

基于 UE4 的战场态势可视化技术

计算机科学与技术

学生 唐世龙 指导老师 李辉

【摘要】战场态势系统中渲染的目标繁多，场景复杂，与此同时场景渲染的逼真性、流畅性都需要得到保证，而开发引擎 UE4 具备渲染效果真实、运算能力强、易用性强等特点，极好地满足了态势系统的这些需求，但是 UE4 在应对全球级态势方面的能力稍显不足，而且编程难度较高，如果适配到面向部队训练、指挥的战场态势，开发工作量较大，因此，本文针对这些不足，提出了一种基于仿真开发引擎 UE4 的态势仿真系统架构，将系统分为控制端模块、网络指令系统模块、球幕系统模块、态势生成模块，使得态势系统各部分之间耦合性更低、可维护性更高。本文研究了如何在 Cesium for Unreal 插件以及 CesiumLab 等工具的综合配合下，在 UE4 的环境下搭建一个高精密度的全球大地形。研究了如何利用 UE4 中强大的粒子系统、体积云等组件来搭建一个具有极高仿真度以及良好交互性的自然天气仿真系统。研究了如何通过 Qt 工具以及 UE4 的结合使用，开发一个能够接受网络包并将其解析为各种指令的网络指令系统，以对态势生成系统中各个军事设备以及战场环境进行实时的控制。以及研究了如何基于 UE4 中的蓝图系统，对战场基本军事作战目标：飞机、士兵、武器，进行模拟与控制。

【关键词】战场态势；虚幻引擎；天气模拟；军事目标模拟；武器模拟。

插图索引

图 2.1	态势系统的组成结构	9
图 2.2	系统的执行流程	10
图 2.3	球幕系统效果演示	11
图 3.1	Cesium for Unreal 原理图	12
图 3.2	在 GEBCO 中下载高程以及影像数据	13
图 3.3	Cesium Lab 的切片流程	13
图 3.4	在 UE4 中输入 Layer.json 文件地址	14
图 3.5	不同 LOD 下的地形	14
图 3.6	Jet 中的定位组件	14
图 3.7	瑶湖机场-建筑群	15
图 3.8	瑶湖机场-桥	15
图 3.9	瑶湖机场-植物	15
图 4.1	网络系统调用关系图	16
图 4.2	GameManager 绑定代理事件	18
图 5.1	体积材质域的输出节点	21
图 5.2	低云层基本形状纹理贴图	22
图 5.3	高层云基本形状纹理贴图	22
图 5.4	云的形状展示	23
图 5.5	云实现的原理图	23
图 5.6	云基本形状贴图的 UV 坐标变化	24
图 5.7	噪声体积纹理	24
图 5.8	添加云噪声前后的变化	24
图 5.9	战机从云中穿过	25
图 5.10	雾中的士兵正在巡逻	25
图 5.11	霾的效果展示	25
图 5.12	不同等级的雨	26
图 5.13	不同等级的雪	26
图 6.1	骨骼重定位	27
图 6.2	对各处骨骼的变化进行混合	28
图 6.3	起落架状态机	28
图 6.4	Pitch、Roll 动画混合	29
图 6.5	飞机尾焰的级联粒子	29

图 6.6	飞行中尾焰的效果	29
图 6.7	飞机的毁伤特效	30
图 6.8	士兵的动作状态机	30
图 6.9	士兵的各种姿势	31
图 6.10	士兵行走的混合空间	31
图 6.11	士兵警戒中水平、垂直角变化曲线	32
图 6.12	士兵巡逻的逻辑流程图	32
图 6.13	士兵射击的逻辑流程图	33
图 6.14	无动力炸弹的投放与爆炸图	35
图 6.15	导弹制导的逻辑流程	36
图 6.16	导弹的两个阶段	36
图 6.17	幻影 2000 战机发射火箭弹	37
图 6.18	士兵正在发射子弹	37

表格索引

表 4.1	CPacket 定义	16
表 4.2	SendPacket 与 OnRecvData 函数	17
表 4.3	指令相关结构体	18
表 4.4	指令的定义	20
表 7.1	WeaponParent 的组件	34
表 7.2	WeaponParent 的虚函数	34

目录

第 1 章 绪论	1
1.1. 研究背景与意义	1
1.2. 国内外研究现状	2
1.2.1. 国内研究现状	2
1.2.2. 国外研究现状	5
1.3. 论文的主要工作及章节安排	7
第 2 章 态势仿真系统整体架构设计	8
2.1. 态势仿真系统渲染引擎 虚幻 4	8
2.1.1. 虚幻 4 介绍	8
2.1.2. 虚幻 4 作为态势仿真系统渲染引擎的优势	8
2.1.3. 虚幻 4 的蓝图系统(Blueprints)	8
2.2. 态势仿真系统的整体框架设计	9
2.2.1. 框架设计图	9
2.2.2. 系统执行方案	10
2.2.3. UE4 的控制管理模块 (Game Manager)	11
2.3. 球幕系统	11
第 3 章 全球大地形以及静态态势的搭建	12
3.1. Cesium for Unreal	12
3.1.1. 全球地形搭建的过程	12
3.1.2. Cesium 坐标系统	14
3.2. 静态场景的搭建	15
第 4 章 网络指令系统	16
4.1. 网络指令系统结构	16
4.1.1. 网络指令系统介绍	16
4.1.2. 网络指令系统调用关系图	16
4.1.3. 网络指令系统实现	16
4.2. 指令分类	18
4.2.1. 相关结构体	18
4.2.2. 指令定义	18
第 5 章 天气模拟	21
5.1. 云	21
5.1.1. UE4 体积云组件与体积材质	21
5.1.2. 云的形状	21

5.1.3. 云的实现原理图	23
5.1.4. 云的移动	23
5.1.5. 云的动态效果	24
5.1.6. 战机云中穿越效果	25
5.2. 雾	25
5.3. 霾	25
5.4. 雨、雪	26
第6章 作战目标模拟	27
6.1. 作战目标的控制与管理	27
6.2. 飞机模拟	27
6.2.1. 飞机骨骼动画	27
6.2.2. 飞机特效	29
6.2.3. 飞机运动控制	30
6.3. 士兵模拟	30
6.3.1. 士兵的状态	30
6.3.2. 士兵行走	31
6.3.3. 士兵警戒	32
6.3.4. 士兵巡逻	32
6.3.5. 士兵射击	33
6.4. 武器模拟	34
6.4.1. 武器的控制与管理	34
6.4.2. 无动力炸弹	35
6.4.3. 导弹	35
6.4.4. 火箭弹	37
6.4.5. 子弹	37
第7章 结语	38

第 1 章 绪论

1.1. 研究背景与意义

虚拟仿真技术在军事领域中扮演者极其重要的角色，近些年来随着计算机仿真技术的发展，众多国家开始意识到虚拟仿真技术在军事领域的潜力。

在传统的军事训练中，训练者都是在真实的军事设备中进行排演，这种实战式训练虽然能够很好地达到训练效果，但是它对场地要求极高，同时训练之后对军事设备的维护将是一比不小的开支。除此之外，实战式训练受天气、时间、场地等各种不稳定因素的影响，实际训练效果并不能得到很好地保证，同时实战式训练也不能保证训练者的人身安全。因此，军事训练与排演迫切地需要一个能够有效地避免现实不稳定因素的影响，同时能够提供多种训练环境的系统。而虚拟仿真技术提供了一条可行的路径。利用虚拟仿真技术，我们可以搭建一个虚拟战场态势系统，这样训练者不用走出室外便可以完成各式各样的训练。待学员将各种技能掌握熟练之后，再进行实战式训练，这样不仅大大降低了训练成本，同时也能保障学员的人身安全。

在战场态势系统中，所有目标应当统一使用经纬度以及高度进行定位，这样指挥官便可以清楚地知道每一个目标在全球的位置，以方便对其进行部署安排。UE4 中的坐标系统是一种简单地以厘米为尺度的坐标系统，如若直接使用该坐标系统，会导致多系统之间的空间数据不兼容，位置定位有偏差等问题。同时在态势系统中，为方便指挥官清楚地了解全球各个位置的地形、水域、卫片影像等信息，我们需要对全球范围的大地形进行建模与编辑，虽然在 UE4 中内置了一个地形编辑器，但是它只能建立小范围的地形，如果对全球地形进行建模，将会耗费巨大的时间成本，而且不能保证地形的准确度。综上所述，在 UE4 系统中建立一个统一的、精确的、以经度、纬度、高度作为定位尺度的坐标系统，以及一个可以自动加载地形数据并构建全球大地形的管理工具，是必须的。

战场态势系统要素包含多种，环境与目标是最其中最重要的部分。对各种天气的模拟，能够为训练人员提供多种训练环境，让他们在与虚拟环境之间的交互中，积累作战经验，培养对不同环境的应对能力。对于指挥官，天气模拟能够使指挥官更清楚地了解战场的基本环境条件，根据环境制定合理的作战策略。而战场目标作为态势系统中的动态因素，也是战场态势的重要组成部分之一，如果没有战场目标，战场中的其他因素将变得没有任何意义。战场中的目标包括飞机、士兵、车辆、武器等，他们每一个都时时刻刻影响着战场局势的走向，对目标进行仿真模拟，有助于指挥官更清楚地把握各个目标的状态，并及时采取正确的应对策略。对于训练人员，真实的目标模拟，更能让训练人员身临其境地感受到战场的氛围感，以取得更好地训练效果。

本文就以上态势系统存在的问题与需求，设计了一个基于 UE4 的战场态势可视化系统，

在设计中，为了使模块之间耦合性更低、可维护性更高，我们把系统划分成控制端模块、球幕系统模块、态势生成模块以及网络指令控制模块，其中态势生成模块以及网络指令控制模块由本人完成实现。各模块的主要功能如下：①控制端模块：负责向各个态势生成计算机发送网络指令包。②网络指令系统：部署在每一台态势生成计算机中，它负责实时侦听控制端传来的控制指令包，并对其进行解析，判断指令的类型，之后根据解析后的结果来执行指令。③球幕系统：负责将 10 台态势生成计算机生成的态势进行混合，无缝拼接于球形屏幕上。④态势生成模块：它是 UE4 部分的总控制器，它与网络指令系统一同部署在每一台态势生成计算机中，它包含四个控制器成员，分别对士兵、武器、天气部分进行控制与管理。

在态势系统中的态势生成模块，以及网络指令模块中，本人实现的主要功能如下：

(1) 全球范围大地形的搭建。为解决 UE4 中全球大地形构建的困难，我们从互联网中下载了高精度的地理信息数据，并利用 CesiumLab 工具对其进行处理与切片，之后在 UE4 中，用 Cesium for Unreal 插件对数据进行加载与显示，并且利用该插件中提供的 CesiumGlobeAnchorComponent 可以完美解决 UE4 中的坐标转换问题。

(2) 实现网络指令系统。态势系统各模块之间的通信交由网络系统来进行处理，由控制端控制态势的生成以及球幕的初始化等设置。各个模块之间各司其职，互不干涉，这大大提高了各个模块的通用性。

(3) 天气现象仿真模拟。本态势系统提供对多种天气现象的控制，例如雨、雪、云、雾、霾等。通过天气系统，我们可以模拟出多种天气现象：晴天、阴天、暴风雨来临、闪电等现象，使得战场环境更具多样性、复杂性。

(4) 目标模拟。本系统对飞机、士兵以及武器目标进行了模拟。目标的创建与更新可以由控制端发送的指令进行控制，也可以自行进行仿真运动。

1.2. 国内外研究现状

1.2.1. 国内研究现状

仿真技术作为目前现代战场的重要技术，受到国内外的高度重视与广泛研究，呈现出研究热点频发、系统设计纷纭的局面。众多高校纷纷投入相关研究，而计算机领域也将仿真技术在现代战场中的运用作为一项严重重点进行挖掘。目前，国内相关专业人员对仿真技术在战场模拟方面的运用做出了大量系统化的阐述，研究较多。其中阐述多以仿真技术的现实运用为主，侧重于结合我国目前军事发展的新趋势及军事模拟战场的现实运用，以案例分析、系统设计、原理阐述为重点，从现实运用的角度出发对仿真技术在模拟作战场地中的重要意义进行论述。

罗博峰《虚拟现实技术训练应用研究综述》中就从虚拟现实概念的阐述出发，重点从五个方面阐述虚拟现实技术训练应用研究的现状。该论文认为，虚拟现实(Virtual Reality,

VR)概念从 1984 年被 William Gibson 提出以来,在系统仿真领域造成了巨大的影响。并由此对虚拟现实技术的运用进行了解读。

该论文先从理论角度指出虚拟现实技术的特点,即沉浸性、交互性、想象性,随后指出,虚拟现实技术训练是指以虚拟现实技术为支撑,创建提供模拟实兵、实装、实弹的虚拟环境,培训和提高受训者作战能力的训练活动,属于模拟训练。具有安全保密、场景多变、费效比高可重复性等优点。罗博峰先生根据训练的目的、条件、内容以及方式方法,结合当前 VR 技术在部队训练中的应用研究现状,文章将其划分为 5 个方面:即虚拟战场环境仿真、指挥决策训练模拟、装备操作训练模拟、战役战术训练模拟、军事游戏训练模拟。

孙峥皓在《关于虚拟战场态势仿真系统设计研究》则是从现代战争的突出特征入手,重点从模拟设计的角度出发,分析了虚拟战场态势仿真系统在现代战场中的运用。

该文章指出,现代战争是复杂条件下的一体化联合对抗,在陆、海、空、天、网电多个维度发生,作战体系结构及其在强对抗环境下的弹性将是影响战争胜负的关键因素。而战场态势瞬息万变,抽象、信息量匮乏的战场态势展示已不能适应现代战争的需要。由此作者提到向指挥员及各类作战人员提供动态、高逼真、模拟真实环境的战场态势画面,增强战场感知能力,提高反应速度的重要性与必要性。

文章从系统设计层面出发,重点模拟战场态势仿真系统总体设计。主要包含功能需求与开发流程设计。作者认为,一个完整的战场态势展示系统包括三维场景构建,在场景中添加实体参战模型,为模型添加特效动作,通过脚本控制来实现模型的数据驱动,通过 UI 界面来实现作战流程控制及视角控制等。依托该原理对开发流程设计进行阐述。随后论述了如何实现虚拟战场态势仿真系统。第 1 点即是地形场景创建,第 2 点为海洋模型创建,第 3 点为大气模型创建,第 4 点为参战实体创建,第 5 点为 Unity3D 场景融合,第 6 点为人机交互界面设计,并就以上几点进行了原理阐述与实现通道的论述。

笔者认为,该文章侧重从设计研究的角度进行系统模拟,具有极强的实用性与参考性。

游雄在《基于虚拟现实技术的战场环境仿真》中提到,战场环境仿真在内容上包括战场环境感知和战场环境数据仿真,并重点论述如何运用虚拟现实技术来实现战场环境感知仿真。

该文章从战场环境的构成出发,指出战场环境具有多维性、互动性的特点,而战场环境仿真是指运用仿真技术来描述战场环境,并建立了大量的结构图,对各要素进行阐述,讲述虚拟战场环境系统的基本构成。

游雄认为,虚拟战场环境系统由软件系统、数据库系统和硬件系统 3 部分构成。根据虚拟战场环境的应用需求,以上 3 个部分就有不同的组合方式,进而构成不同的应用系统。

该论文理论性强,对虚拟战场技术的重要理论进行了详细而严谨的阐述,在理论阐述之中也着重强调虚拟战场技术的现实运用,并依托系统的构建实现理论的现实价值。

李宁在《虚拟作战战场环境的研究与实现》中则重点阐述了虚拟作战战场环境的关键

技术。以案例介绍的方式讲述了虚拟战场环境的关键技术,包括虚拟战场环境的结构组成、仿真模型、仿真管理和想定生成、场景管理、虚拟士兵的行为表示、实时战场环境声音仿真、人机交互方式等。并在此基础上,介绍了已研制完成的基于多 PC 机联网具有沉浸与交互特征的多用户同时参与的含虚拟士兵的虚拟作战演示与观众参与系统。

该文章注重虚拟作战战场环境的实现,构建了大量数据列表,进行了虚拟作战环境的现实模拟,重点在于将原理进行实际运用。

李金刚先生在《虚拟现实技术在军事领域的应用》对虚拟现实技术进行了定义,重点分析了虚拟现实技术在军事中的应用。

该文章指出,虚拟现实技术在军事领域上有着广泛的应用和特殊的价值。如虚拟战争中作战指挥的模拟,新式武器的研究、开发以及使用,军事技术培训等都可以使用虚拟现实技术。基于虚拟现实的军事训练技术已经相当成熟。而且虚拟现实的优点就是费用低,并且不会对环境造成其他危害。同时,虚拟现实技术在军事模拟中的应用,可以将过去简单、枯燥的训练变得生动、有趣并且可视化,从而大大提高了训练效率。指明虚拟现实技术为作战模拟提供战场环境、为单兵模拟训练和指挥员训练提供场所、为诸军兵种联合虚拟演习提供平台。

徐享忠、郭齐胜在《军用仿真发展现状与展望》中以综合介绍论述的方式,阐述了军用仿真技术的发展现状。

该文章对仿真的内涵进行了阐述,认为仿真有多种分类方法,常见的分类依据包括模型的类型、组成单元的性质、仿真计算机的类型、仿真系统的用途及所属应用领域。仿真具有经济、安全、前瞻、可重复、无破坏、受环境制约小等诸多优点,但也存在不足,主要体现在:对于复杂系统的建模,无经验或观测样本可供借鉴;仿真可信性难以保证,分解之后难以确保如实集成;计算量巨大,分解细化、描述精细与计算复杂相互矛盾。

该文章对仿真技术在国内的发展作出了说明。文章指出,国内将仿真学科成为仿真科学与技术,2009 年,中国系统仿真学会向教育部提交了《设置“仿真科学与技术”一级学科的建议书》,建议在我国高等院校设置“仿真科学与技术”一级学科;2010 年,中国系统仿真学会发布《仿真科学与技术学科发展研究报告》蓝皮书,认为“仿真科学与技术是以建模与仿真理论为基础,以计算机系统、物理效应设备及仿真器为工具,根据研究目标建立并运行模型,对研究对象进行认识与改造的一门综合性、交叉性学科”,并对仿真科学与技术的理论、方法体系进行归纳总结。

总的来说,国内对仿真技术在军事战场相关领域的研究呈现出多面开花、重视实用性的态势,其中对仿真技术的来源及阐述仍然是基础研究,仿真技术在虚拟战场模拟中的现实运用始终是研究者关注的核心之一。理论的发展趋势、不同国家之间的军事运用等为学者所注意。

正如贾连兴等人在《作战仿真研究热点及发展》中指出的那样,目前,基于军事网络

的仿真研究还处于起步阶段。其主要原因是网络技术本身还很不成熟，国内外与其相关的应用研究大都处于概念研究、体系结构设计、原型系统的试验等阶段。尽管目前网络技术还不是很成熟，但它将是网络技术的发展方向，及早展开基于网络的军事应用仿真研究，对于我军将来抢占信息制高点，推动仿真技术在军事领域的应用具有非常重要的意义。

众多参考文献的一个共识是，仿真技术在虚拟战场的具体运用中具有极高的价值，对于提升军事实力、提升作战的效能、维护国家安全具有重要的参考价值。

1.2.2. 国外研究现状

总体来看，国外对仿真技术模拟战场运用的研究具有理论研究更成熟、实际运用更广泛的特点。换言之，国外高度重视仿真技术在模拟战场中的运用，并进行了大量广泛而深入的理论研究和实案运用。各高校、国防部都高度重视仿真技术的军事运用，并由此呈现出技术领先、理论先行、案例运用广泛且具有参考性的特点。

特别说明，美国国防部高度重视仿真技术的发展，近十多年来，美国一直将建模与仿真列为重要的国防关键技术。

孙峥皓在《关于虚拟战场态势仿真系统设计研究》指出，在美国著名的红旗军演中，各作战单位在真实的作战区域中移动，在指挥部的三维仿真软件上同步显示这些运动，同时允许所有训练弹与实弹的模拟投放。国内对战场态势仿真系统的研究起步较晚，同时用来实现态势仿真的方式也较分散。有些使用国外仿真软件进行二次开发，如 SKYLINE、wordwind、arcgis、OSG 等。有些使用商业软件开发，采用底层的 OpenGL 图形应用程序接口来仿真海上多艘舰艇进行各种不同编队排列行进的过程，其优点是跨平台特性，缺点是无法提供图形界面或用户输入函数，开发难度大。有些利用 Vega Prime 仿真平台开发。

李金刚先生在《虚拟现实技术在军事领域的应用》指出，当前国际上，虚拟现实技术的研究大致可以分为以下三类。第一类是研究虚拟人机交互界面，这一类人员大多是由研究图形学、行为学或是心理学的专家组成；第二类是研究虚拟现实系统的构造技术，主要研究内容包括实时生成有真实感的图形画面的算法和硬件体系结构，较有代表性的研究机构是美国的北卡大学(University of North Carolina)；第三类着重于研究虚拟现实的应用。在这一类研究人员中，人工智能界显示出了高度的热情。有许多的人工智能理论专家和机器人专家参与研究。较有代表性的研究机构有美国的卡内基梅隆大学。

徐享忠、郭齐胜在《军用仿真发展现状与展望》中指出，美国国防部(Department of Defense, DoD) 长期重视军用仿真技术。关于仿真学科，国外主要称其为“建模与仿真”，并受到到高校重视，积极推进建模与仿真技术的研究实践。美军还建立了联合部队和军种 2 个层次的作战实验室。联合作战实验由联合部队司令部(原大西洋司令部) 负责，下辖“联合作战中心”、“联合 C4 IS R 作战中心”和“联邦作战实验室”等联合作战实验机

构。军种作战实验室包括:陆军太空与导弹作战实验室(Space and Missile Defense Battle Lab, SMDBL)、海军作战中心(Navy Warfare Center, NWC)、空军太空作战实验室(Air Force Space Battle Lab, AFSBL)、美海军陆战队作战实验室(USMC Warfighting Lab, USMCWL) 和网络战实验室(Cyber Battle Lab, CBL) 等。

该论文提到,自 20 世纪 80 年代中期美国国防部提出先进分布仿真(Advanced Distributed Simulation, ADS)概念以来,分布式仿真已逐渐成为仿真领域研究的热点和复杂系统研究的重要工具。

孙柏林、靳大安在《外军建模与仿真综述》一文中重点论述了美国军队 M & S 的情况。

该文章指出,美军高度重视仿真与模拟,将仿真与模拟应用 21 世纪新的作战能力形成过程之中。从外军 M & S 概况、美军 M & S 的组织机构、美军 M & S 的技术发展、美军 M & S 的主要系统、美军 M & S 的基本应用等方面对美军 M & S 的情况进行了说明,即美国军队在作战建模与仿真领域经过几十年的建设与探索,已经有了十分深厚的基础,并且也从这一领域获取了重要的成果,特别是在帮助高级指挥官决策方面。目前,美军正针对未来战争的需求,继续向更深的层次迈进。

焦楷哲等人在《国外军用虚拟训练系统研究》一文中则重点介绍了虚拟训练系统的基本概念和基本组成,综述了美国、俄罗斯、英国、法国、德国等军事强国军用虚拟训练系统的应用研究现状。

该文章指出,美国是最早开展虚拟训练研究和应用的国家,其技术和装备一直处于国际领先地位。20 世纪 80 年代,美军开始将计算机作战虚拟训练作为军事训练的基本手段,同时还通过卫星联网和计算机模拟实现大规模模拟军事演习。俄罗斯同样是世界上的虚拟训练大国和强国。他们的先进武器装备几乎都编配有相应的虚拟训练器/系统,并且正在朝着通用化和嵌入式的发展方向发展。英、法、德等西欧国家也十分重视虚拟训练。他们将虚拟训练器/系统发展作为参加军备竞赛的极重要方面,虚拟训练技术及应用始终处于世界先进行列,并形成了十分繁荣的西欧“虚拟训练器/系统”商品市场。

田赤军等人在《国外军用训练模拟技术发展分析》指出,国外长期以来将军事训练模拟技术作为重大战役的战术、战法演练,提高军事训练质量,检验装备的效能、降低军事训练消耗、推动武器装备发展,快速形成战斗力的重要手段。在此方面投入了大量的人力、物力,财力,推动了军用训练模拟技术的长足进步和发展,经历了人工模拟、半自动模拟和计算机模拟的不同发展阶段,取得了丰硕的成果,在日常及战时军事训练中发挥了重要作用。重视训练模拟的顶层规划和系统建设,注重一体化试训平台研制及标准规范制定与推广,利用虚拟现实技术,提升训练模拟系统的逼真度。特别指出的是,一直以来,美军致力于虚拟现实技术的研究,加紧研发各种虚拟训练系统,以提高美军的战斗力,应对日益复杂的战场态势,虚拟现实技术在虚拟战场环境、近战战术训练、单兵模拟训练和诸军兵种联合战略战术演习等军事领域的应用较为广泛。

综上，国外的研究同样以实用性为主，重视仿真技术在实际作战中的运用，侧重于运用具体的模型进行作战演练，具有极强的现实性，受到高度重视。

1.3. 论文的主要工作及章节安排

本文章节的主要安排如下：

第一章绪论，主要阐述了本课题的研究背景以及在国内外的研究现状。以及对主要的研究内容做一个大致的介绍。

第二章整体架构设计，主要介绍了本系统采用的渲染开发系统虚幻 4 的特点和作为态势仿真系统的优势，最后介绍了 UE4 蓝图系统的优势。随后展示了态势系统的整体框架设计图以及系统的执行方案图，介绍了 **GameManager** 在整个系统中的地位与角色。最后介绍了 **JTRS-320** 球幕系统的组成以及作用，并展示了球幕系统的效果图。

第三章全球大地形以及静态态势的搭建，其中介绍了 **Cesium for Unreal** 插件的由来以及其提供的功能。之后阐述了如何从网上下载初始地理信息数据，以及通过 **CesiumLab** 对地形数据进行切片，最终通过 **Cesium for Unreal** 进行地形数据的加载与显示。最后介绍了静态场景瑶湖机场的搭建，展示了它的桥、建筑、植物的效果图。

第四章网络指令控制系统，最开始简单地介绍了网络指令系统，并展示了网络指令系统的调用关系图，之后分别对该系统中的各个类的作用于联系进行了分别的介绍。最后通过表格的方式对态势系统中的指令进行了详细地介绍。

第五章天气控制，介绍了 UE4 中、体积云组件、天空大气组件、指数高度雾组件等，以及利用它们实现的天气系统。介绍了利用体积云组件实现的云系统，该云系统支持对云形状、云高、云厚等参数的修改。介绍了基于指数高度雾组件的雾系统，之后阐述了通过修改天空大气组件中的 **Mie** 散射范围的方法实现霾。最后介绍了利用 **SkyCreator** 插件来对雨、雪进行模拟仿真。

第六章作战目标模拟，主要阐述了基于 UE4 中的蓝图系统对两种军事作战目标：飞机、士兵的实现的原理。该章分别介绍了两种目标骨骼动画制作与动画混合，介绍了 UE4 中的骨骼重定位技术，它大大提高了骨骼系统的复用性。介绍了飞机的两种运动方式：由指令控制或者由 UE4 自行模拟。之后介绍了士兵的它的行为状态机以及行走、巡逻、设计、警戒动作的实现原理。之后阐述了本系统中的武器系统的实现原理，介绍了 **WeaponParent** 的组件构成以及虚函数的作用。之后介绍了无动力炸弹利用发射物移动组件实现下落的原理。再之后介绍了导弹运动的两种方式：指令控制与自动制导，分别对这两种方式进行了详细介绍。最后，由于子弹与火箭弹的实现简单，简要的介绍了他们的发射特点。

第 2 章 态势仿真系统整体架构设计

2.1. 态势仿真系统渲染引擎 虚幻 4

2.1.1. 虚幻 4 介绍

UE (Unreal Engine) 是目前世界最知名的、最受欢迎的顶尖游戏引擎, 它在全球商用游戏引擎市场中占有 80% 的份额。自 1998 年正式诞生至今, 经过不断的发展, 虚幻引擎已经成为整个游戏界——运用范围最广、整体运用程度最高、次世代画面标准最高的一款游戏引擎。UE4 是美国 Epic 游戏公司研发的一款 3A 级次时代游戏引擎。UE4 中基于物理的渲染技术、高级动态阴影选项、屏幕空间反射以及光照通道等强大功能将帮助创作者灵活而高效的做出令人赞叹的内容。许多我们耳熟能详的游戏大作, 都是基于虚幻引擎诞生的, 例如: 双人成行、鬼泣 5、质量效应、战争机器、爱丽丝疯狂回归等等。

2.1.2. 虚幻 4 作为态势仿真系统渲染引擎的优势

①**画面效果**, 评价一个仿真系统好坏的标准, 画面效果必然是其中最重要的指标之一。UE4 采用了目前最新的即时光迹追踪、HDR 光照、虚拟位移等新技术, 使其画面效果完全达到了 3A 游戏标准, 这为我们搭建一个逼真的战场态势系统提供了技术支撑。

②**运算力**, 战场态势复杂多变、目标繁多, 如何在保证性能的同时, 做到不掉帧、无延时、高仿真度, 是一个困难且重要的问题。UE4 能够每秒钟实时运算两亿个多边形计算, 极好地满足了战场态势系统的这一需求。

③**开源**, UE4 的底层代码完全开源, 开发者能够学习自定义并调试整个虚幻引擎, 这使得市面中出现了很多基于 UE4 的各种插件, 这些插件的出现极大的提高了 UE4 项目的开发效率, 使得代码编写更加地灵活。本态势系统使用到的核心插件 Cesium for Unreal 插件便是其中之一。

④**易用性**, 在 UE4 中, 我们完全可以只用依靠蓝图进行开发, 如今十分火爆的《绝地求生》就是一款纯蓝图开发的游戏。蓝图系统利用节点与连线的可视化方式建立代码块之间的联系, 极大地降低了代码编写的门槛, 快速制作出原型并推出交互效果。

2.1.3. 虚幻 4 的蓝图系统(Blueprints)

蓝图是 UE4 中特殊的一种资源, 它是一类完整的游戏性脚本系统, 它提供一个直观的、基于节点的界面, 利用连线将节点、事件、功能和变量连接在一起, 从而创建出复杂的游戏元素。它为关卡设计师和游戏开发者提供一个在虚幻编辑器中快速创建和迭代游戏可玩性的工具, 而不需要写一行代码。

蓝图系统主要有以下四个优势: ①**开发方便**: 蓝图提供一个直观的、基于节点的界面, 不需要写一行代码, 便可以利用蓝图实现各种功能需求。②**面向组件**: 蓝图系统是面向组件的, 蓝图可以对各个各个不同的组件分别操作, 同时每一种组件都有自己不同的功能,

使得蓝图系统具有极高的逻辑性，内聚性。③**开发效率高**：蓝图的编译速度极快，通常能够在一秒钟之内编译完成，同时通过视口界面，能够直观得看见 Actor 的变化。

2.2. 态势仿真系统的整体框架设计

2.2.1. 框架设计图

本仿真系统一共分为四部分：①**控制端模块**：负责向各个终端发送网络指令包。②**网络指令系统**：部署在每一台终端电脑中，它负责实时侦听控制端传来的控制指令包，并对其解析，判断指令的类型，之后根据解析后的指令结果，调用 Game Manager 提供的相应的接口。③**球幕系统**：负责将 10 台态势生成计算机生成的态势进行混合，无缝拼接于球形屏幕上。④**态势生成模块（Game Manager 控制）**：它是 UE4 部分的总控制器，它与网络指令系统一同部署在每一台终端电脑中。它包含四个控制器成员，分别对士兵、武器、天气部分进行控制与管理。它向网络指令部分提供控制接口，以对场景进行控制。

四部分系统全部位于同一局域网内。系统的整体框架图如下：

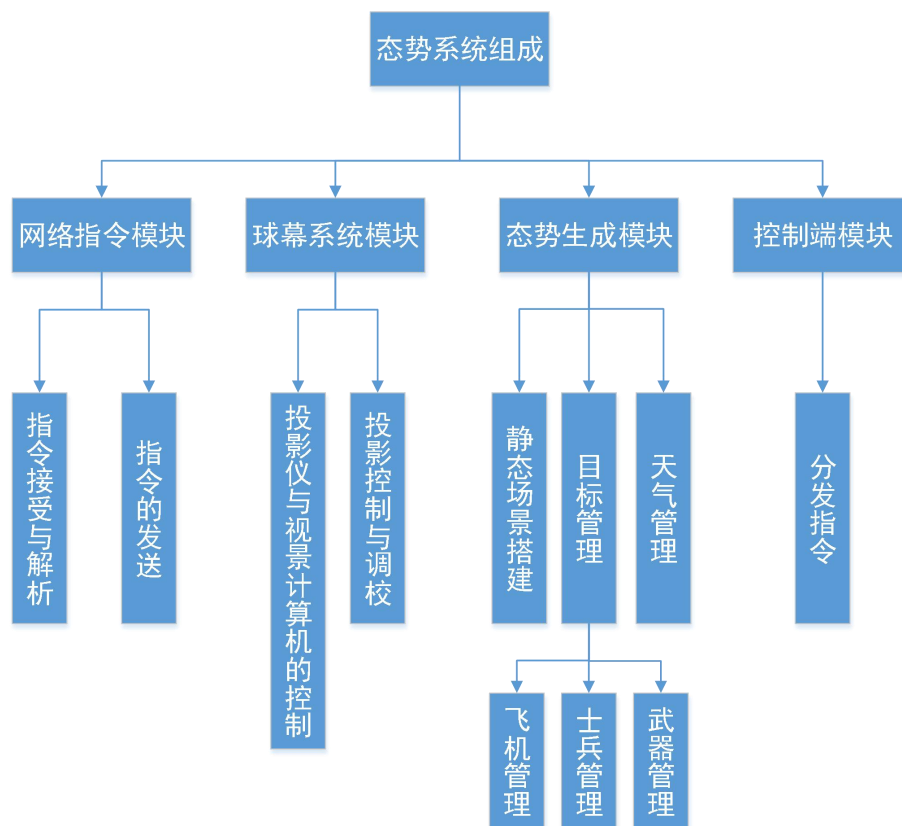


图 2.1 态势系统的组成结构

2.2.2. 系统执行方案

系统启动初期，会对球幕系统进行初始化。球幕系统中的所有的投影仪以及各席位的态势生成计算机会被开启，各席位计算机的态势生成系统程序启动，并根据控制端发送的作战想定进行初始化。

系统启动后，由控制端向各席位计算机发送控制网络指令包，各席位计算机同步地对网络包进行解析并判断指令的类型，通过通过各席位中的态势生成系统的 **GameManager** 对象来执行各条指令。

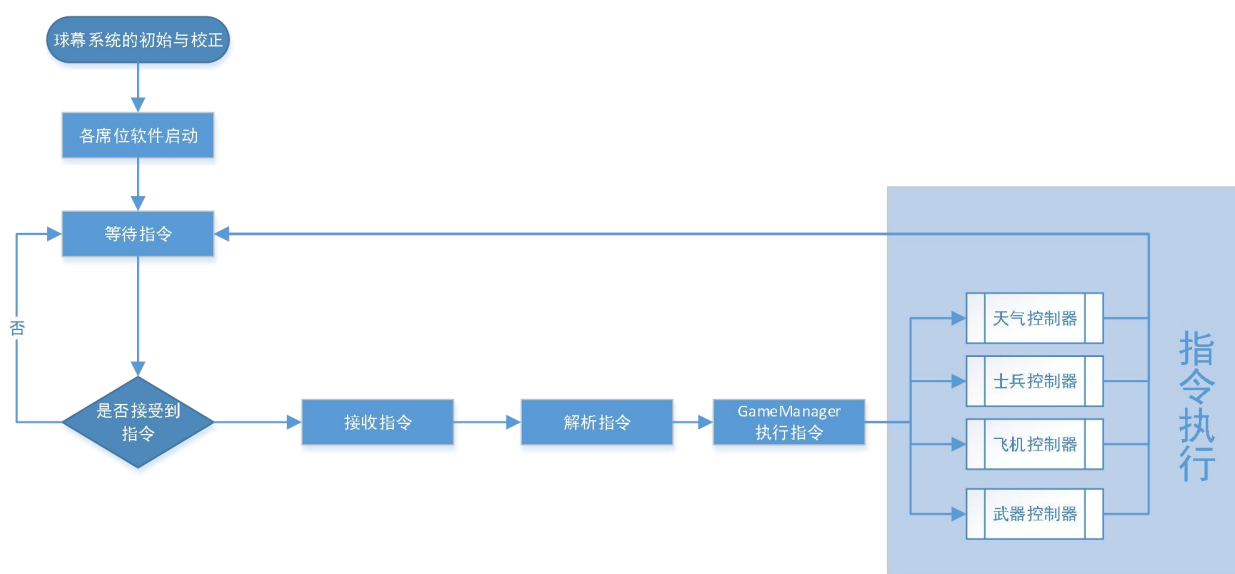


图 2.2 系统的执行流程

2.2.3. UE4 的控制管理模块（Game Manager）

Game Manager 作为 UE4 部分的总控制器，它负责各种控制器的生成、管理与控制以及网络指令系统的初始化，同时它也需要负责态势生成系统与网络指令系统之间的交互，一方面它为指令系统提供相应的命令接口，它也需要向网络命令系统发送系统当前的各种状态数据，例如发送当前存在的目标的空间状态信息、相机空间位置信息、天气状态信息等。

2.3. 球幕系统

根据球幕系统需求和功能划分，球幕系统由球幕系统硬件、投影控制与调校软件和中央控制程序配套软件组成。硬件设计采用 10 台投影机投射到正投屏幕上，进行无缝拼接，具备超大的可视角度，满足超高分辨、超大可视角度、大幅图像图形显示的需求。

球幕系统示意效果图如下：



图 2.3 球幕系统效果演示

第 3 章 全球大地形以及静态态势的搭建

3.1. Cesium for Unreal

Cesium 是 AGI 公司计算机图形开发小组于 2011 年研发的三维地球和地图可视化开源 JavaScript 库，Cesium 为三维 GIS 提供了一个高效的数据可视化平台。Cesium 用于地理数据可视化，它支持海量数据的高效渲染，支持时间序列动态数据的三维可视化，具备太阳、大气、云雾等地理环境要素的动态模拟和地形要素的加载绘制。

在虚幻引擎中，Cesium 提供了一个非常强大的地理数据可视化插件 Cesium for Unreal。Cesium for Unreal 将 Cesium 与虚幻引擎，这两个非常强大的引擎的进行结合，解锁了虚幻引擎的 3D 地理空间生态系统，它将一颗高精度的 WGS84 地球、开放 API、空间索引开放标准（如 3D Tiles）、基于云端的真实世界 3D 内容与虚幻引擎的力量相结合。

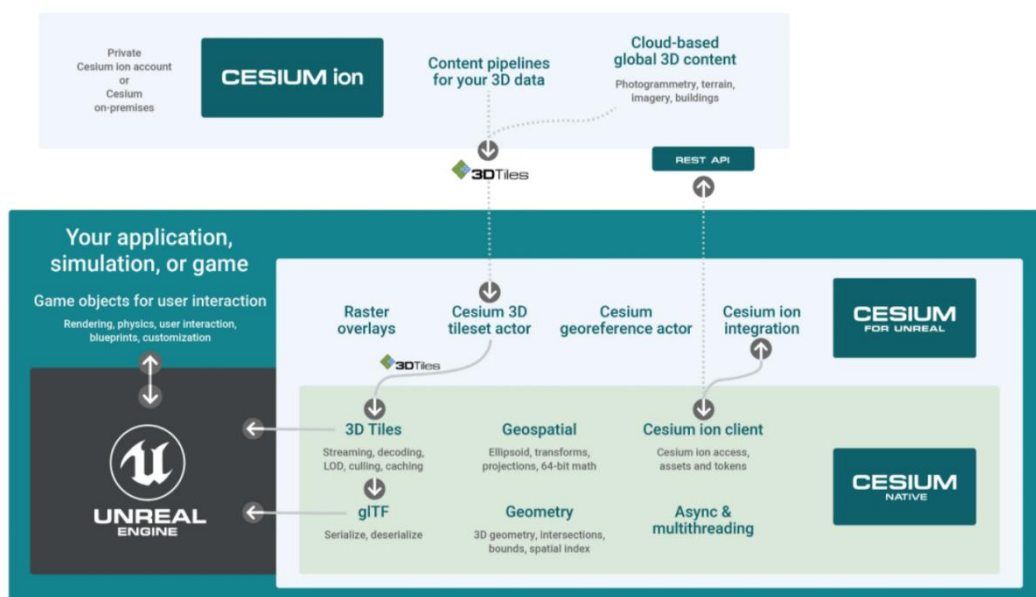


图 3.1 Cesium for Unreal 原理图

Cesium for Unreal 提供两种地理信息数据的方式：①通过访问 Cesium ion 网站，直接获取各式地球影像、地形、矢量等在线数据，读取的速度受制于网速快慢。②用户自己提供地理信息数据，用户可以离线使用 Cesium，本系统采用的是这种方式。

3.1.1. 全球地形搭建的过程

(1) 高程、影像原始地形数据下载

本系统的数据均在 The General bathymetric Chart of the Oceans (GEBCO) 网站 (<https://download.gebco.net/>)中下载，该网站支持对全球任意位置的高程以及影像数据进行

下载。进入该网页后，按住 Ctrl 与鼠标左键便可对全球的区域进行选择，点击 Add to basket 加入下载列表，再点击 Download your data，由于地形影像数据的选择下载需要服务器进行计算，我们需要输入自己的邮箱地址，下载地址会发送至邮箱。



(a) 选取数据的区域

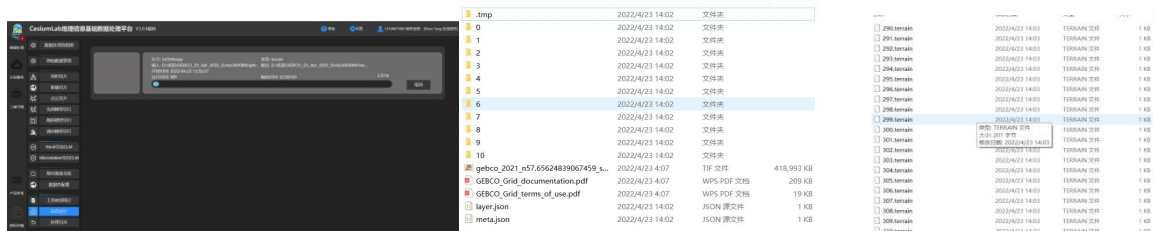
(b) 下载数据

图 3.2 在 GEBCO 中下载高程以及影像数据

(2) 利用 CesiumLab 对数据进行切片

Cesium Lab 是一个强大地理信息处理软件，它主要有两种处理功能：

①数据处理，Cesium Lab 能够对地形、影像、点云数据进行切片。原始地形数据的文件格式为 TIF，TIF (Tag Image File Format) 图像文件是图形图像处理中常用的格式之一。而 Cesium 只能对 Terrain 瓦片格式进行载入并显示，所以我们需要使用 CesiumLab 将 TIF 文件进行切片。②分发服务，CesiumLab 数据分发服务能够在本地创建一个地理信息访问端口，供局域网内的用户使用。之后打开 CesiumLab，导入从网上下载的 TIF 文件并提交处理，CesiumLab 会自行进行切片。切片后根目录中会生成多个文件夹，其中每一个文件夹代表一层 LOD，每个文件夹中存放着 Terrain 瓦片文件。



(a) 切片过程

(b) 切片结果

(c) 切片生成的 Terrain 文件

图 3.3 Cesium Lab 的切片流程

切片结果的文件夹中会生成一个 Layer.json 文件，该 json 文件为地理服务入口文件，用以来描述地形服务的基本信息和结构。

(3) 载入 Cesium 插件

进入虚幻引擎并启用 Cesium for Unreal 插件，新建 Cesium3DTileset Actor，在它的 Url

中输入 Layer.json 的地址，Cesium 插件将会自动访问该地形信息服务文件，并在视口中将地形信息进行加载。

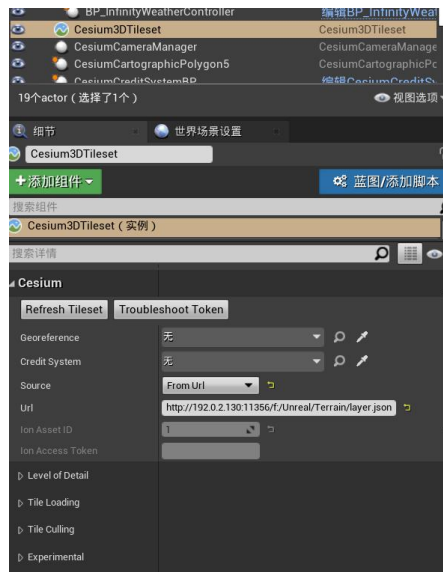


图 3.4 在 UE4 中输入 Layer.json 文件地址



图 3.5 不同 LOD 下的地形

3.1.2. Cesium 坐标系

UE4 中的坐标系不适用于态势系统，UE4 中的坐标系是以厘米为单位，XYZ 轴来描述目标的位置信息。而在战场态势系统中，是通过经纬度与高度来描述目标的位置的，Cesium for Unreal 提供了一种定位系统，它能够将 UE4 中的坐标转化为经纬度与高度。

在每一个需要被定位的目标类中加入 CesiumGlobeAnchorComponent 组件，该组件能够自动地将该目标的 UE4 坐标映射为全球经纬度坐标。

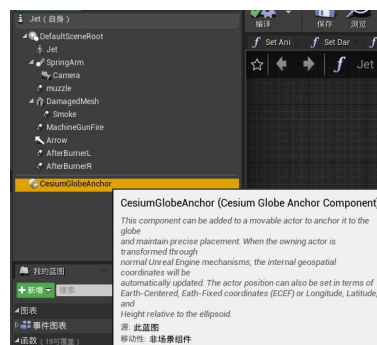


图 3.6 Jet 中的定位组件

3.2. 静态场景的搭建

为了给目标提供一个活动的示例场景，本系统对瑶湖机场周围的环境进行了建模，它的主要组成因素如下：

(1) 建筑



(a) 商业区

(b) 住宅区

(c) 厂房区

图 3.7 瑶湖机场-建筑群

(2) 桥



图 3.8 瑶湖机场-桥

(3) 植物

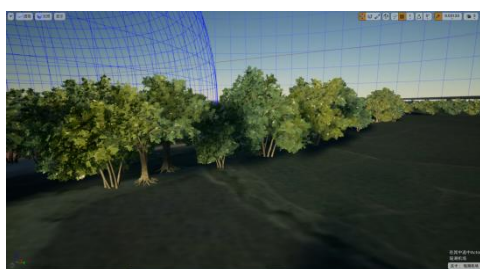


图 3.9 瑶湖机场-植物

函数名	函数描述	参数名	参数类型	参数描述
SendPacket	向控制端发送网络包	Pk	CPacket	网络包信息
		bReliable	bool	是否进行可靠传输
OnRecvData	接受网络包	Pk	CPacket	接收的网络包

表 4.2 SendPacket 与 OnRecvData 函数

SendPacket 函数在态势系统向控制端发送信息时使用，例如发送气象、心跳包、目标高度等信息。OnRecvData 函数用于接受网络包，并将网络包添加在 Receiver 的网络包列表中。

(3) Receiver

Receiver 继承至 Qt 中的 QThread 类，它提供了一个与平台无关的管理线程的方法。Receiver 线程开启后，Receiver 不断地从网络包列表中读取并解析网络包，解析完成后它再调用各个指令的回调函数来让态势生成系统做出相应改变。

(4) UE_NetProcess

由于在 Qt 与 UE4 中，存在一些重复的结构体而导致程序编译错误，便需要将 Qt 部分的代码以及 UE4 中的代码分别进行存放，UE_NetProcess 负责的是将 Qt 中接口进行再封装，方便 UE4 部分的调用。

(5) NetWorkMgr

NetWorkMgr 为 UE4 部分的网络管理器，它负责通过调用 DelegateObject 中的代理事件来实现 Receiver 中的各个指令的回调函数，并对回调函数的绑定。回调函数的实现如下，以天气特效的回调函数为例：

```
int weatherType = tWeatherEffectOrder.weatherType;
FStringsGrade= UCastClass::getUCastClassTarget()->QStringToFString
(tWeatherEffectOrder.sGrade);
AsyncTask(ENamedThreads::GameThread, [weatherType, sGrade]()
{
    UDelegateObject::getDelegateEventTarget()->
    onWeatherEffectChange.Broadcast(weatherType, sGrade);
});
```

(6) DelegateObject

DelegateObject 负责代理 GameManager 中的各项事件，方便在 C++ 部分控制 GameManager 蓝图。在 DelegateObject 对事件代理器的声明，GameManager 能够在事件代理器上绑定相应的代理事件，通过对事件代理器的调用便可以在 C++ 中对 UE4 中的蓝图进

行调用。DelegateObject 对事件代理器的声明如下，以天气特效为例：

```
//天气特效
```

```
DECLARE_DYNAMIC_MULTICAST_DELEGATE_TwoParams(FWeatherEffectOrdr,int,  
weatherType, FString, sGrade);
```

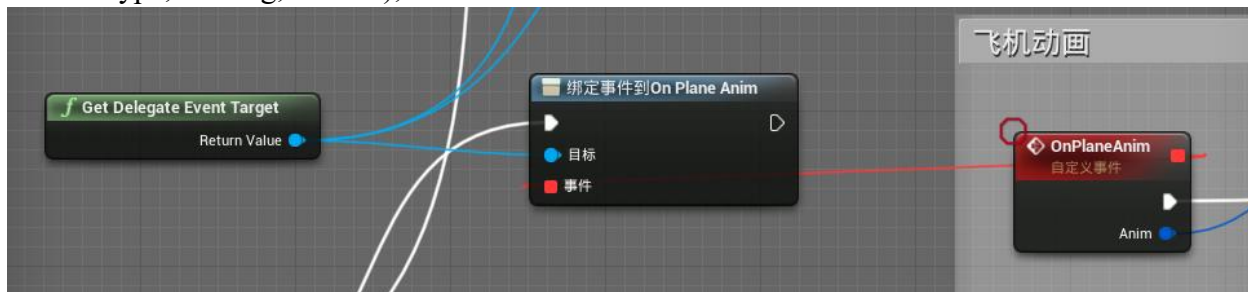


图 4.2 GameManager 绑定代理事件

4.2. 指令分类

4.2.1. 相关结构体

结构体名称	结构体描述	成员名	成员类型	成员描述
UEstTypeBaseInfo	目标的基本信息	nTypeID	int	目标的类型 ID，用于判断为飞机、士兵。
		StrTypeName	FString	目标的种类，用于判断该目标为哪种类型的飞机、士兵。
		nFraction ID	int	目标阵营 ID
		StrCountry	FString	目标所属国家
		StrOrg	FString	目标所属组织
UEstObjStatus	目标当前的状态信息	IPos	FString	目标当前经纬度以及高度的字符串信息，各项之间以逗号为隔。
		IHpr	FVector	目标姿态角度
		DSpd	float	目标当前速度，单位 m/s
		DOil	float	目标当前油量，单位加仑

表 4.3 指令相关结构体

4.2.2. 指令定义

接口名	接口描述	参数名	参数类型	参数描述
CreateOrUpdateObj	创建目标	nObjID;	FString	目标独一无二的 ID

		BaseInfo	UEstTypeBaseInfo	目标的基本信息
		ObjStatus	UEstObjStatus	目标状态信息
UpdateTime	调整 UE4 世界的时间。	Year	FString	年份
		Month	FString	月份
		Day	FString	日期
		Hour	FString	小时
		Min	FString	分钟
		Second	FString	秒钟
PlaneAnim	设置飞机动画	sID	FString	飞机 ID
		Left Flat Angle	float	左平尾翼角度
		Right Flat Angle	float	右平尾翼角度
		Rudder Angle	float	方向舵角度
		Left Landing Gear Angle	float	左起落架角度
		Right Landing Gear Angle	float	右起落架角度
		Front Landing Gear Angle	float	前起落架角度
		Left Flap Angle	float	左襟翼角度
		Right Flap Angle	float	右襟翼角度
		Left Front Flap Angle	float	左前襟翼角度
		Right Front Flap Angle	float	右前襟翼角度
		Left Aileron Angle	float	左副翼角度
		Right Aileron Angle	float	右副翼角度
		Front Wheel Angle	float	前轮角度
		Speed Reducer Angle	float	减速板角度
ViewEffect	设置观察特效	Mode	float	切换特效的模式
WeatherEffect	设置天气特效	Type	int	天气特效类别（云、风、电、冰雹等）
		Grade	int	天气特效等级
ViewChange	设置自由视点	IPos	FString	视点的经纬度以及高度的字符串信息，各项之间以逗号为

				隔。
		IHpr	FVector	视点的姿态角度
FollowView	视点跟踪	SHandle	FString	目标 ID
		IHpr	FVector	视点的姿态角度
		IsSync	bool	姿态是否与目标保持一致
SetCloudAltitude	设置云高云低	fHigh	float	云最低海拔,单位 m
		fLow	float	云最高海拔,单位 m
SetWind	设置风速风向	fSpeed	float	风速, 单位 m/s
		fHead	float	风向, 单位度
Gatlin	发射子弹	StrObjID	FString	子弹发射方 ID
		IPos	FString	子弹生成的位置
		wID	FString	子弹 ID
RequestAlt	获取目标的海拔高度	IDArray	TArray<FString>	目标 ID 数组
LaunchWeapon	发射武器	LauncherID	FString	发射方 ID
		WeaponID	FString	武器 ID
		WeaponType	int	武器类型 0,1,2,3 分别为子弹、无动力炸弹、导弹、火箭弹
		TargetLocation	FString	武器类型为导弹时, 目标位置的经纬度以及高度的字符串信息。
		TargetID	FString	武器类型为导弹时, 目标 ID
		bAutoPilot	bool	是否自动导航
SoldierAnim	设置士兵动作	SoldierID	FString	士兵 ID
		AnimationType	int	动作的 ID
		bUse	bool	开始或者停止

表 4.4 指令的定义

第 5 章 天气模拟

5.1. 云

5.1.1. UE4 体积云组件与体积材质

若要使态势系统中的云更加真实，同时能够达到目标从云中穿过的效果，就必须规定系统中的云不能是后期处理形成，而应该是实实在在存在在三维空间中，为了达到这个目的，便需要用到 UE4 中的体积云插件。UE4 中的体积云组件是基于物理的云渲染系统，该系统使用材质驱动方法，允许设计师自由地创建项目所需的任意类型的云。云系统处理动态实时设置，并通过使用实时捕获模式的天空大气 (Sky Atmosphere) 和天空光照 (Sky Light) 进行补充。

UE4 中的材质分为表面、体积、后期处理等若干类，而体积云的材质域为体积材质，体积材质的输出节点包括如下几个：

- ①反射率，定义体积材质的整体颜色，通过此输出可以控制云呈现的颜色。
- ②自发光颜色，控制材质自身发光颜色，通过此输出可以控制夜间云的发光颜色。
- ③消光，可以模拟光略过体积表面时引起的颜色的变化，从而更真实的模拟云与太阳光之间的光照互动。
- ④环境光遮挡，模拟表面缝隙中出现的自阴影。

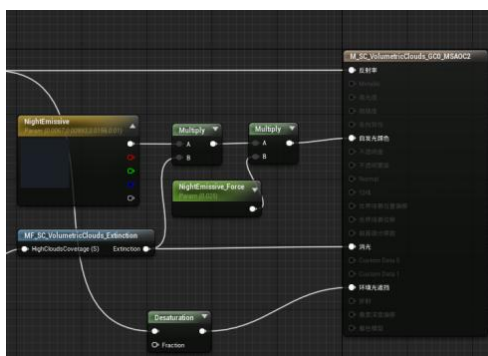
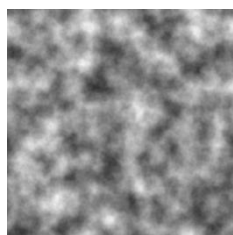


图 5.1 体积材质域的输出节点

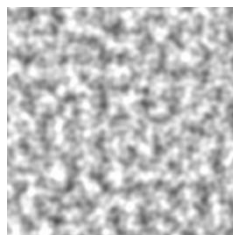
5.1.2. 云的形状

现实中，云的形状千变万化，同时各个高度的云层特点也不尽相同，为了能够生成不同高度、不同形状的云，本系统设计了两个云层：HightCloudLayer 与 LowCloudLayer，两种云层的特点各不相同。

①**LowCloudLayer** 为低云层，它分为主云层与次云层，次云层是主云层的一个附加，它能够模拟现实世界中最常见的层云以及层积云。低云层的形状及高度由以下三个基础形状贴图进行控制：



(a) 主云层



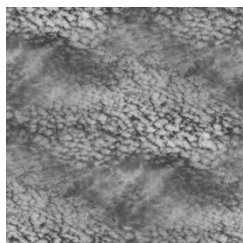
(b) 次云层

图 5.2 低云层基本形状纹理贴图

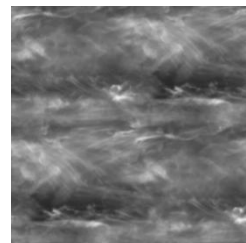
②**HighCloudLayer** 为高云层，它一共有三种云形状可选：绵长纤维状云、卷积云、烟雾状散云。



(a) 烟雾状散云



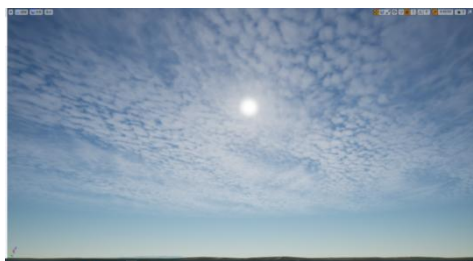
(b) 卷积云



(c) 绵长纤维状云

图 5.3 高层云基本形状纹理贴图

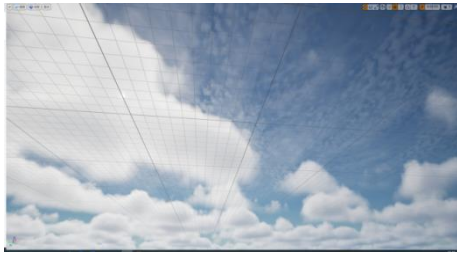
云系统支持对云的各种参数进行调整，包括云层的浓密程度、高度、覆盖率、绵延广度、各种形状云所占百分比等参数。通过改变这些参数，可以模拟自然中各种云。本态势系统主要对 4 种常见的云进行仿真：卷积云、纤维状积云、层云以及层积云。以下为 UE4 中的效果图：



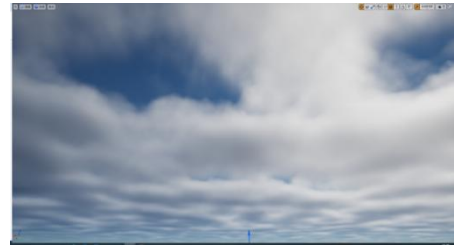
卷积云



纤维状卷云



层云



层积云

图 5.4 云的形状展示

5.1.3. 云的实现原理图

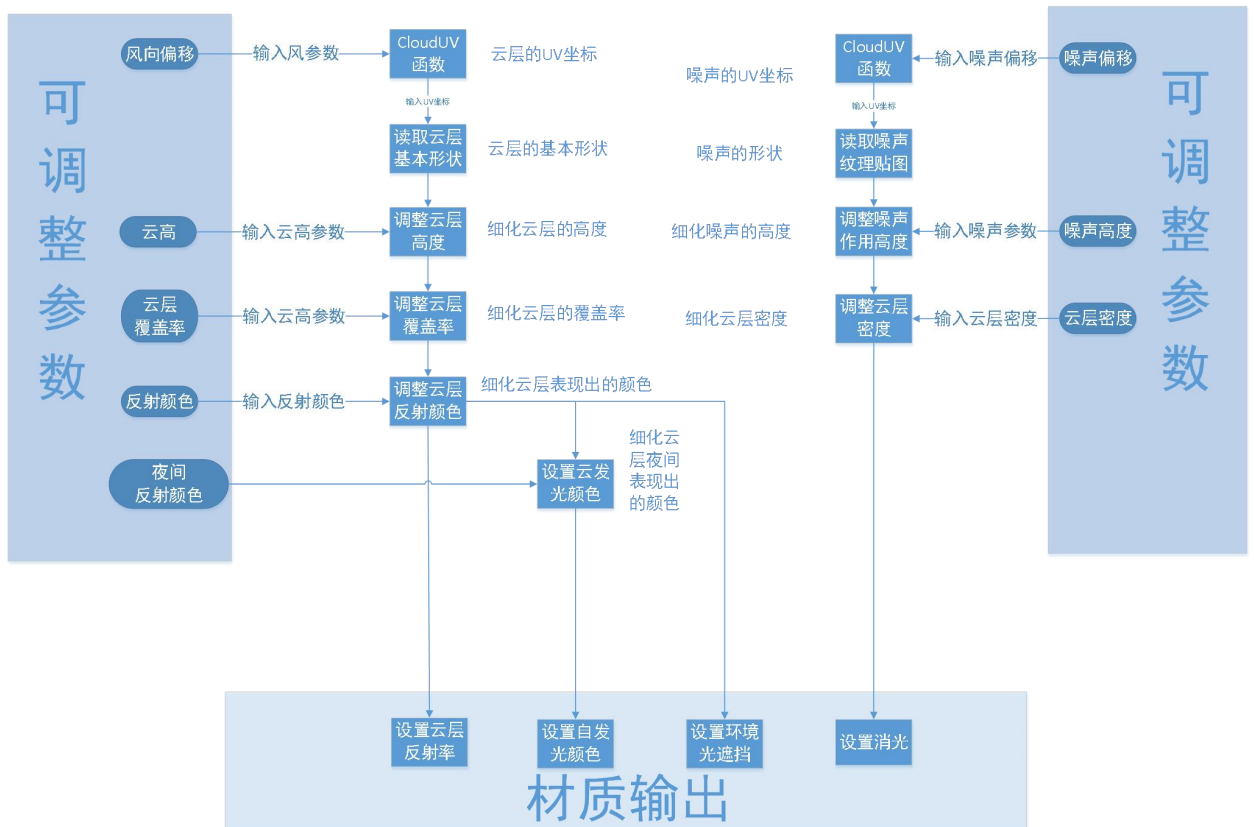


图 5.5 云实现的原理图

5.1.4. 云的移动

云的形状图为静态的，如果要模拟云随着风飘动的效果，就需要实时更新并改变云形状纹理贴图的 UV 坐标。在材质蓝图中 MF_SC_CloudUV 用于处理 UV 坐标的变化。它接收初始云位移以及风向位移参数。在 Weather Controller 中，根据设定的风向偏向角以及风速的大小，计算偏向向量，并更改云材质实例的 LowCloudMap_Offset 参数，以实现云的偏移。

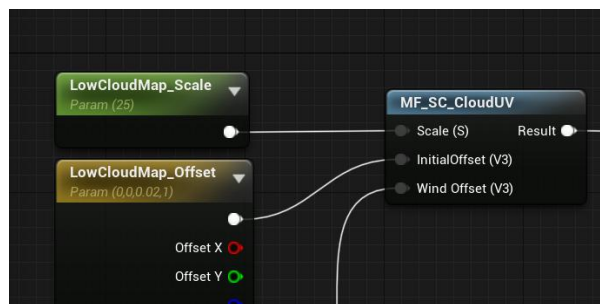
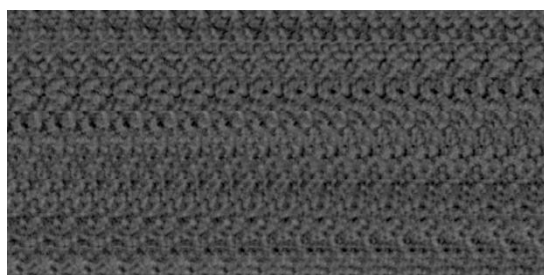


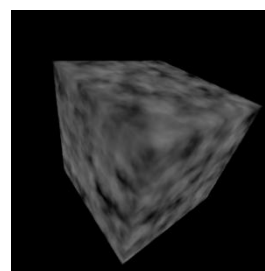
图 5.6 云基本形状贴图的 UV 坐标变化

5.1.5. 云的动态效果

现实中的云的形状不是一成不变的，同时云的轮廓并不会非常鲜明，是朦朦胧胧的。要实现这种效果，需要用到体积噪声纹理，增加云形状的随机性。所谓噪声，可以看做是一个随机数生成器，在云材质中，为了使得云的形状无规律，形状更多变，变需要云噪声来实现这种效果。云噪声作为噪声的一种，在 UE4 中表现为一种体积纹理。



(a) 噪声体积纹理的展开图



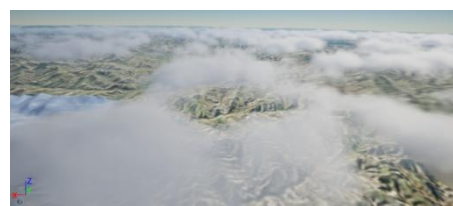
(b) 表现为立方体的噪声体积纹理

图 5.7 噪声体积纹理

将图 5.6 (b)进行水平方向的横切，切出来的切片就是图 5.6(a)中铺开的图像。



(a) 添加噪声前



(b) 添加噪声后

图 5.8 添加云噪声前后的变化

将噪声附加至云的基础形状后，云层变得更加细碎、朦胧。同时，我们可以通过改变噪声纹理的 UV 坐标变化，使得云层“摆动”起来。在 Weather Controller 中每帧更新云材质实例的噪声水平与垂直方向的向量偏移，可以让每一时刻的云噪声都不相同。

5.1.6. 战机云中穿越效果

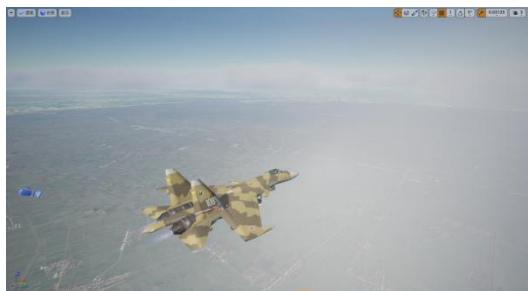


图 5.9 战机从云中穿过

5.2. 雾

雾通过 UE4 中的指数高度雾组件实现。该组件模拟一种可以随着高度衰减的雾气，营造场景的氛围感。通过调整组件中的雾最大不透明度参数，可以得到不同等级的雾气效果。



图 5.10 雾中的士兵正在巡逻

5.3. 霾

态势系统的天空大气模拟利用了 UE4 中的天空大气组件，该组件能够提供一种光光进入行星大气层中参与介质发生的散射的近似散射。

光与大气中悬浮的灰尘、花粉、空气污染物等更大的粒子相互作用会产生 Mie 散射，这也是自然中霾的产生。通过改变天空大气组件中的 Mie 散射范围，可以得到不同等级的霾效果。



(a) 一级霾



(b) 二级霾

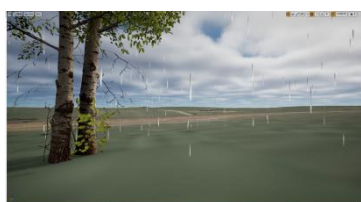
图 5.11 霾的效果展示

5.4. 雨、雪

本态势系统利用 Sky Creator 插件来对雨、雪天气现象进行模拟，SkyCreator 是一款强大的天气模拟插件，它内置了一个天气粒子系统，能够对雨、雪进行非常逼真的模拟。

在 Weather Controller 中加入 UNiagaraComponent 组件，将该组件中的 Niagara 系统资产设置为 SkyCreator 提供的 SC_WeatherFX_System 文件，SC_WeatherFX_System 可以对雨、雪粒子的多种参数进行调整，例如生成速率、重力大小、生命周期、雨滴雪花的大小、雨溅的大小、风速风向等参数。

(1) 雨



(a)一级雨



(b) 二级雨



(c) 三级雨

图 5.12 不同等级的雨

(2) 雪



(a)一级雪



(b) 二级雪



(c) 三级雪

图 5.13 不同等级的雪

第 6 章 作战目标模拟

6.1. 作战目标的控制与管理

三种目标的控制与管理交由各自的 Controller 进行管理，GameManager 通过调用三种 Controller 的接口来实现控制。

6.2. 飞机模拟

6.2.1. 飞机骨骼动画

(1) 骨骼统一与动画重定向

本系统一共支持 F16、F18C、F22、幻影 2000、歼-15、歼-16、歼-20 等战机的模拟，若是为每一种骨骼的战机制作一个动画蓝图，将会耗费许多的时间与精力，所以本系统在对战机进行建模时，统一了每一种机型的骨骼命名以及骨架结构。因为在 UE4 中，只有骨骼架的结构完全一致，才能使用动画重定向的功能。

动画重定向允许将一套骨骼上的动画资源重定向至另外一套骨架结构相同的骨骼中。在使用这一项功能时，由于各种战机的骨骼位置以及旋转角依然存在细微的差异，会导致骨骼错位的现象，所以在重定向之后，依然需要对各种机型的动画蓝图进行细微的调整。



(a) 将 F16 的骨骼重定向至 F18C

(b) F16 重定位后左右平尾翼发生错位

图 6.1 骨骼重定位

(2) 骨骼控制

飞机的机翼一共包括左右平尾、左右副翼、左右襟翼以及左右前襟翼，飞机的其他骨骼包括减速板、左右前起落架等。由于每一处骨骼的位移以及旋转角度不同，需要将各处骨骼进行混合，这就需要用到 UE4 中内置的函数“每个骨骼的分层混合”。在该函数中键入骨骼名称以及混合比例，便可以在同一时刻改变不同位置的骨骼的状态，而不会发生动画冲突。

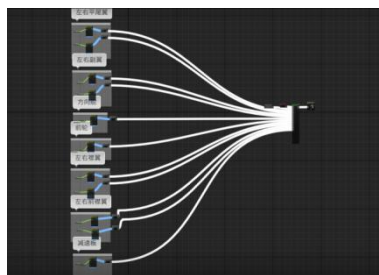
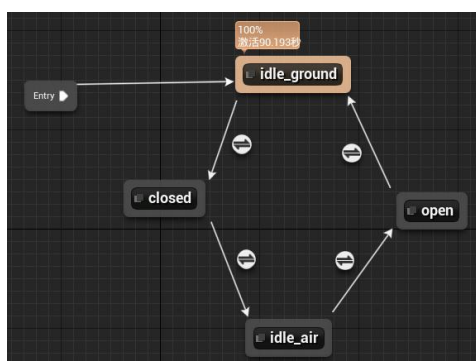


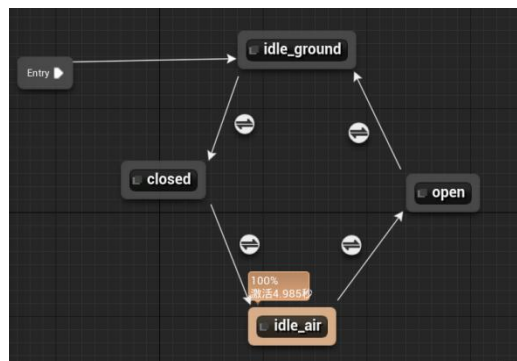
图 6.2 对各处骨骼的变化进行混合

(3) 起落架状态机

飞机的起落架的收放的状态机如下图：



(a)idle_gorund: 起落架放下



(b)idle_air: 起落架收起

图 6.3 起落架状态机

飞机的初始状态为 idle_ground，飞机的起落架处于打开状态，当飞机的 Wheel_Open 状态位被置为 false 时，飞机开始播放起落架收起的动画，当动画播放完毕后，飞机又切换至 idle_air 状态，即飞机飞行的状态，起落架打开的原理同理。

(4) 飞机动画的两种状态

本系统为飞机的动画提供了两种状态：

①由主控命令控制：飞机的所有动画均有主控发送的 PlaneAnim 指令进行控制，该指令的参数包括飞机所有的骨骼参数，例如左右襟翼、左右平尾翼等。在这个状态中，bAutoAnim 状态位被置为 false，动画蓝图实时监听 Jet 对象的骨骼变量，再更新自身的变量，从而实时地对指令做出响应。

②由飞机的姿态角度控制：飞机的所有动画均由飞机的姿态角度 HPR 进行控制。在这个状态中，bAutoAnim 状态为被置为 true，动画蓝图实时监听 Jet 对象的 HPR 角度，并更新自身的变量。根据飞机机翼空气动力学原理，分别创建左转、右转、上升、下降的动画帧，并利用 UE4 的动画混合文件对动画帧进行混合。只需要传入飞机的 HPR 角度，动画混合文件能够根据角度自动进行差值计算，并更改飞机的机翼姿态。



图 6.4 Pitch、Roll 动画混合

6.2.2. 飞机特效

(1) 飞机尾焰

飞机尾焰是由粒子系统制作，它被绑定在飞机尾部排气筒位置。它一共由六种级联粒子组成：尾焰内焰，尾焰外焰以及四个光晕。



图 6.5 飞机尾焰的级联粒子



图 6.6 飞行中尾焰的效果

(2) 战损飞机特效

当飞机被击中摧毁后，将飞机的骨骼网格体设置为不可见，同时将其替换为毁伤模型，同时将火焰燃烧的特效启用，这样一个简易却真实的毁伤特效便做好了。



图 6.7 飞机的毁伤特效

6.2.3. 飞机运动控制

GameManager 中的 OnCreateOrUpdateObj 接口用于创建或更新目标，GameManager 创建出飞机之后，一直不断地调用 Jet 对象的 Update 函数以控制飞机的位置、姿态角、速度等参数。

6.3. 士兵模拟

6.3.1. 士兵的状态

士兵一共有稍息、蹲、爬、使用 iPad、使用望远镜、掉落、死亡、开枪、换弹、巡逻、警戒、搜寻，12 种状态，由一个复杂的士兵动作状态机来负责管理士兵的状态，状态机的内容如下图。

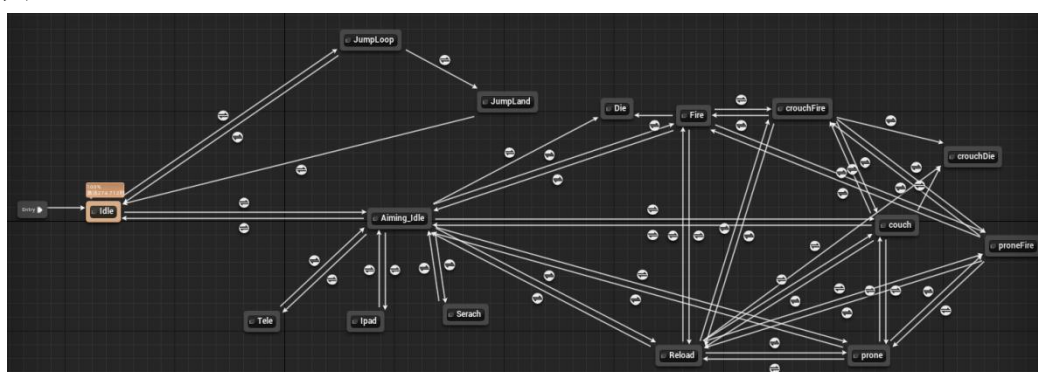


图 6.8 士兵的动作状态机

士兵最开始处于 idle 状态，即稍息放松状态，当某一个状态位发生改变时，士兵将会立马转移至该状态，以下是士兵的主要状态：

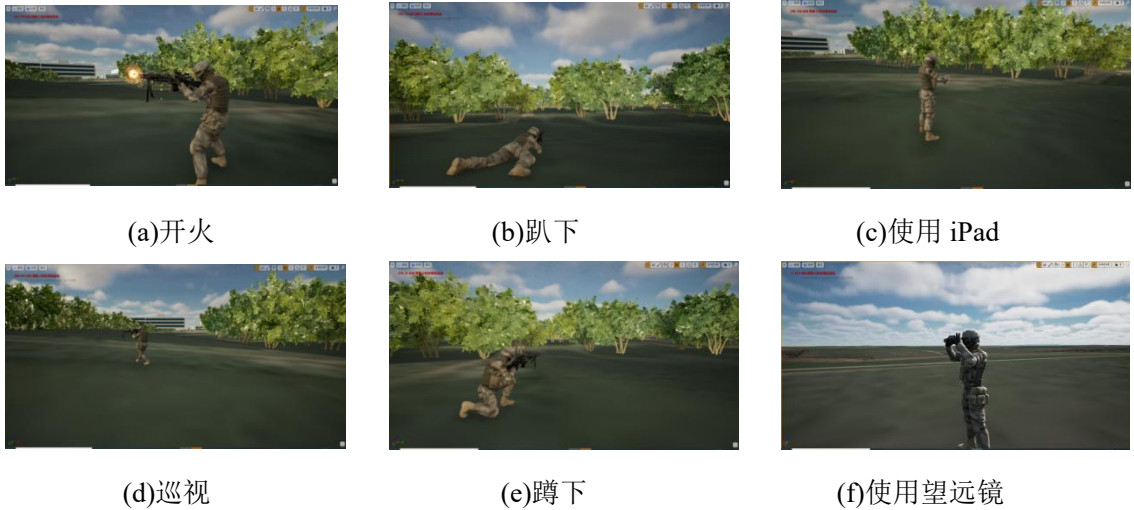
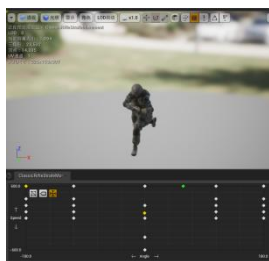


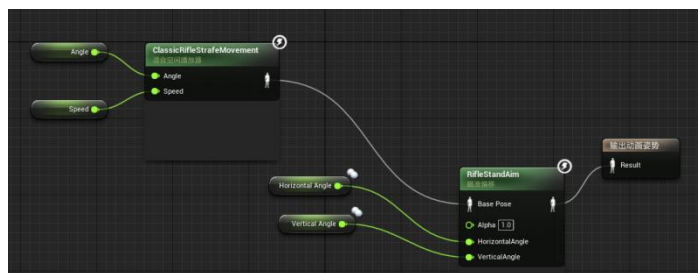
图 6.9 士兵的各种姿势

6.3.2. 士兵行走

首先确定士兵行走的两种参数：行走的速度 **Speed**（单位为 **cm/s**）、行走的方向 **Angle**（单位度，向着正前方时为 0° ，顺时针为正，逆时针为负），以这两个参数作为控制士兵行走动画的参数。依照士兵骨骼分别创建士兵奔跑、行走、慢走、停止的动画序列，以及士兵向左、向右、向前、向后行走的动画，之后创建二维动画混合空间文件，命名为 **ClassicInfantryMovement**，此文件用以控制士兵的行走移动。以 **Angle** 为横坐标，**Speed** 为纵坐标，根据动画与速度方向的关系，分别将各种动画序列拖至混合空间中恰当的点，之后在士兵的动画蓝图中调用 **ClassicInfantryMovement**，将参数 **Angle** 与 **Speed** 传入。经过测试，士兵能够根据给出的方向与速度作出正确的动作。



(a)ClassicInfantryMovement 混合空间



(b)调用 ClassicInfantryMovement

图 6.10 士兵行走的混合空间

6.3.3. 士兵警戒

士兵处于警戒状态时，需要下蹲并向四处瞄准。要实现此功能，需要用到 TimeLine 节点。在 Vigilance 事件后边创建并双击 TimeLine 节点，TimeLine 节点中横轴为时间，纵轴为变化的参数，点击左上方添加浮点型轨道，并在曲线中添加三个浮点节点(0,-64), (10,64), (20,-64)，意即在 20 秒内，士兵的瞄准水平角由-64° 平滑移动至 64°，再平滑移动至-64°。

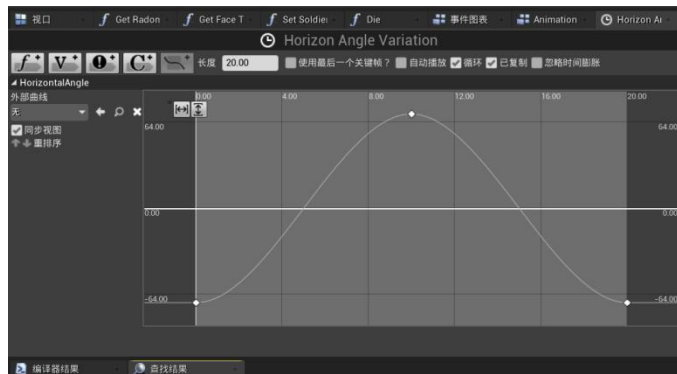


图 6.11 士兵警戒中水平、垂直角变化曲线

6.3.4. 士兵巡逻

士兵巡逻一共分为两种：

(1) **指定路线巡视**，由指令给定的若干路径点进行巡逻，指令可以指定循环巡逻还是单次巡逻。循环巡逻时，当士兵到达最后一个路径点时，将会将目标点索引置为 0，重新进行巡逻，直至指令命令停止巡逻。

(2) **中心巡逻**，指令告知士兵一个中心点以及中心范围，士兵将会一直在中心点的一定范围之内进行巡逻，每次到达目标点之后，便会在中心范围内随机加入一个巡逻点，直至指令命令停止巡逻。

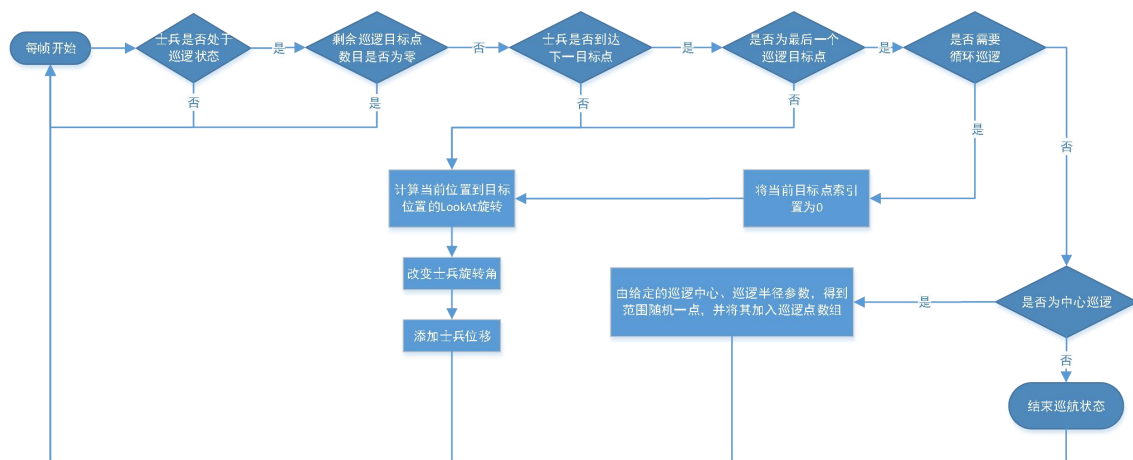


图 6.12 士兵巡逻的逻辑流程图

6.3.5. 士兵射击

士兵由指令控制开火，士兵会朝着当前瞄准的方向进行射击。它的主要实现逻辑如下：

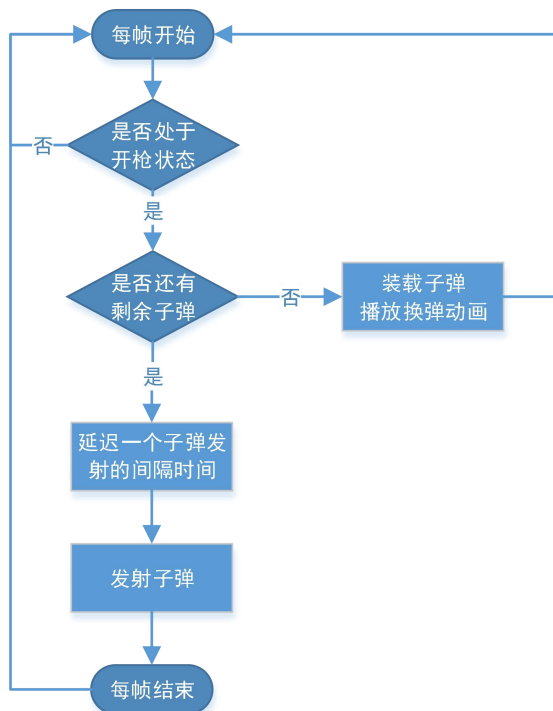


图 6.13 士兵射击的逻辑流程图

6.4. 武器模拟

6.4.1. 武器的控制与管理

本系统一共实现了子弹、炸弹、火箭弹、导弹四种武器的模拟。所有的武器继承于一个父类 **WeaponParent**，它的基本组件如下表所示：

组件名	组件类型	组件作用
Sphere	SphereCollisionComponent	处理武器类的碰撞
CesiumGlobalAnchor	CesiumGlobalAnchorComponent	用于定位武器的全球位置

表 7.1 WeaponParent 的组件

WeaponParent 包含以下虚函数，所有继承至 **WeaponParent** 的类都需实现这些函数。

虚函数名	函数描述	参数名	参数类型	参数描述
MultiParticle	武器生成特效	Location	FVector	特效的位置
		Rotator	FRotator	特效的旋转角
Delete	武器消亡时调用函数	Location	FVector	武器消亡时所处的位置
Update	武器受控制器控制，每帧进行更新	LLH	FString	武器的经纬度及高度的字符组合，间隔符为“，”
Init	武器初始化调用函数	None	None	None

表 7.2 WeaponParent 的虚函数

6.4.2. 无动力炸弹

无动力炸弹只能由飞机目标进行投放，无动力炸弹的初始速度为飞机目标投放时的速度。在投放之后，炸弹做抛物线运动，当撞击至地面或者目标时，便会发生爆炸。炸弹的重力由组件 **ProjectileMovement** 进行处理，该组件专门用来控制发射物移动，之后的火箭弹、导弹、子弹均会用到此组件。



(b)战机投发无动力炸弹

(c)无动力炸弹爆炸

图 6.14 无动力炸弹的投放与爆炸图

6.4.3. 导弹

导弹具有两种运动模式，一种为指令控制导航，另一种为自动制导。当导弹由指令控制时，**GameManager** 调用导弹的 **Update** 函数控制导弹的每一帧的运动；当导弹为自动制导模式时，也有两种制导模式，一种为目标的位置，一种为目标的 ID，导弹制导的流程如下：

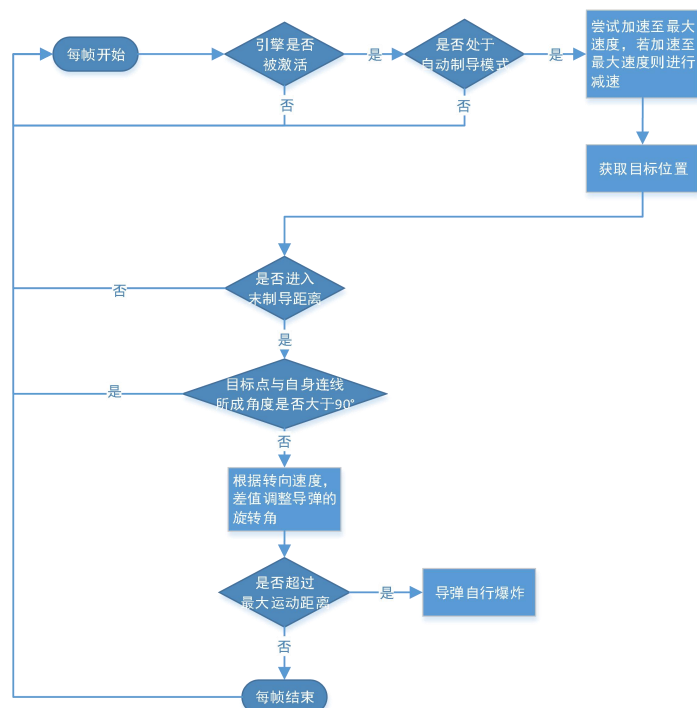


图 6.15 导弹制导的逻辑流程

导弹处于制导模式时，导弹的运动可分为三个阶段。

(1) 引擎未点火

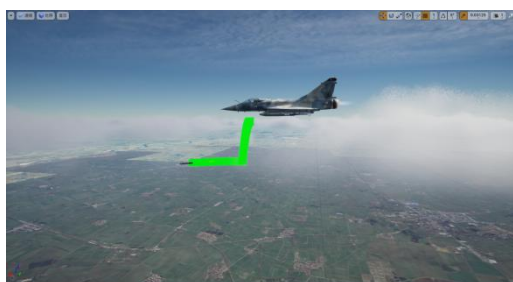
导弹会做一段抛物线运动，此时导弹只受到重力的影响。

(2) 加速状态

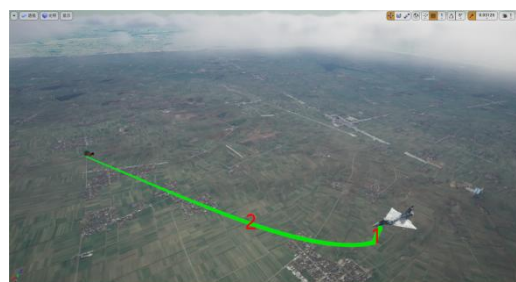
引擎点火之后，导弹便会尝试自行加速至最大速度，如果达到最大速度，将会慢慢进行减速，直至速度降为 0。

(3) 制导阶段

当目标点进入末制导范围时，导弹根据目标的 ID 实时获取目标的位置信息，判断导弹与目标之间所成的夹角是否大于 90 度，如若大于 90 度，说明无论导弹的旋转角无论如何调整都无法命中目标，这时将不会对导弹进行任何操作，直至它运动至最大距离自行爆炸；如若角度小于 90 度，说明导弹依旧可以调整自身旋转角以命中目标，导弹将会根据自身的转向过载调整旋转角度。



(a)阶段一：导弹只受重力影响



(b)阶段二：制导

图 6.16 导弹的两个阶段

图中绿色线条所示为导弹运动的路径，当导弹处于 1 号数字标识的航迹时，导弹正处于自由落体阶段，此时导弹引擎并未工作；当导弹处于 2 号数字标识的航迹时，目标点已经进入末制导距离，它会根据目标位置实时更新自身的旋转角，直至运动方向与目标点处于一条直线上。

6.4.4. 火箭弹

火箭弹与导弹相似，只不过火箭弹不存在引擎未点火状态以及制导。火箭弹的引擎在战机发射时就会被点燃，火箭弹只存在加速过程，当加速至最大速度（燃料耗尽）时，将会减速，直至超过自身最大运动距离。



图 6.17 幻影 2000 战机发射火箭弹

6.4.5. 子弹

子弹在发射时就会达到最大速度，运动过程中不断减速，直至速度减为 0 为止。

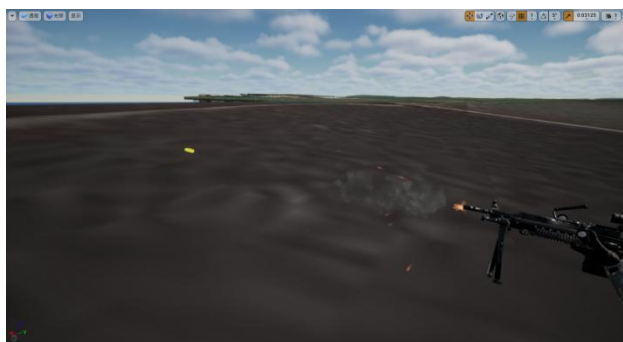


图 6.18 士兵正在发射子弹

第 7 章 结语

本文就以上态势系统存在的问题与需求,设计了一个基于 UE4 的战场态势可视化系统,在设计中,为了使模块之间耦合性更低、可维护性更高,我们把系统划分成控制端模块、球幕系统模块、视景生成模块以及网络指令控制模块。其中视景生成模块以及网络指令控制模块由本人完成实现。态势系统中本人实现的主要功能如下:

- (1) 全球范围大地形的搭建。为解决 UE4 中全球大地形构建的困难,我们从互联网中下载了高精度的地理信息数据,并利用 CesiumLab 工具对其进行处理与切片,之后在 UE4 中,用 Cesium for Unreal 插件对数据进行加载与显示,并且利用该插件中提供的 CesiumGlobeAnchorComponent 可以完美解决 UE4 中的坐标转换问题。
- (2) 实现网络指令系统。态势系统各模块之间的通信交由网络系统来进行处理,由控制端控制态势的生成以及球幕的初始化等设置。各个模块之间各司其职,互不干涉,这大大提高了各个模块的通用性。
- (3) 天气现象仿真模拟。本态势系统提供对多种天气现象的控制,例如雨、雪、云、雾、霾等。通过天气系统,我们可以模拟出多种天气现象:晴天、阴天、暴风雨来临、闪电等现象,使得战场环境更具多样性、复杂性。
- (4) 目标模拟。本系统对飞机、士兵以及武器目标进行了模拟。目标的创建与更新可以由控制端发送的指令进行控制,也可以自行进行仿真运动。

参考文献

- [1] 石露.基于 Unity 跨平台坦克虚拟驾驶态势仿真系统研究[D].西安:西安电子科技大学,2020:1-15
- [2] 李宁.分布式虚拟战场态势仿真系统研究[D].上海:上海交通大学,2015:9-13
- [3] 贾连兴,单维峰,鲁云军,翟成功,刘华.作战仿真研究热点及发展[J].系统仿真学报,2007(14):3148-3149
- [4] 田赤军,李艳雷,唐成,耿化品.国外军用训练模拟技术发展分析[J].战术导弹技术,2015(4):108-109
- [5] 孙柏林,靳大安.外军建模与仿真综述[J]. 计算机仿真, 2002(1):4-6
- [6] 徐享忠,郭齐胜.军用仿真发展现状与展望[J].装甲兵工程学院学报,2019(1):75-77
- [7] 黄进.空战战场环境模拟关键技术研究[D].四川:电子科技大学,2015:2-8,15-20
- [8] 孙峥皓,杨利民,廖馨,张琦.关于虚拟战场态势仿真系统设计研究[J].计算机仿真,2018(12):309-312
- [9] 李强. 基于 OGRE 3D 引擎的漫游系统的研究[D].太原:太原理工大学.2007,07
- [10] 孙家广,胡事民.计算机图形学基础教程[D].北京:清华大学出版社,2005,7-15
- [11] 邸锐(免费打工仔). OGRE 3D 游戏开发框架指南[M].北京:电子工业出版社,2005,5-28
- [12] 张永亮, 苏剑飞, 郑慧娟, 许世勇.某型自行火炮指挥车训练仿真系统设计[J]. 火炮发射与控制学报, 2009(04):50-54.

致谢

时光如白驹过隙，我已经在四川大学度过了四年本科生活，值此本科毕业论文结束之际，谨向所有在本科阶段所有帮助我的老师，同学表达我最真挚的问候与感谢。

首先我要感谢的是我的论文导师，他在我做毕业设计的过程中，给了我许多非常有用的建议。当我的研究在遇到各项问题时，他鼓励我努力研究、专心思考，直到最后将问题都解决。在此，谨对老师的谆谆教诲表达最真挚的感谢。

然后我要对跟我共同开发本系统的师兄李子奇、韩勇军以及师姐陈莉楠表达衷心的感谢。陈莉楠师姐在建模方面帮助了我很多，为系统提供了很多精美的飞机以及士兵模型，同时在场景搭建方面也功不可没。李子奇师兄在网络部分的搭建帮助了我很多，很多时候都是他的话点醒了我，让我知道了问题所在。韩勇军师兄在系统测试、以及球幕展示方面帮助了我很多，为我介绍了很多这方面知识，让我们的系统能够很好在球幕上进行显示。

之后我要感谢我的学校——川大，是川大为我提供了这样一个学习的平台，让我认识到了很多厉害的同学，让我在知识的海洋里遨游，丰富了我的知识与见识，感谢川大对我的培育。

最后我要感谢我自己，谢谢自己在很多个忙碌的日子不懈努力，认真完成各项系统需求，不断地学习进步。