

Метод непрерывной детализации трехмерных моделей растительности

Н. А. Елыков, И. В. Белаго, С.А. Кузиковский
{nicolas, bel, stas}@sl.iae.nsk.su

Институт Автоматики и Электрометрии СО РАН
Новосибирск, Россия

Аннотация

В работе предложен метод для высококачественной визуализации больших непрерывных пространств 3D растительности.

Ключевые слова: системы виртуальной реальности, визуализация 3D растительности.

1. ВВЕДЕНИЕ

Современные графические ускорители для персональных компьютеров позволяют решать задачи, требующие генерации изображения визуальной обстановки фотографического качества в реальном времени.

Высококачественная визуализация больших непрерывных пространств 3D растительности является одним из требований при создании большинства тренажеров наземного транспорта, а также многих приложений виртуальной реальности.

Представление больших пространств леса в виде отдельно стоящих моделей деревьев, обладающих приемлемым визуальным качеством для отображения вблизи от наблюдателя, невозможно по причине огромного количества примитивов, необходимых для визуализации. Представление больших пространств леса, в виде текстурированной поверхности, приемлемое для визуализации леса находящегося вдали от наблюдателя, по мере приближения к нему становится неприемлемым. Таким образом, необходимо непрерывное представление визуальной модели 3D растительности, позволяющее, с одной стороны, отображать их в реальном времени с использованием современных графических ускорителей, и обладающее, с другой стороны, приемлемым визуальным качеством на всем пространстве визуализации. В данной работе представлен метод для высококачественной визуализации больших непрерывных пространств 3D растительности, позволяющий при этом равномерно распределять нагрузку по визуализации между CPU и GPU.

Описываемый метод состоит из двух представлений: представление леса на дальнем плане, представление леса на ближнем плане, а так же перехода из одного представления в другое.

2. ПРЕДСТАВЛЕНИЕ ЛЕСА НА ДАЛЬНОМ ПЛАНЕ

Вдали лес можно представить, в виде текстурированной соответствующей текстурой, 3D моделью земли, однако подобное представление не обладает

необходимым объемом, характерным для леса. Для придания объема использовался метод, предложенный в [1], для визуализации меха. Лес при этом представляется набором слоев повторяющих геометрию земли, и сдвинутых относительно нее либо вверх, либо по направлению нормали на соответствующие расстояния см. Рис 1. Причем, для того что бы не производить детализацию геометрии слоев леса одновременно с детализацией геометрии земли, а так же для того, что бы избежать излишнего роста объема базы данных, геометрия слоев генерировалась при помощи *Custom Vertex Shader* из геометрии земли при отображении.

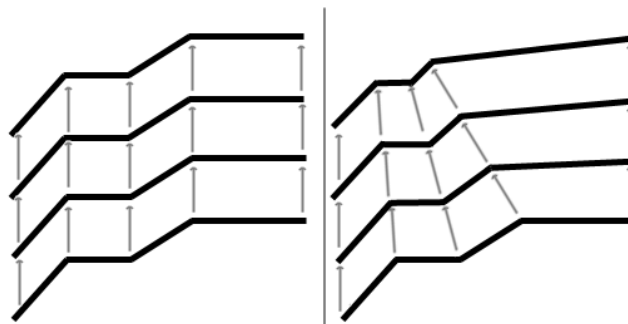


Рис.1. Слева сдвигка слоев деревьев относительно геометрии земли вверх, справа сдвигка по нормали.

Каждый из слоев текстурируется специальной текстурой T_i представляющей горизонтальный срез одного или нескольких деревьев см. Рис. 2.

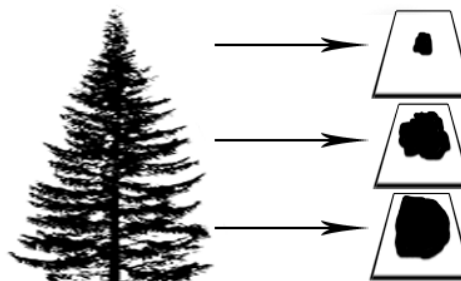


Рис 2. Генерация текстур слоев леса

Для придания большего объема текстуры нижних слоев затемняются. Что бы достичь большего визуального разнообразия, слои леса также текстурируются еще одной текстурой – Gr (см. Рис 3), которая RGB каналами

подкрашивает деревья, а A каналом “вырезает” поляны, причем эта же текстура, с теми же что и на слоях текстурными координатами, используется и для визуализации земли. Таким образом, финальный цвет i -го слоя деревьев вычисляется с помощью формулы:

$$TreesColor_{iRGB} = 2 * T_{iRGB} * Gr_{RGB}$$

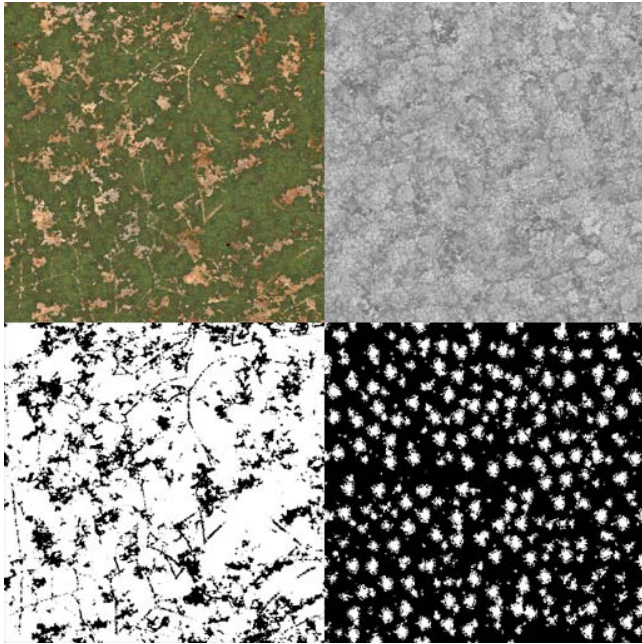


Рис. 3 Текстуры Gr и Tr (сверху RGB каналы, снизу альфа канал).

$$TreesColor_{iA} = T_{iA} * Gr_A$$

При этом пространственное распределение деревьев будет задаваться с помощью формулы:

$$TreesDistribution = \frac{1}{n} \sum_{i=0}^{n-1} TreesColor_{iA}$$



Рис. 4. Слева - артефакты на силуэтах гор. Справа – артефакты на силуэтах гор замаскированы системой вертикальных загоронок.

Однако при использовании данного метода на силуэтах гор, а так же, когда направление взгляда параллельно слоям леса, возникают визуальные артефакты

(см. Рис 4). Для того что бы скрыть большинство из подобных артефактов было предложено использовать систему вертикальных загоронок. Загородки – набор вертикальных двухсторонних граней генерируемых на ребрах 3D модели террейна и имеющих характерную высоту деревьев (см Рис 5с).

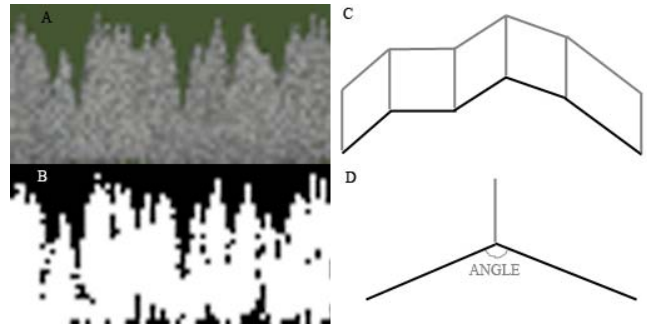


Рис 5. А. RGB каналы текстуры S . В. Альфа канал текстуры S . С. Система вертикальных загоронок построенных на ребрах террейна. Д. Критический угол между треугольниками, на общем ребре которых может быть поставлена загородка.

Загородки текстурируются текстурой S , представляющей собой вертикальный срез одного или нескольких деревьев см. Рис 5ab. Так же для уменьшения визуальных отличий между слоями леса и вертикальными загорокками, на загородки накладывается, с теми же текстурными координатами, что на земле и на слоях леса, текстура Gr , а текстурные координаты для текстуры S подбираются так, что бы как можно более точно соответствовать текстуре пространственного распределения деревьев - $TreesDistribution$. Кроме того, что бы избежать визуальных отличий в освещении загоронок, нормаль к загородкам, используемая для освещения, либо направляется вертикально вверх, либо перпендикулярно террейну на котором находится загородка. Финальный цвет вертикальных загоронок вычисляется по формуле:

$$Color_{RGB} = 2 * S_{rgb} * Gr_{RGB}$$

$$Color_A = S_A * Gr_A$$

Для того, что бы уменьшить количество граней образующих систему вертикальных загоронок, загородки генерируются только на тех ребрах террейна, у которых угол между гранями, содержащими данное ребро, меньше критического см. Рис. 5d. Критический угол выбирается эмпирически в зависимости от характера террейна для которого генерируется система загоронок. Так же из системы исключаются полностью прозрачные загородки, т.е. те загородки которые полностью попадают на поляну.

При использовании системы вертикальных загоронок, когда ориентация загородки параллельна направлению взгляда, появляются визуальные артефакты связанные с трilinearной фильтрацией текстур. Для того, что бы избавиться от подобных артефактов, введена зависимость прозрачности вертикальных загоронок от угла

между касательной к загородке \vec{T} и направлением на наблюдателя \vec{V} : $A = 1.0 - \text{asb}(\text{dot}(\vec{T}, \vec{V}))$.

Прозрачность вычисляется в каждой вершине загородки с использованием *Custom Vertex Shader* и интерполируется внутри грани. Объединив загородки с близкими касательными в группы и сохранив для каждой из групп конус касательных, можно производить раннюю отбраковку полностью прозрачных групп загоронок.

Описанное представление больших пространств леса, в виде комбинации системы слоев и вертикальных загоронок, применимо только для отображения дальних планов, поэтому необходимо плавно управлять прозрачностью данного представления леса в зависимости от расстояния до наблюдателя. Кроме того, что бы уменьшить артефакты связанные с клипированием геометрии дальней плоскостью отсечения, прозрачность слоев леса, системы вертикальных загоронок и земли увеличивается по мере приближения к дальней отсекающей плоскости. Таким образом прозрачность задается в следующем виде:

$$Alpha = \begin{cases} 0 & \text{если } 0 < d < FadeInStart \\ \frac{d - FadeInStart}{FadeInEnd - FadeInStart} & \text{если } FadeInEnd < d < FadeOut \\ 1 & \text{если } FadeOut < d \end{cases}$$

,где d – расстояние до наблюдателя, $FadeInStart$ – расстояние на котором начинают появляться слои и загородки, $FadeInEnd$ – расстояние на котором слои и загородки полностью появились, $FadeOut$ – расстояние на котором слои и загородки начинают исчезать, что бы избежать артефактов связанных с отсечением дальней плоскостью, $FarPlane$ расстояние до дальней отсекающей плоскости.



Рис 7. Пример работы слоев леса и системы вертикальных загоронок.

Для дополнительной оптимизации система слоев и вертикальных загоронок разбивается на пространственные

группы, и если данная группа является полностью прозрачной - она не визуализируется, кроме того, данная группировка используется для отбраковки граней, не попавших в область видимости.

3. ПРЕДСТАВЛЕНИЕ ЛЕСА НА БЛИЖНЕМ ПЛАНЕ

В предыдущей главе рассмотрено представление больших лесных массивов, однако он не предназначен для использования вблизи от наблюдателя. Для представления леса вблизи от наблюдателя используется набор трехмерных моделей деревьев помещенных на террейн [2]. Переход к представлению на дальнем плане осуществляется через плавное изменение прозрачности трехмерных деревьев. При этом прозрачность меняется следующим образом:

$$Alpha = \begin{cases} 0 & \text{если } d > FadeInEnd \\ 1 & \text{если } d < FadeInStart \\ \frac{d - FadeInEnd}{FadeInStart - FadeInEnd} & \text{иначе} \end{cases}$$

Причем т.к. область исчезновения трехмерных моделей деревьев, и область появления слоев леса и вертикальных загоронок перекрываются, то количество визуальных артефактов, появляющихся при движении наблюдателя, минимально.

Для уменьшения визуальных артефактов связанных с переходом из одного представления в другое модели деревьев расставляются в соответствии с текстурой пространственного распределения *TreesDistribution* описанной в предыдущей главе, цвет дерева определяется текстурой *Gr*, само дерево освещается в соответствии с нормалью к терреину в точке в которое дерево помещается.

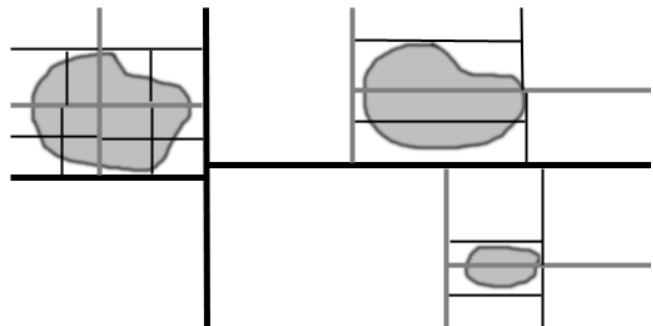


Рис 8. Пример построения kd-tree.

Трехмерные модели деревьев можно визуализировать за N вызовов к графическому ускорителю, где N – пропорционально количеству деревьев которое необходимо отобразить в данный момент времени. Т.к. число деревьев для отображения достаточно велико, то такой подход ведет к значительному снижению производительности [3]. С другой стороны можно на этапе подготовки объединять трехмерные модели деревьев в группы, что бы визуализировать их за один вызов к

графическому ускорителю, однако такой подход приводит к значительному увеличению объема базы данных.

Для решения данной проблемы авторы проводили группировку трехмерных моделей на этапе отображения. Для этого на этапе подготовки лес пространственно разбивался на *kd-tree*, таким образом, чтобы лист дерева был либо пуст, т.е. не содержал ни одного дерева, либо имел характерные размеры 200 – 300 метров и содержал как можно больше деревьев см. Рис. 8. Каждый непустой лист содержит несколько деревьев, заданных своей позицией относительно границ листа, нормалью к поверхности в данной точке, цветом дерева и номером трехмерной модели в списке всех моделей деревьев. Причем, для экономии дискового пространства позиция и нормаль задавались с пониженной точностью. На этапе отображения *kd-tree* рекурсивно обходится, при этом если узел не попадает в область видимости, то он и его потомки не обходятся. Все трехмерные модели деревьев из видимых и непустых узлов *kd-tree*, собираются, сортируются в порядке “от глубины к себе” и визуализируются за один вызов к графическому ускорителю.

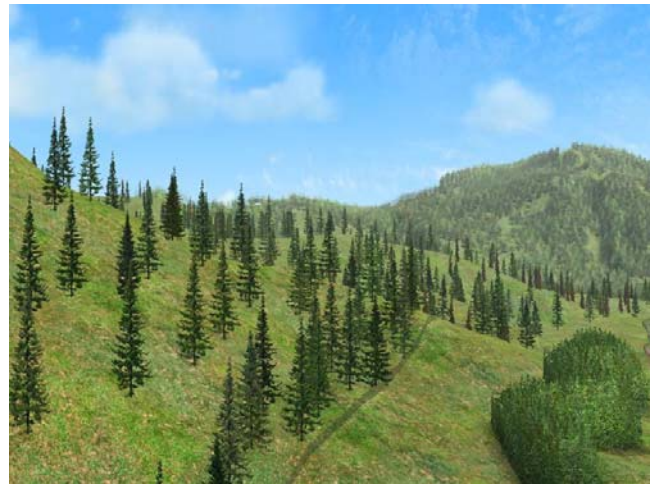


4. ЗАКЛЮЧЕНИЕ

Авторами данной работы был разработан метод непрерывной детализации трехмерных моделей растительности в системах виртуальной реальности, обладающий приемлемым визуальным качеством на любом расстоянии до наблюдателя.

Предложенный метод учитывает особенности архитектуры современных видеоускорителей и позволяет равномерно распределить нагрузку по отображению между *CPU* и *GPU*. При этом метод не требует значительных ресурсов для визуализации.

Реализация метода показала, что визуализация участка лесистой поверхности 7500x7500 м.² на машине *Pentium IV – 2.4 GHz*; с видео ускорителем *NVidia GeForce 4* происходило со скоростью ≈60 кадров в секунду. Такой скорости в большинстве случаев оказывается достаточно для эффективного применения метода в системах синтеза визуальной обстановки.



5. ЛИТЕРАТУРА:

- [1] Jerome Lengyel, Praun Emil, Finkelstein Adam, Hoppe Huges. “Real-Time Fur over Arbitrary Surfaces” *ACM Symposium on Interactive 3D Graphics 2001*, 227-232.
- [2] Kurt Pelzer, Piranha Bytes. “Rendering Countless Blades of Waving Grass”, “*GPU Gems*“, Addison Wesley, 2004, 107-121
- [3] Francesco Carucci. “Inside Geometry Instancing”, “*GPU Gems 2*“, Addison Wesley, 2005, 47-67

Об авторах:

Ельков Николай Александрович– м.н.с. ИАиЭ СО РАН

E-mail: Nicolas@sl.iae.nsk.su

Белаго Игорь Викторович– н.с. ИАиЭ СО РАН

Кузиковский Станислав Александрович– н.с. ИАиЭ СО РАН

A method for continuous level of detail for 3D vegetation model

Abstract

We propose a method for high quality visualization of large continuous areas covered with 3D vegetation.

Keywords: virtual reality, real-time rendering, 3D vegetation visualization.

Elykov Nikolay – Institute of Automation and Electrometry SB RAS. E-mail: Nicolas@sl.iae.nsk.su

Belago Igor– Institute of Automation and Electrometry SB RAS.

Kuzikowski Stanislaw – Institute of Automation and Electrometry SB RAS.