

СПОСОБЫ ОПТИМИЗАЦИИ КОМПЬЮТЕРНЫХ ИГР

В этой статье рассматривается, что такое оптимизация и наиболее популярные методы применения её в компьютерных играх.

ВВЕДЕНИЕ

Оптимизацией игры называется процесс повышения производительности для улучшения игрового процесса и визуального восприятия. Количественной характеристикой оптимизации зачастую выступает значение FPS (Frames Per Second – количество кадров в секунду). Этот параметр напрямую влияет на восприятие изображения, поскольку чем кадров больше, тем плавнее движется картинка. Показатель хорошей оптимизации – работающая игра работает с одинаковой частотой кадров на широкой линейке аппаратных конфигураций, включая низкопроизводительные.

I. ГРАФИЧЕСКАЯ ОПТИМИЗАЦИЯ

Наиболее распространённой техникой оптимизации графики является **Level of Detail**. Она организует переключение между различными степенями детализации объекта в зависимости от расстояния до него. Важно уточнить, что технология Level of Detail не влияет на качество текстур, а лишь на дальность отображения более качественного изображения. Схожим образом работает метод **MIP Mapping**, однако он обрабатывает текстуры объектов, а не полигоны. Технология MIP Mapping'a основывается на создании так называемой MIP-пирамиды, состоящей из одинаковых текстур различного разрешения.

Менее распространённым методом оптимизации текстур является использование **текстурных атласов** – особых изображений, состоящих из ряда различных игровых текстур. Используя данную технологию, программа получает сведения в виде координат текстурного атласа, по которым располагается необходимая для отрисовки текстура. Это значительно ускоряет систему.

Одним из наиболее значимых методов оптимизации является **Culling** – алгоритм отсека невидимой геометрии. Такие алгоритмы могут учитывать взаимное перекрытие и другие особенности динамических объектов. Самый простой и эффективный алгоритм отсека –

View Frustum Culling – отрисовка только тех объектов, которые находятся в поле видимости камеры. Чаще всего VFC используется в паре с Occlusion Culling – алгоритмом отсека, который отключает рендеринг объектов сцены, закрытых другими объектами.

II. ОПТИМИЗАЦИЯ ПРОГРАММНОГО КОДА

Архитектура современных игр выстраивается с учетом **многопоточности**, т.е. разбиением работы на независимые части и их параллельным выполнением. Независимые друг от друга объекты возможно разбить на группы одинакового размера и обработать параллельно на разных ядрах процессора, нагружая их равномерно. К некоторым областям программы многопоточность просто неприменима: две функции, результат одной из которых зависит от другой, разбить на потоки невозможно.

Метод **кэширования** применяется в тех случаях, когда количество объектов быстро изменяется. Во время запуска игры выделяется область памяти, в которую заносится базовый набор объектов. Созданные в кэше объекты извлекаются из него по мере необходимости, а затем помещаются обратно, когда перестают быть нужными.

III. ВЫВОДЫ

В современном мире важность компьютерных игр возросла, поскольку они получили практическое применение в таких отраслях, как киноиндустрия, военное дело, системы обучения и многие другие. Поэтому оптимизация компьютерных игр имеет большое значение, так как она улучшает игровой процесс и повышает качество пользовательского опыта.

Список литературы

1. Heckbert, P. S. Survey of texture mapping // Computer Graphics Lab. – 1986. – с. 207-212.
2. Williams, L. J. Pyramidal Parametrics // Computer Graphics Lab. – №17 – 1983. – 11 с.
3. Lengyel, E. Oblique View Frustum Depth Projection and Clipping // Journal of Game Development – №2 – 2005. – с. 5-16.

Кассирова Александра Денисовна, студентка 1 курса факультета ИТиУ БГУИРа, sashakassirova@mail.ru.

Шнип Александр Дмитриевич, студент 1 курса факультета ИТиУ БГУИРа, ashnip@mail.ru.

Научный руководитель: Шатилова Ольга Олеговна, старший преподаватель кафедры вычислительных методов и программирования БГУИР, магистр технических наук, o.shatilova@bsuir.by.