轻量级 Web3D 地铁火灾逃生在线规划关键技术

艾子豪^{1,2}, 胡永豪³, 闫丰亭³, 张惠娟³, 王冬青³, 青胜蓝³,
朱合华³, 贾金原^{3*}

1. 吉林动画学院虚拟现实学院, 长春 130012

2. 长春理工大学计算机科学技术学院, 长春 130022

3. 同济大学软件学院, 上海 201804

* 通信作者. E-mail: jyjia@tongji.edu.cn

收稿日期: 2018–10–17; 接受日期: 2018–11–12; 网络出版日期: 2019–04–11

中央基本业务经费基金 —— 科技创新计划 (批准号: 0200219153)、中央基本业务经费基金 —— 重点领域交叉合作项 (批准号: 2100219066)、吉林省科技攻关计划项目课题 (批准号: 20170203004GX) 和吉林省科技攻关计划项目课题 (批准号: 20170203003GX) 资助

摘要 基于移动互联网浏览器实现地铁站逃生模拟训练是一种高效率的火灾逃生训练方式. 然而, 由于地铁站规模庞大且火灾情景复杂, 在线逃生路径规划仿真平台模型因数据规模大, 其在基于有限网络带宽传输以及渲染能力较弱的浏览器上运行时, 速度将非常缓慢甚至无法运行. 为解决此问题, 本文针对轻量级 Web3D 地铁火灾逃生路径在线规划平台实时在线关键技术进行了研究. 首先, 针对大规模地铁站 BIM 静态场景数据, 通过语义和体素化成分检验的轻量化方法对其进行了轻量化处理. 同时, 针对大规模虚拟化身的在线渲染, 基于数据拆分并灵活组合思想, 通过对虚拟化身的几何体信息和虚拟化身的动画数据进行数据管理, 实现了大规模虚拟化身在线渲染的轻量化处理, 进而实现了轻量级人群可视化; 其次, 针对动态烟气数据, 提出了基于烟气冗余消除和归一化的轻量化处理方法, 并基于精灵纹理粒子系统构建了轻量级烟气场景, 实现了轻量级烟气可视化; 最后, 基于上述一系列轻量化处理的 Web3D 地铁场景中的逃生路径规划问题研究, 本文提出了基于虚拟足迹聚类的蚁群优化算法 eAACO (evacuation based on adaptive ant colony optimization), 该算法通过 VR 设备获取真实人群逃生路径, 实现对路径数据筛选和聚类以形成骨干路径, 并与蚁群算法 (ACO, ant colony optimization) 相结合, 设计了逃生路径规划的最优方案. 实验表明, 上述关键技术的实现较好解决了大规模地铁站火灾逃生路径规划 Web3D 模拟平台的实时在线处理问题.

关键词 Web3D, 地铁站 BIM, 轻量级烟气可视化, 轻量级人群可视化, 虚拟足迹聚类, 火灾逃生路径规划, 蚁群优化算法

引用格式: 艾子豪, 胡永豪, 闫丰亭, 等. 轻量级 Web3D 地铁火灾逃生在线规划关键技术. 中国科学: 信息科学, 2019, 49: 405–421, doi: 10.1360/N112018-00275
Ai Z H, Hu Y H, Yan F T, et al. Key technology of lightweight Web3D online planning of metro fire escape (in Chinese). Sci Sin Inform, 2019, 49: 405–421, doi: 10.1360/N112018-00275

1 引言

1.1 技术研究所面临的挑战

公共场所发生火灾, 往往伴随着大量人员伤亡, 给人民群众生命财产安全造成极大威胁. 近年来发生在韩国和俄罗斯的两起特大型地铁火灾, 其造成的财产损失十分巨大, 但俄罗斯政府由于经常开展地铁逃生训练, 因此事故中地铁站内的 5000 名乘客均安全撤离, 仅有 11 人受轻伤, 而韩国政府由于对国民较缺乏安全逃生的训练, 致使地铁站内的 600 名乘客中死亡人数达到了 198 人, 受伤 146 人. 由此可见, 逃生演习是有效减少损失的预先措施, 然而, 在现实生活中, 一场大规模真实人群的火场逃亡演习, 往往会消耗大量人力物力, 并干扰人们正常生活和社会工作, 因而, 基于计算机模拟软件进行火灾路径的模拟仿真显得非常高效和必要. 目前基于现有技术开发这种 Web3D 火灾模拟平台面临着诸多挑战, 主要表现在如下 4 方面:

挑战 1. 地铁火灾的火、烟气与大规模密集人群动态情景在 Web 浏览器端的实时可视化是一个挑战, 其一直是影视特效和游戏特效的难点. 首先, 第一个难点在于火、烟气本身没有固定的形状, 传统办法是通过大量的“粒子”组成点云来模拟这两者. 火灾场景中火和烟是必不可少的, 传统粒子模拟方式通过大量“有形”模拟“无形”方法, 所产生的计算、渲染压力很大, 浏览器有限的图形运算能力在面对大量运算时, 会容易出现崩溃的情况. 另外, 大规模虚拟化身模型的数据传输和动态渲染也非常具有挑战性, 原始的单个虚拟化身具有 5736 个顶点, 11382 个三角面, 数据量为 9.42 MB, 而浏览器缓存仅有 2 GB (iPhone 等苹果手机的浏览器为 512 MB), 100 个虚拟化身的数量完全突破了移动端缓存的容量上限. 因此, 如何找到一种仅用少量粒子就能最大限度保证烟火形态的方法来平衡本火灾模拟平台的运行效率和烟火的真实感间的矛盾, 以及找到一种针对虚拟化身模型的轻量化处理方法尤为重要.

挑战 2. 如何将具有庞大数据量的 FDS^[1~3] 数据轻量化, 并应用于本实验平台的烟气运动、弥漫效果中是本文的第 2 个挑战. 为了能够让场景中烟气的运动、弥漫效果与实际烟气更加符合, 本文引入了 FDS^[4] 火灾动力学模型作为参考. FDS 是目前世界上公认的火灾动力学模型, 但是通过使用 FDS 软件 Pyrosim^[5] 计算后会产生大量的数据, 在以本文中的场景作为计算场景的情况下, 一个火灾案例发生 5 min 所产生的数据量就达到了 7 GB, 如此庞大的数据会给浏览器缓存和网络带宽造成巨大压力. 因此, 如何对 FDS 数据进行轻量化处理, 也是本文要面临的挑战.

挑战 3. 如何为虚拟化身设计一个适合的寻路算法, 并使之能够实现大规模地铁站场景下的大规模人群的火灾逃生情景, 是本文面临的第 3 个挑战. 传统的蚁群算法 (以下简称 ACO 算法)^[6] 所生成的计算结果会具有相对的盲目性, 并且在保证计算结果精确的同时也会使得用户的等待时间更长, 而且通过大量“蚂蚁”不断对场景空间进行路径探寻和计算迭代, 所形成的路径足迹又往往与真实人类的路径足迹不太一致. 因此, 如何找到一种计算量小又能够真实模拟人类逃生的寻路算法就十分重要.

挑战 4. 如何将上述提到的动态火、烟气、大规模虚拟化身与大型静态 BIM^[7~9] 地铁站场景相整合, 并实现虚拟化身的寻路效果, 以及动态烟气与 FDS 动力学数据的结合, 这是本文的最终挑战. 如上文所阐述, 动态火、烟气的粒子运算及渲染, 大规模动态虚拟化身在线传输及渲染, 两者任何一方对于 Web3D 程序的开发都是很大的挑战, 但作为基于虚拟足迹聚类的轻量级 Web3D 地铁火灾逃生路径在线规划平台, 还需要有大规模的地铁站 BIM 场景作为依托, 为了实现烟气的真实蔓延效果, 虚拟烟气必须要与 FDS 数据相结合. 为了实现虚拟化身的逃生, 就必须针对火灾场景给出逃生路径规划算法. 这一切功能的实现, 需要我们对任何一项挑战都要做到轻量化处理和计算资源、传输资源的合理

化分配。

本文主要工作有如下几个方面。首先,在静态场景数据的处理方面,针对地铁站大规模静态场景数据,通过基于语义和体素化成分检验的轻量化方法,对大规模场景数据进行了轻量化处理;第2,在动态场景数据的处理方面,针对大规模虚拟化身的在线渲染,我们通过分析虚拟化身的几何体信息和虚拟化身的动画数据,提出了数据拆分并有机组合的思想,实现了大规模虚拟化身的轻量化处理;第3,针对动态烟气数据,提出了基于烟气冗余消除和归一化的轻量化处理方法,并基于精灵纹理粒子系统,构建了轻量级烟气场景;第4,在前3步的轻量化处理基础上,有效实现了对大规模场景数据、虚拟化身和动态烟雾数据的在线实时渲染;最后在虚拟化身的智能寻路方面,基于上述一系列轻量化处理的 Web3D 地铁场景中的逃生路径规划问题,提出了基于虚拟足迹路径聚类的蚁群优化算法 eAAO (evacuation based on adaptive ant colony optimization),该算法基于 VR^[10] 设备获得的真实人群逃生路径,对路径数据进行筛选和聚类以形成骨干路径,并与 ACO 算法相结合,实现了逃生路径的最优规划。综上所述,我们最终在 Web 浏览器端,实现了基于大型地铁站中大规模人群实时在线的火灾逃生场景。

1.2 研究综述与相关工作

针对上述挑战,本文针对地铁火灾逃生路径在线规划平台^[11]的如下相关技术进行综述。

主流的火灾应急软件技术。目前,为加强公共安全建设,世界各国科研机构都在相继研究自己的火灾逃生平台。英格兰格林威治大学 (University of Greenwich) 开发了 EXODUS^[12] 和 SMARTFIRE^[12] 应急模拟系统,英国 Legion 公司开发了 Legion^[13] 系列软件系统, Crowd Dynamics 公司开发了 Myriad^[14] 软件以及美国 Massive 公司开发了 Massive^[15] 软件。此类软件较逼真地实现物理模拟效果和精准数值计算能力,但其是单机软件平台,无法进行网络交互,同时在运行时对硬件资源要求比较高。

主流的火灾计算技术。此外,以权威火灾动力学模型 FDS 为基础的火灾场景模拟软件 PyroSim 被广泛应用于各类火灾情景模拟的相关工作中,其生成的火灾模拟数据非常真实、精确,收到了相关行业机构的认可与好评,但其产生的数据量非常庞大,无法实时传输,计算时间非常慢,根本无法实现网页的实时在线渲染。

主流的火灾情景可视化引擎技术。其三,基于 Unity3D^[16], Unreal^[17], Three.js^[18] 等游戏引擎的流行,更多开发者能够方便地开发针对生产安全和公共安全的 3D 模拟软件产品^[19]。此类软件虽然实现了对 PC 或移动互联网支持,但常规运用引擎开发的火灾模拟系统,需要本地安装才能运行,无法提供高效、轻量、便捷的运行方式,而传统针对网页版的系统开发的火灾模拟系统,又常常需要针对不同平台浏览器安装相应的插件,使用范围狭窄且用户学习成本高。即使目前主流针对 HTML5 技术的引擎技术避免了插件安装,但由于图形处理能力及渲染效率不佳,因此,无法做到大规模场景下的火灾仿真模拟。

本文提出基于虚拟足迹聚类的轻量级 Web3D 地铁火灾逃生路径在线规划平台,以解决当前公共安全火灾模拟软件产品中的不足,该软件 and 当前市面模拟平台产品对比如表 1 所示。

2 技术路线

本文基于 B/S 架构实现轻量级 Web3D 地铁火灾逃生路径在线规划平台。在浏览器端,使用 Java

表 1 各火灾逃生模拟平台对比
Table 1 Comparison of fire evacuation simulation platforms

	Stand-alone system	On-line system	Web-side system	The Web-side system in this paper
Dependence	Application	Client	Plug-in	No plug-in
User support	Single user	Multi-user	Multiplayer online	Multiplayer online
Smoke simulation accuracy	Very high	High	Normal	High
Scene size	Huge	Medium	Small	Huge
Usability	Low	Medium	High	High
Portability	Low	Medium	High	High

Script 语言和面向 HTML5 [20] 的 WebGL 图形引擎 (Three.JS) 进行 Web3d 开发, 其可实现跨平台并在大多数浏览器上运行, 用户无需安装任何插件. 在服务器端, 使用 JavaEE 和 C++ 语言, 实现对路径规划、静态场景数据和动态烟气数据的可视化预处理. 其具体技术路线如图 1 所示, 其中, 服务器端软件可分为如下模块: (1) BIM 场景轻量级预处理模块; (2) 烟气数据轻量级预处理模块; (3) 大规模虚拟化身的轻量级预处理模块; (4) 基于虚拟足迹聚类的路径规划模块. Web 浏览器端软件可分为如下模块: (1) 静态地铁站的轻量级 Web3D 渲染模块; (2) 纹理粒子模拟动态烟气场景的轻量级 Web3D 渲染模块; (3) Clone 技术优化大规模虚拟化身的轻量级 Web3D 渲染模块; (4) 化身足迹采集以及路径优化模块; (5) 虚拟化身消防逃生在线交互可视化模块.

3 地铁火灾在线逃生规划关键技术

3.1 虚拟逃生足迹采集预处理

为了可以采集到虚拟火灾情况下, 测试人员逃生路径的相关信息, 就必须预先为测试人员提供一个可以进行 WebVR 模拟逃生的虚拟场景 [21, 22]. 该场景中需要包括场景模型、虚拟化身以及虚拟的烟气. WebVR 模拟逃生过程的实现方式, 是在以 Three.JS 引擎所提供的 WebVR 解决方案 [23] 为基础, 让受试人员通过佩戴 VR 头盔设备, 在网页端 3D 虚拟地铁场景中进行火灾逃生训练. 期间, 系统会将受试人员的路径信息在服务器端进行记录, 并将该路径信息用于后续的路径规划工作. 由于该场景需要运行在网页浏览器中, 因此需要对其进行轻量化处理.

3.1.1 基于构件重用的大规模地铁建筑物 BIM 在线可视化

在对整个 Web3D 场景中的 BIM 模型进行轻量化处理的过程中, 通过对建筑模型各个部件的语义信息进行分析 [24], 依据参考文献 [24] 所提供的基本思想, 我们采用自主研发的工具软件 BIM Lightweight Tool 对模型进行轻量化处理, 该工具基于 IFC Engine 开源项目编写制作, 通过对 IFC BIM 文件中的各个模型构件进行几何信息和语义信息的提取, 并在查找处重复构建后仅保留一份“个例”作为样本. 再通过变换矩阵, 依照保留下来的个例样本对其进行复制, 对经过去重轻量化处理的建筑模型进行复原, 通过对 BIM 模型进行数据冗余去除和几何数据重用的方式来实现 BIM 模型的轻量化处理, 使得模型文件的传输和渲染效率大大提高.

我们先后选择了两个具有代表性的地铁站场景作为逃生实验场地, 第 1 个大型地铁站为车公

图 1 (网络版彩图) 系统的架构设计与技术路线

Figure 1 (Color online) Overall technology roadmap

图 2 (网络版彩图) 地铁站场景截图. (a) 车公庙地铁站; (b) 同济大学地铁站

Figure 2 (Color online) Screenshots of two typical metro stations. (a) Che Kung Temple metro station; (b) Tongji University metro station

庙地铁站, 如图 2(a) 所示, 该地铁站作为大型中转型地铁站共有 4 层, 东西长 195.096 m, 南北长 112.452 m, 其模型构建数为 33444 个, 三角面片数为 26645625 个, 数据量为 456 MB. 第 2 个大型地铁站为同济大学地铁站, 如图 2(b) 所示, 该地铁站作为常规性非中转型地铁站共有 3 层, 东西长 624 m, 南北长 48 m, 其模型构建数为 1128 个, 三角面片数为 4755474 个, 数据量为 200 MB. 二者楼层的网格长度均为 0.25 m. 在实际的程序运行过程中, 两个场景的加载时间仅需 1~2 s.

图 3 (网络版彩图) 在线交互式 Web3D 大规模逃生虚拟化身的可视化. 大规模虚拟化身在线渲染 (a) 技术路线和 (b) 效果

Figure 3 (Color online) Online interactive Web3D visualization of large scale escaping virtual avatars. Large scale virtual avatars online rendering (a) technology roadmap and (b) effect

3.1.2 基于人体模型与动作重用的虚拟化身行为轻量化及在线可视化

而在地铁火灾逃生模拟中, 不仅要呈现大型地铁站场景, 也要呈现大规模虚拟化身及其动态逃生足迹. 针对此特点, 本文在研究当前动态模型基础上, 提出了基于骨骼动画模型的虚拟场景及虚拟化身轻量化处理方案, 该方案具体实现过程如图 3(a) 所示, 图 3(b) 为基于本文提出的方案实现的虚拟化身在线渲染效果.

- 步骤 1. 通过 Blender^[25] 工具软件, 在保证模型五官细节的情况下, 对模型进行多边形简化.
- 步骤 2. 构建虚拟化身数据库, 并将数据拆分为模型几何体数据与动画数据.
- 步骤 3. 在虚拟化身数据传输过程中, 依据用户当前位置的可视范围, 进行分布式传输与加载.
- 步骤 4. 将虚拟化身几何体以不同比例、不同外观材质以及不同所在位置与动画数据的不同种动画姿态、播放速度和不同运动朝向相结合, 最终构建出大规模差异化的虚拟化身集群, 并通过 Clone 技术实现优化渲染.

图 3(a) 清晰地阐述了针对大规模虚拟化身集群渲染整体的技术路线, 从图 3(b) 可以看到大规模虚拟化身在线渲染的实际效果.

3.1.3 虚拟烟气场景的轻量级可视化

为能够更加真实地表现烟气场景的动态性, 本文根据真实场景的特定空间几何数据, 采用动态烟气生成工具 FDS 建立 FDS 建筑场景, 并将火源置于易发生火灾的位置^[26~28], 进而模拟不同场景下

图 4 (网络版彩图) 重型 FDS 烟气数据的轻量化预处理. (a) FDS 数据轻量化技术路线; (b) 轻量级烟气渲染全局与局部效果图

Figure 4 (Color online) Lightweight preprocessing of huge FDS data. (a) Technical roadmap of lightweighting FDS data; (b) visual effects of lightweight smoke rendering

不同持续时间的动态烟气场景. FDS 工具虽然可以逼真模拟地铁站发生时的火灾情景, 但其计算产生的烟气数据规模庞大. 因此, 本文提出了动态烟气场景轻量化及在线渲染方案, 其处理过程如图 4(a) 所示, 图 4(b) 为基于本文提出的方案实现的动态烟气在线渲染效果.

步骤 1. 通过对语义上的零密度体素进行过滤来消除 FDS 数据的冗余.

步骤 2. 通过时间线性化采样来减少 FDS 烟气场的帧数据.

步骤 3. 将烟气密度的所有浮点数据统一在 $[0, 255]$ 的透明度区间内.

步骤 4. 将整数烟气数据转换为像素灰度的二进制数据.

步骤 5. 通过 Internet 将二进制数据传输到 Web 浏览器.

步骤 6. 通过精灵 (sprite) 纹理粒子^[29] 构建出轻量级烟气.

在动态烟气场景中, 只有非零密度的烟气数据才对 Web3D 可视化有意义, 因此, 过滤所有零密度烟气量数据可以减少烟气数据量. 本文提出通过加大线性化采样时间间隔, 以去除相同或密度接近的帧数据. 此外, 对应于像素灰度级 (0~255), 本文通过归一化公式将烟气密度浮点数据转换为 $2^8 = 256$ 级透明度的整数数据, 如图 5 所示, 并将这 256 级整数烟气密度体积数据转换成二进制数据进一步减少数据量, 以便更容易地传输到 Web 浏览器.

透明度由以下公式计算:

$$\text{Opacity} = \frac{2^8(\text{SD}_c - \text{SD}_{\min})}{\text{SD}_{\max} - \text{SD}_{\min}}, \quad (1)$$

其中 SD_c 是某点的烟气密度, 而 $\text{SD}_{\min}$ 和 $\text{SD}_{\max}$ 分别是 SD_c 的最小值和最大值.

此外, 在上述的实现过程中, 针对虚拟烟气的效果, 本文采用基于精灵纹理粒子的虚拟烟气制作方案. 使得粒子纹理管理更加统一, 大大提高了程序开发和程序运行的效率. 具体来说, 本文将 50 个

图 5 (网络版彩图) 将烟气密度归一化为 2^8 个透明度级别Figure 5 (Color online) Normalization of smoke density at the 2^8 levels of transparency

图 6 (网络版彩图) 虚拟足迹聚类分析与逃生路径规划技术路线

Figure 6 (Color online) Virtual traces clustering and firing evacuation path planning

Sprite 纹理粒子作为一个烟气管理单元, 即烟团, 每个精灵都绑定有从互联网获得的烟气纹理贴图素材. 虚拟场景中整齐排列多个这样的群集, 并填满整个场景. 在每个群集中, 50 个 Sprite 粒子在程序范围内执行随机圆周运动. 由于 Sprite 粒子系统带有“广告牌”和“自适应纹理尺寸”功能, 本文通过将 FDS 烟气数据的密度值进行归一化, 并使其与精灵粒子的纹理透明度相映射. 最终更真实有效地展示出场景中“充满烟气”的效果.

3.2 基于聚类路线跟踪的在线逃生路径规划

在本文提出的火灾逃生路径在线规划平台中, 测试人员通过佩戴 VR 设备进入 WebVR 火场, 并在多次路径规划中获得了逃生路径数据^[30~32], 该数据可为路径规划提供指导^[33~35]. 实验表明, 将 ACO 算法与测试人员逃生路径相结合时, 该算法即 eACO (evacuation based on adaptive ant colony optimization) 算法可比 ACO 算法更快地找到最优路径. 本文以“虚拟足迹”表示测试人员逃离虚拟火场时的有价值路径数据, 并基于有价值的历史足迹, 进而获取这些足迹集群. 在每个足迹群集中, 都可从群集中选择出其中的骨干足迹, 即骨干路径, 所有有价值的历史足迹和骨干路径都将是 eACO 算法路径规划的启发式路径^[36]. 本文提出的基于虚拟足迹路径聚类的蚁群优化算法 eACO 实现过程如图 6 所示.

3.2.1 在线虚拟路径采集方法

本文提出的火灾逃生路径在线规划平台中的虚拟足迹采集, 通常是在大型地铁站场景中进行, 其逃生模拟效果如图 7 所示. 本文分别以上文中所提到的车公庙地铁站和同济大学地铁站作为虚拟逃

图 7 (网络版彩图) 通过 WebVR 设备在线采集虚拟逃生路径

Figure 7 (Color online) Online collecting virtual traces of passenger evacuation paths with WebVR devices

图 8 (网络版彩图) 记录的逃生路径足迹

Figure 8 Collected virtual traces of fire evacuation

图 9 (网络版彩图) (a) 有效路径; (b) 逃生失败的无效路径; (c) 超时路径; (d) 出现冗余的无效路径

Figure 9 (Color online) (a) Valid path; (b) invalid path due to failure of escaping; (c) invalid path due to timeout; (d) invalid path due to redundancy

生实验场地: 首先将这两个地铁场景中测试人员的逃生足迹数据记录在 WebVR 端, 然后传输到服务器端, 图 8 为记录的逃生路径足迹示意图。

针对图 8 中的逃生路径足迹, 可根据如下情况对路径信息进行预处理:

(1) **有效路径.** 由于该种路径可以在制定的时间阈值内到达安全出口, 对火灾逃生起到重要的参考作用. 如图 9(a) 所示, 因此, 本文将对指定时间阈值内的有效逃生足迹路径进行保留。

(2) **逃生失败的无效路径.** 由于该种路径无法到达安全出口, 对火灾逃生没有参考价值. 如图 9(b) 所示, 因此, 本文对最终不能到达出口的逃生足迹路径进行删除。

(3) **超时路径.** 由于该种路径即使可以到达安全出口, 但是所用时间超出了指定的时间阈值, 进而使得逃生几率降低. 如图 9(c) 所示, 因此, 本文将未能在指定时间阈值内到达出口的足迹路径进行删除。

(4) **出现冗余的无效路径.** 由于该路径在制定的时间阈值内到达安全出口, 即使其中包括一些回路和冗余路径, 但依然具有参考价值. 如图 9(d) 所示. 因此, 本文针对虽然在指定时间阈值内到达出口, 但是其中包含有冗余路径或环路的足迹路径, 进行了环路去除和冗余路径平滑处理。

3.2.2 逃生骨干路径的抽取

经过上述对虚拟足迹进行采集, 并对路径信息进行筛选后, 剩下的人员逃生路径将被聚类, 其聚类算法过程如下:

经过算法 1 处理后, 每个集群都将与一个称为该集群骨干的代表性路径相关联. 在具体实现过程

图 10 (网络版彩图) 骨干路径提取

Figure 10 (Color online) Backbone path extraction

图 11 (网络版彩图) 蚂蚁依据场景中的烟气数据进行路径规划

Figure 11 (Color online) Ant planning evacuation path in terms of smoking scenario

算法 1 虚拟逃生足迹的骨干路径抽取

输入: 一组在线地铁逃生足迹 $P = \{P_A, P_B, \dots, P_i, \dots, P_n\}$;

输出: 从集群逃生足迹中提取的骨干路径.

步骤 1: 检查 P 的所有逃生足迹的有效性;

步骤 2: 删除无法逃离 P 的无效足迹;

步骤 3: 通过删除循环和多余分支来平滑有效足迹;

步骤 4: 从 P 中选择空间最短或时间最快的一条迹线 $P_A = \{A_0, A_1, A_2, \dots, A_i, \dots, A_m\}$;

步骤 5: 计算 P_A 与 P 中其余足迹之间的平均距离, 假设 $P_B = \{B_0, B_1, B_2, \dots, B_j, \dots, B_l\}$ 是另一条足迹, 则两者之间的平均距离定义为

$$\overline{\text{Dist}_{AB}} = \begin{cases} \frac{1}{m} \sum_{k=1}^m |A_k - B_k|, & \text{if } m < l, \\ \frac{1}{l} \sum_{k=1}^l |A_k - B_k|, & \text{if } l < m; \end{cases} \quad (2)$$

步骤 6: 针对 P 中的每对 n 路径重复上述步骤 5, 然后存储 n 路径中的 $n \times n$ 矩阵. 然后设置经验阈值 τ , 并且使用著名的 Lloyd 算法将 n 路径划分为基于 τ 的簇;

步骤 7: 设 $C = \{P_0(u), P_1(u), P_2(u), \dots, P_j(u), \dots, P_m(u)\}$ 为包含 m 个路径 $P_0 \rightarrow P_m$ 的簇, $P_j(u)$ 是参数曲线, 其中参数 u 是曲线的归一化弧长.

最终, C 的骨干路径就是另一条参数曲线 $P_{\text{backbone}}(u) = \frac{1}{m} \sum_{i=0}^{i=m} P_i(u)$.

中, 本文采用了算术平均方法. 经过几次 WebVR 逃生路径规划之后, 每个人都会对这些足迹进行评估, 然后最佳逃生足迹将被保留. 如图 10 描绘了其余集群的骨干路径提取, 在本例中, 该簇具有 5 条路径 $P_0 - P_4$, 且提取的骨干路径为加粗红色曲线.

3.2.3 基于骨干路径的 eAACO 逃生路径规划算法

在完成骨干路径提取后, 本文提出将骨干路径信息与 ACO 算法相结合的启发式蚁群优化算法——eAACO 算法. eAACO 算法将轻量级地铁车站场景地图和轻量级烟气障碍地图作为输入, 并为现场每个虚拟化身规划动态逃生路线. 具体来说, 基于动态烟气数据, eAACO 算法为每个虚拟化身每 0.7 s 规划一次最优逃生路径.

本文提出的 eAACO 算法, 其核心思想是如何确定蚂蚁下一步的运动方位. 本文选用 1.6 m 高度的烟气数据进行二维地形图的绘制, 并对这个地图上的任何网格进行相应区域的烟气密度检测, 如果烟气密度大于 32 mg/m^3 [37,38], 其将被标记为障碍物, 否则, 其被认为是可访问的网格. 如图 11 所示, 蚂蚁会依照寻路算法, 在寻路网格上对四周 8 个方向的网格进行寻路, 直至到达安全出口为止.

表 2 两种寻路算法的时间消耗对比 (s)

Table 2 Comparison of time consumption between two algorithms (s)

	100 people	200 people	300 people	400 people	500 people
Traditional ACO algorithm	273.722	467.396	707.625	985.688	1261.27
The eAAO algorithm in this paper	68.875	65.655	69.5	70.21	70.5

该算法的主要过程如算法 2 所示.

算法 2 eAAO 地铁站逃生路径规划算法

输入: 骨干路径和历史聚类逃生足迹;

输出: eAAO 逃生路径.

步骤 1: 获得 Grid_{ij} 的烟气密度 ($\text{sd}_{G(ij)}$), 初始化一些常数和变量作为 eAAO 路径寻找的输入数据;

步骤 2: 计算距离启发式 $\eta_{ij} = \frac{1}{d(G(i,j), \text{Exit})}$ 以选择下一个相邻网格; 计算 $\Delta\tau_{G(ij)}$ 作为每次迭代的信息素增量:

$$\Delta\tau_{G(ij)} = \begin{cases} \sum_{k=1}^m \Delta\tau_{G(ij)}^k, & \text{if Grid}(i,j) \text{ has no smoking neighbors,} \\ \sum_{k=1}^m \frac{\Delta\tau_{G(ij)}^k}{2}, & \text{otherwise;} \end{cases} \quad (3)$$

步骤 3: 依据阈值 τ 更新骨干路径 P 的所有网格的信息素:

$$\tau_{G(ij)}(t) = \rho \times \tau_{G(ij)}(t-1) + \Delta\tau_{G(ij)}; \quad (4)$$

步骤 4: 若当前位置 $L_n = \text{exit grids}$, 重复以下步骤直至蚂蚁到达安全出口:

(1) 计算 $\varphi_{G(ij)} = \eta_{G(ij)} + \text{sd}_{G(ij)}$;

//每隔 Δt 时间, 从 Grid_{ij} 得到烟气数据, 形成相应的密度数据.

//将该值转换为倒数, 并记录为该项的启发式值.

(2) 计算 $\tau_{G(ij)}^{vb}(t) = \text{vb}_{G(ij)} + \tau_{G(ij)}(t)$; //将骨干路径的值与最后一次迭代增益值相加.

(3) 计算 8 个邻居点的概率:

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{G(ij)}^{vb}(t) + \varphi_{G(ij)}}{\sum_{s \in \text{allowed}_k} (\tau_{is}(t) + \varphi_{is})}, & j \in \text{allowed}_k; \\ 0, & \text{otherwise;} \end{cases} \quad (5)$$

(4) 选择一个未被占用且未被访问的 L_n 的邻居节点, 并将其记录为 L_{n+1} ;

(5) $L_n = L_{n+1}$;

步骤 5: 在上面的步骤 4 中让所有蚂蚁按照从 (1) 到 (5) 的步骤不断重复, 直到 eAAO 收敛;

步骤 6: 从上述第一次 eAAO 迭代中产生的路径里选择出最佳路径;

步骤 7: 让各只蚂蚁为各自路径上爬行过的路径释放相同数量的信息素, 并将最优路径上的 τ 值更新为原来的 1.2 倍;

//对 τ 值的更新, 可以使得下次进行迭代计算时, 路径数据更具导航的参考性.

步骤 8: 所有蚂蚁继续进行路径迭代, 直到最优路径结果趋于稳定, 并返回以 L 为开始位置的最佳路径.

我们的实验平台采用 CPU: Xeon E5 2620 v3 2.4 GHz 内存: 6 GB 的云端服务器, 分别使用 ACO 算法和与骨干路径信息相结合的 eAAO 算法, 对虚拟化身数量为 100, 200, 300, 400 和 500 的情况进行了寻路计算, 并将计算结果生成的所需时间进行了记录 (如表 2 所示), 由表 2 中结果可知, 采用 eAAO 算法可以大大降低因虚拟化身增加所产生的寻路计算压力, 提高了系统的运行效率, 降低了用户的等待时间.

通过上述计算数据可以得出, 将骨干路径信息作为运算参考依据并与 ACO 算法相结合, 所生成的 eAAO 寻路算法在保证 ACO 算法精确性的同时, 降低了运算时间与运算规模. 所生成的最佳路径, 使得虚拟化身能够在指定时间阈值内, 以最短的路径长度到达安全出口位置. 对真实的火灾逃生情况起到了良好的参考作用.

表 3 测试环境硬件配置
Table 3 Hardware configuration of test environment

	PC testing environment	iOS testing environment	Android testing environment
CPU	i7-7700HQ	Apple A10	Snapdragon 820 (MSM8996)
Memory	16 GB	2 GB	4 GB
GPU	Nvidia GTX M1070	PowerVR GT7600	Adreno530
OS	Windows 10 64 Bit	iOS 11.4	Android 6.0
Network	4G wireless network	4G wireless network	4G wireless network

表 4 内存占用情况 (MB)
Table 4 Occupied memory (MB)

	100 people	200 people	300 people	400 people	500 people
PC	293	354	401	411	412
iOS	221	299	282	284	314
Android	322	339	346	358	372

表 5 渲染帧率情况 (Hz)
Table 5 Rendering frame rate or frequency (Hz)

	100 people	200 people	300 people	400 people	500 people
PC	86	65	52	45	35
iOS	37	25	22	18	15
Android	26	26	23	21	18

4 实验结果

本火灾逃生路径规划平台基于 B/S 模式, 所提出的整套解决方案的实施都基于表 3 所示的服务器和 Web 终端环境.

本文提出了用于评估的 3 个指标: (1) 场景加载时的内存占用情况与渲染帧数; (2) 加载烟气数据后的渲染帧数; (3) 多人火灾情境下逃生时的渲染帧数.

4.1 内存加载占用情况与渲染帧数指标实验

本文将针对大规模 BIM 场景模型和虚拟化身的轻量化效果进行测试, 测试的平台为 PC 服务器和 Web 移动端, 测试的项目为在程序开启寻路功能后的内存的占用和渲染帧率.

首先, 我们分别在两个平台上一次加入 100, 200, 300, 400 和 500 个虚拟化身, 并记录下两个平台的内存占用情况, 实验数据如表 4 所示, 测试中用的浏览器均采用针对各自操作系统的 Chrome 浏览器最新版, 从测试结果可以看出, 在 100 到 500 虚拟化身区间内, 所占内存都在 512 MB 以内, 并且在 iOS 移动端平台上运行时, Three.JS 也对内存做了相应的优化.

渲染帧率的相关数据如表 5 所示, 在原计划要达到的 100 个虚拟化身的情况下, PC 端可以轻而易举地达到 86 帧, iOS 端可以达到 37 帧, 达到了日常要求的 60 帧和 30 帧标准. 即使在 500 个虚拟化身的情况下, 帧数也分别达到了 35 帧和 15 帧.

图 12 (网络版彩图) 基于 Web 浏览器的实时渲染烟气场景

Figure 12 (Color online) Real time rendering of smoke scene based on Web browser. (a) $t_0 = 60$ s (78 FPS); (b) $t_1 = 120$ s (77 FPS); (c) $t_2 = 180$ s (76 FPS); (d) $t_3 = 240$ s (78 FPS); (e) $t_4 = 300$ s (78 FPS); (f) $t_5 = 360$ s (77 FPS); (g) $t_6 = 420$ s (79 FPS); (h) $t_7 = 480$ s (78 FPS)

表 6 FDS 数据轻量化处理前后对比

Table 6 Comparison of FDS data before and after lightweight processing

	FDS data 1	FDS data 2
Raw FDS data volume (MB)	85.3	229.2
Lightweight FDS data volume (MB)	3.26	7.76

4.2 加载烟气数据后的渲染帧数指标实验

本实验通过精灵纹理粒子在大型地铁站场景中构建出火灾发生时烟气弥漫的效果. 通过使用 FDS 工具, 我们获得了一组针对本地地铁站场景的相应烟气数据. 通过对该烟气数据的轻量化和归一化处理, 最终我们得到了所需的轻量化烟气数据, 通过将这些轻量化烟气数据与粒子纹理的透明度相对应, 最终在地铁站场景中实现了烟气弥漫的效果. 轻量级烟气在本实验中要达到两个目的: (1) Web3D 场景上的可视化; (2) 在路径规划中作为障碍物.

在上述轻量化工作之后, 轻量级烟气数据量已经大大减少 (例如, 在我们的实验中, 只有原始重量级数据的 1/30), 并且可以充分确保可视化效果.

如图 12(a)~(h) 所示, 将场景中虚拟化身数量设定为 100 人的情况下, 自火灾发生的 8 min 时间内, 页面刷新率始终保持在 70 FPS (frames per second) 以上. 随着时间的推移, 烟气逐渐浓烈, 但页面刷新率始终变化不大, 依旧维持在 75 FPS 左右. 由此可见轻量级烟气在效果演示方面的稳定性, 同时由表 6 中通过对于原始 FDS 数据的数据量与轻量化之后的数据量相对比, 也可以体现轻量化烟气的轻量化性.

4.3 多人火灾情境下逃生渲染帧数指标实验

为了测试 eAACO 算法的运算效率以及对渲染帧数的影响, 我们设定了以下实验案例, 在带有烟气的地铁站场景下, 我们设定烟气会从地铁站一侧的底层向整个地铁站蔓延, 并且场景中有 300 名虚拟化身正在通过 eAACO 算法按照动态优化规划的路线进行逃生, 在常规 3D 图形程序的运行过程中, 如果运行帧率在 30 FPS 以上, 即可确定为流畅运行. 如图 13(a)~(f) 所示, 在整个程序的运行过程中全景渲染速率都在 41~58 FPS 的范围内.

由于在路径规划过程中, 使用重复信息素的多线程算法提高了路径规划性能, 并在大规模虚拟化

图 13 (网络版彩图) 在 eAACO 算法驱动下的地铁站中 300 个虚拟化身的可视化逃生过程与帧率

Figure 13 (Color online) Performance data on online fire escaping of 300 virtual passengers using our proposed eAACO algorithm. (a) $t_0 = 0$ s (41 FPS); (b) $t_1 = 10$ s (43 FPS); (c) $t_2 = 20$ s (46 FPS); (d) $t_3 = 30$ s (50 FPS); (e) $t_4 = 40$ s (54 FPS); (f) $t_5 = 50$ s (58 FPS)

身的渲染过程中, 使用 Clone 渲染方式的多线程算法提高了渲染性能, 因此, 火灾发生时, 页面刷新率保持在 40 FPS 以上, 随着虚拟化身模拟的乘客安全撤离火灾现场, 最终页面刷新率为 58 FPS.

我们目前已将火灾逃生模拟平台项目工程上传至互联网¹⁾, 欢迎大家进行体验.

5 结论

针对地铁车站火灾逃生在线 VR 路径规划平台, 特别是为解决大型地铁站计算机模拟火灾逃生数据量庞大的瓶颈问题, 在虚拟 Web3D 技术的基础上, 我们提出了一套轻量化解决方案. 其中, 我们为大型地铁站 BIM 静态场景、虚拟化身模型数据以及动态 FDS 烟气数据提供了轻量级解决方案. 首先, 对 BIM 格式的大规模地铁站模型进行轻量级的预处理, 使数据量减少了大约 10 倍 (在我们的实验中). 对于虚拟化身模型的轻量级处理, 通过将虚拟化身的几何数据进行简化, 并与动画数据进行分组管理, 再在浏览器端进行灵活组合, 使得模型文件数据量缩小为原来的 1/10. 然后, 在烟气数据进行冗余数据去除和数据标准化、轻量化之后, 烟气数据量减少 200 倍 (再次在我们的实验中). 通过这种方式, 所有的场景数据和烟气数据现在都可以通过互联网传输并实时上传到网页上进行渲染. 最后, 利用本文提出的 eAACO 算法和多线程机制, 可以在 3 s 内完成大型地铁站的逃生路径规划. 在上述 3 项关键技术的基础上, 开发了基于 WebGL 的原型平台, 用于 Web 上的虚拟火灾逃生路径规划, 任何人都可以参与到虚拟现实的在线消防逃生路径规划中, 而无需下载和安装插件. 通过我们的实验证实, eAACO 算法在网络上的刷新率能够满足公众对虚拟网络路径规划的需求.

尽管在现阶段取得了上述成绩, 但是目前在研究方面仍有很多不足, 首先在 WebVR 技术的实现方面, 由于 Three.JS 在所提供的相关接口技术方面还存在很多不太完善的地方, 使得目前程序的稳定性欠佳, 这也使得采集人员逃生的数据效果往往不理想. 因此, 对于 WebVR 的人员模拟逃生的功能, 目前还处在比较初级的原型阶段. 另外, 针对大规模虚拟化身的人数, 依然无法达到真实客运高峰的 3000 人, 截至目前, 我们所能达到的实际极限人数仅有 1000 人. 针对这两个问题, 我们将会在未来继续开展针对 WebVR 真实人员路径数据采集方面和虚拟化身模型轻量化方面的工作, 提高 WebVR 相关软件的稳定性和地铁站大规模人员逃生情景的真实性.

1) <http://smart3d.tongji.edu.cn/MixChinaVR/FESDemo.html>.

参考文献

- 1 Thunderhead Engineering Consultants. FDS. 2016/2/29. <https://www.thunderheadeng.com/pyrosim/>
- 2 Thunderhead Engineering Consultants. FDS. 2017/4/18. <https://www.thunderheadeng.com/pyrosim/>
- 3 Benkoussas B, Djedjig R, Vauquelin O. Numerical assessment of conventional regulation effectiveness for smoke removal from a two level underground station. *J Fundam Appl Sci*, 2016, 8: 401–425
- 4 Cha M, Han S, Lee J, et al. A virtual reality based fire training simulator integrated with fire dynamics data. *Fire Saf J*, 2012, 50: 12–24
- 5 Thunderhead Engineering Consultants. FDS. 2018/11/7. <https://www.thunderheadeng.com/pyrosim/>
- 6 Chiang C W, Huang Y Q, Wang W Y. Ant colony optimization with parameter adaptation for multimode resource-constrained project scheduling. *J Intell Fuzzy Syst*, 2008, 19: 345–358
- 7 WIKIPEDIA. BIM. 2017/4/19. https://en.wikipedia.org/wiki/Industry_Foundation_Classes
- 8 Lin Y H, Liu Y S, Gao G, et al. The IFC-based path planning for 3D indoor spaces. *Adv Eng Inf*, 2013, 27: 189–205
- 9 WIKIPEDIA. IFC. (2017/4/19). https://en.wikipedia.org/wiki/Industry_Foundation_Classes
- 10 Xu Z, Lu X Z, Guan H, et al. A virtual reality based fire training simulator with smoke hazard assessment capacity. *Adv Eng Softw*, 2014, 68: 1–8
- 11 Sharakhov N, Nicholas P, Sforza P M. SpeedSpy: a mobile Web3D platform for visualizing broadband data. In: *Proceedings of the 18th International Conference on 3D Web Technology*, 2013. 208
- 12 University of Greenwich. EXODUS/SMARTFIRE. 2018. <http://www.fseg.gre.ac.uk/index.html>
- 13 Copyright Bentley Systems. Legion. 2018. <http://www.legion.com>
- 14 Crowd Dynamics. Myriad. 2017. <http://www.crowddynamics.com/>
- 15 Massive Software. Massive. 2017. <http://www.massivesoftware.com/>
- 16 Unity Technologies. Unity3D Engine. 2018. <https://unity.com/>
- 17 Epic Games. Unreal Engine. 2018. <https://www.unrealengine.com/>
- 18 Mr.Doob. Three.js. 2018. <https://threejs.org/>
- 19 Chiu Y P, Shiao Y C. Study on the application of unity software in emergency evacuation simulation for elder. *Artif Life Robot*, 2016, 21: 232–238
- 20 W3C. HTML5 Introduction. 2017/4/19. https://www.w3schools.com/html/html5_intro.asp
- 21 Diez H V, Garcia S, Mujika A, et al. Virtual training of fire wardens through immersive 3D environments. In: *Proceedings of the 21st International Conference on Web3D Technology*, 2016. 43–50
- 22 Kinatder M, Ronchi E, Nilsson D, et al. Virtual reality for fire evacuation research. In: *Proceeding of Federated Conference on Computer Science and Information Systems*, 2014. 313–321
- 23 Spec Editor's Draft. WebVR. 2018/2/10. <https://webvr.info/samples/>
- 24 Liu X J, Xie N, Tang K, et al. Lightweighting for Web3D visualization of large-scale BIM scenes in real-time. *Graph Model*, 2016, 88: 40–56
- 25 Blender.org. Blender. 2018. <https://www.blender.org/>
- 26 Benkoussas B, Bouhdjar A, Vauquelin O. Numerical security assessment in case of fire in underground transport spaces. *Nat Technol*, 2015, 12: 45–59
- 27 Song Y Q, Gong J H, Li Y, et al. Crowd evacuation simulation for bioterrorism in micro-spatial environments based on virtual geographic environments. *Saf Sci*, 2013, 53: 105–113
- 28 Kompenhans J, Raffel M, Dieterle L, et al. Particle image velocimetry in aerodynamics: technology and applications in wind tunnels. *J Vis*, 2000, 2: 229–244
- 29 Mr.Doob. Sprite Smog Modeling and Rendering. 2018/2/10. <https://threejs.org/docs/index.html#api/objects/Sprite>
- 30 Chen W, Lu J H, Kong D K, et al. GameLifeVis: visual analysis of behavior evolutions in multiplayer online games. *J Vis*, 2017, 20: 651–665
- 31 Cierpka C, Kähler C J. Particle imaging techniques for volumetric three-component (3D3C) velocity measurements in microfluidics. *J Vis*, 2012, 15: 1–31
- 32 Duo Q, Shen H Z, Zhao J D, et al. Conformity behavior during a fire disaster. *Soc Behav Personal*, 2016, 44: 313–324
- 33 Xie C, Ma G X, Li Q, et al. Visual exploration of tsunami evacuation planning. *J Vis*, 2016, 19: 475–487
- 34 Lu D Y. Information-theoretic exploration for texture-based visualization. *J Vis*, 2017, 20: 393–404
- 35 Tian Y, Zhou T S, Yao Q, et al. Use of an agent-based simulation model to evaluate a mobile-based system for

- supporting emergency evacuation decision making. *J Med Syst*, 2014, 38: 149
- 36 Gogna A, Tayal A. Metaheuristics: review and application. *J Exp Theor Artif Intell*, 2013, 25: 503–526
- 37 Inthavong K, Tao Y, Petersen P, et al. A smoke visualisation technique for wake flow from a moving human manikin. *J Vis*, 2017, 20: 125–137
- 38 Guest J, Eaglin T, Subramanian K, et al. Interactive analysis and visualization of situationally aware building evacuations. *Inf Vis*, 2015, 14: 204–222

Key technology of lightweight Web3D online planning of metro fire escape

Zihao AI^{1,2}, Yonghao HU³, Fengting YAN³, Huijuan ZHANG³, Dongqing WANG³, Shenglan QING³, Hehua ZHU³ & Jinyuan JIA^{3*}

1. *School of Virtual Reality, Jilin Animation Institute, Changchun 130012, China;*

2. *School of Computer Science and Technology, Changchun University of Science and Technology, Changchun 130022, China;*

3. *School of Software Engineering, Tongji University, Shanghai 201804, China*

* Corresponding author. E-mail: jyjia@tongji.edu.cn

Abstract Virtual training of metro fire evacuation definitely becomes considerably more efficient and convenient immigration of trainees from the PC desktop to mobile web browsers. However, web browsers face extreme difficulty in sustaining online visualization of large-scale building information modeling (BIM) data on metro, FDS smoking data, crowding passengers, and planning of fire evacuation path simultaneously due to the weak rendering capability and limited networking bandwidth. Herein, a novel lightweight Web3D technology roadmap is proposed to address these problems. First, the lightweight preprocessing of metro BIM data is performed by semantic and geometric retrieval of repetitive entities. Second, crowding passengers are visualized online by reusing several avatars and behavior randomly in terms of textures and scales in a lightweight manner. Third, substantial smoking data are reduced by redundancy elimination and voxel normalization and rendered online with spirit texture particle. Finally, the backbone escape paths are extracted by clustering the virtual evacuation traces that are collected by virtual reality devices, such as HTC VIVE, as initial heuristics, and a lightweight metro fire evacuation planning evacuation based on adaptive ant colony optimization is realized on mobile web browsers. Numerous tests have been conducted online, and experimental results show that the proposed method can be loaded remotely in a few seconds and visualized on mobile web browsers smoothly.

Keywords Web3D, metro BIM, lightweight smoking visualization, lightweight crowding visualization, virtual traces clustering, online planning of fire evacuation path, ant colony optimization

Zihao AI was born in Changchun, Jilin Province, China in May, 1989. He is currently a joint master candidate of Jilin Animation Institute and Changchun University of Science and Technology. His research interests include lightweight Web3D game technology and online visualization of fire evacuation planning.

Yonghao HU was born in Zhangjiajie, Hunan Province, China in December, 1995. He is currently a master's candidate in School of Software Engineering, Tongji University. His research interests include lightweight WebBIM and WebGIS.

Fengting YAN was born in Rizhao City, Shandong Province, China in January, 1980. He is a lecturer at the Shanghai University of Engineering Science and obtained his Ph.D. degree from School of Software Engineering, Tongji University. His research interests include WebVR and intelligent escape planning.

Jinyuan JIA was born in Leling, Shandong Province, China in November, 1963. He is currently a professor at School of Software Engineering, Tongji University. His research interests include lightweight Web3D and WebVR/WebGame/WebBIM.