

白 皮 书



基于实时渲染的高真实度 Groom: 虚幻引擎中的毛发运用

目录

引言	3
为什么会有此白皮书?	4
DCC应用中的毛发	5
术语	5
Groom工作流程	6
Groom最佳实践	7
虚幻引擎中的毛发实时渲染	10
优化	10
导入流程	11
Groom资产编辑器	11
表面附着	12
光源近似	14
渲染	16
纤细发丝的光影	17
模拟	17
皮肤暗化	18
使用案例：用MetaHuman Creator制作人类毛发	19
斜刘海Groom	19
短卷Groom	22
使用案例：狐獴项目中的动物皮毛与羽毛	27
狐獴皮毛	29
老鹰羽毛	31
总结	32
其它学习资料	33



引言

头发和毛皮都是由数千根具有一定特征的彩色发丝构成的复杂有机结构。我们能够一眼认出毛发的独特光泽，是因为发丝本身拥有独特的半透明与反光属性，而且容易形成发束并卷曲在一起。这些发丝还会对重力或风吹等外力产生反应；由于生活中我们一直在观察各种各样的毛发，所以对这些效果早已习以为常。

然而，生成逼真的CG毛发一直是一个难点，并且是已经持续了数十年的研究，其中涉及的不仅包括游戏与影视中的动画角色，还牵涉到军事、工业、动物科学及其他领域中的培训模拟。

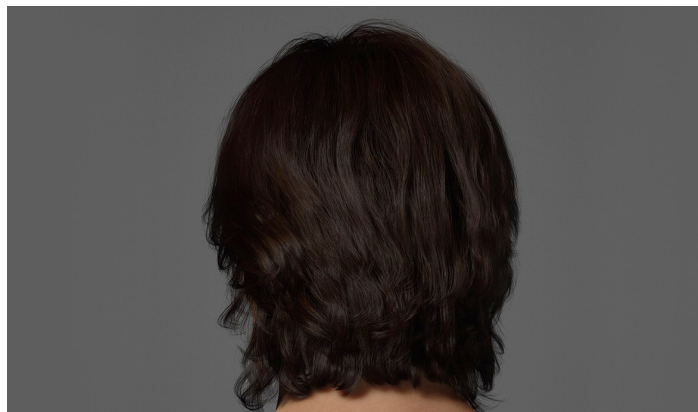


图1: DCC应用中的毛发导线，以及虚幻引擎中的最终渲染效果。



创建计算机生成（CG）毛发的最大难点在于如何瞒过肉眼。然而，由于计算机处理能力有限，在完成耗时良久的渲染流程之前，CG美术师通常无法看到CG毛发的最终色彩、场景环境中的阴影，或是头发对光照和运动的反应。在渲染结束后，美术师终于可以看到改进的空间，却不得不对设置进行微调，然后再次渲染。因此，要制作精美的CG毛发与毛皮，整个流程通常费时费力。

正是出于这个原因，Epic Games最近推出了在虚幻引擎中实时渲染毛发的功能，让美术师能够立即在屏幕上看到最终的结果，无需为渲染苦苦等待。

为什么会有此白皮书？

毛发的渲染通常会消耗大量时间，虽然实时渲染是众望所归的解决办法，但为了确保其高效运行，制作出最逼真的毛发或毛皮，在虚幻引擎中设置时就必须慎重考虑多种参数。

本白皮书旨在介绍虚幻引擎中实时渲染毛发的流程，并在文档的内容之外，探讨美术师们在使用该功能时可能遭遇的美术与技术考量。

DCC应用中的毛发

虚幻引擎中的毛发是在Maya、3ds Max或Houdini等数字内容创作（DCC）应用中创建的。毛发会附着在人脸或动物模型等多边形表面上。

用于制作毛发的常用工具包括**XGen**、**Yeti**和**Ornatrix**等DCC插件，还有**Houdini**所包含的Hair Utils工具。除了毛发之外，这些工具也能制作鸟类的羽毛。

在DCC应用中完成毛发的设置后，就可以将其导入到虚幻引擎中，以进行实时渲染。

术语

为了介绍DCC应用的方法论与考量因素，以及虚幻引擎中的扩展工具组，我们会在此简短回顾一下与制作CG毛发相关的术语和流程。

毛发导线、引导曲线（Guide hair, guide curve） - 代表一组毛发生丝样条类型的图元。毛发工具包括了操控毛发导线的机制，能够确定最终毛发（插值毛发）的表现——插值毛发的形状取决于和最近的引导曲线的加权平均距离，其长度通常由引导曲线的长度决定。

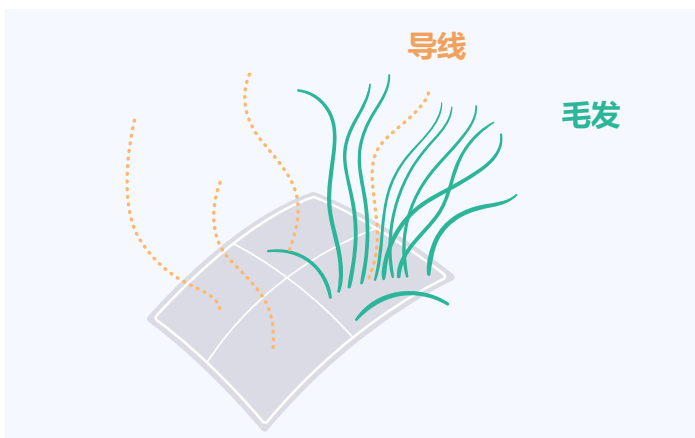
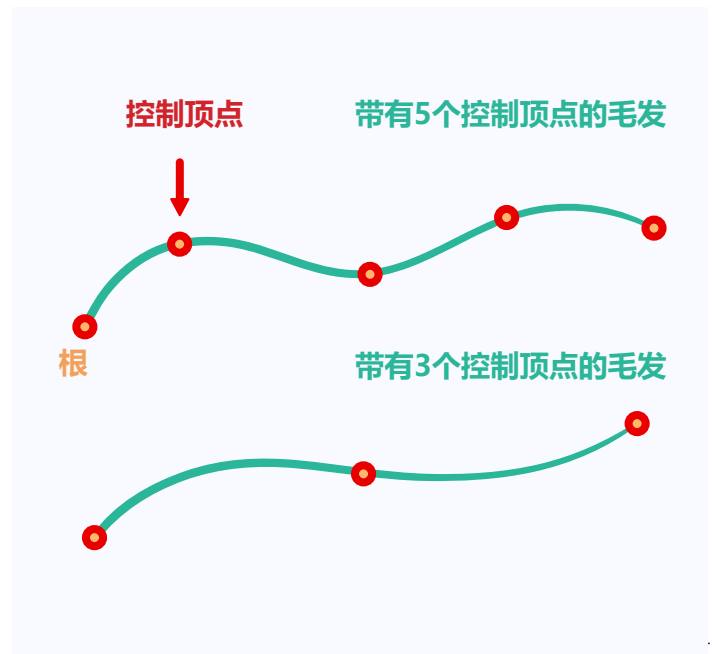


图2：导线和毛发的关系

控制顶点（Control vertices） - 引导曲线上会放置一个或多个控制顶点，以便在曲线上控制该点。毛发导线上的控制顶点数量越多，毛发导线的形状就更为精确。然而，控制顶点的增多会增加处理时间，从而降低性能。



造型（Styling） - 沿着特定的方向对毛发导线或插值毛发进行推拉、塑形和卷曲等操作，使CG毛发更符合参考图片。

Groom - 定义毛发生长和自然毛发属性的过程，此外还包括毛发的造型过程。其中包括了设置发丝的厚度及卷曲聚丛的倾向。由此产生的设置及保存设置的文件被称为**Groom文件**。在DCC应用中，使用毛发导线和参数设置就能实现Groom。

Groom工作流程

Groom工作流程发生在DCC应用中，此时模型尚未导入到虚幻引擎中。作为起点，先使用一组毛发导线来确定形状，然后会在网格体区域中创建体积。

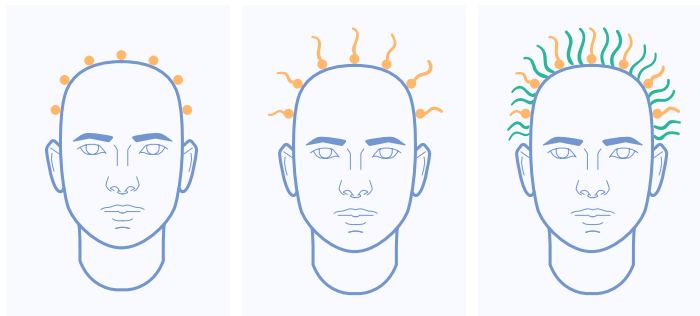


图4：人脸形状毛发的引导毛发

可以使用Groom笔刷定位、雕刻和缩放可塑性的引导曲线，Groom笔刷可以卷曲、修剪甚至移除毛发。

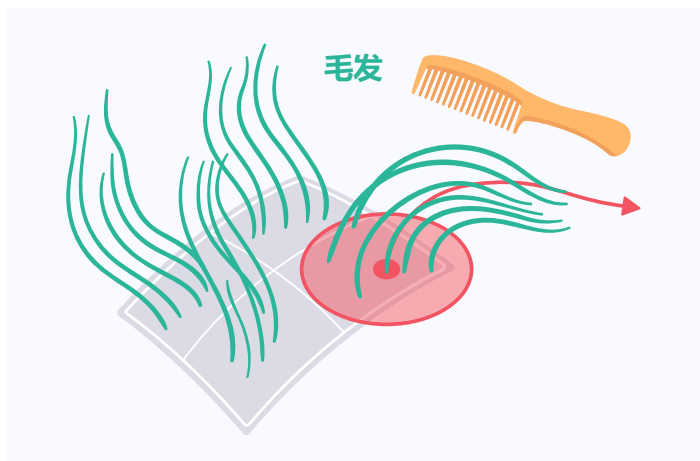


图5：虚拟Groom笔刷

在雕刻和毛发造型笔刷的帮助下，引导曲线会被塑造成发型或毛皮样式。渲染毛发的色彩可以通过材质参数或毛发采样的纹理确定。

由于角色的不同区域会有大相径庭的特征，Groom设计师可能会为人类的头部、眉毛、胡须和体毛，以及动物的头部、身体、腿部和尾巴创建单独的Groom。

在创建较为复杂的Groom时，可以将其分解为不同的部分，这样将资产导入虚幻引擎后就能更轻松地管理多个分组。

程序化造型

DDC应用中的毛发工具包含许多程序化造型参数，可以用来调整毛发导线或插值毛发的效果。这些参数的名称通常与现实生活中的发型设计术语相仿，比如**修剪 (trim)**、**卷曲 (curl)** 和**卷发 (frizz)**。每个程序都有相关的一个或多个数值，以表明毛发导线或发丝的相对变动程度。

修剪 (trim) 值会限制毛发导线或插值毛发的长度。**卷曲 (curl)** 设置会获取卷曲的数量和范围，并相应地执行。**鬍曲 (frizz)** 或**噪点 (noise)** 会根据输入的数值，使曲线的控制顶点位置随机化。

对于此类参数，控制顶点的数量会直接影响到曲线可以支持的细节程度。由于复杂的形状需要大量的控制顶点，你也许不得不经常重建曲线，才能确保控制顶点在整根毛发中均匀分布。

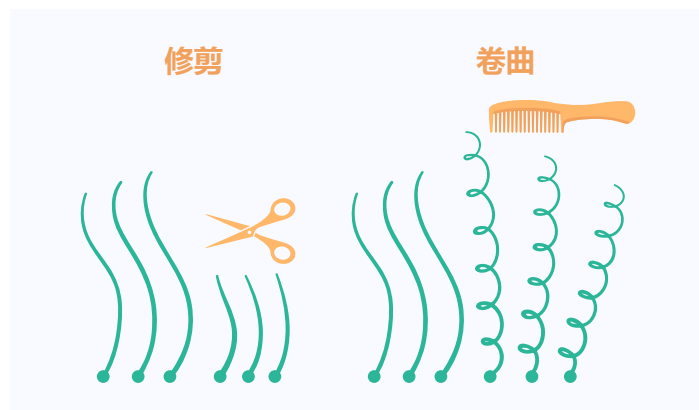


图6：程序化造型工具

密度与厚度

渲染带有毛发的图片时，系统需要在每个像素中渲染多根可见的发丝。人类的头部长有数十万根发丝——在虚幻引擎里，人类发型的合理发丝数量大约为5万根——每根发丝的平均直径约为100微米。（动物毛皮的发丝数量和平均直径会根据动物而大相径庭。）

最适宜的Groom会找到密度与厚度的平衡点，在真实感和性能之间把握平衡。

聚丛

聚丛是毛发自然发生的现象，常见于发束、单独卷曲或潮湿的毛皮。添加聚丛水平的主要目的是为较大的形体增添细节、变化和层次，以获得更为逼真的外观。

在CG中，聚丛算法会在渲染期间将发丝的尖端拉扯到一起，由此打造出这种效果。

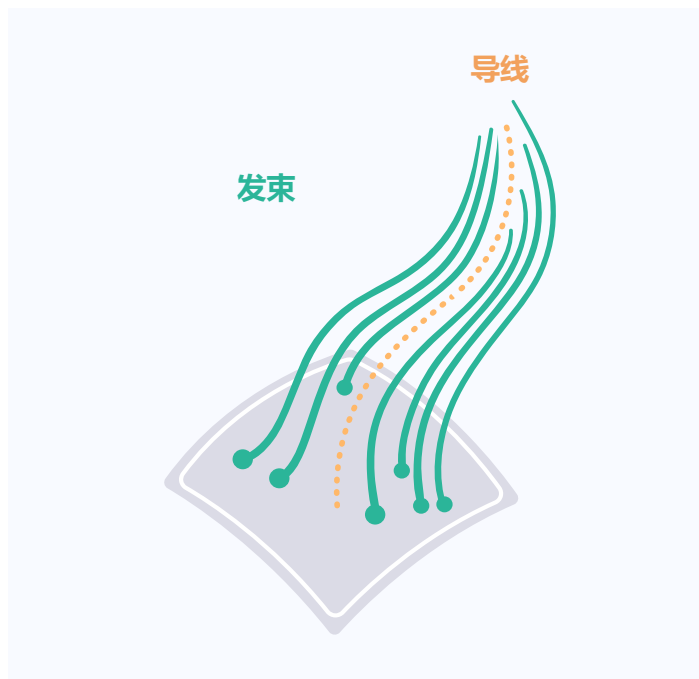


图7：聚丛将发丝尖端聚拢

为了让毛发产生层次效果和自然结构，聚丛又被细分成三个等级：大聚丛、中聚丛和小聚丛，我们也会将其称为主要聚丛、次要聚丛和三级聚丛。主要聚丛是主要的毛发发丝集群，从中会衍生出次要聚丛和三级聚丛。

何时使用这些额外的聚丛并没有固定规则，一切都要看对应的参照物。现实中的聚丛效果往往细节丰富，（因此）要完全获得参照图片中的效果会十分困难。最好先从较大的聚丛结构入手，然后再着手处理更细微的聚丛效果，直到整体结构符合参照图片中的效果。

程序化与人工方法

程序化方法会使用参数（修改器、表达式）来放置毛发，并设置一根或多根毛发的卷曲和长度等属性。人工方法与之相反，需要手动放置毛发，并使用鼠标刷对单根或小群毛发进行雕刻，以匹配参照物。

Groom通常包含这两种方法，从程序化转变为人工控制的时机取决于Groom设计师，可以使用一系列造型修改器程序化地定义简单引导曲线的最终状态，也可以人工放置每一根引导曲线并精雕细琢为最终造型。

程序化方法倾向于处理较短的毛发。通常情况下可以先使用尽可能多的程序化方法来填充Groom区域，随后再手动放置发束和单独的毛发。

如果角色距离摄像机较远，或者处于高速运动的序列中，程序化方法通常就是最优解。在大部分情况下，与使用人工方法构建的资产相比，程序化Groom的图层或节点网络都会更少。

虽然人工方法颇耗时间，但对需要近距离观摩的主角类资产而言，这是必需的选择。凭借这一技巧，Groom设计师可以手动放置毛发来匹配具体细节，从而获得理想的效果。例如，在为影片制作视觉效果时，Groom设计师通常会手动调节每一根睫毛来匹配演员的容貌，或者手动移动每一根毛皮发丝来匹配参照物的毛皮结构。

Groom最佳实践

参照

要获得最出色的逼真度，最好的方法莫过于寻找参照物并仔细观察细节。收集高清图片可以确保纤细发丝和微妙层次等特写细节能够清晰展现。寻找能够覆盖多个角度的参照是重中之重，因为它们能够透露Groom的综合体积与形状。



图8: 狐獴参考照片 (左侧) 和渲染 (右侧)

从多个不同的来源选择有趣的结构并结合参考，就能创造出基于现实而又独一无二的Groom。在Groom设计流程中，Groom设计师可以将关键参照图片分作主要参照，将其他参照图片作为次要参照。

假如Groom包含大量方向性变化，则可以在平面图像中勾勒出变化方向。这一技巧能够加快进程，让你更快地将引导曲线布置在正确的区域中。



图9：勾勒方向变化

生长网格体

虽然可以直接将Groom附着到模型的主要部分上，但使用生长网格体会更为轻松——这个独立的几何体唯一的作用就是“生成”毛发。当你之后将Groom导入虚幻引擎时，就能将Groom绑定在主要网格体的皮肤上。

如果模型的设计仍在变动，使用生长网格体就能减少转化次数（将Groom重新配置/转化到网格体上）。生成多个生长网格体，让每个网格体代表Groom的独立区域，就能更轻松地聚焦各个区域的Groom。

让Groom网格体单独分离开来的另一项好处是，那些无需毛发的几何体区域，例如眼球、眼眶、口腔内部、牙齿、衣物等，都可以被移除或隐藏起来。

如果网格体的UV布局中有部分区域分辨率不足，则可以使用一组单独的UV来绘制Groom的属性贴图。



由虚幻引擎实时渲染的猎犬毛发，作者：Jelena Jovanovic - Digitalbite

虚幻引擎中的毛发实时渲染

虚幻引擎主要执行优化、渲染、模拟以及毛发LOD的生成，以及有限的造型控制。在理想情况下，你需要先确定好Groom的造型，再将其导入到虚幻引擎中。在初次导入之后更改造型的情况并不罕见，此时只需要在数字内容创作应用中更改造型，随后重新导入Groom即可。

虚幻引擎使用**Alembic**文件格式作为毛发导入语言，并会通过命名规则来辅助导入**XGen**、**Ornatrix**、**Yeti**和**Houdini** Hair Utils等数字内容创作工具的静态Groom。

在DCC应用中创建Groom文件并导出为Alembic格式时需要注意，你要确保将特定属性导出到Alembic文件中，详情请参见本白皮书的“导入流程”章节。Ornatrix和Yeti可以将这些属性导出到Alembic文件中，以便后续导入虚幻引擎。

优化

就基于发丝的毛发系统而言，高密度的Groom可以包含数十万甚至百万根发丝，每根发丝也会包含数十个控制顶点。这两点共同构成了导入、渲染和模拟的性能影响因素。

在影片特效等离线渲染流程中，通常会参照真实世界中的（毛发）密度和宽度来尽可能营造真实感。但在实时渲染中，虽然你也可以一开始参照就现实中的密度和宽度，以求尽可能还原参考资料中的效果，但通常最后都需要对性能进行评估并进行优化。

用于实时渲染的毛发，可以在导入虚幻引擎之前，就在DCC应用中完成优化。优化时可以先从控制顶点的数量入手。如果毛发短而直，就可以大幅减少控制顶点，并且不会产生太多的视觉差异。如果毛发长而蓬乱，而且比较复杂，就要避免减少过多的控制顶点，以免有损整体的结构。

下一步就是优化密度和宽度。首先将密度减半，将宽度调整为两倍，并根据需要重新评估。在选取人类毛发的平均宽度时，通常可以先从0.008厘米开始，而发际线处、鬓角处和散落的纤细发丝通常会使用更低的宽度值。

视觉感知的密度和实际的密度之间存在一种平衡。较粗的发丝可以阻隔光线和直接可见的皮肤，从而实现更浓密的Groom外观。通常来说，在降低密度的同时，最好通过提高宽度作为补偿，直到在性能与真实性之间取得平衡。

如果资产距离摄像机非常近，这种方法未必能够奏效，因为宽度将看起来不够真实。LOD可以在摄像机远离时通过发丝剔除和加厚解决这一问题，在改善性能的同时，确保Groom的宽度与密度能尽可能地贴近现实。如需进一步优化，你可以回归传统，选择基于发片的方法，使用少量大型平整的板片提供大量单独毛发的大致形状和运动。虚幻引擎也为这些方法提供了可用的功能。

导入流程

要将Groom文件导入到虚幻引擎中，你必须先将它从DCC应用中导出为Alembic格式，随后才能将其导入虚幻引擎，并成为Groom资产。

虚幻引擎并不会根据引导曲线和插值规则重建Groom，而是将引导曲线和发丝一同导入，精准地还原制作的Groom。

在导入流程中，导入者需要查找属性和分组并将它们导入进新的Groom资产中，并且这些属性和分组需要遵循“**Alembic for Grooms规范**”一文中所述的命名规范。实施该方案能将**宽度**和**颜色**属性以及模拟与插值毛发所需的**导线**属性导入到虚幻引擎里。单个Alembic文件中的多个毛发分组可通过**group id**支持。在这些属性中，**根UV**属性会获取底层表面（如皮肤表面）的UV，能够根据底层表面，在Groom中产生部分空间变化。

如果Alembic文件只包含了曲线，但没有遵照Alembic导出规范，Groom文件依然可以导入到虚幻引擎中，但将不包含自定义属性。

要了解关于导入流程的详情，请参考虚幻引擎文档中的**毛发渲染**主题。

Groom资产编辑器

Groom资产编辑器可以在选中Groom资产时使用，编辑器中的控制选项能够在虚幻引擎中对Groom的上色、优化和其他方面予以协助。

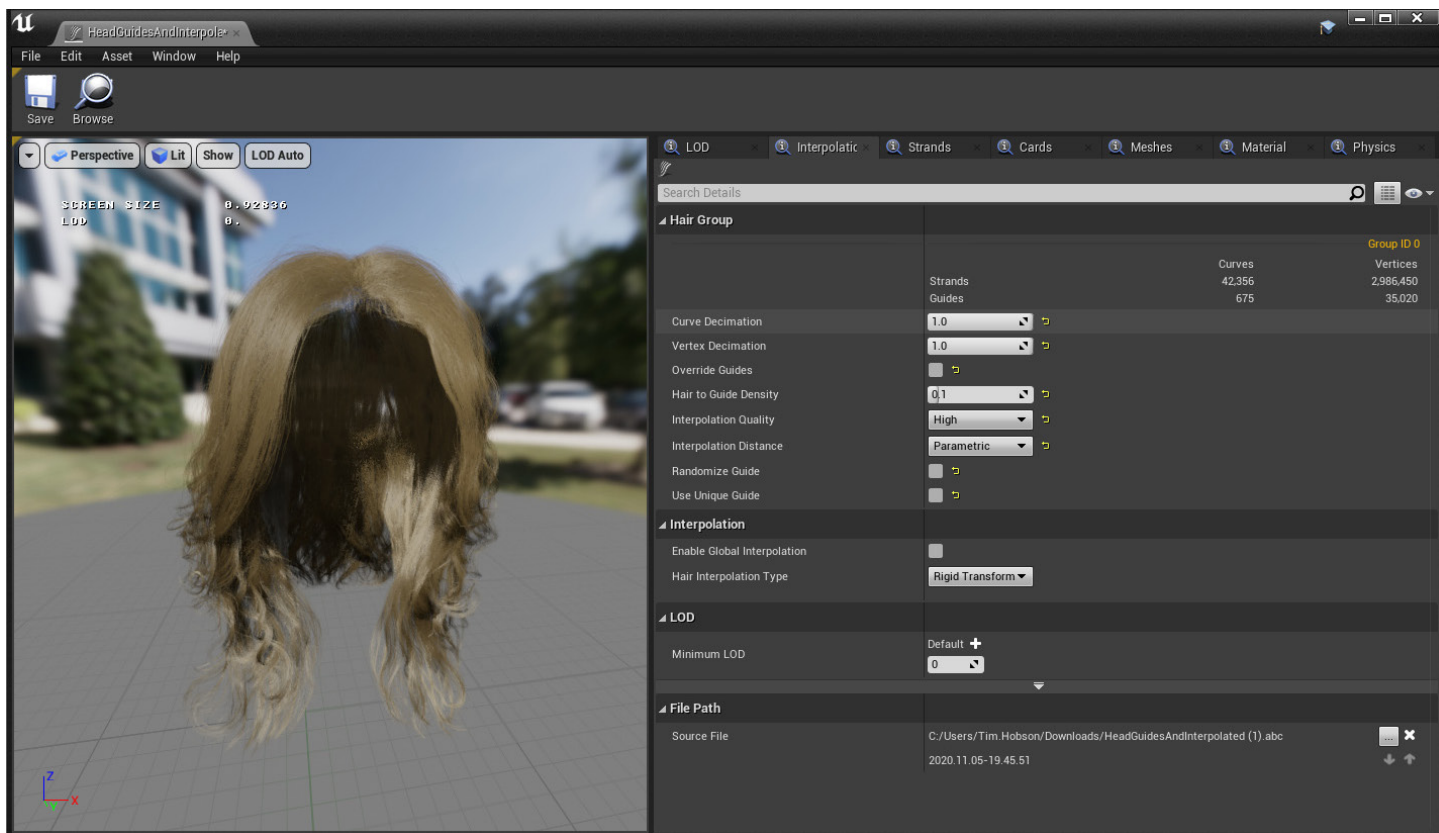


图10: 虚幻引擎中的Groom资产编辑器

虽然Groom系统的主要设计目的是处理由发丝构成的Groom，但是出于可伸缩性，也支持发片和网格体的几何形式。为了方便设置，所有几何体表现种类都可以在单个资产（Groom资产）和组件（Groom组件）来管理所有这些几何体表示。编辑器还提供了一个LOD面板，允许你配置每个LOD应该使用的几何体呈现和删减发丝的方式。Groom资产可以由多个分组构成（例如一个刘海分组、一个马尾分组等），每个分组都可以有自己的LOD设置。

要了解关于使用该功能的更多信息，请参考虚幻引擎文档中的[Groom资产编辑器用户指南](#)和[毛发属性参考](#)。

表面附着

Groom通常会附着在绑定好的**骨骼网格体组件**上。

要将Groom附着在蒙皮表面上，就需要在层级中将Groom组件附着到骨骼网格体组件下，还需要提供一个绑定资产（请参考虚幻引擎文档中的[毛发属性参考](#)）。绑定资产存储了毛发发丝在目标骨骼网格体上投射的信息。

在附着过程中,我们会为每个发根以及与之最近的网格体三角形计算出**重心坐标**。当网格体在动画期间发生形变时，毛发根部会始终被约束到骨骼网格体上。

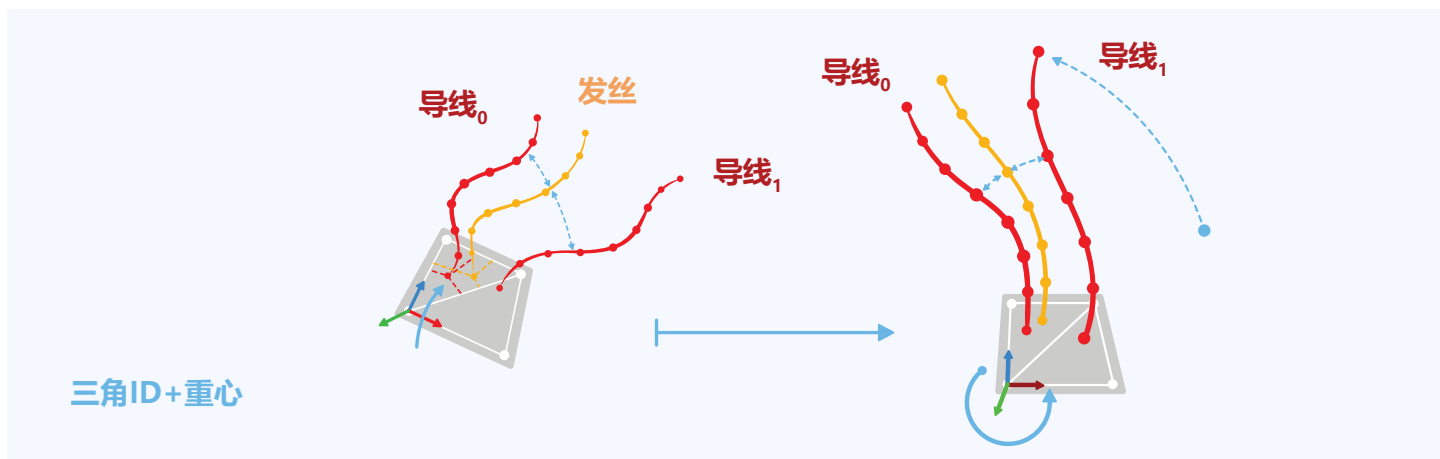


图11: 发丝如何在皮肤形变期间“扎根”在网格体三角形上的图解

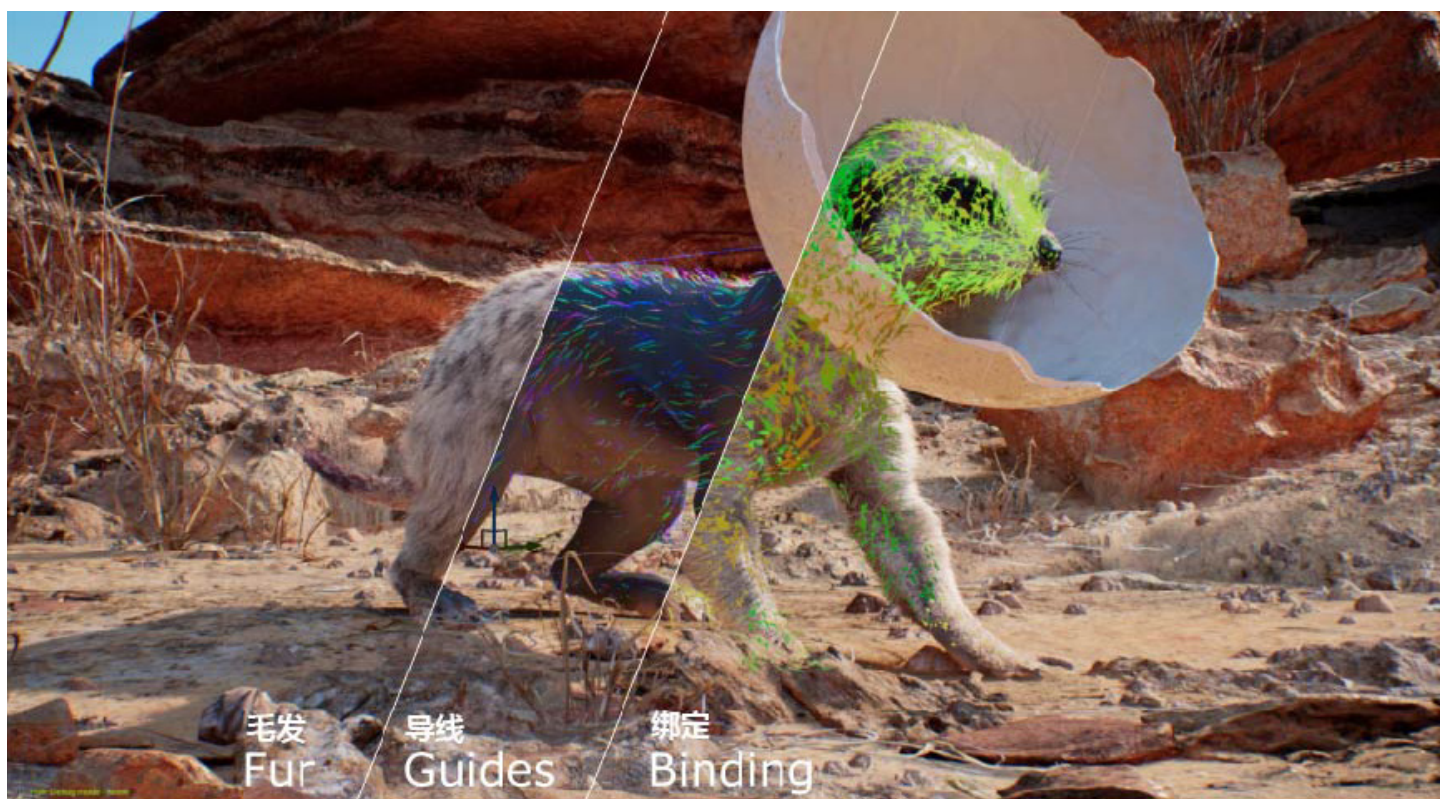


图12: 狐獾角色的毛发、导线和绑定

皮肤形变

当皮肤表面形变时，附着在表面上的发丝可能会缩紧，或是移动到不自然的位置上。睫毛、眉毛或胡须等精细的Groom有时可能难以渲染出真实感，因为它们都是由纤细的发丝在不平坦的皱缩皮肤表面上构成的。

为了解决这些问题，Groom可以选用全局形变约束。即使皮肤严重形变，这一约束也能迫使毛发尽可能地保留原状。一组径向基函数（RBF）会在附着的骨骼网格体的多个点位进行预运算，然后在运行时应用，以维持原本的Groom位置。

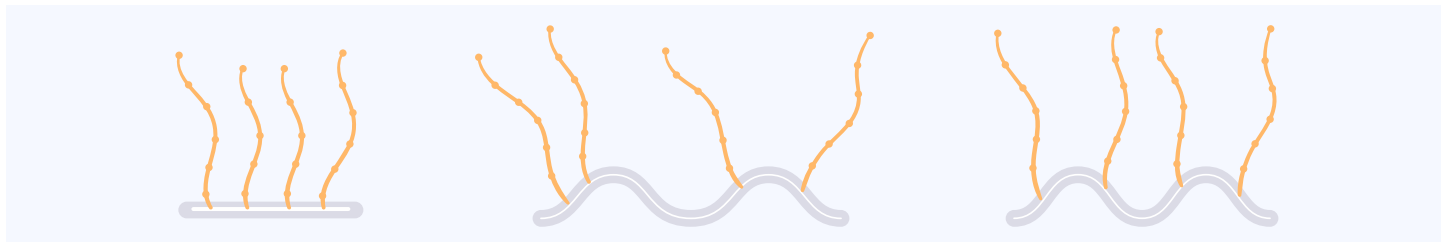


图13: 普通状态下的皮肤毛发（左侧），没有RBF约束条件时的皮肤形变（中间），带有RBF约束条件的相同皮肤/毛发（右侧）

光源近似

毛发的着色方法旨在尽可能模拟光线照射在真实毛发上的复杂表现，同时不会超出实时渲染的能力范畴。

在现实中，我们看见的毛发色彩会受到许多因素的影响：毛发本身的色彩、光线照射的方式、光的折射与散射，以及其他发丝折射的光线对其产生的影响。

下图简单展示了光线照射到毛发的发丝时发生的情况。发丝既有反光性，也有透光性。部分光线会被发丝直接反射，形成光泽和高光；其他光线则会被发丝吸收，随即散射。

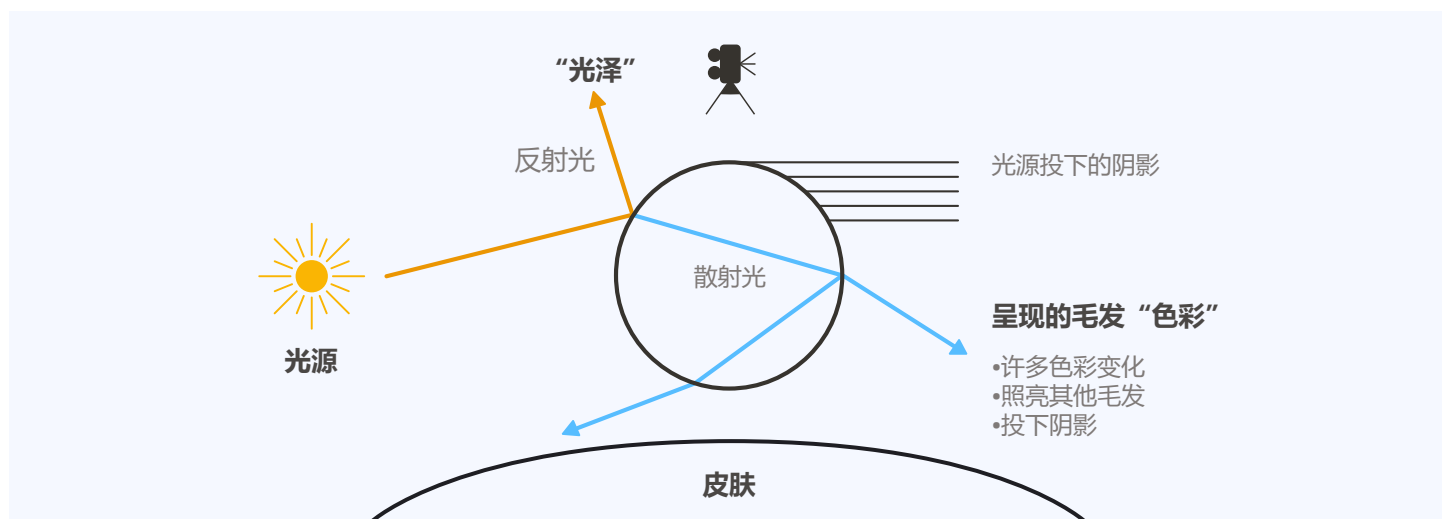


图14: 毛发的横截面图，以及在光线照射下的活动。

被发丝吸收的光线有两种散射方式：可以直接从发丝的另一侧射出，也可以在发丝内反弹后射出。这两种光线散射方式被称为“纤维散射”，以表示毛发独特而复杂的外观。这个现象众所周知，名为“双向散射分布函数”（Bidirectional Scattering Distribution Function）的数学函数也对此进行了充分描述。

另外，从纤维中逸出的光线可以散射到其他纤维中，从而导致“纤维间散射”。这种效果更为复杂，因此计算起来耗时更久。

在考虑光线影响毛发渲染的各种方式时——包括阴影、环境光遮蔽、背光等——散射光对数千根毛发发丝的作用会变得极为复杂。例如，人类只需要看一眼外表，就能轻易而举地分辨出仿冒与真实的动物毛皮，这主要是由于光对毛皮的影响方式（而不是发丝的色彩）。如果我们要创造出栩栩如生的CG毛发，就必须考虑到所有这些光照效果。

早期的CG毛发着色方式会尝试模拟光线照射到毛发时的物理效果，虽然效果确实非常逼真，但对计算的要求巨大，渲染时间极长。然而，一些较快的“捷径”运算已经被开发出来，其效果已经非常接近于真实的物理效果。虚幻引擎使用了众多近似手段，使得实时渲染时也能得到逼真的毛发效果。



图15：头发的CG渲染，结合了反射光（最明亮的区域）、散射光（中等亮度、呈现头发色彩）和只有少量光线触及的黑暗区域

以**双重散射（dual scattering）**近似为例，该方法会将复杂的散射现象分解成两个部分：全局散射会粗略/宏观地评估光线穿过纤维/毛发的方式；局部散射会通过简单的模型评估光线在单根纤维周围的反弹。与计算每个光源在一系列毛发上产生的效果相比，计算双重散射效果所需的时间会大幅减少，所获得的效果却与其他运算密集型的方法相似。

另一种能够在近似表现毛发真实感时节省时间的方法是使用**深度不透明贴图（deep opacity maps）**。这种方法会为每个光源创建专属的阴影贴图，将毛发吸收信息存储在多个阴影贴图图层里。由于贴图会跟随散发毛发的物体的形状，所以不需要太多的贴图，也能实现多个不透明阴影贴图所能取得的真实感，让这种方法的运算变得更为快捷。

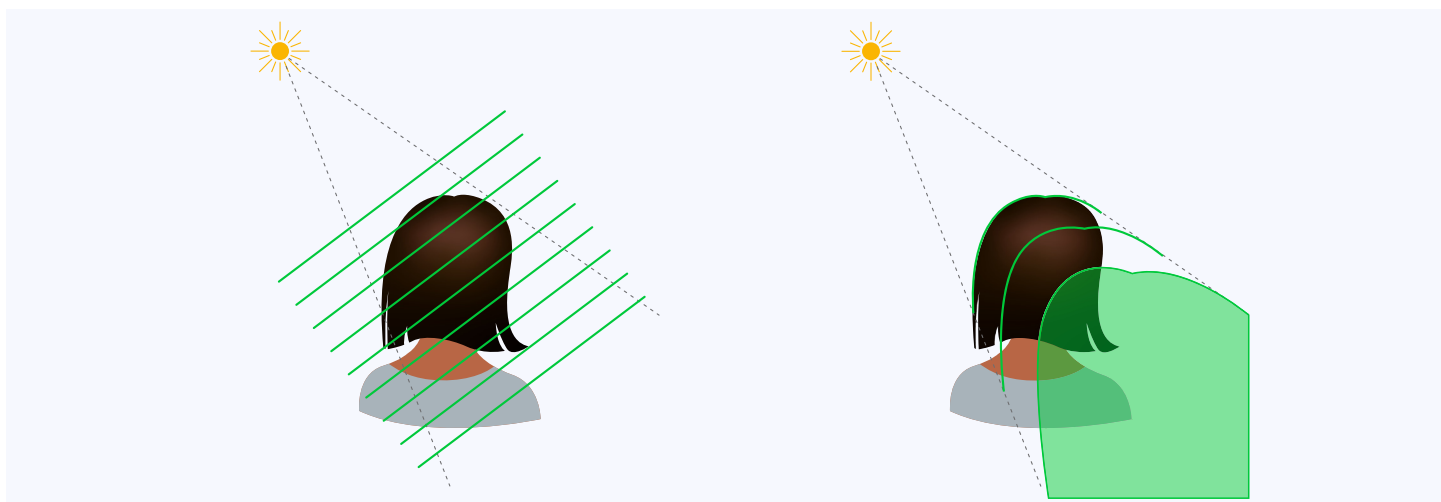


图16：不透明阴影贴图（左侧）和深度不透明贴图（右侧）的呈现

更多有关虚幻引擎近似表现实时光照行为的技巧，可参考本白皮书的[其他学习资料](#)部分。

密度体积

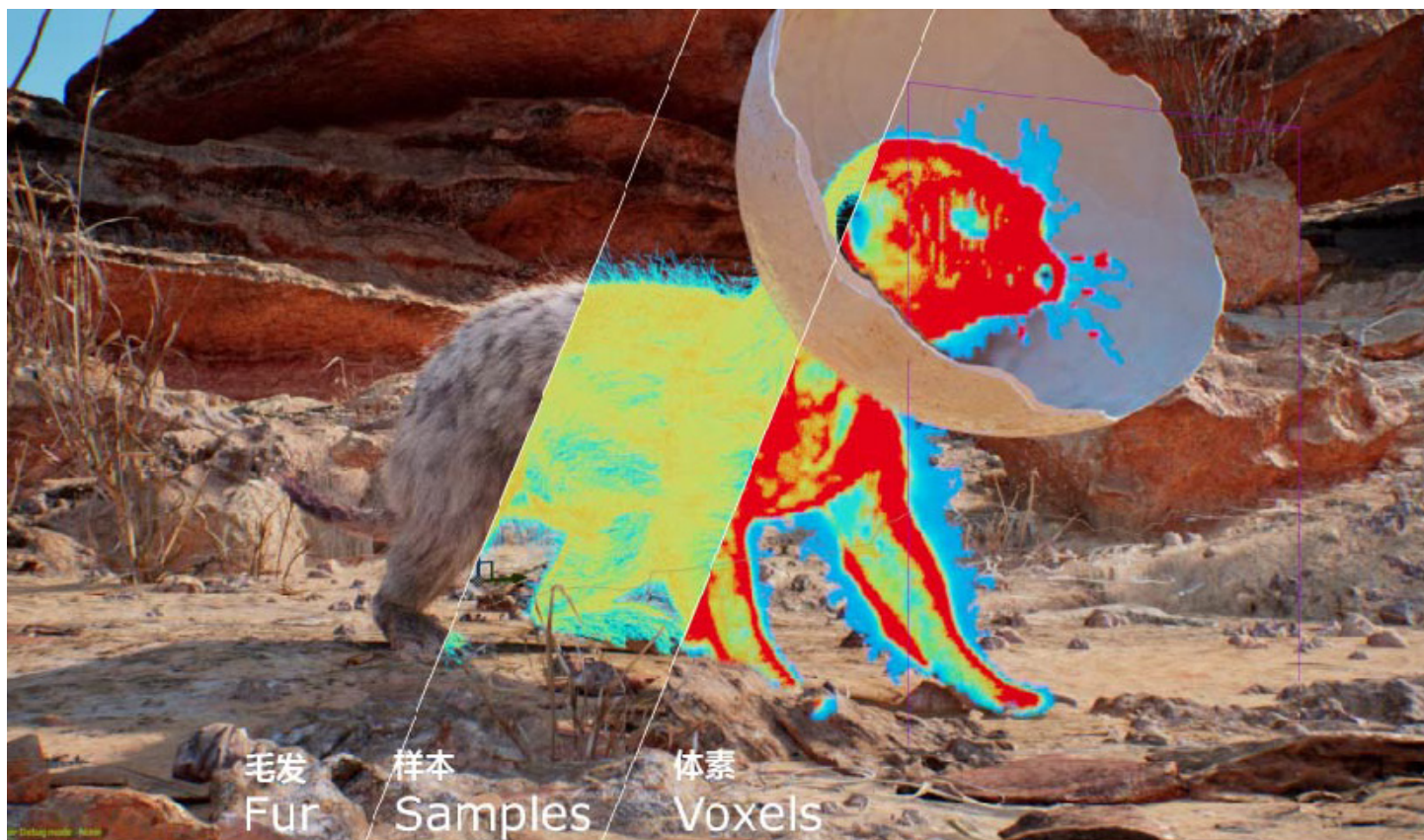


图17：密度体积、样本和毛皮成果

在虚幻引擎近似表现光照行为的技巧中，另一种方法是在每一帧中创建虚拟密度体积。密度体积提供当前视口中的皮毛密度或“厚度”信息（见图17），例如比较狐獴腿部中央毛发的“厚实”外观（绿色）和腿部边缘飘拂的毛簇（蓝色）。这些反馈会为渲染系统提供必要的信息，以迅速地近似出光线在这类毛皮密度上产生的效果。密度体积会使用体素（小型的3D占位方块），将发丝在密度体积中体素化。

密度体积是毛发着色、环境光遮蔽和环境光照的默认机制，能够获得逼真的可信的背光、自阴影和照明效果。然而在虚幻引擎中，美术师还可以选择使用深度不透明贴图，对每个光源进行设置。

在为可见发丝计算完光照后，系统就会进入下一个步骤：渲染。

渲染

为了虚幻引擎中进行渲染，导入的毛发会被转换成单独的多段线。这些曲线会被重新划分为小簇的发丝分组，以便在LOD选择和剔除时使用。

当实时系统渲染每个像素中的多根纤细发丝时，有可能会出现锯齿问题。毛发的渲染可以使用多种方法，具体取决于性能和品质预算。对于实时渲染，虚幻引擎默认的主要可视性算法结合使用了**MSAA**（多重采样抗锯齿）可视性缓冲来提取最初可见的发丝，同时也使用了亚像素覆盖缓冲。这一结合实现了性能与品质的平衡。

在制作动画时，我们更愿意牺牲性能来换取质量，因此可以使用不同的方式，动态地构建出一列可见发丝。这可以得到更好的排序和半透效果。对于这两种算法，我们都会在压缩后输出每个像素的可见发丝列表，以减少每个像素中的采样数量。

Groom会被分成小簇以运算LOD和剔除信息。在渲染Groom时，这些小簇可以根据屏幕覆盖率和可见性/遮蔽性进行细粒度的LOD选择。这样我们就能根据屏幕上实际可见的内容，相应地更改渲染与性能开销。这种方法对于大型动物的皮毛渲染尤其适用，因为Groom可能只有一部分离摄像机很近，而其他部分距离很远。

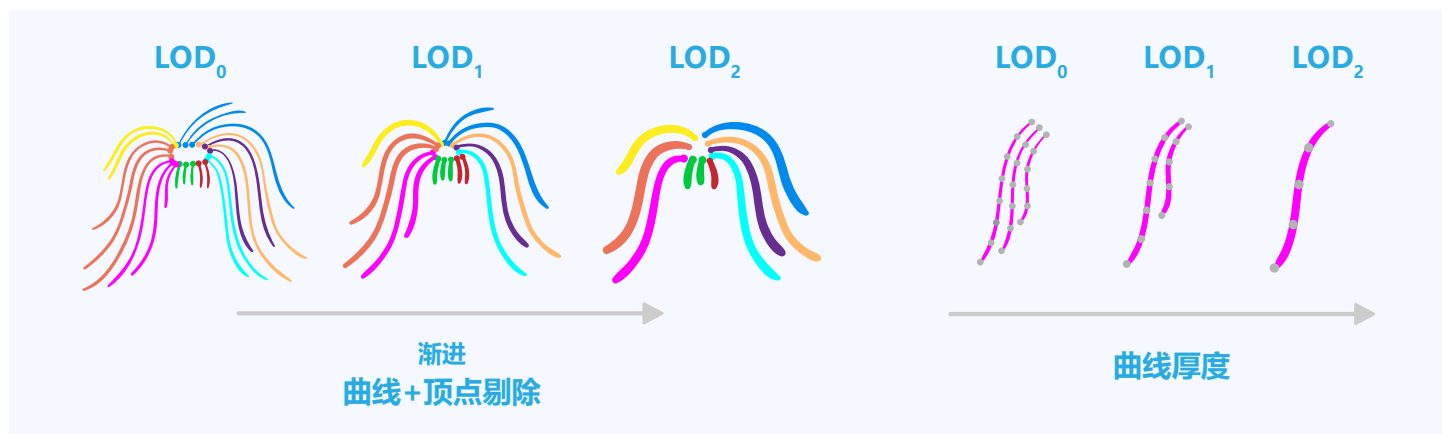


图18：配制毛发LOD的方法

出于可扩展性，Groom系统支持不同的几何表现形式，例如发片和网格体。当Groom只覆盖到几个像素，或者为了在性能较低的平台上运行时，可以切换为更简单的渲染方式。

纤细发丝的光影

由于阴影贴图或毛发体素分辨率不足，睫毛或胡须等纤细发丝在定向光源下投射的阴影可能无法保真。为了应对这一问题，在启用光线追踪时，可以直接使用毛发几何体来投射阴影，让纤细的结构能够投射出更清晰锐利的阴影。但这么做必须谨慎，因为毛发几何体越是复杂，这一选项的性能开销就越是高昂。

人的脸部其实覆盖着被称作“绒毛”的纤细发丝。和其他毛发一样，绒毛会被所有的环境光源照亮。由于其本身颜色较浅，其光照中不可忽视的一部分来自于皮肤表面的反光。在虚幻引擎中，Groom可以选择采用这种特殊的光照成分。

模拟

对头部和身体移动、风吹、重力等因素做出反应时，真实的毛发会以特别的方式运动。毛发由于这些外力所产生的动画通常是通过基于物理的模拟实现的，而这非常依赖运算。然而，为了在动画中实现照片级的真实感，在制作这些受到外力和运动影响的CG毛发模拟时，就需要创作出栩栩如生的动画。

毛发数量繁多，并且物体碰撞和自我碰撞的运动都非常复杂，因此很难模拟。毛发的动力学依赖于复杂的相互依赖关系。就动物皮毛而言，除了毛发本身的模拟作用，有很大一部分毛发动态效果是由肌肉负责驱动的。

用于模拟的导线（不一定要与毛发导线相同）会驱动相关插值曲线的形状和运动。导线的选择可能比导线的数量更重要——质量的重要性远胜于数量。

Groom中的Groom导线通常会提供某种结构，例如聚丛、方向变化、流动等。结构通常需要在模拟期间维持，所以Groom导线是模拟的良好起点。如果这些Groom导线无法提供足够的分辨率，用户可以使用Groom衍生的其他模拟导线来予以补充。如果你的主要关注点是性能，就可以减少数量以缩短运算时间。

虚幻引擎中的毛发模拟是作为**Niagara VFX**系统的一部分实现的，出于性能原因，只能在GPU上使用。解算器以XPBD（基于位置的顺应动力学模拟）为基础。为了解算所有约束条件，用户必须提供子步骤的数量和解算器迭代，以及所有发丝、碰撞和本构模型参数。解算器会以重力作用下的原始网格体（闲置姿势）为目标，以确保最终的造型尽可能地还原原本的造型。

当创建物理资产并将其加入骨骼网格体时，模拟解算器会处理身体与物理资产图元的碰撞。

在计算自碰撞时，会基于一个平均速度场计算。该速度场是通过将粒子光栅化到一个规则的体素网格体而建立的。对于本构模型，最大的挑战通常是寻找良好的毛发组构材质模型，使发丝能够在较低的解算器迭代次数下，表现得栩栩如生。这些模型控制着发丝在模拟期间的扩展、弯折和扭曲。

我们正在不断测试新的组构材质模型，看看哪些能够实现毛发的质量和性能最佳平衡。科瑟拉弹性杆和角弹簧方式已经可以在Groom资产物理属性中使用了。更多关于这些技巧的信息，可见本白皮书文末的[其他学习资料](#)部分。

皮肤暗化

在摄像机靠近人类毛发与肤色相差较大的Groom时，毛发发丝中最明显的部分就是毛干的基底。由于人类的皮肤天生比较透明，人们会注意到皮肤底下的毛球，也就是“毛囊”。毛囊会使皮肤产生微弱但可见的暗化。为了模拟这一点，Groom可以生成毛囊纹理，并在皮肤着色器中使用（Groom资产可选择该选项）。

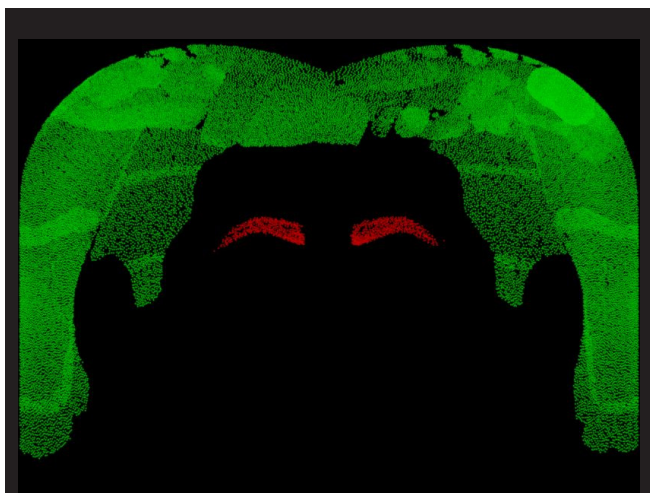


图19：毛囊贴图



观看视频

使用案例：用MetaHuman Creator制作人类毛发

为了帮助大家进一步理解实时渲染毛发，我们在此展示一些虚幻引擎中的使用案例，以及相应的实现细节。

为了展现人类毛发的Groom，我们来看一些基于**MetaHuman Creator**创建的数字人类Groom示例。这些都可以下载。

MetaHuman Creator是用于构建完整数字人类，包括面部与肢体绑定的全新工具。人物模型可以在虚幻引擎中创建动画，并且带有很多预定义的头发和脸部毛发样式。

MetaHuman Creator中的毛发选项是在DCC应用中预先创建的、随后导入虚幻引擎中的Groom。借助MetaHuman Creator中的控制选项，你可以通过混合匹配来创造毛发、眉毛、髭须和胡须的造型。在生成角色之前，你可以在MetaHuman Creator中调整Groom的颜色。

这里所提及的Groom都是使用Maya中的XGen插件制作的。

斜刘海Groom

斜刘海Groom由前方的斜刘海和后方的自然卷发构成。

斜刘海 Groom



面部Groom

Groom美术师: Marija Blašković

由于前方是刘海，后方是卷发，我们可以合理地将造型分别划入不同的图层（即分组）。图层与理发师在理发是使用的 workflow 类似，都是自下而上构建的。这一技巧有助于在向上构建造型的同时，保障毛发的精妙分层。

通过观摩参照物、观察相似发型中的卷曲造型，我们可以提出多个原则，以帮助我们构建这种发型。卷曲通常是不完全的，虽然毛发倾向于构成清晰明确的纠结螺旋，但依然是三开的，尤其是翩翩长发的上层图层。卷曲的根部通常会失去形状，逐渐地沿着发丝扭曲、相互纠缠在一起，形成有趣的形状。



图20：上色后的导线图层

发束会被分解成两三个子发束图层，每根发丝都会有随机的偏移值。另外，在不同的子发束图层中会设置不同的发束比例，以避免与主发束形状雷同。这样就能进一步分解形状，让发束有更多变化。

高频噪点可以分解笔直发丝的统一外观，发丝尖端还会添加额外噪点，进一步打破发束。拓宽发束尖端和随机化长度也能打造出更自然、更逼真的卷曲。

最后，我们会添加两到三个低频噪点图层，以加入完全杂乱蓬松的发丝图层。这些图层对于打造真实的外观功不可没。

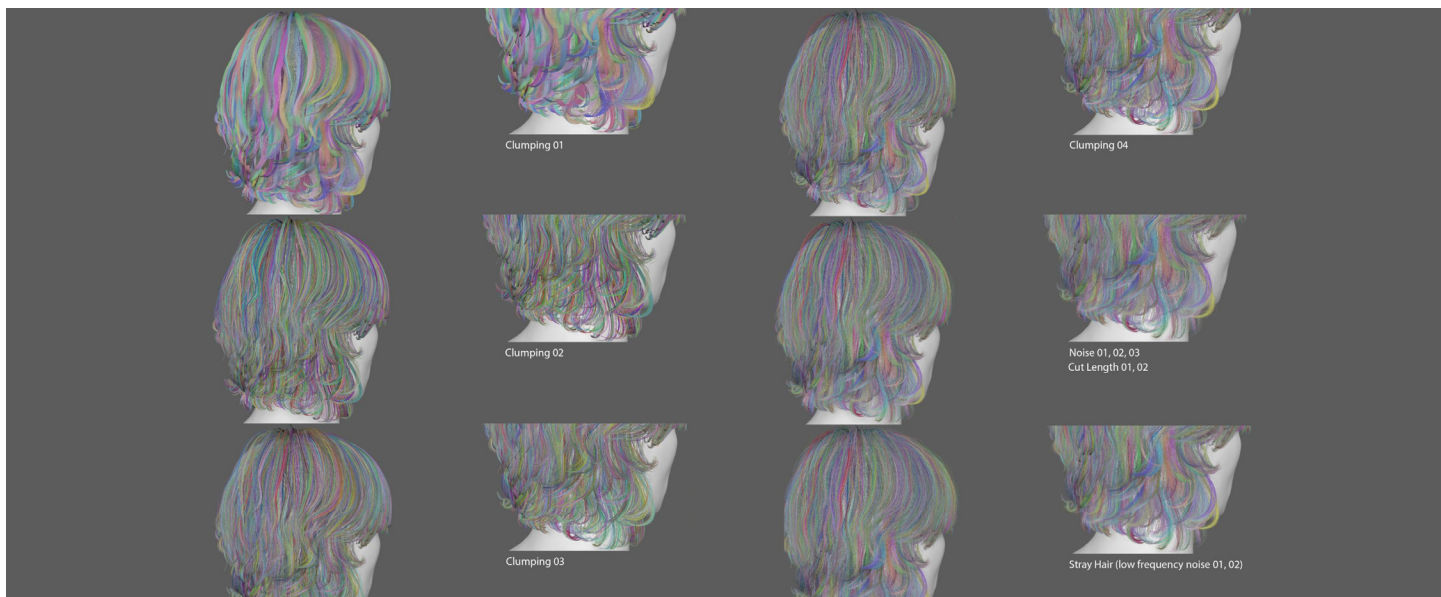
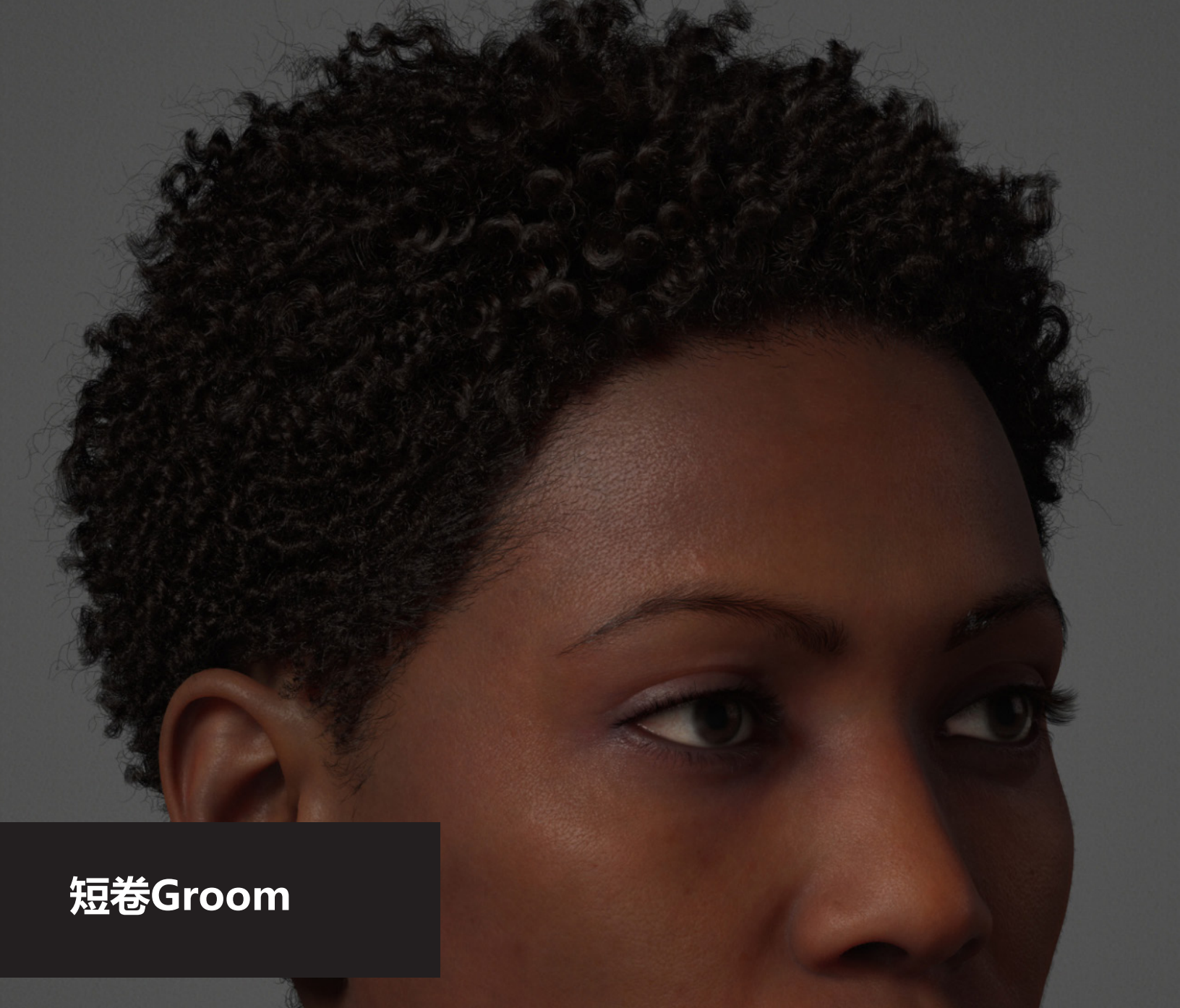


图21：堆叠后的XGen修改器，用于产生聚丛和噪点效果



短卷Groom



短卷Groom

Groom美术师: Nikola Milosavljević

短卷Groom

短卷Groom是一种带有浓密、纠缠的螺旋卷曲的短发发型。为了看起来更自然，卷曲必须具有丰富的变化，从各个角度都能清晰地看见许多不同的卷曲。前额的发际线也必须清晰可见。

为了制作初始的导线，我们选择了基于属性的Groom，使用了粗略密度遮罩和雕刻的体积网格体，也就是XGen中的MeshCut修改器。这能让我们迅速布置出毛发长度和Groom的综合形状。

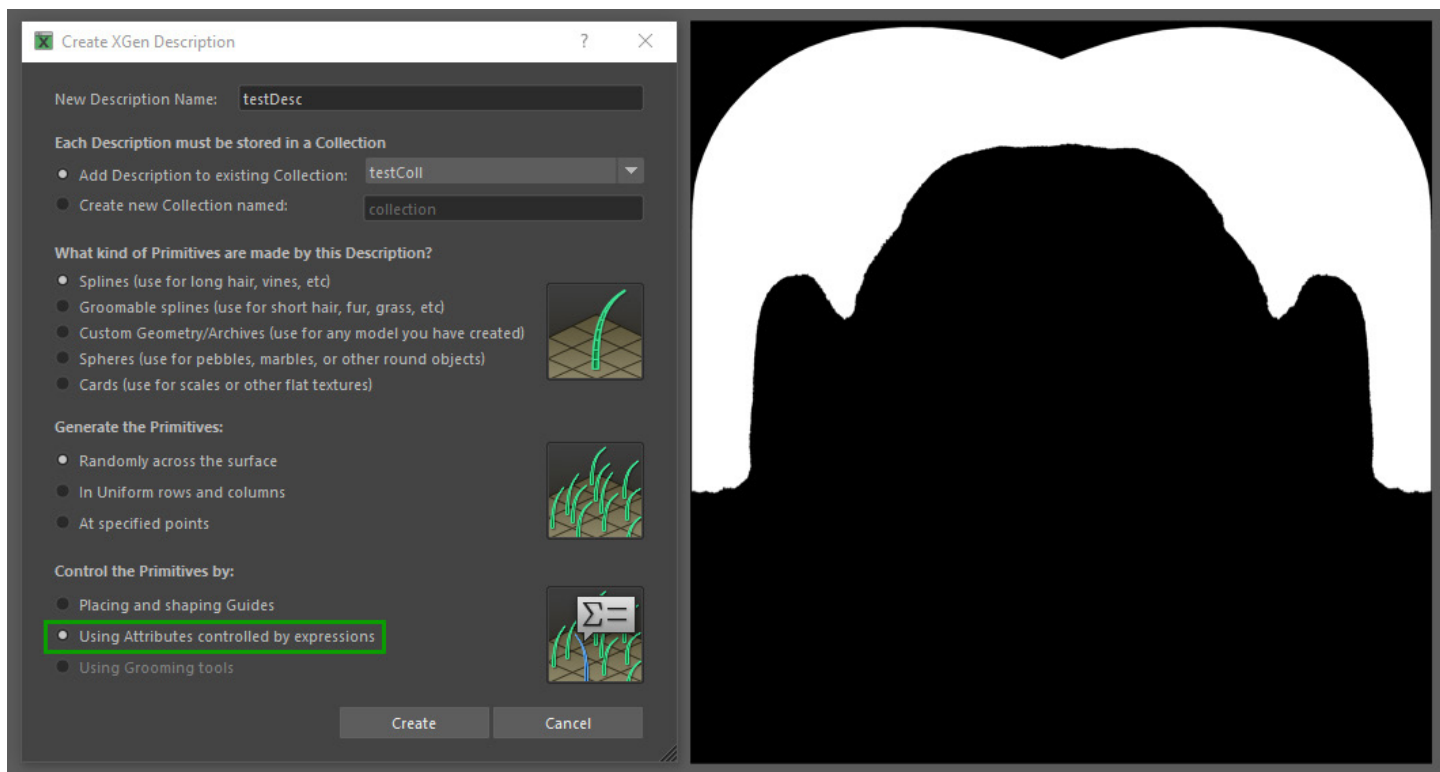


图22: XGen中的密度贴图

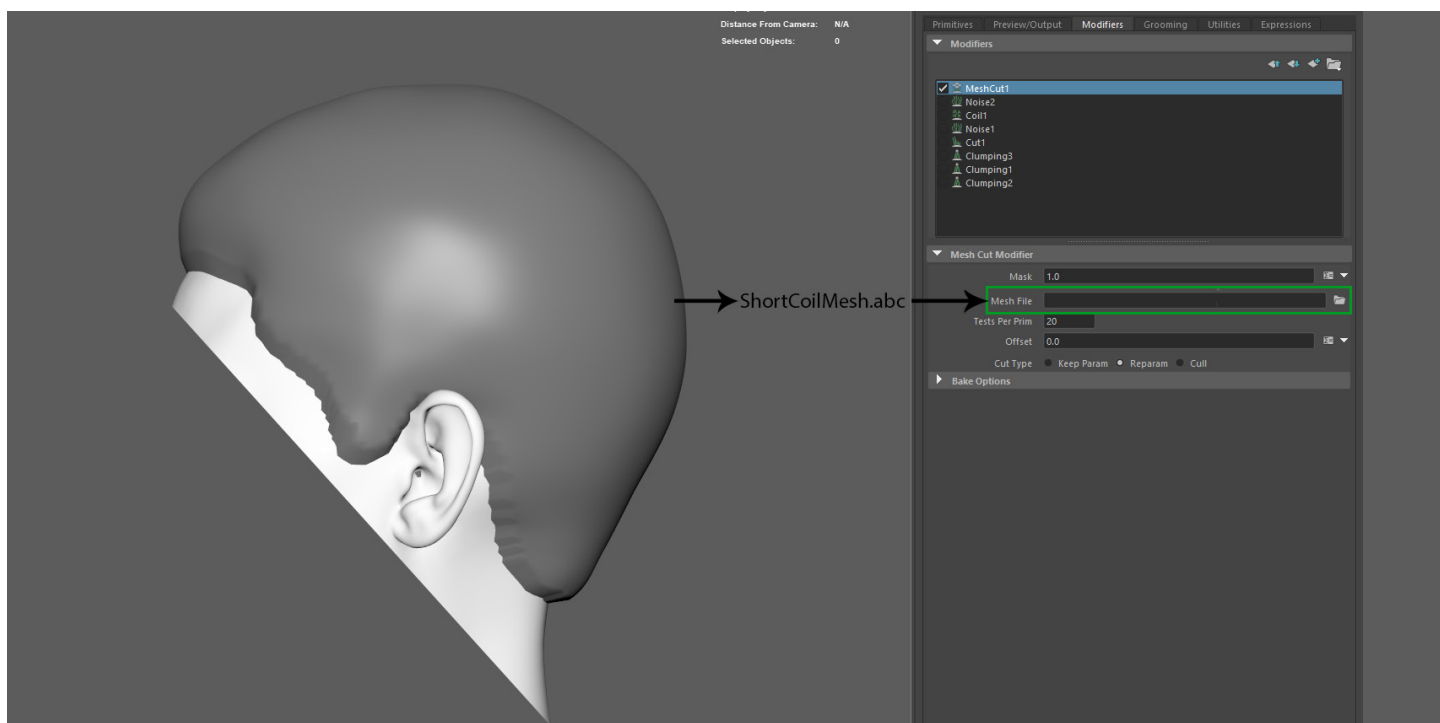


图23: 人物头部Groom的初始定义区域

一定比例的毛发会作为导线而被导出，以便在新的样条描述（Groom中的一个图层或分组）中使用。

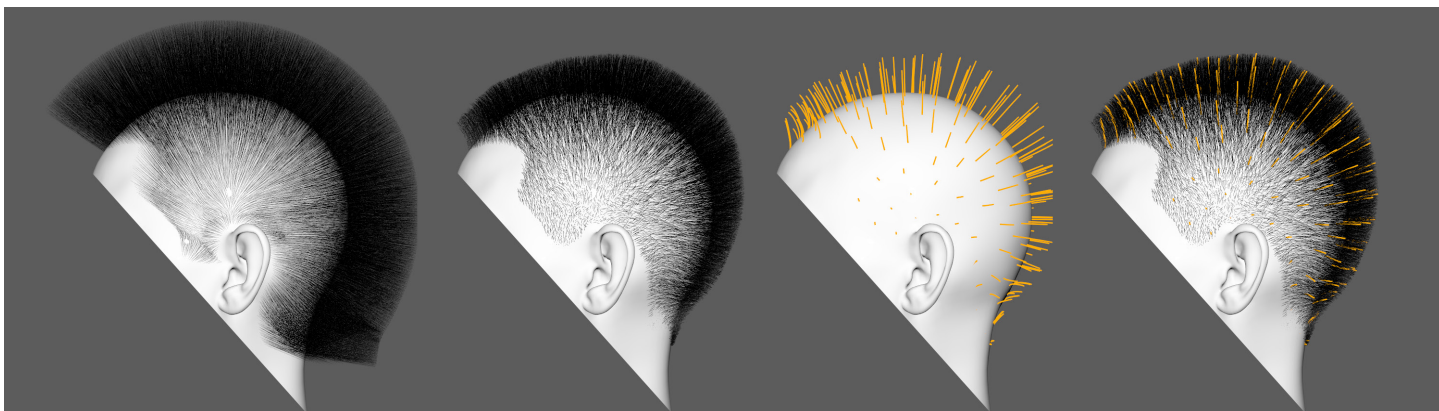


图24：用作引导毛发而导出的粗略Groom图层

雕刻新的描述，再创建基础聚丛贴图，并分为三个不同的图层来进一步控制弯卷厚度、形状、弯曲数量等参数。使用表达式来随机遮罩导出到新图层上的部分卷曲导线。这些表达式会在基础聚丛贴图的ID上创建遮罩。

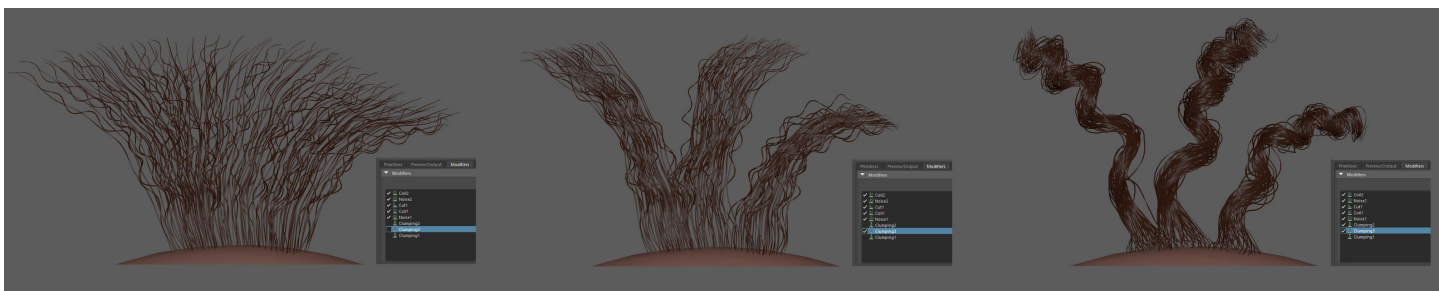


图25：三种类型的卷发Groom图层

弯卷会以程序化方式生成，在发根与发尖之间运用表达式来混合弯曲数量，并且可以略微控制混合在导线上的位置（靠近发根、发尖或中间某处）。禁用**统一控制顶点**选项可以获得更好的结果，弯卷中的形状也会更精确分明。启用**统一控制顶点**选项则会让形状更平均。

修改器控制顶点数量参数同样有着重要的作用。初始数值可以是80到100个控制顶点，卷曲的细节程度比较适宜，也不会占用过多的系统资源。控制顶点数量随后会被优化到60至70个，以便导出。

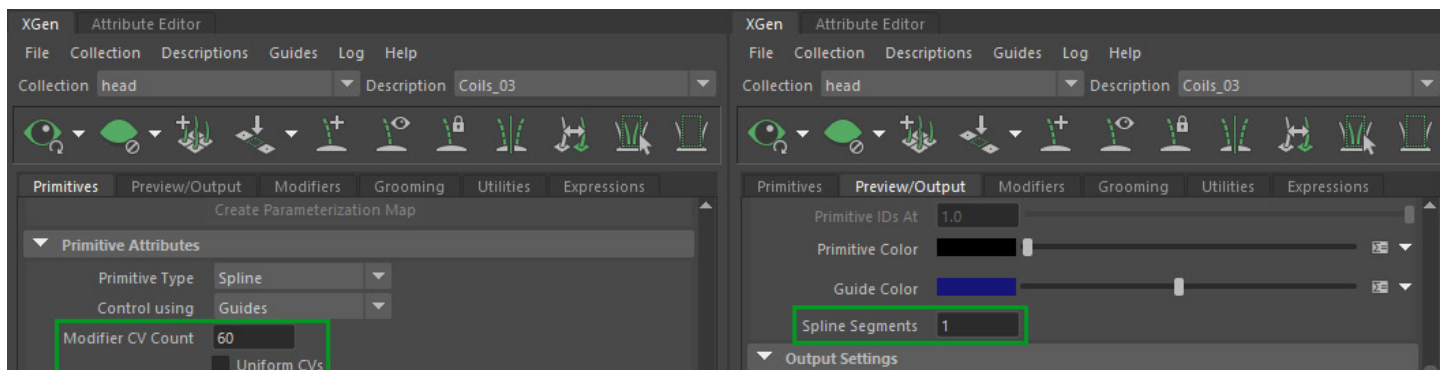


图26：XGen中的控制顶点修改

每个弯卷描述都由两个聚丛修改器控制。一个修改器的聚丛强度较低，以吸引发丝靠向导线。另一个修改器则会负责卷曲。在此基础之上，我们还使用了层叠噪点、卷曲和剪切修改器，以打破单调的外观。

描述的基础聚丛贴图衍生的贴图定义了聚丛强度。纹理被载入Mari纹理绘画软件中，投射到头皮网格体上，并且应用了**边缘检测**过滤器。随后使用**HSL**和**对比度**调整图层，产生一张结构类似于浓缩单元或蛇皮的灰度图。我们随后会从Mari中导出灰度图，在Photoshop中进一步修改，在每个单元的中央到外部边缘创造出渐变效果。

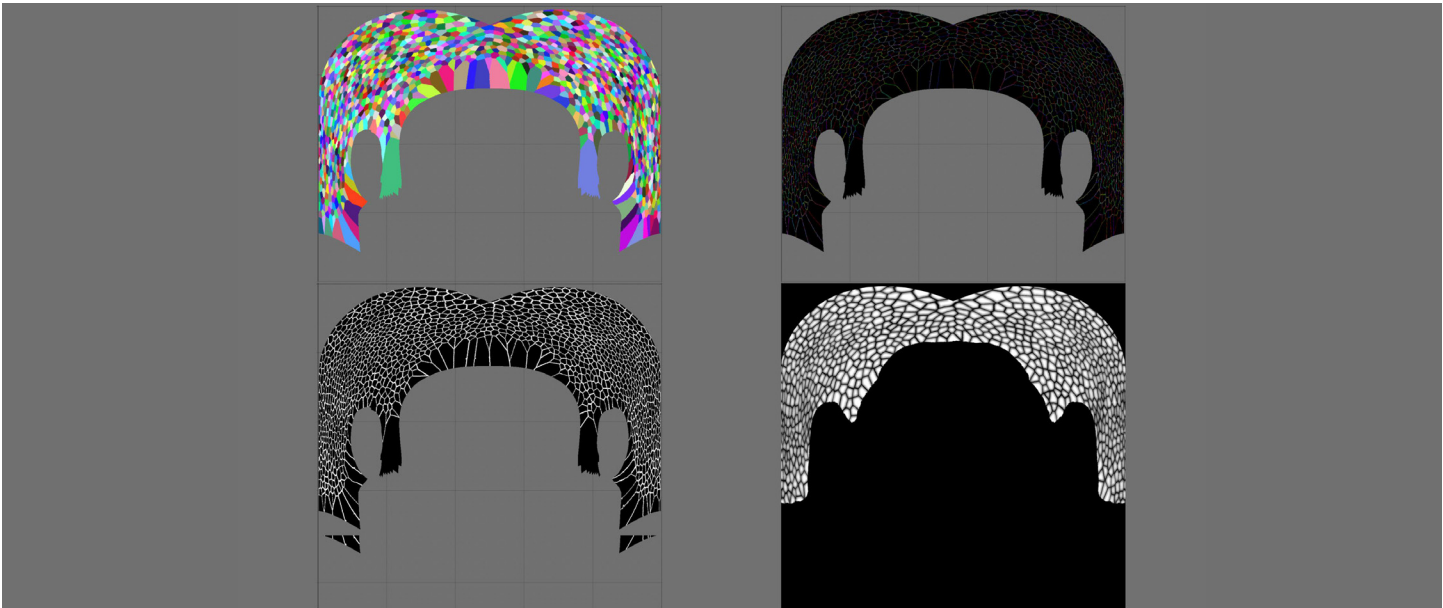


图27：左上角起顺时针：聚丛贴图、边缘检测过滤器、HSL和对比度贴图，以及经Photoshop修改的最终聚丛贴图

填充

在卷曲无法填入头皮的时候，我们有时就需要依靠填充毛发。填充描述的密度贴图的创建方式和弯卷的聚丛强度贴图相同，但是单元的边框会调成白色，单元本身则是黑色。贴图会将毛发生长限制于白色区域中，只在必要的地方增添密度。通常必须人工进行的绘画就能以此实现自动化。贴图还可以通过XGen的噪点功能另行调整。

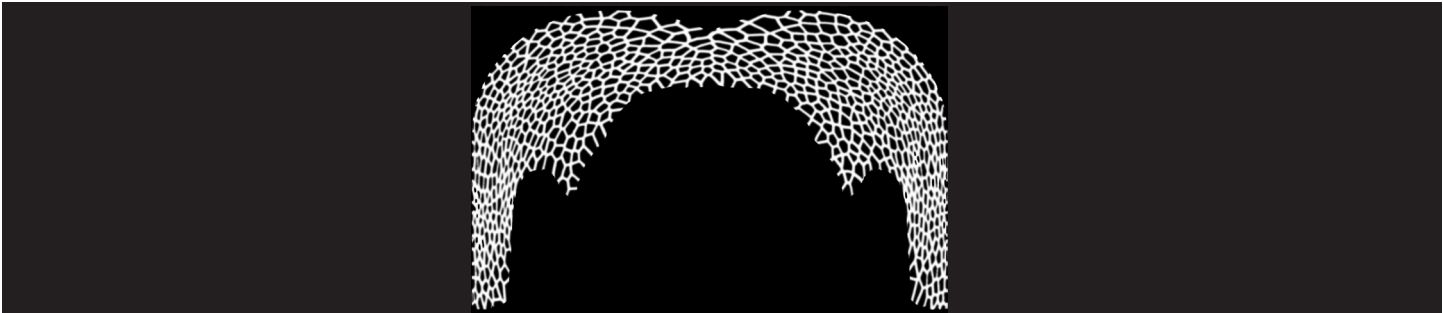


图28：填充贴图

填充毛发的弯卷效果比卷曲的弯卷效果更为松弛，噪音和弯卷修改器则会产生变化。



观看短片

使用案例：狐獴项目中的动物皮毛与羽毛

对于皮毛和羽毛使用示例，我们会看看Weta Digital的实时动画短片作品[Meerkat](#)中的狐獴和老鹰。这部影片讲述了一只狐獴和一只老鹰之间令人捧腹的遭遇。狐獴拥有实时渲染的皮毛，而老鹰则长着实时渲染的羽毛。这两个动物的Groom都是在Maya中使用Yeti插件制作的。

着色器在创建时尽可能模拟真实皮毛和羽毛的生物与形态原则。如果要真实重现狐獴皮毛和老鹰羽毛上的花纹，就必须对生长模式、与色素沉积相关的蛋白质和色素释放时间有着充分的了解。

举个例子，尾巴细长的狐獴（细尾狐獴）背部带有的斑驳图案就是特定的毛发成长图案和定时黑色素沉积结合的作用，使毛发的发丝受到至少两种灰黑色基因色素带的影响。因此，着色器必须能够控制身体周围的毛发颜色，还需要控制发丝颜色的时间选择。



图29：狐獴的参考照片

影片的两个主角，狐獴和老鹰，有多次同屏出场。但因为它们这两个动物体型差异较大，所以毛皮的光照和阴影都变得更为复杂。

如本白皮书的**光源近似**部分所述，Groom投射的阴影和光线在Groom中的传播都需要依靠体素结构。体素的大小决定了光照和阴影的精准度。体素太大会导致“斑驳”的错误外观，但体素太小又需要分配大量内存。

为了应对后一个问题，我们需要依靠稀疏体素结构，使分配仅在需要时进行。分配是由GPU驱动的，并且基于的是Groom导入期间所生成的簇。这些发束的绑定信息定义了Groom占据的空间和需要分配内存的位置。除了节约内存之外，稀疏结构还能跳过空白区域，实现更快的光线追踪。

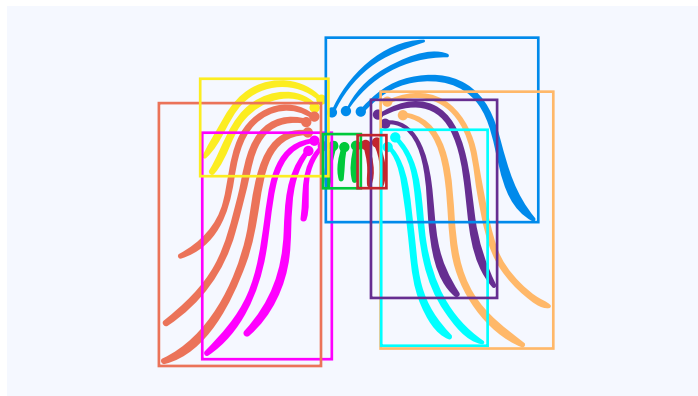


图30：稀疏体素结构仅在需要时分配内存发束

环境光遮蔽和天光需要运用这个结构来投射正确的遮蔽效果，在Groom中正确传播光线。虚幻引擎提供了多种计算天光贡献的方法，以便在精准度和速度方面有所取舍。在**狐獴 (Meerkat)** 项目中，我们使用了最准确的版本，通过投射多道光线穿过毛皮，妥善地整合了天光在毛皮和羽毛材质上的贡献，然后精准计算散射光线。

这两个角色的纹理都是直接使用Substance Painter绘制在角色皮肤上的。这些纹理都是包含了RBG遮罩的低分辨率贴图，并会在着色器中通过遮罩控制色彩和图案。这里有两个主要遮罩——一个定义了控制着身体色彩空间的区域，另一个定义了控制发丝与条纹计时的区域。

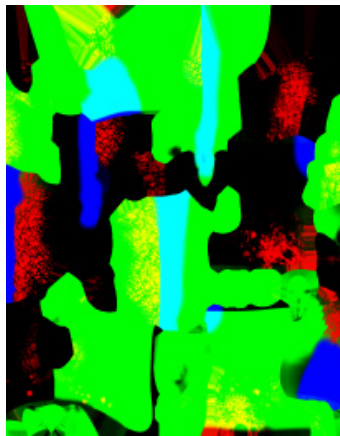
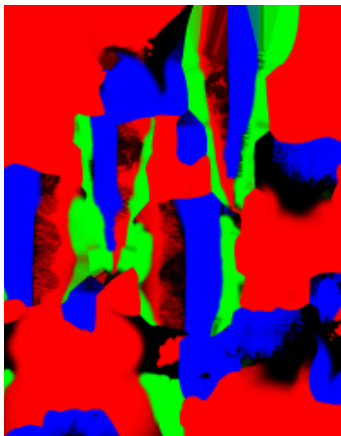


图31: 老鹰的空间区域 (左侧) 和定时区域 (右侧) 遮罩

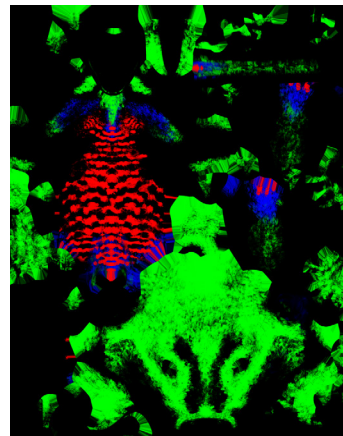
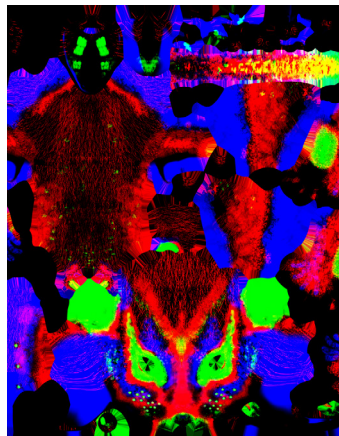


图32: 狐獴的空间区域 (左侧) 和定时区域 (右侧) 遮罩

纹理贴图会根据皮肤的UV空间生成，能够使用**根UV**属性定义整根发丝的色彩。

狐獴皮毛

由于狐獴的动作很快，为了营造出准确的动态模糊和抗锯齿，就必须将上一帧的光照重新投射到当前帧中，因此精准的运动向量非常关键。为了实现这一点，此前的发丝位置会被缓存，运动向量的运算于是变得和位置差异运算一样简单。

虚幻引擎中的皮毛实时性能非常关键，因此选择合适的皮毛密度非常重要。狐獴的皮毛密度只有普通影片特效中的25%。降低密度并权衡取舍宽度有助于让实时的毛发更逼真、更精细。

狐獴身体的Groom是作为一个整体单独设计的，但对于精细的脸部毛发，例如胡须、睫毛和其他独特元素，我们都单独创建了分组。



图33：虚幻引擎界面中的狐獴Groom

就这只狐獴而言，胡须和睫毛都是手动布置的引导曲线，然后在Yeti节点网络中将其直接转换为发丝。

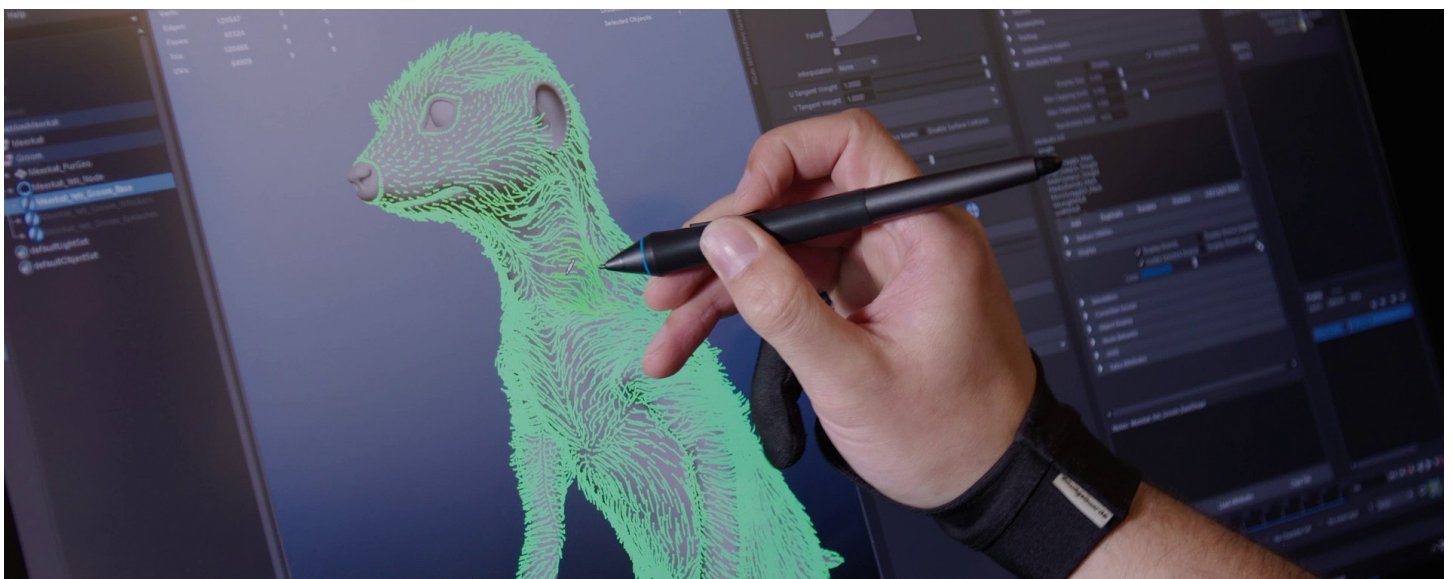


图34：绘制狐獴的引导皮毛

老鹰羽毛

羽毛的复杂本性使老鹰变成了一大独特的挑战。Yeti会使用羽毛图元来构建羽毛，羽轴实际上等同于“发丝”，羽支则生长在羽轴上，随后羽毛会在老鹰全身实例化。

在老鹰飞向、行走与移动时，每根羽毛都会因为动画而形变，这意味着多根羽毛都需要一个绑定。另外，在虚幻引擎中，发丝只能绑定在几何体上。于是我们为每根羽轴创建一条实例几何体，并且在Maya中完成绑定。羽支会作为毛皮导入到虚幻引擎里，随后绑定在几何体羽轴上。

老鹰羽毛需要手动放置，才能在确保结构和外观的同时，满足绑定的要求。



图35: 绘制羽支位置

对羽毛而言，羽轴和羽支的色素施放时间决定了羽毛上看似复杂的图案的构成。羽毛着色器遵循的空间与计时区域原则与狐獴的毛发发丝相同。主要差别在于，羽轴旁羽支的朝向和位置极为复杂。羽毛上所见的图案和羽毛几何体密不可分，由于它们会对最终外观产生深远的影响，所以必须特别关注他们的构建。

羽毛着色器原本是在多边形平面上绘画的色彩贴图在Substance Painter中的投射。老鹰全身共有13种不同类型羽毛。这些贴图随后还会使用羽轴和羽支的UV坐标重新映射。

不幸的是，这么做立即就会引发问题，羽毛的图案很难产生自然的变化，也难以通过迅速高效的方式控制色调和数值。一个更为程序化的方式则是在发丝中使用色素沉积原则，以产生更为有机的外观。



图36：最终实时渲染中的老鹰视图

我们在导入流程的部分中谈到，纹理贴图是使用皮肤的UV空间生成的，并运用**根UV**属性定义整根发丝的色彩。对狐獴皮毛来说，这么做非常自然。但对于羽毛来说，**根UV**参数只能定义单独的整根羽毛。

为了获得特定的图案，我们就需要更多毛发发丝的属性：一个属性会定义羽轴旁的羽支位置（0为根部，1为尖端），另一个属性会指明羽毛所属的一侧（内侧与外侧羽毛）。色彩斜坡回用于创建毛发发丝的色彩图案/数值，以控制发丝上的条纹。

总结

实时渲染毛发系统依然是个新鲜事物，还有许多发展的余地。

在未来发布的毛发系统中，Epic Games的目标是聚焦质量与性能。对于毛发的着色和渲染，我们将通过提升参数化和光照响应来改进光源近似。

我们还希望改进碰撞响应和材质，并进一步优化处理大型高密度Groom时的性能。

我们预期会在虚幻引擎的未来版本中添加更多功能，让前沿的实时渲染毛发系统能够更进一步。



其它学习资料

公式

[双向反射分布函数 \(BRDF\)](#)

[径向基函数 \(RBF\)](#)

技术论文

[人类头发纤维的光线散射](#)

[虚幻引擎中基于物理的毛发着色](#)

[双重散射近似](#)

[深度不透明贴图](#)

[基于位置和方向的科瑟拉弹性杆](#)

[卷发的艺术模拟](#)

相关软件

[MetaHuman Creator](#)

关于本文

作者

Gaëlle Morand

Charles de Rousiers

Michael Forot

贡献人员

Shawn Dunn

Nathan Farquhar

Darrin Wehser

Marija Blašković

Nikola Milosavljević

编辑

编辑: Michele Bousquet

排版: Jung Kwak