

«Программирование с использованием OpenGL. Визуализация растительности»

Грызлов Александр Викторович

От визуализации растительности существенно зависит качество визуализации сцены в целом. Если мы рассматриваем естественные ландшафты, то растительность занимает существенную часть пространства и требования к качеству визуализации здесь особенно высокие. В игровых приложениях данной тематике уделяется мало внимания из-за того, что повышение качества визуализации растительности влечет к замедлению основного игрового процесса, а в приложениях данного типа это неприемлемо, так как основой здесь служат именно игровые процессы, а процессы визуализации имеют лишь второстепенное значение. В приложениях, которые направлены на визуализацию все совсем наоборот. Здесь все время направлено на детальную прорисовку картинке, именно такие приложения мы и будем рассматривать. Должны присутствовать возможности передвижения по сцене, изменение параметров сцены (интерактивное) и др.

Существующие методы, позволяющих визуализировать растительность имеют некоторые достоинства и недостатки. Каждый из них основывается на понятии «уровень детализации».

Мой алгоритм основан на методе визуализации «пучков травы», представленных как множество элементов, несколько напоминающих травяные пучки. Такой пучок состоит из шести взаимно пересеченных плоскостей, на которые накладывается текстура травы. Этот подход достаточно реалистично визуализирует траву при сравнительно небольшой нагрузке на графический процессор. Возможна реализация воздействия ветра и моделирования травы различных природных видов. Не представляет трудности моделирование травы различной высоты.

Текстура накладывается таким образом, чтобы каждая отдельная плоскость имела видимые и невидимые пиксели, что реализуется при помощи алгоритма альфа отсечения.

В программе был использован алгоритм сдвига плоскостей. Происходят периодические колебания, зависящие от времени. Предусмотрена начальная фаза сдвига, с которой начинается движение. В результате чего воздействие ветра на траву выглядит достаточно реалистично.

Для построения сложных трёхмерных сцен без уменьшения FPS реализован алгоритм замены удалённых от камеры «пучков» травы так называемыми «билбордами».

Билборд – это поверхность, вектор нормали которой сонаправлен с вектором ближайшей плоскости отсечения.

Используя функцию нахождения круга заданного радиуса с центром в точке наблюдения, определяем «пучки» входящие в круг, а остальные объекты заменяем.

Получившаяся сцена не намного уступает по качеству отрисовки сцене без применения билбордов. Однако имеется неприятный недостаток связанный с видимой подменой объектов находящихся на окружности. И всё же этот недостаток, с которым приходится мириться несоизмерим с тем достоинством, которое мы получаем.

В программе реализован простой движок, который предоставляет возможность управления камерой наблюдателя при помощи клавиатуры. Управление камерой позволяет наблюдать траву с разных позиций: заглянуть вглубь травы, опустив взгляд «под ноги», осмотреть траву вдали, забравшись на холм.

Для реализации ландшафта в программе применён способ его отрисовки при помощи карты высот.

Карта высот представляет изображение в RAW формате, с палитрой оттенков серого цвета для определения глубины/высоты точки ландшафта. Более светлые оттенки отображают более высокие участки.

Генерация травы также предусматривает различные вариации вида травы и её положения:

- наложение заранее подготовленной текстуры по её номеру
- различную фазу сдвига для реализации воздействия ветра
- различный угол поворота пучка травы
- случайное изменение оттенка травы
- случайное отклонение координат «посадки»

Преимущества алгоритма:

- § Высокая скорость визуализации
- § Обзор травы с различных позиций и углов камеры
- § Реалистичность при обзоре со средних дистанций
- § Возможность исполнения травы различных видов

Можно улучшить алгоритм, применяя комбинацию известных простых методов визуализации. Тем самым добиться большей реалистичности без ухудшения общей производительности сцены. Данные методы можно также применять для визуализации других сложных природных объектов, например, таких как деревья. Так же возможно разработать новый подход, основываясь, к примеру, на фрактальной геометрии. Стоит отметить что задача визуализации природных объектов актуальна, и не ограничивается лишь компьютерными играми. Она находит своё применения в приложениях для моделирования ландшафтного дизайна, в воссоздании памятников архитектуры и геоинформационных системах.