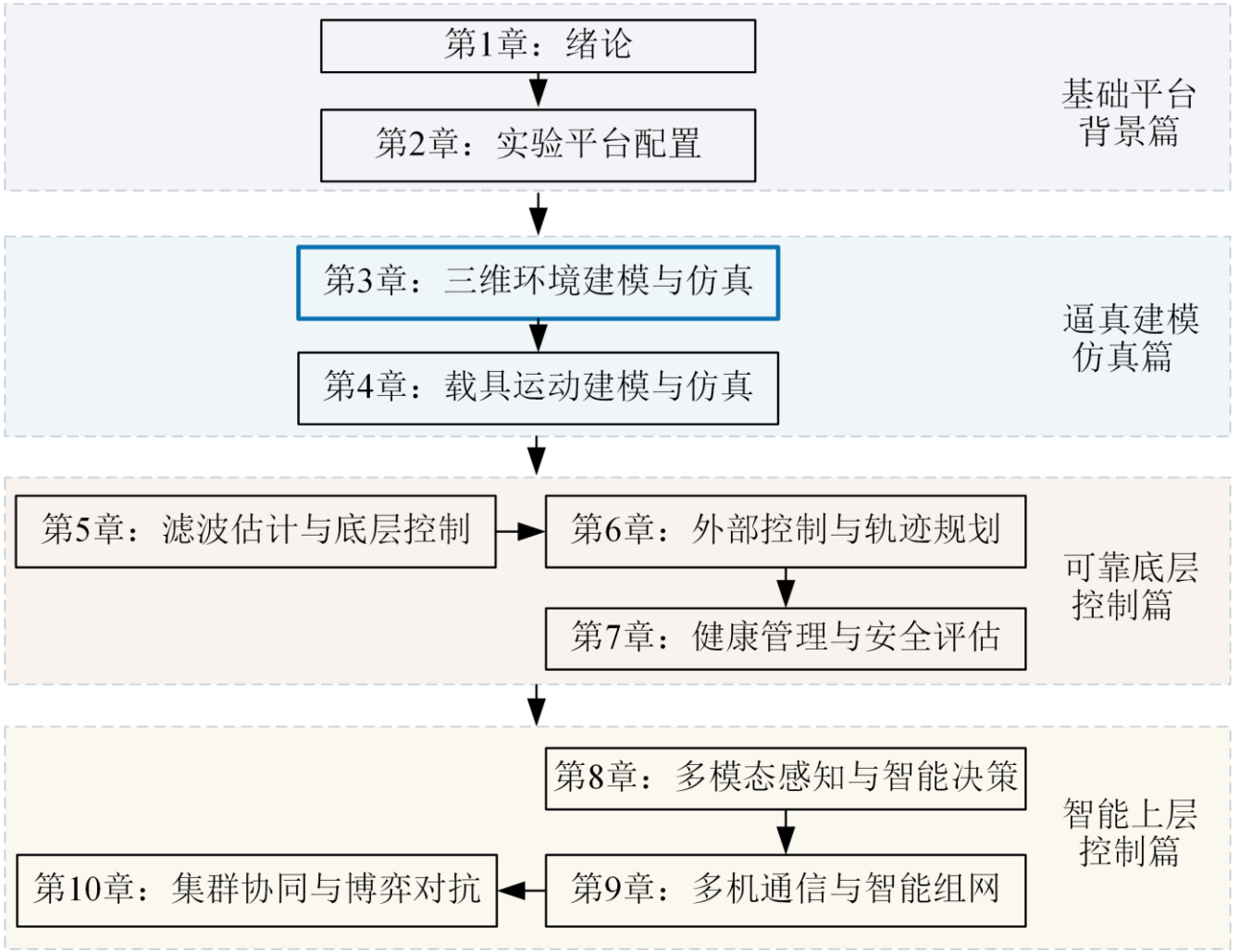
智能无人系统控制实践

第三章　三维环境建模与仿真

作者：戴训华　中南大学
主页：faculty.csu.edu.cn/daixunhua
邮箱：dai.xh@csu.edu.cn
平台官网：rflysim.com

配套教材





教材：戴训华，全权，任雪峰. 智能无人系统控制实践[M]. 北京：电子工业出版社，2026.

本章学习目标

学完第三章能够完成什么

1. 解释几何、材质、场景与传感器观测的关系
2. 按资源来源选择建模、处理与场景制作工具
3. 配置对象与接口，获取有坐标和时间含义的数据
4. 复现配套例程，保存场景并解释一次改动的结果

三维仿真的价值，在于可重复地改变环境并观察算法的反应。

本章目录

三维仿真基础与工具选型

第三章 三维环境建模与仿真

3.1 三维仿真基础与工具选型

3.2 UE场景运行渲染与交互

3.3 多模态传感器数据生成

3.4 分布式仿真架构与数据接口

3.5 场景与载具资源开发

3.6 例程体系与综合实践

3.7 习题案例分析与课程设计

本节 5 个教学单元

3.1.1 背景知识与三维仿真流程
教材对应：第3.1.1—3.1.2节

3.1.2 三维仿真流程·模型建立与编辑
教材对应：第3.1.2节·模型建立与编辑

3.1.3 三维仿真工具·建模与场景制作
教材对应：第3.1.3.1、3.1.3.3节

3.1.4 三维仿真工具·其他辅助工具
教材对应：第3.1.3.3节；采集见3.1.2

3.1.5 三维仿真工具·平台比较与选型
教材对应：第3.1.3.1—3.1.3.2节

3.1.1 背景知识与三维仿真流程：三维仿真的资源与观测

资源、场景、状态与观测

- **三维资源：**网格、材质、纹理或机器人结构等数据，可在不同场景中复用。
- **场景：**把地形、建筑、载具、光源和传感器组织到同一空间中。
- **运行状态：**记录对象的位置、姿态与部件运动；由运动模型或脚本持续更新。
- **传感器观测：**在指定状态与配置下生成图像、深度、点云等算法输入。
- **任务例子：**同一小屋模型在路边放两次形成场景；载具前进改变状态，相机随之看到不同遮挡。

资源描述对象，场景组织对象，状态与传感器共同决定观测。

3.1.1 背景知识与三维仿真流程：任务环境与观测条件

环境差异怎样进入无人系统任务



图：不同环境中的三维场景

- **空间条件：**地面起伏、建筑间距和障碍位置决定可行路径。
- **观测条件：**纹理、反光和阴影影响目标识别与定位。
- **任务条件：**同一场景可服务空中、地面等不同载具。

地形与遮挡影响通行，表面与光照影响视觉线索。

3.1.1 背景知识与三维仿真流程：仿真与AI控制闭环

三维环境与感知控制闭环

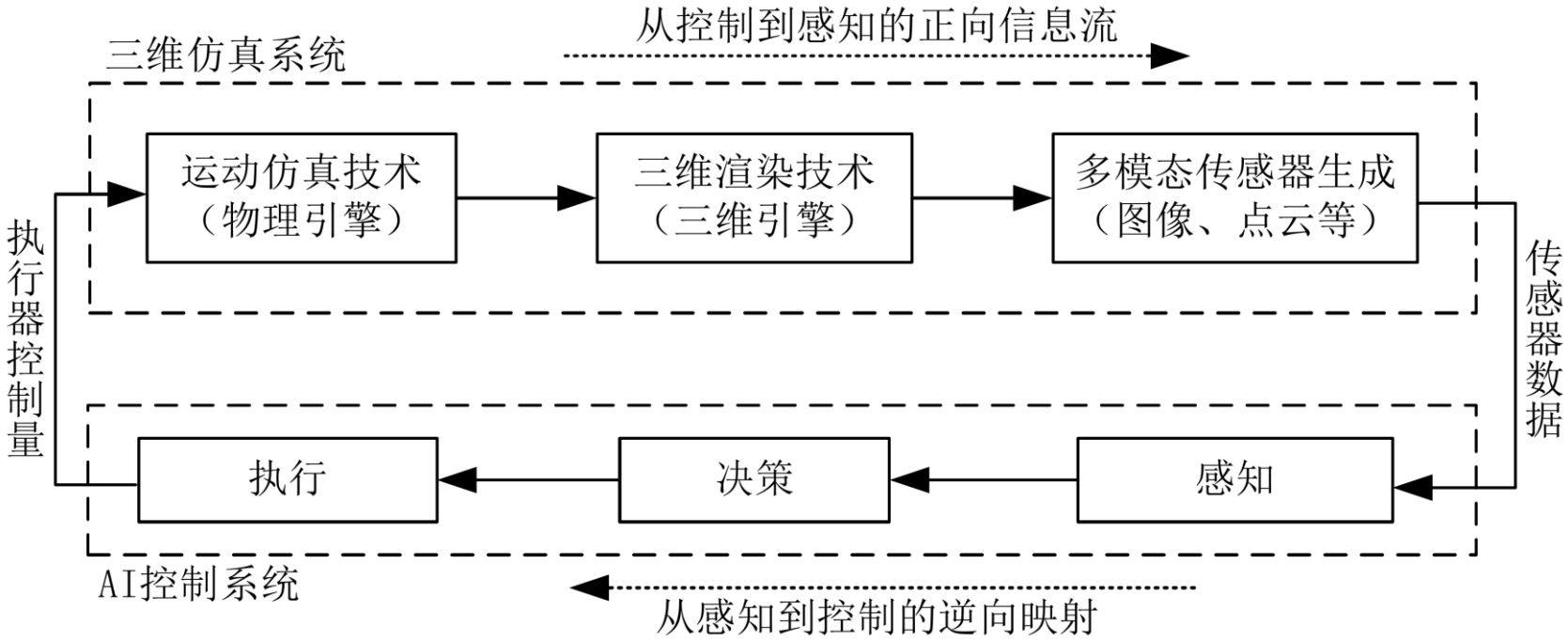


图3.1 三维仿真与AI控制系统关系

状态进入场景，场景形成观测，观测参与感知、决策与控制。

3.1.1 背景知识与三维仿真流程：任务作用

同一场景怎样支撑感知、规划与训练

- **感知与融合：** 固定相机和激光雷达配置，比较遮挡变化后的观测。
- **规划与协同：** 重放同一障碍布局，比较路线和多载具避碰结果。
- **训练与评估：** 只改变天气或光照，观察算法性能是否稳定。
- **人机交互：** 显示状态与事件，让操作者检查、干预并复盘决策。

三维仿真为同一任务提供可重复的环境、观测和交互条件。

3.1.1 背景知识与三维仿真流程：渲染、运动与传感器

渲染、物理与传感器的职责

环节	输入与产物	任务中的作用
渲染	场景、光照与相机 → 图像	提供可见外观
物理与运动	力、控制与约束 → 运动状态	计算运动与接触响应
传感器	场景与采样配置 → 观测数据	形成图像、距离等观测
交互与接口	按键、脚本与消息 → 运行操作	连接算法和实验条件

图像、运动响应与传感器数据来自不同计算环节。

3.1.1 背景知识与三维仿真流程：三维仿真流程

资源、场景与观测串起仿真流程

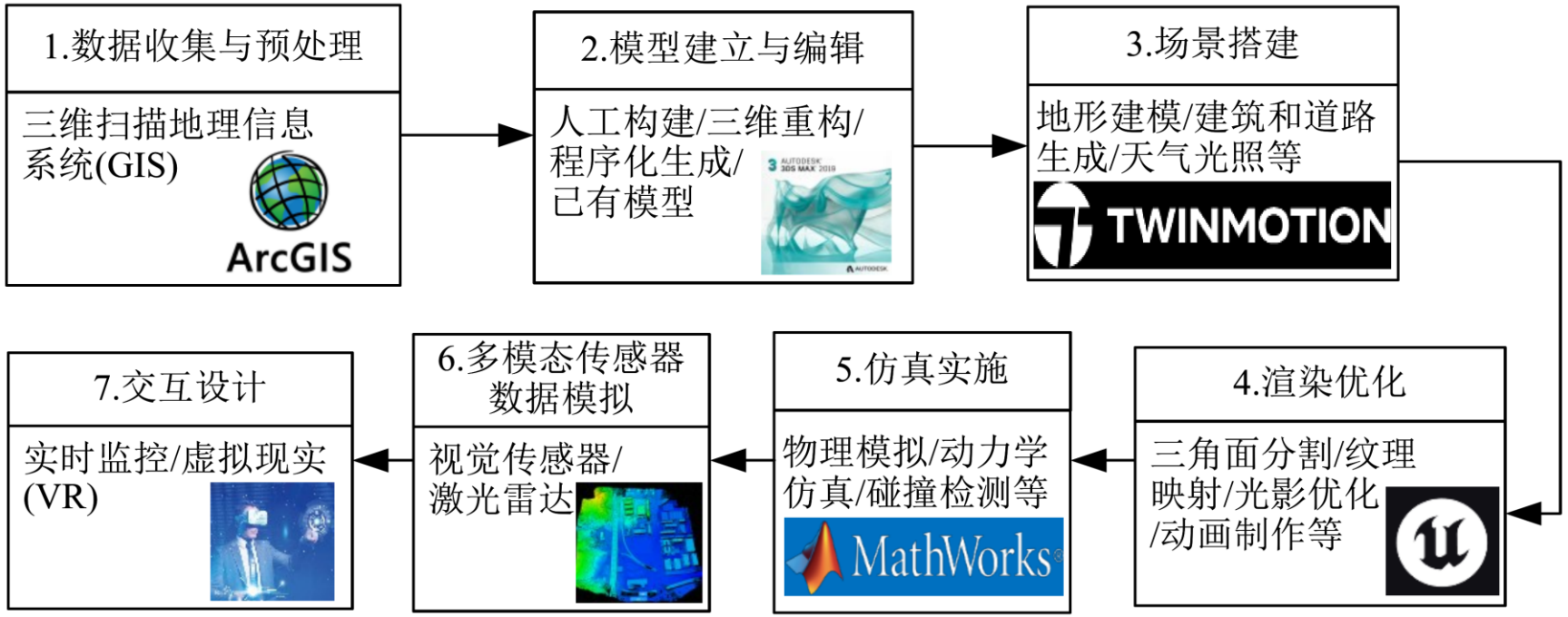


图3.2 三维仿真实现路线图

按已有输入进入相应阶段；静态背景可以跳过动作制作。

3.1.1 背景知识与三维仿真流程：前三阶段

从现实数据到可运行场景

- **数据收集：**扫描、影像或地理数据进入清理与坐标核对。
- **模型建立：**把形状、外观和动作整理成可复用的对象资源。
- **场景搭建：**按尺度与坐标放置地形、建筑、载具、光源。
- **交付检查：**确认原点、单位和材质，再进入运行与观测环节。

每个阶段交付可检验的资源；已有资源时可跳过重新采集。

3.1.1 背景知识与三维仿真流程：数据来源与资源复用

三种资源起点对应不同工作量

资源起点	新增工作	本章对应例程
已有模型或场景	导入、放置与参数配置	loadSimpleMesh / StarterContent
设计几何	建模、部件组织、材质	3dsMaxUsage / SketchUpUsage
真实环境数据	自采：扫描与清理 复用：加载与定位	自采：3DScannerScene 复用：Cesium

先复用可用资源，缺少几何或真实环境时再增加制作环节。

3.1.1 背景知识与三维仿真流程：实时性与渲染质量

真实感与运行速度的任务取舍

- **视觉识别：**保留目标纹理、边缘和重要受光变化，使目标在图像中可辨。
- **通行与避障：**保留障碍轮廓、距离和通道宽度，远处微小装饰可以简化。
- **计算成本：**可见几何、光照和多相机视图增加渲染负担，影响数据到达速度。
- **比较方法：**固定布局与相机轨迹，只改变一个因素，才能解释结果差异。

保留会改变任务判断的细节，把计算量花在有效观测上。

3.1.1 背景知识与三维仿真流程：技术难点

逼真场景为何还要管时序与算力

- **实时性：**相机30帧/秒时，每帧总预算约33毫秒。
- **多传感器：**图像与点云采样频率不同，需核对时间戳和坐标。
- **大场景：**细节、范围和动态对象越多，加载与更新越吃紧。
- **算力竞争：**渲染、物理与算法共享CPU/GPU，须按目标观测分配。

真实感、时序一致和运行速度需要按任务共同取舍。

本章目录

三维仿真基础与工具选型

第三章 三维环境建模与仿真

3.1 三维仿真基础与工具选型

3.2 UE场景运行渲染与交互

3.3 多模态传感器数据生成

3.4 分布式仿真架构与数据接口

3.5 场景与载具资源开发

3.6 例程体系与综合实践

3.7 习题案例分析与课程设计

本节 5 个教学单元

3.1.1 背景知识与三维仿真流程
教材对应：第3.1.1—3.1.2节

3.1.2 三维仿真流程·模型建立与编辑
教材对应：第3.1.2节·模型建立与编辑

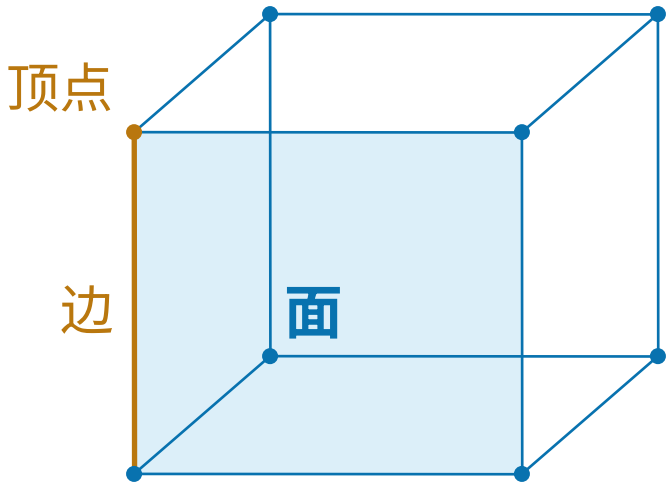
3.1.3 三维仿真工具·建模与场景制作
教材对应：第3.1.3.1、3.1.3.3节

3.1.4 三维仿真工具·其他辅助工具
教材对应：第3.1.3.3节；采集见3.1.2

3.1.5 三维仿真工具·平台比较与选型
教材对应：第3.1.3.1—3.1.3.2节

3.1.2 三维仿真流程·模型建立与编辑：网格与拓扑

网格：顶点、边、面与连接关系



示意：浅蓝区域表示一个面

· **顶点与边：** 顶点记录三维位置；边连接顶点，勾勒表面的边界。

· **面与拓扑：** 面表达表面区域；面之间的连接关系决定开口与连续性。

· **编辑结果：** 移动顶点会改变形状，删除一个面会使表面出现开口。

顶点给出位置，拓扑给出连接，网格共同确定物体表面。

3.1.2 三维仿真流程·模型建立与编辑：几何与外观

几何密度与表面外观



(a) 较少多边形　(b) 较多多边形　(c) 三维模型

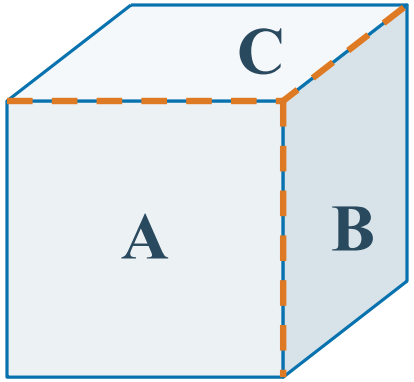
图3.3 几何形状定义和拓扑关系示意图

- **轮廓：** 多边形过少时，曲面和边缘更容易显出棱角。
- **外观：** 颜色与受光还由纹理和材质决定，并非只由面数决定。

3.1.2 三维仿真流程·模型建立与编辑：UV与材质

UV展开：三维表面怎样对应二维纹理

① 三维模型



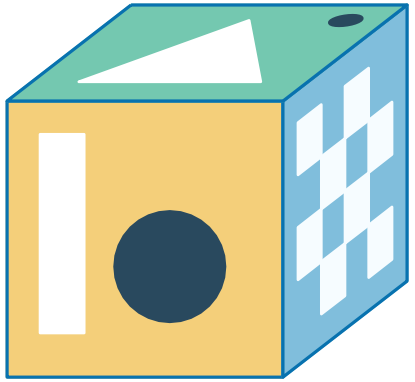
展开

② UV岛叠加在已有纹理上



取样

③ 贴图后的模型



示出前、右、上三面

橙色虚线为展开接缝

三个面均显示对应纹理

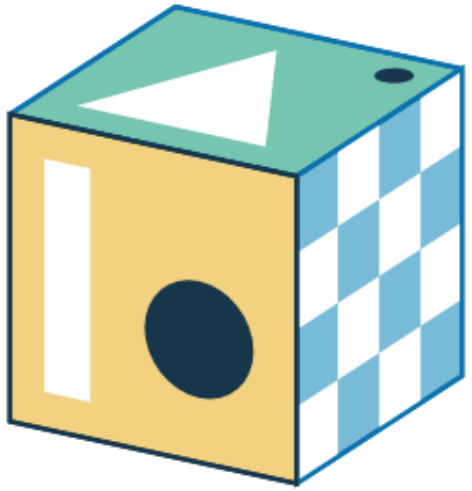
- **UV展开：**沿接缝把三维表面排成二维UV岛，确定纹理坐标（U、V）。
- **纹理取样：**按U（横向）、V（纵向）读取图像颜色，贴回对应表面。
- **材质：**组合纹理与粗糙度等参数，规定表面如何响应光照。

几何决定表面位置，UV定位纹理，材质决定受光表现。

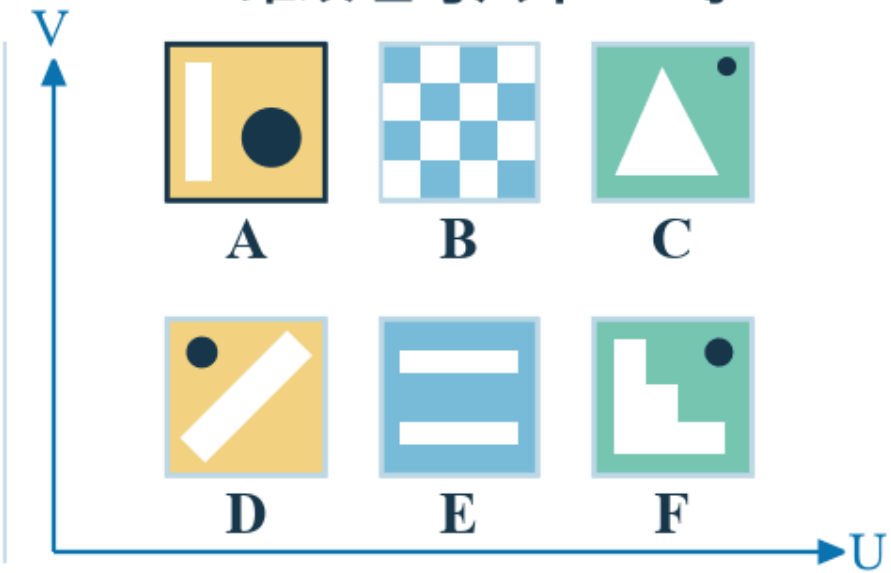
3.1.2 三维仿真流程·模型建立与编辑：UV与材质

沿接缝展开，再按UV把纹理贴回表面

贴图后的立方体



二维纹理与六个UV岛



A前 B右 C上 D左 E下 F后

图：六面展开是裁片类比；实际UV编辑建立坐标对应，不改变模型外形

3.1.2 三维仿真流程·模型建立与编辑：资产与实例

模型文件、引擎资产与场景实例

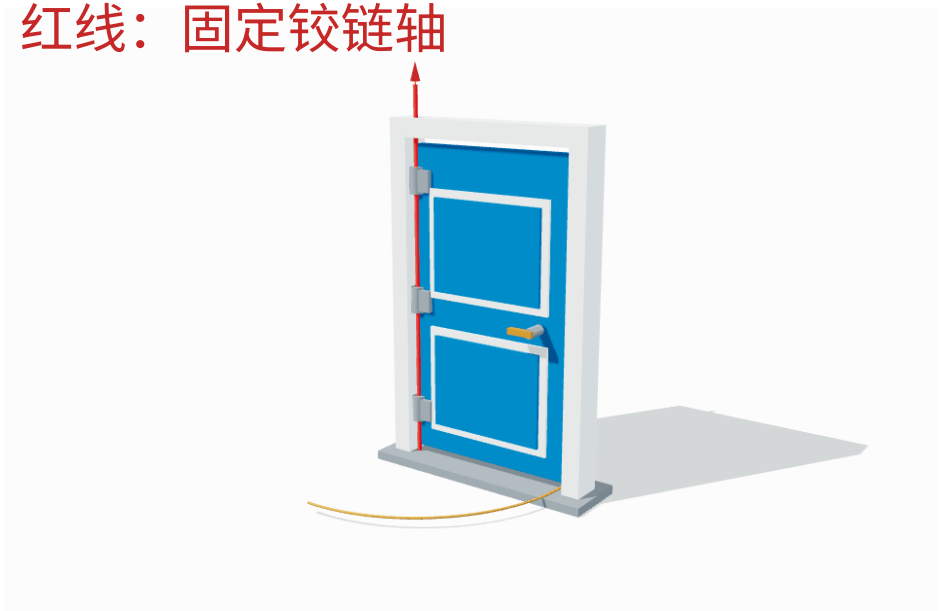
box.fbx
交换文件 ➡ 网格资产
可复用资源 ➡ 两个场景实例
分别放置

- **虚幻引擎（UE）资产：**导入后的网格与材质可被多个对象共享，避免重复制作。
- **独立变换：**实例分别记录位置、旋转和尺度；重摆一个实例不改变源文件。

同一个资产可被多个实例引用，各实例有自己的位置和状态。

3.1.2 三维仿真流程·模型建立与编辑：尺寸与转轴

尺寸、原点和转轴决定导入后的几何关系



图：2m门板绕侧边铰链开合（示意）

- **尺寸：**2m与2cm相差100倍，导入单位影响与载具的相对大小。
- **原点：**底部原点便于落地放置，中心原点便于对称旋转。
- **部件转轴：**门在铰链处转动，旋翼在电机轴处转动。

导入前校准尺度、原点与转轴，物体才能正确落位并运动。

3.1.2 三维仿真流程·模型建立与编辑：文件用途

常见文件格式按数据用途区分

类型	常见格式	主要保留的信息
网格交换	FBX / OBJ / STL	以几何为基础；外观与动作随具体格式保存
运行时资源	glTF / GLB	场景层级、网格与表面数据
机器人描述	统一机器人描述格式（URDF）等	部件、关节与惯性
制作工程与场景	SketchUp（SKP） Twinmotion（TM）	保留工具中的编辑组织

选格式要看交接目标：外形、机器人结构与场景组织各不相同。

3.1.2 三维仿真流程·模型建立与编辑：网格交换格式

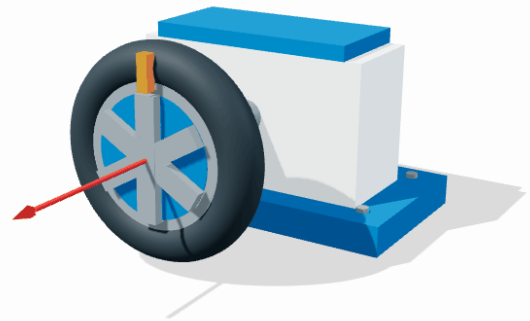
FBX、OBJ与STL分别交接什么

格式	主要交接内容	具体选择与条件
FBX	网格、层级、动画； 可转换的材质信息	带部件和动作交接； 复杂材质可能需要重建
OBJ	静态几何与UV； 外观常依赖MTL和纹理	带外观的静态网格； 附属文件一起保存
STL	以三角面保存表面几何	只需交接形状； 不承担完整外观和动画

3.1.2 三维仿真流程·模型建立与编辑：URDF结构

用连杆与关节描述机器人

连续旋转关节：不设角度限位



图：底座与车轮的关节运动（示意）

URDF入门模型与源码：https://github.com/ros/urdf_tutorial

· **结构文件：**采用可扩展标记语言（XML），用标签记录部件与连接关系。

· **连杆（link）：**一个刚性部件，可引用网格作为外观。

· **关节（joint）：**连接父子连杆，规定相对位姿、运动类型和轴。

URDF把部件连接成结构树，网格文件负责部件外形。

3.1.2 三维仿真流程·模型建立与编辑：URDF属性

URDF中的外观、碰撞与惯性



结构示意：三类信息分别参与计算

- **visual**：可见几何、材质和相对位姿，决定部件怎样显示。
- **collision**：接触检测使用的几何，常采用更简单的形状。
- **inertial**：质量、质心和转动惯量，参与动力学计算。

同一连杆可用精细外观、简化碰撞体和独立惯性参数。

3.1.2 三维仿真流程·模型建立与编辑：URDF复用边界

URDF跨平台复用的范围

平台	使用方式	需要继续配置的部分
RflySim	导入URDF机器人资源	平台控制、观测与实验配置
Isaac Sim	转换成通用场景描述（USD）资产	驱动、碰撞与传感器属性
MuJoCo	读取URDF；原生模型使用MuJoCo模型格式（MJCF）。	执行器、接触与仿真选项

MuJoCo模型与URDF扩展：https://mujoco.readthedocs.io/en/stable/modeling.html

复用机器人结构与网格，再按目标平台配置驱动、接触和传感器。

本章目录

三维仿真基础与工具选型

第三章 三维环境建模与仿真

3.1 三维仿真基础与工具选型

3.2 UE场景运行渲染与交互

3.3 多模态传感器数据生成

3.4 分布式仿真架构与数据接口

3.5 场景与载具资源开发

3.6 例程体系与综合实践

3.7 习题案例分析与课程设计

本节 5 个教学单元

3.1.1 背景知识与三维仿真流程
教材对应：第3.1.1—3.1.2节

3.1.2 三维仿真流程·模型建立与编辑
教材对应：第3.1.2节·模型建立与编辑

3.1.3 三维仿真工具·建模与场景制作
教材对应：第3.1.3.1、3.1.3.3节

3.1.4 三维仿真工具·其他辅助工具
教材对应：第3.1.3.3节；采集见3.1.2

3.1.5 三维仿真工具·平台比较与选型
教材对应：第3.1.3.1—3.1.3.2节

3.1.3 三维仿真工具·建模与场景制作：工具分工总览

从几何制作到运行：按任务选工具

几何制作
按需选工具 ➡ **场景环境**
Twinmotion可选 ➡ **UE编辑器**
组织可运行场景 ➡ **RflySim3D**
运行与观测

- **几何：** 3ds Max、Blender或SketchUp；制作网格、材质与纹理。
- **资产复用：** 已有模型或Fab资产可直接导入UE，跳过从零建模。
- **环境：** 在UE中布置场景；Twinmotion可补材质、植被与光照。
- **运行：** UE组织对象与行为；RflySim3D连接运动状态与观测。

几何、环境和运行各有产物；已有资产可复用，无需走完全部工具。

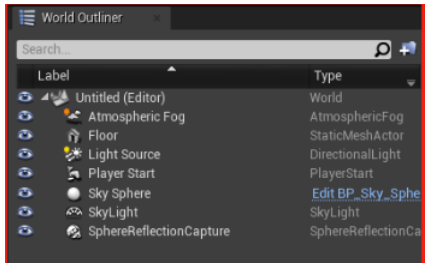
3.1.3 三维仿真工具·建模与场景制作：UE编辑与运行

UE编辑器与RflySim3D运行程序

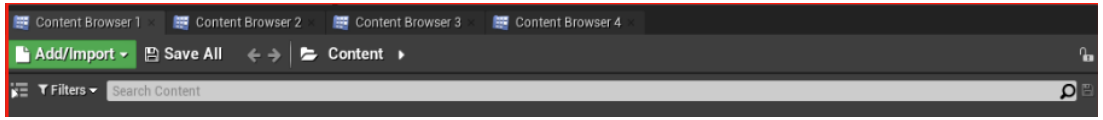
场景视口
预览三维场景



对象列表
场景对象（Actor）



内容浏览器：管理模型与材质资源



图：UE编辑器的三个区域（局部）

· **虚幻引擎（UE）**：组织场景、渲染、交互与可扩展行为。

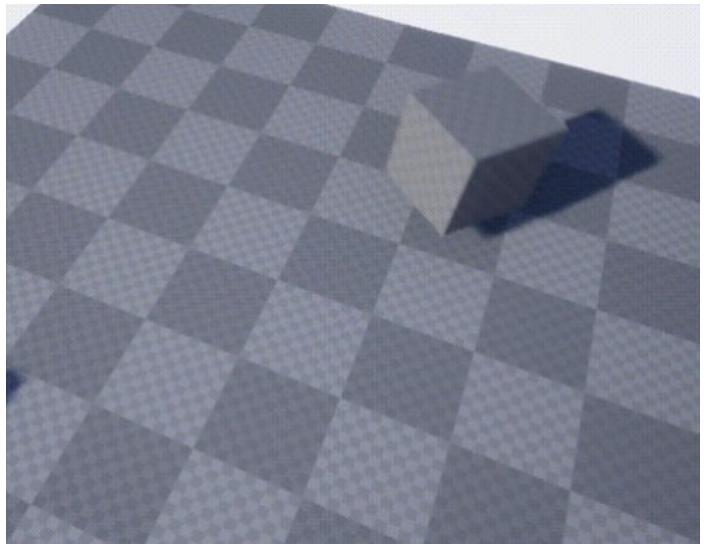
· **编辑侧**：盒体例用UE4.27.2；新建空白工程，制作关卡与蓝图。

· **运行侧**：RflySim3D加载资源，连接对象状态与传感器。

编辑器制作场景与行为，运行程序加载资源并响应实验输入。

3.1.3 三维仿真工具·建模与场景制作：蓝图与C++

蓝图与C++怎样分工



图：蓝图盒体例程中的运行场景

- **蓝图：**可视化脚本组织事件、变量和操作，便于查看行为关系。
- **C++：**扩展组件、插件与接口，可与蓝图配合。
- **运行验证：**编译保存蓝图，拖入关卡后点击Play，观察盒体旋转。

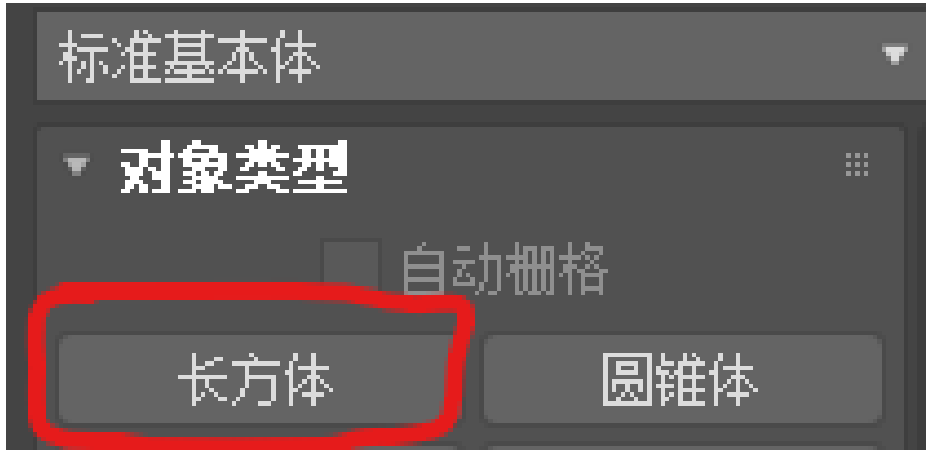
UE关卡、资产与蓝图文档：https://dev.epicgames.com/documentation/en-us/unreal-engine

蓝图便于连接场景行为，C++便于扩展底层能力与原生接口。

3.1.3 三维仿真工具·建模与场景制作：3ds Max建模

精细网格与部件怎样编辑

创建面板：标准基本体 → 长方体



图：长方体创建选项（局部）

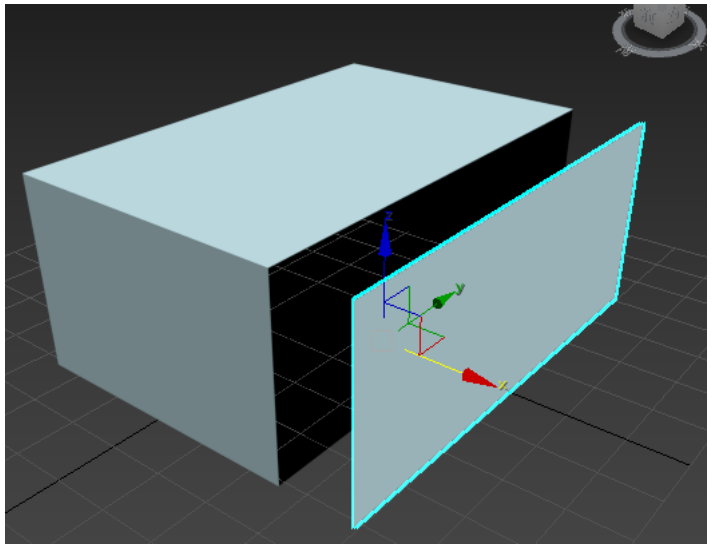
- **职责：** 建立和修改网格，组织独立部件与动画。
- **优势：** 多边形编辑和变换控制细，适合载具部件加工。
- **产物：** 保留制作工程，导出box.fbx等交换资源。

3ds Max建模与导出帮助：https://help.autodesk.com/view/3DSMAX/2024/ENU/

3ds Max把尺寸与部件要求转成可交换的模型资源。

3.1.3 三维仿真工具·建模与场景制作：3ds Max对象组织

分离对象与合并网格



图：盒体面板分离为独立对象

- **分离（Detach）：**分离为对象后，选定面可单独移动或旋转。
- **附加（Attach）：**多个部分合为网格，便于统一管理。
- **迁移应用：**机身与舵面分开，后续再配置转轴和控制。

独立运动的部分保留独立对象，静态部分可按需要合并。

3.1.3 三维仿真工具·建模与场景制作：模型交接

3ds Max模型交接：几何与依赖一起保存

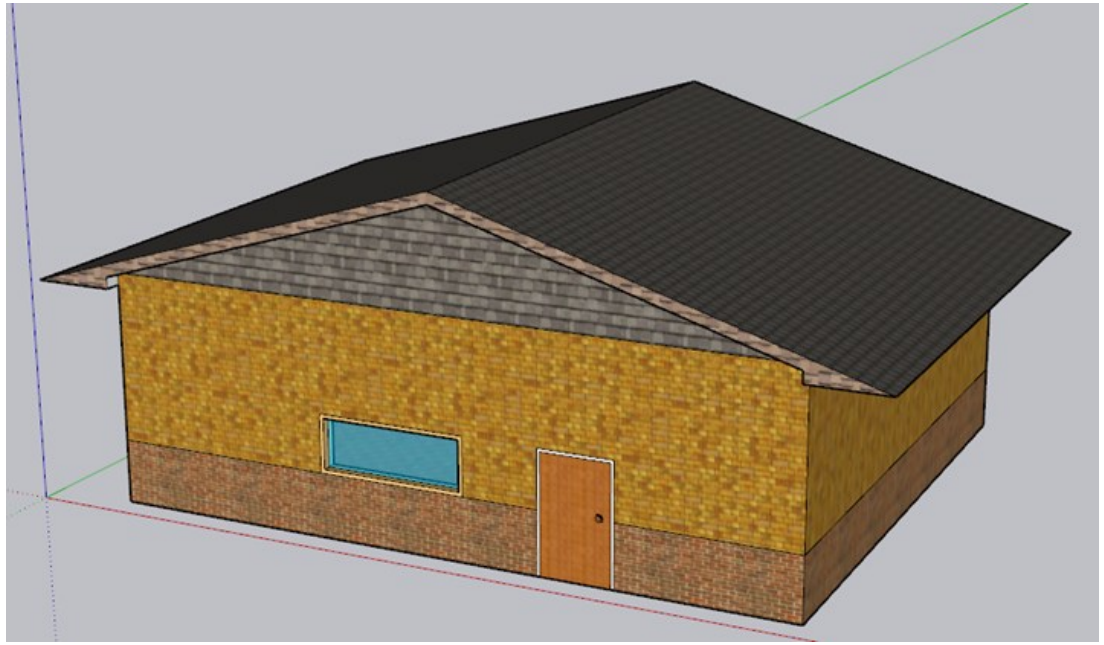
制作工程 尺寸与部件	→	FBX及依赖 交换数据	→	目标引擎 资产与材质

- **几何交接：**保留单位、朝向和部件命名，导入后用已知尺寸检查比例。
- **外观交接：**纹理可关联或嵌入；复杂材质需按目标引擎重新建立。

交付模型时保留编辑工程和纹理依赖，换平台后才便于修复。

3.1.3 三维仿真工具·建模与场景制作：SketchUp建筑

从平面轮廓快速建立建筑



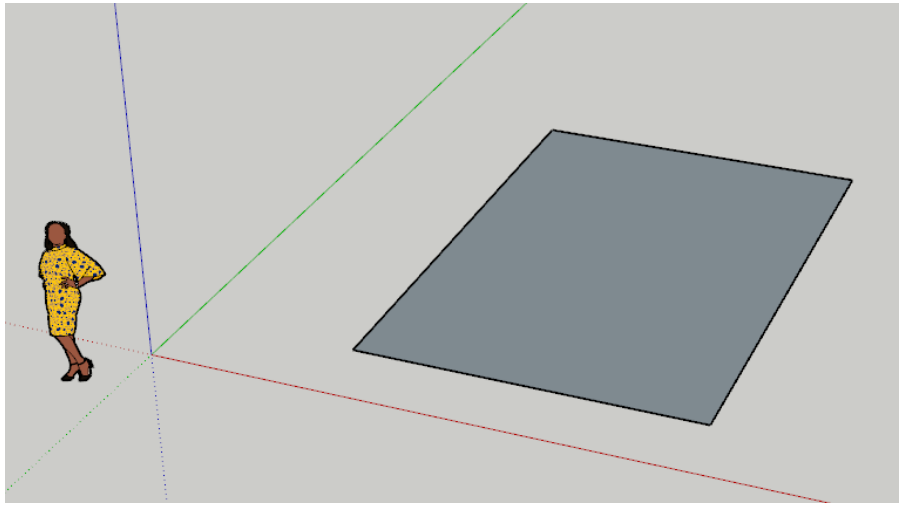
图：SketchUpUsage的house.skp

- **建模思路：**平面轮廓推拉成三维体量。
- **适用任务：**小屋、房间与建筑布局，便于快速修改尺寸。
- **交接产物：**house.skp保存模型组织，供后续场景装饰。

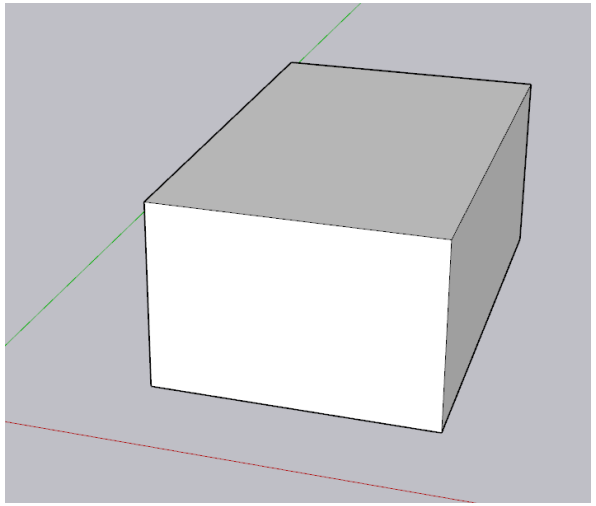
SketchUp推拉建模教程：https://help.sketchup.com/en/sketchup/pushing-and-pulling-shapes-3d

3.1.3 三维仿真工具·建模与场景制作：SketchUp推拉

平面怎样通过推拉变成立体



矩形面：建立平面轮廓



推拉后：形成盒体

- **基本过程：**封闭轮廓 → 生成面 → 推拉厚度 → 调整尺寸。
- **建筑迁移：**墙体、门洞和屋顶先定几何，再添加纹理与环境。

3.1.3 三维仿真工具·建模与场景制作：SketchUp对象复用

SketchUp群组与组件组织建筑

墙体与屋顶
群组隔离编辑

同型门窗
组件定义复用

→ 同一建筑模型
保留对象层级

- **群组：**把一部分几何包在一起，减少相邻边面相互牵连。
- **组件：**多个实例共享定义；修改共享定义会同步影响对应实例。

SketchUp组件与实例：https://help.sketchup.com/en/sketchup/components

重复门窗用组件共享定义，改一处可同步更新各实例。

3.1.3 三维仿真工具·建模与场景制作：Blender替代路线

开源工具支持网格建模与资产制作

- **功能：**网格、UV、材质、骨骼和动画可在同一工具中制作。
- **盒体例子：**对象模式移动整个盒体；编辑模式修改顶点或面，改变盒体形状。
- **课程用法：**用Blender制作盒体并导出FBX，可接续本章的模型导入。
- **保存工程：**.blend保留可编辑对象；FBX用于交换，纹理文件一并交付。

Blender网格编辑手册：https://docs.blender.org/manual/en/4.2/modeling/meshes/index.html

Blender整合建模、材质与动画，可交付同类资源而无需沿用同一软件。

3.1.3 三维仿真工具·建模与场景制作：Twinmotion环境布置

给已有几何布置环境



图：house.skp在Twinmotion中的场景效果

· **输入：** 导入house.skp，保留建筑的对象层级。

· **加工：** 替换墙面材质，再布置草木与灯光，观察外观变化。

· **产物：** 保存house.tm；按配套版本导出面向UE的场景资源。

Twinmotion场景制作课程：https://www.twinmotion.com/learning-resources

建模工具确定几何，Twinmotion快速组织外观、植被与光照。

3.1.3 三维仿真工具·建模与场景制作：Twinmotion对象组织

Twinmotion导入时怎样选择对象组织

几何体导入：合并方式



“保留层次”保留源模型的对象层级。

图：小屋导入的合并选项（局部）

- **保留层级：**沿用源模型对象关系，便于逐部件修改。
- **按材质合并：**同材质几何合并组织，减少管理对象。
- **控制需求：**要分别移动的门、舵面等部件保留独立性。

保留层级便于逐对象编辑，合并适合无需独立控制的部分。

3.1.3 三维仿真工具·建模与场景制作：Datasmith场景交接

把设计场景交接给UE

Datasmith：将设计场景导入UE的一套工具与数据交换方式。

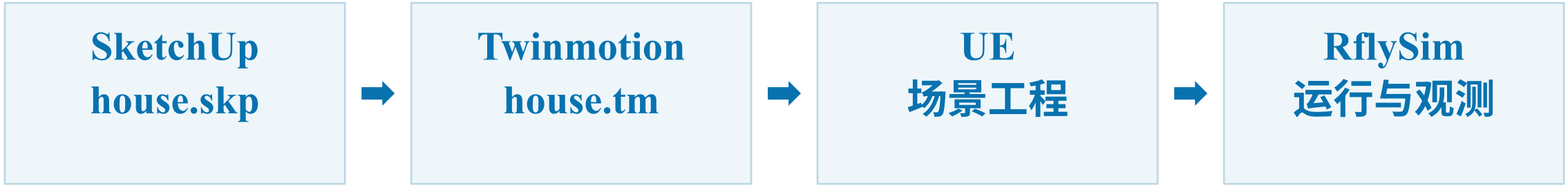
SketchUp / TM
设计与装饰
➡
Datasmith
场景及附属资源
➡
UE工程
继续设置行为

- **交接文件：**.udatasmith保存场景描述，配套资源目录保存网格与纹理，需一起复制。
- **版本配套：**按例程匹配Twinmotion、Datasmith插件与UE；导入后仍需在UE设置行为。

场景描述文件与附属资源一起传递，保留可用的对象组织。

3.1.3 三维仿真工具·建模与场景制作：小屋工具链

小屋案例中的工具分工



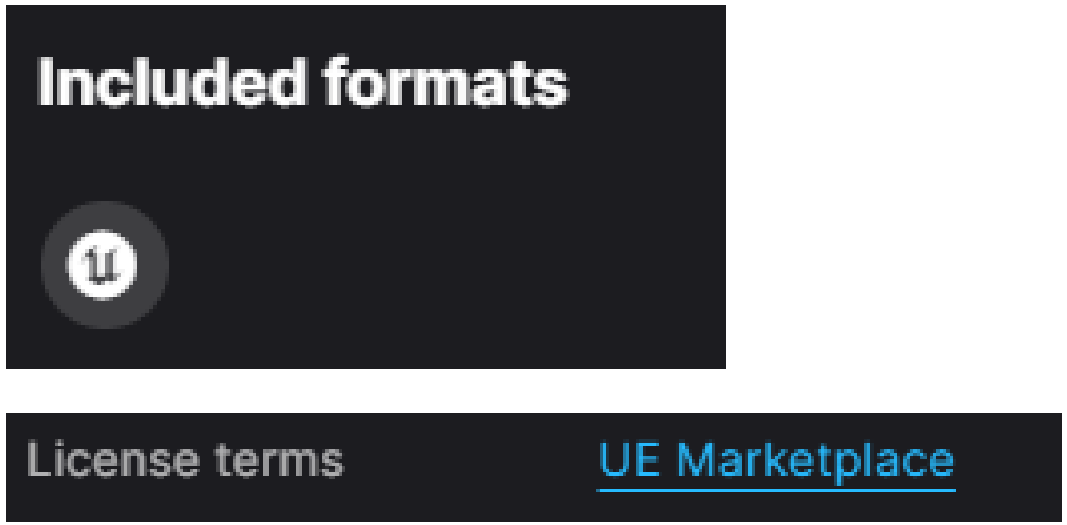
- **SketchUp**：几何与对象层级。
- **Twinmotion**：材质、植被与光照。
- **UE**：场景对象与运行行为。
- **RflySim**：运动状态、控制与观测。

新版Twinmotion到UE的Datasmith交接流程：
https://dev.epicgames.com/documentation/en-us/twinmotion/twinmotion-to-unreal-engine-workflow

工具交接的不只是文件，还包括对象层级、纹理依赖和运行接口。

3.1.3 三维仿真工具·建模与场景制作：Fab资源获取

获取可复用模型与引擎资源



图：Fab资源的格式与许可（局部）

Fab资源检索与下载：https://www.fab.com/

· **资源类型：**模型、材质、场景内容和插件用途不同。

· **技术条件：**格式与UE版本匹配，纹理和插件依赖完整。

· **使用条件：**按资源授权使用与分发，保留来源记录。

资源市场缩短制作时间，筛选依据是格式、版本、依赖与授权。

本章目录

三维仿真基础与工具选型

第三章 三维环境建模与仿真

3.1 三维仿真基础与工具选型

3.2 UE场景运行渲染与交互

3.3 多模态传感器数据生成

3.4 分布式仿真架构与数据接口

3.5 场景与载具资源开发

3.6 例程体系与综合实践

3.7 习题案例分析与课程设计

本节 5 个教学单元

3.1.1 背景知识与三维仿真流程
教材对应：第3.1.1—3.1.2节

3.1.2 三维仿真流程·模型建立与编辑
教材对应：第3.1.2节·模型建立与编辑

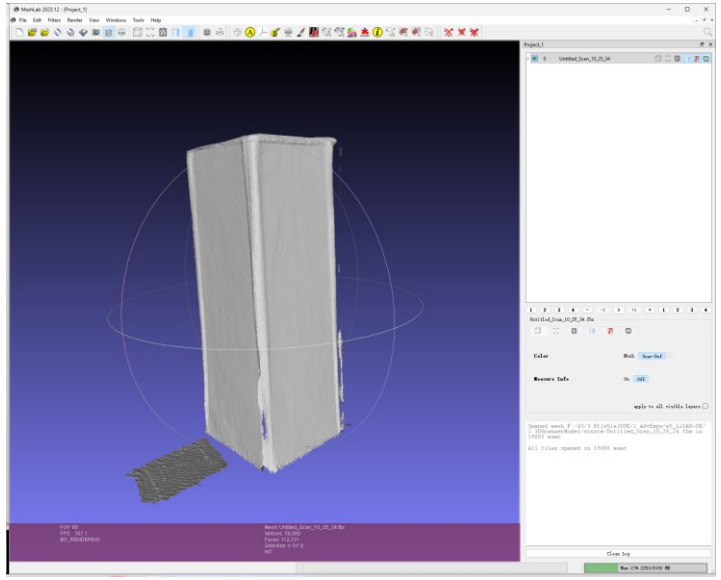
3.1.3 三维仿真工具·建模与场景制作
教材对应：第3.1.3.1、3.1.3.3节

3.1.4 三维仿真工具·其他辅助工具
教材对应：第3.1.3.3节；采集见3.1.2

3.1.5 三维仿真工具·平台比较与选型
教材对应：第3.1.3.1—3.1.3.2节

3.1.4 三维仿真工具·其他辅助工具：实景采集

实景资源：扫描、摄影测量与地理数据



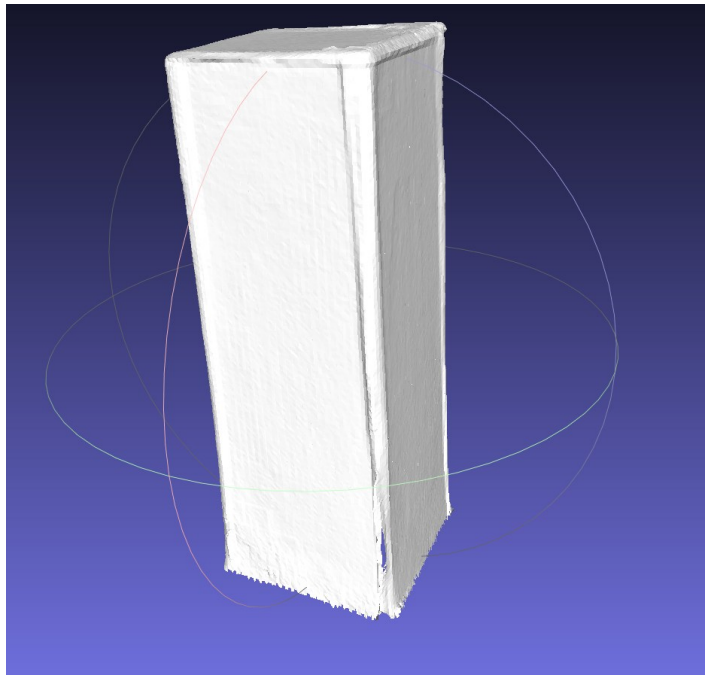
图：扫描模型导入MeshLab（配套例程截图）

- **摄影测量：**多视角照片恢复几何与纹理；MeshLab可清理网格。
- **深度与激光扫描：**采样形成点云或网格；Open3D可处理点云。
- **地理数据：**保留位置、高程基准；支持大范围场景。Cesium加载，GIS与CityEngine辅助城市建模。

实景采集提供几何和外观，清理与定位把数据变成可用资源。

3.1.4 三维仿真工具·其他辅助工具：扫描网格修复

MeshLab：清理与修复扫描网格



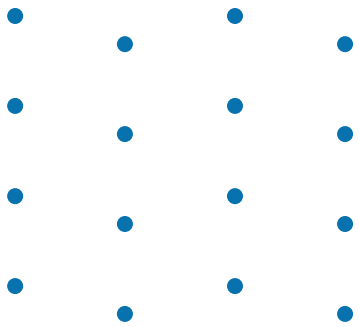
图：例程中清理后的扫描网格

- **清理：**删除背景碎片和孤立几何，检查表面朝向（法线）。
- **修复：**对采集缺口补面，保留任务需要的真实开口。
- **交接：**导出OBJ等资源，同时保留纹理与材质依赖。

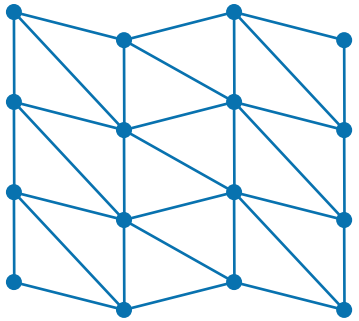
MeshLab网格处理与扫描教程：https://www.meshlab.net/

3.1.4 三维仿真工具·其他辅助工具：点云与网格

点云与网格保存的信息不同



采样点



连接后的网格

示意：同一组位置，不同的数据组织

· **点云：** 保存点的位置，可附带颜色、法线等属性。

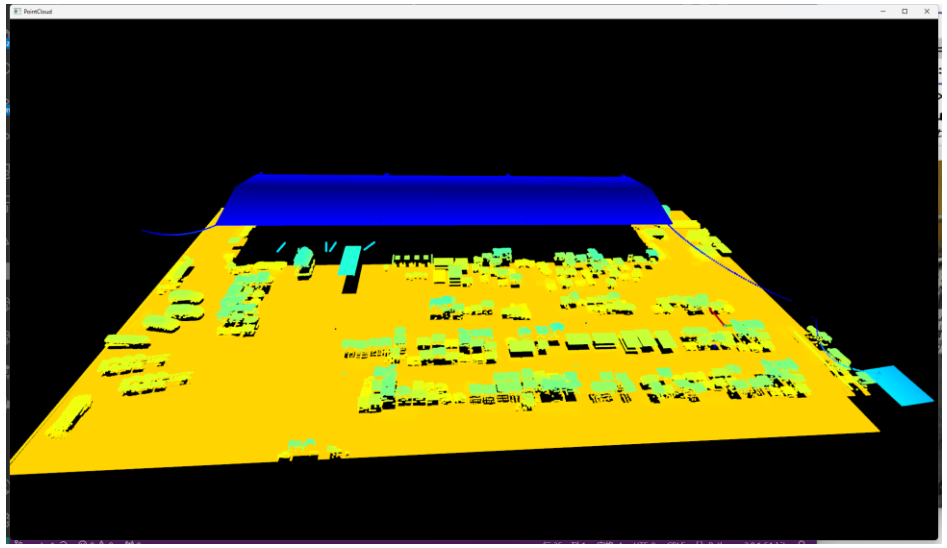
· **网格：** 用面连接顶点，表达和渲染连续表面。

· **高度图：** 每个平面位置记录一个高度，不能同时表达桥面与桥下地面。

点云保存采样点，网格增加表面连接，两者服务不同的数据处理。

3.1.4 三维仿真工具·其他辅助工具：Open3D数据处理

在Python中处理三维数据



图：地形例程的Open3D点云显示

· **接口：** Python或C++操作点云与网格，便于写入算法流程。

· **数据处理：** 降采样减少密集点；配准估计两组点云的相对位姿，使重叠表面对齐。

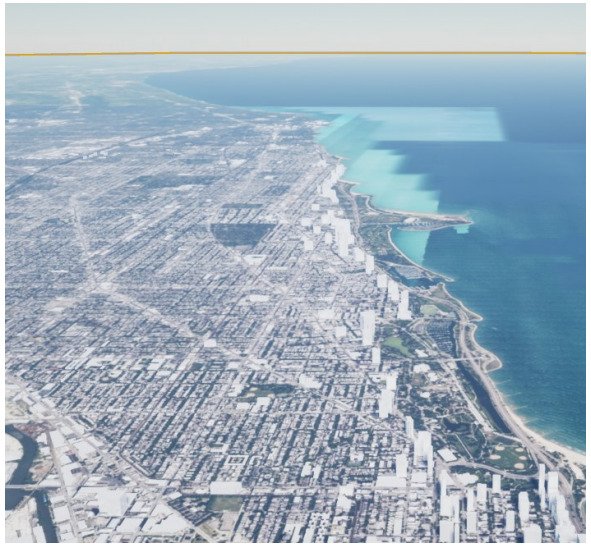
· **例程产物：** VisTerrainPcd.py把地形坐标数组装入点云对象并显示。

Open3D点云处理教程：https://www.open3d.org/docs/release/tutorial/geometry/pointcloud.html

Open3D把几何采样变成可读写、可计算和可视化的数据对象。

3.1.4 三维仿真工具·其他辅助工具：Cesium地理加载

按地理位置加载大范围三维数据



图：Cesium例程加载的城市区域

·**作用：** 在UE中加载地形、影像与3D Tiles等地理资源。

·**优势：** 按位置和视距组织大范围数据，减少逐个手工建模。

·**定位：** 原点对应局部场景位置，高程基准规定高度起算面。

Cesium地形与建筑快速入门：https://cesium.com/learn/unreal/unreal-quickstart/

地理定位与分块加载使大范围环境可以进入局部仿真实验。

3.1.4 三维仿真工具·其他辅助工具：在线与离线资源

Cesium在线资源与离线资源

- **在线加载：**按需从服务获取数据，网络与访问权限影响资源是否完整。
- **离线加载：**将已授权的数据部署到本地，按确定版本组织实验。
- **数据准备：**CesiumLab等工具可转换和发布三维地理数据，再由Cesium加载。
- **复现实验：**记录区域、来源和版本；离线专题须匹配所装版本的例程与授权。

在线服务便于获取与更新，离线数据便于固定实验环境。

3.1.4 三维仿真工具·其他辅助工具：规则化城市

CityEngine与GIS：批量构建城市环境

GIS数据 地块、影像与坐标	➡	CityEngine规则 建筑生成	➡	UE / RflySim 场景与实验

- **地理信息系统（GIS）：**管理空间参考、区域范围和地理数据处理。
- **规则建模：**对地块轮廓应用楼高、层数等规则，生成建筑体量与立面。

CityEngine规则建模学习：
https://learn.arcgis.com/en/paths/learn-the-basics-of-procedural-modeling/

GIS管理空间数据，CityEngine用规则把区域几何扩展为建筑。

本章目录

三维仿真基础与工具选型

第三章 三维环境建模与仿真

3.1 三维仿真基础与工具选型

3.2 UE场景运行渲染与交互

3.3 多模态传感器数据生成

3.4 分布式仿真架构与数据接口

3.5 场景与载具资源开发

3.6 例程体系与综合实践

3.7 习题案例分析与课程设计

本节 5 个教学单元

3.1.1 背景知识与三维仿真流程
教材对应：第3.1.1—3.1.2节

3.1.2 三维仿真流程·模型建立与编辑
教材对应：第3.1.2节·模型建立与编辑

3.1.3 三维仿真工具·建模与场景制作
教材对应：第3.1.3.1、3.1.3.3节

3.1.4 三维仿真工具·其他辅助工具
教材对应：第3.1.3.3节；采集见3.1.2

3.1.5 三维仿真工具·平台比较与选型
教材对应：第3.1.3.1—3.1.3.2节

3.1.5 三维仿真工具·平台比较与选型：引擎与框架

三维引擎、图形工具包与框架

选择	层级与组织方式	适用条件
UE	引擎：关卡、场景对象（Actor）、组件与蓝图	已有UE资产或RflySim工程
Unity / Godot	引擎：场景、对象或节点与脚本	已有脚本经验与相应项目生态
OpenSceneGraph（OSG）/ Panda3D	OSG：图形工具包 Panda3D：三维引擎	已有程序化开发代码或自定义渲染需求
Delta3D	仿真应用框架	已有框架项目与接口依赖

成熟资产与团队代码决定迁移成本，引擎选择服务于项目交付。

3.1.5 三维仿真工具·平台比较与选型：仿真平台

仿真平台按模型与实验接口选用

平台	主要用途	学习时关注的接口
RflySim	无人系统开发与验证	状态、控制、传感器与半实物
Gazebo	机器人仿真与系统集成	模型、传感器与通信
CarSim	车辆动力学与联合仿真	车辆输入输出与模型参数
AirSim原项目	无人机/车辆与视觉实验	图像、状态与外部控制

专用平台提供领域模型，通用平台侧重对象扩展与系统集成。

3.1.5 三维仿真工具·平台比较与选型：Isaac Sim与MuJoCo

按视觉数据与接触控制任务选择仿真工具

- **Isaac Sim**：用USD组织场景，集成物理、传感器与合成数据工具。
- **MuJoCo**：用MJCF组织多关节模型，计算接触与控制响应。
- **抓取控制**：已有MJCF模型时，可在MuJoCo中迭代接触与执行器控制。
- **视觉任务**：需要组织相机、光照与标签数据时，可选Isaac Sim的相应工具。

Isaac Sim传感器配置与数据读取：
https://docs.isaacsim.omniverse.nvidia.com/latest/sensors/index.html

资源能复用，仿真器的场景表达、执行器和实验工作流各有侧重。

3.1.5 三维仿真工具·平台比较与选型：资源工具选型

资源制作工具按任务选用

需要解决的问题	任务与工具对应	得到的产物
建筑与载具部件	简单建筑：SketchUp 精细部件：3ds Max / Blender	几何与独立部件
材质与环境布置	外观布置：Twinmotion 行为设置：UE	可编辑场景
网格与点云处理	交互式网格清理：MeshLab 程序化点云处理：Open3D	整理后的几何数据
大范围地理场景	加载地理数据：Cesium 规则城市：CityEngine＋GIS	定位后的区域资源

缺什么数据，就学习产生或修复该数据的工具。

3.1.5 三维仿真工具·平台比较与选型：例程学习入口

第三章例程中的工具学习入口

例程文件夹	入门内容	可检查的产物
1.UEBlueprintUsage 2.3dsMaxUsage	行为与几何编辑	盒体蓝图 / box.fbx
4.SketchUpUsage 5.TwinmotionUsage	建筑与场景装饰	house.skp / house.tm
3.CesiumForUnrealUsage	地形与城市加载	有地理参考的场景
6.EpicFabUsage	资源筛选与获取	符合工程要求的资源

例程广场 → 第3章 → 0.ApiExps → e0_DevToolsUsage；先读对应Readme.pdf。

先完成一个小产物，再进入对应的完整资源开发实验。

3.1.5 三维仿真工具·平台比较与选型：资源实践

通过复用、建模与扫描准备场景资源

资源起点	第三章内的相对目录	得到的产物
已有模型	1.BasicExps/e4_CusLoadObj/ 2.loadSimpleMesh	test.glb 场景实例
手工建模	0.ApiExps/e0_DevToolsUsage/ 4.SketchUpUsage → 5.TwinmotionUsage	house.skp house.tm
室内扫描	2.AdvExps/e5_LiDAR-UE/ 2.3DScannerScene	扫描场景

统一入口：例程广场 → 第3章；按上表目录进入，先读对应Readme.pdf。

已有资源先跑通，资源不足再制作，最后把观测接入算法。

3.1.5 三维仿真工具·平台比较与选型：数据实践

读取地形点云、载具状态与图像

数据任务	第三章内的相对目录与脚本	得到的数据
地形点云	1.BasicExps/e3_RflySim3DTerrainPcd 脚本：VisTerrainPcd.py	点云对象 三维显示
状态与图像	0.ApiExps/e6_RflySim3DCtrlAPI/ 10.GetCamObjDemo 脚本：GetCamObjDemoWithCam.py	状态元数据 图像数组

统一入口：例程广场 → 第3章。前者处理已有几何，后者请求场景状态与观测。

地形点云用于几何处理，状态与图像接口把场景数据送入算法。

3.1.5 三维仿真工具·平台比较与选型：后续分工

3.1与后续各节的分工

后续内容	在3.1已认识	后续新增知识
3.2 引擎与渲染	网格、材质、UE	对象更新、渲染与蓝图机制
3.3–3.4 观测与接口	观测与资源的区别	成像、采样、坐标与数据格式
3.5 资源开发	格式与工具产物	导入、配置、烘焙与部署
3.6–3.7 实验与应用	任务驱动选型	例程复现、改动与结果解释

资源要进入算法，还须弄清运行状态、采样机制和接口配置。