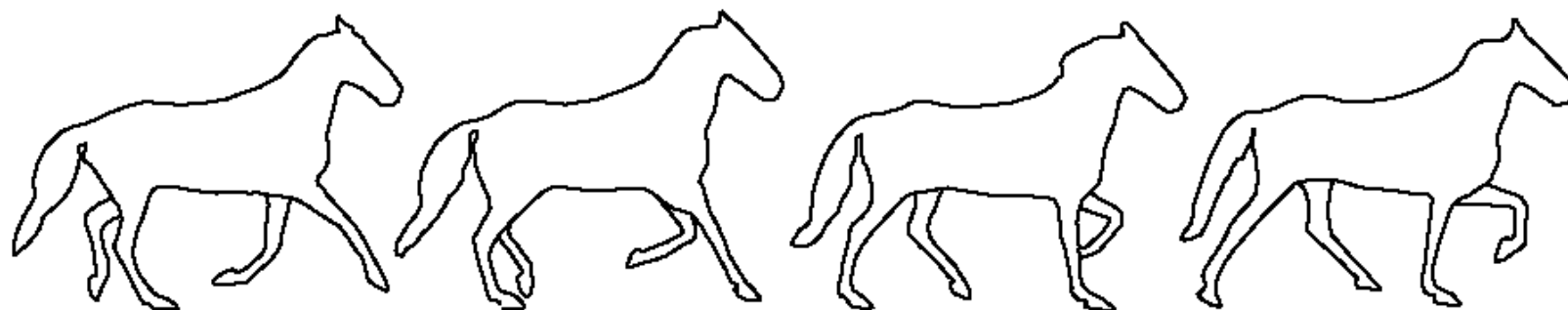


Linear vertex paths

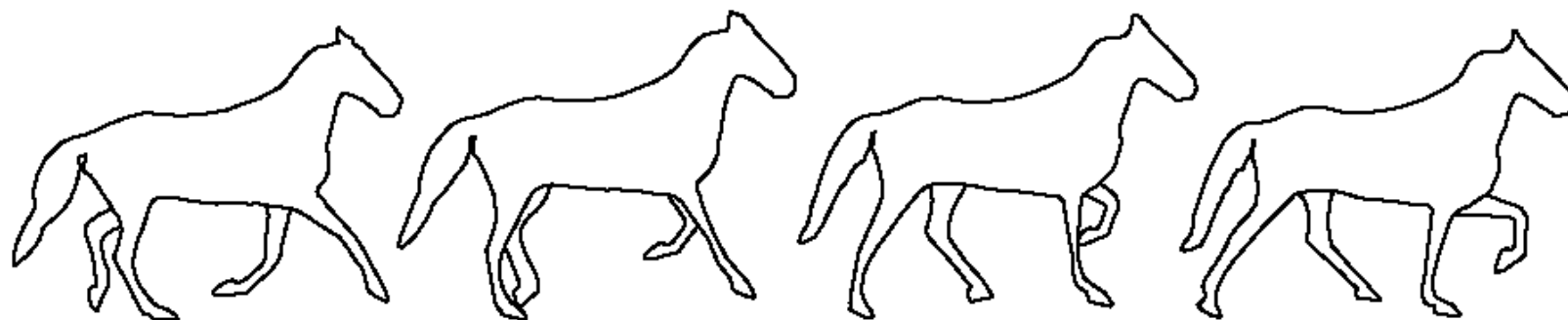
Intrinsic blend

结果

- 实验表明，对于角色动画中的许多应用，内在插值的形状渐变效果良好。



数字化得到的形状(形状取自经典摄影图书《Animals in Motion》)



给定的

内在插值得到的

内在插值得到的

给定的

算法实现

输入：两个关键帧多边形的顶点

- 计算两个关键帧多边形的内在表示;
- 解线性方程组, 计算 λ_1, λ_2 ;
- 计算 S_i ;
- 中间多边形顶点的坐标;

输出：中间帧多边形的顶点

DEMO

Morphing Sequences

内在形状插值法的优缺点

- **优点:**

- 简单;
- 能在一定程度上避免形状插值中出现的收缩和纽结现象,这在二维角色动画中尤其有用。

- **缺点:**

- 顶点对应关系需要用户交互给出;
- 结果依赖于顶点对应关系的好坏;
- 当源和目标多边形中包含短边时,由于计算短边的方向不稳定,中间帧多边形有可能产生较严重的畸变。

基于模糊数学的二维多边形形状渐变

- 1996年, Zhang提出了一种基于模糊数学的二维多边形形状渐变方法。
- 它采用基于模糊数学的方法来建立源多边形和目标多边形**顶点的对应关系**以及**顶点之间的插值**。
- **优点:**
 - 能处理具有不同位置、方向、大小或形状的多边形。该方法不仅稳定, 而且易于推广到曲线形状。

*** Zhang Yuefeng. A fuzzy approach to digital image warping. IEEE Computer Graphics and Applications, 1996, 16(6):34~41**

More 2D Shape Blending DEMO

More Examples

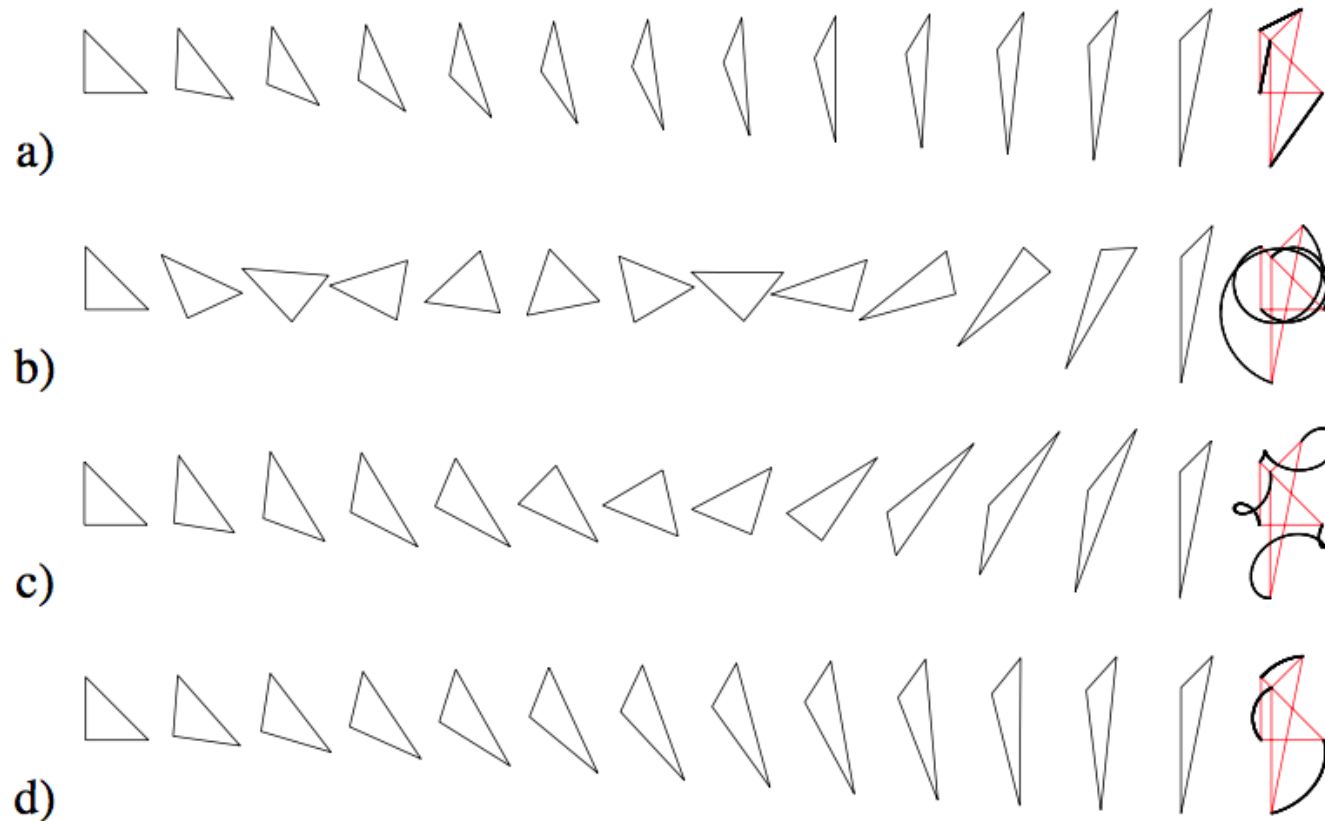


More 2D Shape Blending DEMO



Morphing in After Effects

进一步读的论文



Transformations of a single triangle. (a) Linear vertex interpolation. (b-d) An affine map from the source to the target triangle is computed and factored into rotational and scale-shear parts. Intermediate triangles are constructed by linearly interpolating the angle(s) of rotation, the scaling factors, and the shear parameter. (b) corresponds to Equation 4; (c) shows the results of reducing the overall angle of (b) by subtracting 2π from one of the angles; (d) corresponds to Equation 5 and represents the method of our choice. The last column in all rows shows plots of the vertex paths.

Alexa, M., Cohen-Or, D., & Levin, D. (2000, July). As-rigid-as-possible shape interpolation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (pp. 157-164).

总结

- 基于StyleGan的肖像Morphing是社交媒体中常用的技术。
- 2D Shape Blending 是传统关键帧动画设计中经常碰到的问题，可节约动画师设计中间帧的工作量。**局限性：**自动计算得到的结果不一定是动画师所需要的。



The End