

基于 GPU 的大规模人群疏散模拟 *

李 攀^{a,b}, 彭 伟^{a,b}

(四川大学 a. 计算机学院; b. 视觉合成图形图像技术国防重点学科实验室, 成都 610064)

摘 要: 基于中央处理器(CPU)串行的人群疏散传统方法对于人群规模较少的场景,可以得到良好的疏散模拟效果,但在人群密度较高的场景中,难以达到实时模拟的要求。为了克服上述问题,实现了一种基于图形处理器(GPU)的人群疏散模拟的方法。该方法通过对个体寻径算法的优化,不仅能使个体快速准确地智能寻径,而且将个体寻径独立性与图形处理器高性能计算特性进行结合,充分利用了图形处理器强大的并行计算能力,从而大幅度提高了人群疏散模拟的人群规模,使人群疏散模拟的实时性得到增强。

关键词: 人群疏散模拟; 图形处理器; 个体寻路算法; 并行化仿真

中图分类号: TP391.9 **文献标志码:** A **文章编号:** 1001-3695(2012)03-1166-03

doi:10.3969/j.issn.1001-3695.2012.03.101

Large scale crowd evacuation simulation based on GPU

LI Pan^{a,b}, PENG Wei^{a,b}

(a. School of Computer Science, b. State Key Laboratory of Fundamental Science on Synthetic Vision, Sichuan University, Chengdu 610064, China)

Abstract: Traditional crowd evacuation serial method which is based on the central processor unit can get a good crowd evacuation simulation results for scene with a small scale population. However, it is difficult to achieve real-time simulation requirements in crowd scene. To solve the problem, this paper proposed a crowd evacuation simulation method based on GPU. This method not only provided individual quickly and accurately intelligent pathfinding, but also combined the independence of individual pathfinding with high-performance computing of GPU to make full use of powerful parallel computing capability of GPU. Therefore, it could greatly improve the population scale and real-time feature of the crowd evacuation simulation.

Key words: crowd evacuation; GPU; individual pathfinding algorithm; parallel simulation

0 引言

如今,车站、机场、体育馆等大型公共场所是人群高度集中的场所,也是公共安全事件较易发生、易造成大规模人员伤亡、防控与应急处置较为困难的场所。制定突发事件发生处置预案,保证大型公共设施内人员和财产的安全,成为大型公共场所的核心目标。

实践证明,很多突发事件是不可避免的,只能通过一些实际的做法尽量降低其发生的可能性。比如,完善建筑物防灾设计方案,开展灾害发生时人员疏散演练。虽然实景人员疏散演练针对性强,但由于经费、人员规模、场地限制,更是出于人员安全考虑,无法完全根据相关专家制定的人群疏导方案演练疏导过程,这就大大降低了疏散方案的可靠性和实用性。

正是为了应对这种情况,国内外研究人员将虚拟现实、人工智能等领域相关方法应用到大型公共场所安全领域,从而有效地模拟在火灾、爆炸等突发事件发生时高密度人群的疏导。通过计算机仿真模拟人员疏散可以使研究员快速、方便地模拟各种场景中的人员疏散,根据实验结果评价疏散方案的可行性,评估建筑物布局是否有利于人员疏散。此外,仿真模拟还在费用 and 安全性方面有着显著的优势。近年来,人员疏散仿真软件也在我国得到初步应用,北京奥运会体育馆、上海世博会

场馆、上海环球金融中心等建筑均进行了相关评估,这已经证明了其实用价值。但是,当前国内使用的人员疏散仿真软件仍然存在许多不足,例如:

a) 疏散人员往往缺乏智能寻路行为,不能根据当前建筑物环境选择最优逃生路线。

b) 在仿真区域内,当参与人员人数较多时,不能很好地满足计算需要。

这些缺陷很多时候会使人员疏散评估结果产生偏差,甚至会导致严重错误。

考虑到个体寻径的独立性和图形处理器强大的并行计算能力,本文将两种方法结合,进而实现一种针对大规模人群疏散模拟的实时绘制方法,并克服了以前方法的弊端。与基于 CPU 方式的实现结果进行比较分析,该方案可获得 30 倍左右的加速比。

1 相关理论背景

1.1 GPU 通用计算概述

CUDA 是 NVIDIA 公司推出的图形处理器通用计算平台,它能够有效利用图形处理器强劲的处理能力和巨大的存储带宽进行图形渲染以外的计算。这个架构平台将中央处理器作

收稿日期: 2011-07-07; **修回日期:** 2011-08-23 **基金项目:** 国家自然科学基金资助项目(60903118,60832011)

作者简介: 李攀(1988-),男,四川雅安人,硕士研究生,主要研究方向为 GPU 高性能计算、计算机图形学、虚拟现实(wisesoft2011@gmail.com); 彭伟(1987-),男,四川简阳人,硕士研究生,主要研究方向为计算机图形学、虚拟现实。

为主机端,图形处理器作为协处理器,并由中央处理器和图形处理器协同工作。由中央处理器负责进行逻辑性强的事务处理和串行计算,图形处理器负责处理高度线程化的并行处理任务,两者各自负责完成自己所擅长的部分,同时又弥补了自身的不足。目前其已经广泛应用于图像处理、视频播放、人工智能、模式识别、密码加密、生物计算等领域,并在应用中使中央处理器获得了一到两个数量级的加速。理论上,一块高端图形处理器的单精度浮点处理能力达到甚至超过了同时期中央处理器性能的10倍。在提供相同计算能力的条件下,基于GPU的系统所需的成本与功耗均小于基于CPU的系统。

之前许多研究人员已经提出通过并行计算方法进行大规模人群疏散模拟,他们大多采用了计算机集群系统来实现程序并行,例如文献[1]提出了面向场景的计算机集群处理方式。但是,这样做势必造成系统不能充分利用当前计算机资源,同时,集群之间的信息通信延迟必然也会对人群疏散的实时模拟造成一定影响。

1.2 人物寻径算法

为了模拟人群的智能疏散行为,路径搜索必不可少。在计算机及相关科学领域中有许多寻径算法,A Star算法^[2]就是其中一种非常著名的算法,它也是目前寻找次优开销路径较快的人工智能算法。

A Star算法作为一种启发式搜索算法,它引入了一个特别的估价函数:

$$f(n) = g(n) + h(n) \quad (1)$$

其中: $f(n)$ 是从初始点经由节点 n 到目标节点的估价函数; $g(n)$ 是从初始节点到节点 n 的实际代价; $h(n)$ 是从节点 n 到目标节点最佳路径的估计代价。与其他路径搜索相比,它不会在搜索的过程中选取最佳节点后舍弃其他的兄弟节点,而是在每一步的估价中都把当前的节点与以前节点的估价值比较而得到一个最佳节点,可以有效地防止最佳节点的丢失。同时,它又综合考虑了两种移动开销,一是Dijkstra算法所考虑的,即起点到当前位置的移动开销;另外一个就是从当前位置到目标地点剩余的移动开销。正是由于这些优势,许多游戏中的NPC寻路均采用了A Star算法,如星际争霸、魔兽争霸等。

本文在A Star算法的基础上进行了优化,不仅提高了寻径算法的效率,还实现了人物智能即在复杂场景中选择疏散路径,较真实地模拟了现实中人们通过观察周围环境选择疏散路径的情况。

2 大规模人群疏散模拟算法

考虑到A Star算法存在的局限性,本文首先对该算法进行了优化,大大提高了寻径算法的效率。在此基础上,本文针对传统基于中央处理器串行方式和基于集群并行方式所存在的弱点,结合图形处理器强大的并行计算能力,实现了大规模人群实时疏散模拟,从而用一块支持CUDA的显卡便可拥有与计算机集群相当甚至更高的计算能力,同时也减少了信息通信所带来的延迟,更加方便、有效地实现了实时人群疏散模拟。与传统基于中央处理器串行方式相比,其提高了算法的效率,实现了实时疏散模拟;与传统集群解决方案相比,又减轻了并行负载均衡问题的难度,大大降低了经费投入。

2.1 个体寻径优化算法

A Star算法存在一定的局限性,当来自高级AI模块的查询产生数量极大的A Star算法调用时,会大量耗费中央处理器的资源。当寻路中存在多个起始位置或者多个终止位置时,传统A Star算法势必造成高级AI模块查询数量增加,导致多次A Star算法调用。例如,在当前建筑物场景中,有多个安全出口,然后高级AI模块发出一个请求,个体在这些出口中选择一条开销最小的路径避开障碍物(建筑物、人群等),从而离开建筑物。底层代码会针对每一个出口都调用一次A Star算法,比较之后再返回最小开销路径的出口位置。

为了避免多次调用A Star算法寻优路径所造成时间代价过大,本文将这些起始节点都变为开放节点,并添加到开放列表中,开放列表可以返回一个最佳的起始节点。而多个终止节点则加入到一个终止列表(终止节点只有在被移出开放列表之后才能使用)中,寻径过程中遇到任何一个终止节点都可以停止算法的遍历,从而得到一条次优路径。

经过这样的优化后性能明显提升。假设当前存在 m 个起始节点和 n 个终止节点,任一起始节点到任一终止节点过程中遍历的节点数最大为 k ,则采用传统方法多次迭代的A Star算法,每次查询需要遍历 $m \times n \times k$ 个节点,才能找到起始节点和终止节点直接的较优路径。而使用改进后的方法,由于其聚合了输入参数,找到同样的次优路径只需遍历 k 个节点。从时间复杂度上看,传统方法的时间复杂度为 $O(m \times n \times k)$,而优化后的时间复杂度为 $O(k)$ 。

该算法用到的数据结构说明如下:

open list,保存所有已生成而未考察的节点;

closed list,记录已访问过的节点;

start list,存放所有起始节点;

stop list,存放所有终止节点。

算法执行步骤如下:

a)初始化open list、closed list,将所有起始节点添加到start list,所有终止节点添加到stop list。

b)将start list中所有节点添加到open list。

c)若当前节点不包含在stop list(转步骤d),否则转步骤i)。

d)若open list为空,则转步骤h);否则,选取open list中具有最小 F 值的节点为最佳节点bestNode,并将其放入closed list。

e)若bestNode不包含在stop list中,则扩展之,产生后继节点successor;否则,转步骤i)。

f)对每个successor进行下列过程:

(a)若successor在open list中,且successor的估价值小于open list估价值,则更新open list估价值;

(b)若successor在closed list中,且successor的估价值小于closed list估价值,则更新closed list中的估价值,并把successor节点放入closed list;

(c)若successor既不在open list,又不在closed list中,求successor的估价值,并将successor插入open list中;

g)将bestNode节点插入closed list中,并按照估价值将open list中的节点排序,转步骤c)。

h)不能找到路径,返回失败,算法结束。

i)成功求得最小开销路径,返回成功,算法结束。

经过优化后的 A Star 算法将多个起始节点和终止节点作为参数传递,会增加额外的性能开销。但是,它避免了高级 AI 模块多次调用 A Star 算法造成的大量资源耗费,同时又解决了在出现多个初始位置或多个终止位置时寻径生成最小开销路径的问题。

2.2 人员疏散仿真并行化

在人员疏散仿真过程中,由于人员数量的大规模增加,寻径有其自身独特的特点,它要求在人数众多的情况下搜索时间短,达到实时性。这样就使通常基于单机的串行算法已经不能满足计算需求,程序达不到实时性的要求。

如图 1 所示,在人员疏散仿真流程中,时间开销较大的阶段是人群寻径时间消耗。个体寻径时使用的改进 A Star 算法可以满足使用图形处理器并行计算对数据的两个要求,即数据相关性小、计算密集度高。因此,本文选择利用图形处理器众核优势并行处理个体寻径问题,将每个个体的寻径抽象为图形处理器上的一个独立线程任务。这样就大大缩短了大规模人群寻径时间,从而提高了仿真效率,在场景中人数众多的情况下,实现了人物的实时寻路行为。

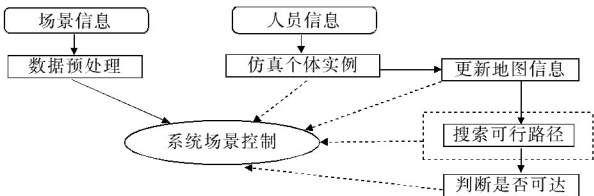


图 1 人群疏散仿真过程

人群疏散仿真算法的主要流程如下:

- a) 在主机端 (CPU) 初始化全局信息,包括场景地图信息、最初人物位置、障碍物位置等。
- b) 为场景中每个个体在主机端分配相应资源,包括初始化各自的开放列表 (open list)、关闭列表 (closed list)、起始点列表 (start list)、终止点列表 (stop list) 等。
- c) 更新客户端 (GPU) 上每个个体信息,利用优化后的寻径算法进行个体独立寻径。
- d) 将客户端计算产生的寻径信息传送到主机端,由主机端进行逻辑处理,更新个体信息,并进行个体避障。若场景中仍然存在未被疏散离开场景的个体,则跳转到步骤 c), 否则算法结束。

2.3 个体避障

由于人物寻径采用并行处理方式且个体路径实时生成,必然造成个体寻径冲突,发生穿人现象。与文献[3]中提出的方法类似,本文采用如下方法避免了此类问题的发生,即由图形处理器实时计算产生的个体寻径信息,传递给主机端后,由中央处理器根据每个个体的运动速度、运动方向等信息最终决定其运动方式。

3 实验结果及分析

本文方法实验硬件环境为: Intel Core 2 Duo E7400 (2.80 GHz) 处理器, 4 GB 内存, NVIDIA GTX285 显卡。

3.1 个体寻径优化实验

本文在多个不同的场景中做了实验,在场景拥有不同起始位置和终止位置的情况下,使用基本的 A Star 算法和本文改进

后的 A Star 算法模拟个体疏散路线。

图 2~7 中绿色方格表示终止位置,红色圆点表示起始位置,灰色方格表示障碍物,橙色方格表示采用基本寻径算法的个体路径,蓝色方格表示采用优化后寻径算法的个体路径 (见电子版)。实验结果表明 (表 1), 使用基本寻径算法和优化后的寻径算法均能正确找到开销较小的次优路径,但随着起始位置和终止位置的增加,会增加寻径时间的开销,而本文所述方法较好地解决了这一问题,大大提高了寻径效率。

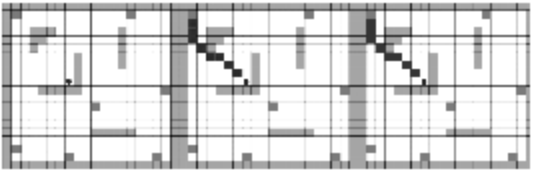


图 2 20×20 场景、1 个起始位置、7 个终止位置



图 3 25×25 场景、7 个起始位置、1 个终止位置



图 4 35×35 场景、6 个起始位置、8 个终止位置

表 1 在不同场景下,本文寻径算法相对于基本寻径算法的加速比

场景大小	起始位置	终止位置	基本寻径算法时间/ms	本文寻径算法时间/ms	加速比
20 × 20	1	7	1.85	0.4	4.600
25 × 25	7	1	3	0.8	3.750
35 × 35	6	8	1 110.5	113.5	9.784

3.2 大规模人群疏散模拟实验

在大小为 100 × 100 的场景中模拟人群疏散情况,实验结果如表 2 所示。

表 2 不同模拟人数下渲染性能比较

人数	地图大小	CPU 方式耗时/ms	GPU 方式耗时/ms
1	100 × 100	0.002	2.061
10	100 × 100	0.012	2.165
100	100 × 100	27.168	2.481
1 000	100 × 100	268	6.013
10 000	100 × 100	2 215	62.511
50 000	100 × 100	12 875	348.614

表 2 的实验结果表明,本文的算法效率相比于基于 CPU 的人群模拟方式,大幅度提高了人群模拟的规模。传统基于 CPU 的人群模拟方式在人数规模很小的情况下,能达到很高的渲染效率,但随着人群规模的大幅度增加,渲染效率则呈线性方式下降。而基于 GPU 的人群模拟方式,在人数达到 1 000 以上时,渲染效率明显优于基于 CPU 的人群模拟方式,在人群规模为 5 万的情况下,基于 GPU 的人群模拟方式加速比达到了 36.9 倍。实验表明在 100 × 100 的场景中,基于 CPU 的人群模拟方式只能有效实时地模拟 3 000 人左右,而基于 GPU 的并行处理方式可实时模拟 8 万人左右。

模拟中型体育馆人群疏散实验结果截图如图 5~7 所示。
(下转第 1194 页)

相似的是 s10_7,平均百分比偏差距离 0.017 051;最不同的是 s17_1,平均百分比偏差距离 1.887 963,两者相差 110 倍,获得了较好的区分度。

表 6 图像 s10_1 匹配结果

序号	待检测图像	对比图像	平均百分比偏差距离 (L_{mped})
1	s10_1	s10_1	0.000 000
2	s10_1	s10_7	0.017 051
3	s10_1	s10_3	0.020 830
4	s10_1	s10_9	0.034 598
5	s10_1	s10_4	0.035 695
6	s10_1	s10_2	0.047 336
7	s10_1	s12_10	0.053 008
8	s10_1	s10_8	0.054 811
9	s10_1	s10_6	0.061 843
10	s10_1	s12_9	0.068 605
11	s10_1	s10_5	0.069 874
12	s10_1	s12_7	0.077 971
13	s10_1	s12_1	0.080 923
14	s10_1	s28_1	0.081 627
15	s10_1	s12_5	0.082 645
16	s10_1	s12_4	0.083 234
17	s10_1	s28_6	0.085 378
...	...	...	...
399	s10_1	s7_10	1.556 354
400	s10_1	s17_1	1.887 963

实验结果表明,同一人不同表情图像的 Trace 特征相似度较高,不同人脸图像的 Trace 特征相差较大。因此基于 Trace 变换的人脸特征,能够有效区分不同人的脸图像,并对同一人脸图像的表情及光照等外在变化能够保持较好的稳定性。

4 结束语

本文提出的基于 Trace 变换的人脸图像识别算法是在对人脸图像进行 Trace 特征提取前增加了对人脸图像的预处理过程。这样,可以将人脸图像的背景、头发、耳朵等外在变化的干扰信息去除,从而保证所提取的人脸描述特征只反映人脸的内在变化,有助于提高人脸识别的精度和准度。通过对仿真结

果的对比分析,人脸图像的 Trace 变换特征能够反映出人脸细节的变化,具有较好的区分能力,对同一人脸不同表情的图像有较好的识别效果,在人脸识别方面具有一定的研究价值。但是,由于人脸的自然特性复杂多变,单纯依赖某一种方法进行人脸识别,其效果有限,因此,需要结合两种或更多不同的人脸特征进行匹配判断以提高人脸识别能力。基于 Trace 变换的方式与人脸识别中已经研究较多的 PCA、SVD 等方法有本质的不同,所以在下一步的工作中,可以考虑结合 PCA 等方法,进一步提高算法的识别能力。

参考文献:

[1] 苏煜,山世光,陈熙霖,等. 基于全局和局部特征集成的人脸识别[J]. 软件学报,2010,21(8):1849-1862.

[2] ANIL K J, ARUN R, SALIL P. An introduction to biometric recognition[J]. IEEE Trans on Circuits and Systems for Video Technology:Special Issue on Image and Video Based Biometrics, 2004,14(1):4-20.

[3] 孙冬梅,裴正定. 生物特征识别技术综述[J]. 电子学报,2001,29(12A):1744-1754.

[4] 尚丽,陈杰,张愉. 人脸自动识别技术综述[J]. 苏州市职业大学学报,2010,21(1):1-12.

[5] 刘鹭,董立文,姜鹏,等. 人脸识别技术的研究与应用[J]. 科技信息,2010(14):492.

[6] 何国辉,甘俊英. PCA 类内平均脸法在人脸识别中的应用研究[J]. 计算机应用研究,2006,23(3):165-169.

[7] 余其兵,欧宗璞. 基于 Trace 变换的人脸认证系统及相关技术的研究[D]. 大连:大连理工大学,2006.

[8] KADYROV K, PETROU M. The Trace transform and its application[J]. IEEE Trans on Pattern Analysis Machine Intelligence, 2001,23(8):811-828.

[9] 吴一全,郝红杰. 一种基于 Trace 变换的数字图像版权保护算法[J]. 中国图象图形学报,2008,5(5):900-905.

[10] 徐艳,陈孝威. 基于差分图像和 PCA 主元分析的人脸检测[J]. 计算机应用与软件,2008,25(8):229-230.

(上接第 1168 页)

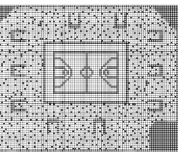


图 5 中型体育馆人群疏散开始时

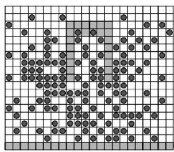


图 6 中型体育馆人群疏散中期个体在门口拥堵

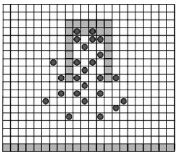


图 7 中型体育馆人群疏散后期个体从门口通过

4 结束语

本文针对大规模人群疏散实现了一种基于图形处理器的并行算法。相比于传统的单机串行方式,本方法可以在人群规模较大时实现人群模拟的实时性。而相比基于计算机集群的方法,本方法又可以充分利用本地计算机资源,同时大大提高了通用性。此外,本文还优化了传统寻径算法,相比传统寻径算法减少了高级 AI 模块寻径调用次数,节省了系统资源,提高了运行效率,得到了更好的效果。但是,该实现方法目前局限于个体独立性运动较强的并行处理,无法很好地模拟人群疏散过程中人的从众心理。今后,还将对该方法作进一步的改进。

参考文献:

[1] 刘晓平,曹力,张高峰. 面向场景的人群疏散并行化仿真[J]. 系统仿真学报,2008,20(18):4809-4816.

[2] HART P E, NILSSON N J, RAPHAEL B. A formal basis for the heuristic determination of minimum cost paths in graphs[J]. IEEE Trans on Systems Science and Cybernetics, 1968,4(2):100-107.

[3] 王洁,王兆其,李淳芃,等. 一种用于群体模拟的分层次避障法[J]. 计算机研究与发展,2007,44(12):2058-2065.

[4] 王兆其,毛天露,蒋洁,等. 人群疏散虚拟现实模拟系统——Guarder[J]. 计算机研究与发展,2010,47(6):969-978.

[5] TREUILLE A, COOPER S, POPOVIC Z. Continuum crowds[J]. ACM Trans on Graphics,2006,25(3):1160-1168.

[6] 方正,卢兆明. 建筑物避难疏散的网格模型[J]. 中国安全科学学报,2001,11(4):10-13.

[7] 徐高. 基于智能体技术的人员疏散仿真模型[J]. 西南交通大学学报,2003,38(3):301-303.

[8] PALLISTER K. Game programming gems 5[M]. [S. l.]: Charles River Media,2005:287-300.

[9] 吴恩华,柳有权. 基于图形处理器(GPU)的通用计算[J]. 计算机辅助设计与图形学报,2004,16(5):601-612.

[10] 刘奥,姚益平. 基于高性能计算环境的并行仿真建模框架[J]. 系统仿真学报,2006,18(7):2049-2051.