

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Нижегородский государственный университет им. Н.И. Лобачевского

**ФОТОРЕАЛИСТИЧНОЕ МОДЕЛИРОВАНИЕ И ВИЗУАЛИЗАЦИЯ
РАЙОНОВ ГОРОДСКОЙ СРЕДЫ**

Учебно-методическое пособие

**Рекомендовано методической комиссией факультета вычислительной
математики и кибернетики для студентов ННГУ, обучающихся
по направлению подготовки 010300
“Фундаментальная информатика и информационные технологии”**

Н. Новгород
2014

УДК 681.3.06(072)
ББК В185
Т38

Т38 Фотореалистичное моделирование и визуализация районов городской среды. Составитель Осипов М.П.: Учебно-методическое пособие – Нижний Новгород: Нижегородский госуниверситет, 2014. – 50 с.

Рецензент: канд. физ.- мат. наук **Л.И. Лебедев**

Учебно-методическое пособие знакомит читателя с основными приемами создания фотореалистичного изображения при 3D визуализации в режиме реального времени. Для усвоения представленного материала пособие снабжено иллюстрациями, подготовленными сотрудниками НИИ ПМК ННГУ.

Разработка соответствует тематике учебно-научного инновационного комплекса УНИК-3 – «теоретический анализ, построение математических моделей, алгоритмов и программного обеспечения, разработка конкретных методик для наукоемких отраслей экономики», который проводится в рамках приоритетного направления развития ННГУ как национального исследовательского университета: «Информационно-телекоммуникационные системы: физические и химические основы, перспективные материалы и технологии, математическое обеспечение и применение».

Предназначено для студентов факультета вычислительной математики и кибернетики ННГУ, специализирующихся в области информационных систем и технологий.

Ответственный за выпуск:
Председатель методической комиссии факультета ВМК ННГУ
к.ф.-м.н, доцент Савельев В.П.

УДК 681.3.06(072)
ББК В185

СОДЕРЖАНИЕ

Введение	5
1. Фотореализм	7
1.1 Основные принципы фотореализма.....	7
1.2 Фотореализм городской среды.....	7
2. 3D визуализация в режиме реального времени	13
2.1 Материалы. Имитация фактуры. Текстуры карты.....	13
2.1.1 Карта цвета.....	13
2.1.2 Карта рассеивания.....	14
2.1.3 Карты зеркального отражения.....	15
2.1.4 Карта отражения.....	15
2.1.5 Карта свечения.....	16
2.1.6 Карта прозрачности.....	17
2.1.7 Карты усечения.....	18
2.1.8 Карты неровностей.....	18
2.1.8.1 <i>Displacement Mapping</i>	18
2.1.8.2 <i>Bump Mapping</i>	19
2.1.8.3 <i>Normal Mapping</i>	20
2.1.8.4 <i>Parallax Mapping</i>	21
2.2 Использование текстур.....	22
2.2.1 Фототекстуры.....	22
2.2.2 Повторяющиеся элементы.....	23
2.2.3 <i>MIP</i> -текстурирование.....	24
2.2.4 Методы фильтрации текстур.....	26
2.2.5 Текстурированный атлас.....	29
2.2.6 Формат хранения графических данных <i>DDS</i>	30
2.3 Низкополигональное моделирование.....	31
2.4 Оптимизация процесса визуализации.....	32
2.4.1 Библиотека моделей и текстур.....	32
2.4.2 Затуманивание и область видимости.....	34
2.4.3 Уровни детализации.....	34
2.5 Реалистичное освещение.....	36
2.6 Методы формирования теней в режиме реального времени.....	39
2.6.1 Проектируемая геометрия.....	40
2.6.2 Карты освещения.....	40
2.6.3 Вершинное освещение.....	41
2.6.4 Теневые объемы.....	41
2.6.5 Теневые карты.....	42
2.6.6 Теневые буферы.....	43
2.7 Воздушная перспектива.....	43
2.8 Постобработка.....	44
2.8.1 <i>Motion Blur</i>	45
2.8.2 <i>Bloom</i>	46

2.8.3 <i>Depth Of Field</i>	46
Список литературы	48

Введение

Высокореалистичная визуализация 3D модели городских ландшафтов в настоящий момент востребована во многих областях. Это архитектура и строительство, образование и культура, реклама и развлечения, прогнозирование чрезвычайных ситуаций (ЧС) и ликвидация последствий ЧС.

3D модели городского ландшафта позволяет:

- Проводить визуально ландшафтный анализ, путём встройки модели будущего объекта (здания) в модель окружающей застройки.
- Проводить презентации планируемой застройки.
- Сохранять памятники архитектуры и строительства, путём создания их электронной 3D модели.
- Проводить презентации города для привлечения инвесторов и туристов.
- Планировать социальную инфраструктуру.
- Проводить инженерные расчёты.
- Изучать возможное развитие чрезвычайных ситуаций, последствий и определение оптимального пути их ликвидации.

При визуализации в режиме реального времени у пользователя появляется возможность погрузиться в пространство модели и перемещаться в ней от первого лица. Он может с любой точки в пространстве модели под любым углом обзора рассмотреть моделируемый объект и прилегающую к нему территорию. Оценить степень их влияния друг на друга. Кроме того, важными преимуществами интерактивных 3D визуализаций являются:

- Работа в реальном масштабе времени с реальным масштабом данных. Возможность полностью погрузиться в пространство модели и перемещаться в ней от первого лица, что позволит передать пользователю наиболее полную информацию об объекте и получить опыт эксплуатации объектов ещё на этапе разработки их концепции.
- Корректировка в режиме реального времени места расположения здания, его габаритов и размеров. Данная возможность позволяет по требованию заказчика (эксперта) погрузить модель здания в указанное место и наглядно продемонстрировать этот вариант расположения с учётом особенностей ландшафта и окружающей застройки.
- Оценка освещённости территории. Интерактивное приложение позволяет оценить степень затенения окружающей застройки в разное время суток.

Фотореалистичная визуализация в режиме реального времени является одной из актуальных проблем в области компьютерной графики и подразумевает собой синтез изображения модели объекта неотличимого от его настоящей фотографии. Режим реального времени предполагает визуализацию сцен со скоростью не менее 24 кадров (изображений) в секунду. Одним из условий достижения фотореализма при визуализации объекта является высокая степень детализации его модели. Процесс визуализации таких моделей является ресурсо-

ёмким. Таким образом, высоко детализированная модель может быть очень сложна для визуализации в реальном времени. Необходимы специальные техники, позволяющие в случае визуализации в реальном времени найти компромисс между фотореалистичностью сцены и ее сложностью.

Основной задачей данного пособия является знакомство студентов с основными приемами создания фотореалистичного изображения при 3D визуализации в режиме реального времени. Для этого произведен целостный обзор методик и рекомендаций, позволяющих добиться фотореализма, который был систематизирован и представлен в доступной форме. Каждый подход сопровождается иллюстрацией примера его применения на 3D модели исторического центра Нижнего Новгорода, созданного сотрудниками лаборатории «Виртуальная реальность – Виртуальный мир» НИИ ПМК ННГУ им. Н.И. Лобачевского.

1. Фотореализм

1.1 Основные принципы фотореализма

Изображение объекта является фотореалистичным, если оно неотлично от фотографического снимка этого объекта. Для достижения данного свойства необходимо выполнение множества факторов, влияющих на достоверность изображения.

В работе «Фотореализм. Профессиональные приемы работы»[1] Биллом Флемингом были сформулированы принципы, которым следует соответствовать при создании фотореалистичного изображения:

1. Ожидания зрителей.
2. Правдоподобие изображения.
3. Беспорядок и хаос.
4. Трещины, прорехи и щели.
5. Пыль, грязь и гниль.
6. Закругленные края.
7. Толщина материала объекта.
8. Зеркальное отражение.
9. Рассеянный свет.
10. Материал поверхности.

По мнению Флеминга, изображение будет фотореалистичным при соответствии как минимум семи предложенным принципам [1]. Используем эти принципы применительно к изображениям городской среды.

1.2 Фотореализм городской среды

При моделировании городской среды существуют два главных условия достижения фотореализма. Это правдоподобность и узнаваемость 3D модели конкретного городского района по его характерным элементам (рис. 1.1).



Рис. 1.1. 3D модель городской среды

У городского жителя уже сложилось определенное видение городского пейзажа. Поэтому модель должна соответствовать ожиданиям зрителя. Расположение объектов на сцене должно быть логически верным. В частности, дом не должен стоять на дороге, схема разметки должна полностью соответствовать дислокации дорожных знаков и светофоров и т.д.

Городской пейзаж характеризуется некоторой упорядоченностью. Это проявляется в таких крупных элементах, как расположение зданий, фонарные столбы, деревья и т.д. В тоже время такие мелкие атрибуты улицы, как грязь, мусор, признаки старения, дождевые подтеки приносят определенный беспорядок в городскую среду. Высокодетализированная модель городского пейзажа, не содержащая этих элементов, как правило, выглядит чересчур идеально. Важно сочетать определенный порядок с хаосом [2]. Включение мелких элементов беспорядка в сцену нарушит монотонность и однообразность сцены и позволит добавить реализма модели городского пейзажа (рис. 1.2).



Рис. 1.2. 3D модель городского пейзажа

Признаки старения – пожалуй, главный фактор, позволяющий добиться фотореализма в сцене. Со временем основание стен часто начинает обрушаться. На стенах и тротуарной плитке появляются различного рода трещины (рис. 1.3). Древесина деформируется, усыхает, трескается, чернеет. Металлическая поверхность покрывается ржавчиной.

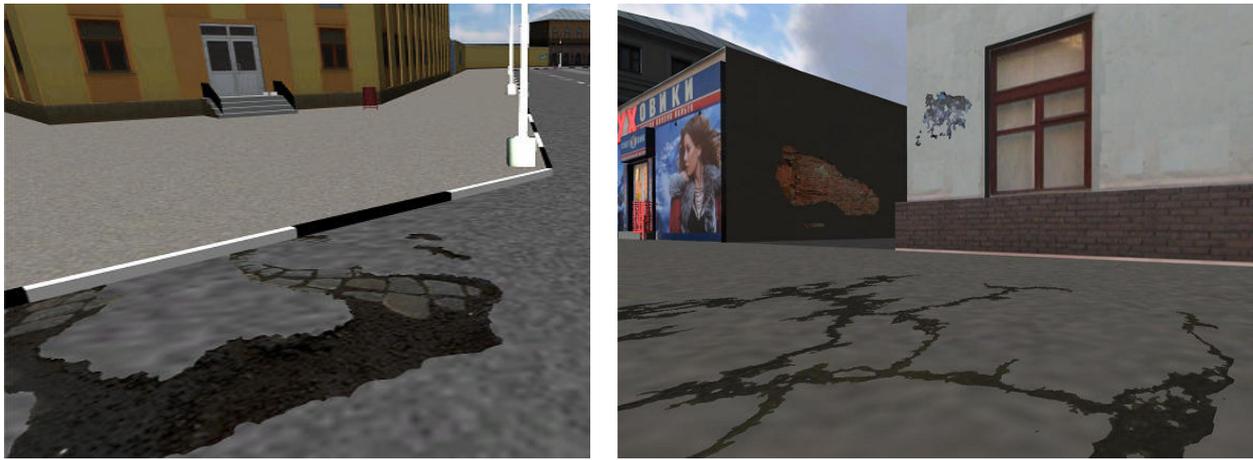


Рис. 1.3. Имитация ям и трещин на тротуарах

Краска и штукатурка на фасаде здания начинают лущиться и облезать (рис. 1.4). Из под облупившейся штукатурки начинают выглядывать кирпичи. Иногда такие места на фасаде здания пытаются подкрасить или замазать, что придает зданию неповторимый облик.



Рис. 1.4. Имитация признаков старения на фасадах зданий

Стекающая со стены грязь, краска, ржавчина образуют грязевые пятна на тротуаре. Следы мазута и машинного масла являются неизменным атрибутом асфальта, также как и тормозной след на перекрестках и поворотах (рис. 1.5).



Рис. 1.5. Добавление в сцену грязевых атрибутов

Также колорит зданию могут придать различные надписи на стенах (граффити), мох и плесень на фасаде здания, подтеки краски и побелки, остатки объявлений на столбах (рис. 1.6).



Рис. 1.6. Добавление надписей на стены зданий и сооружений

Характерной чертой городского пейзажа также является наличие различного вида мусора на улице (окурки, пачки от сигарет, пивные банки, обертки и т.д.) (рис. 1.7).

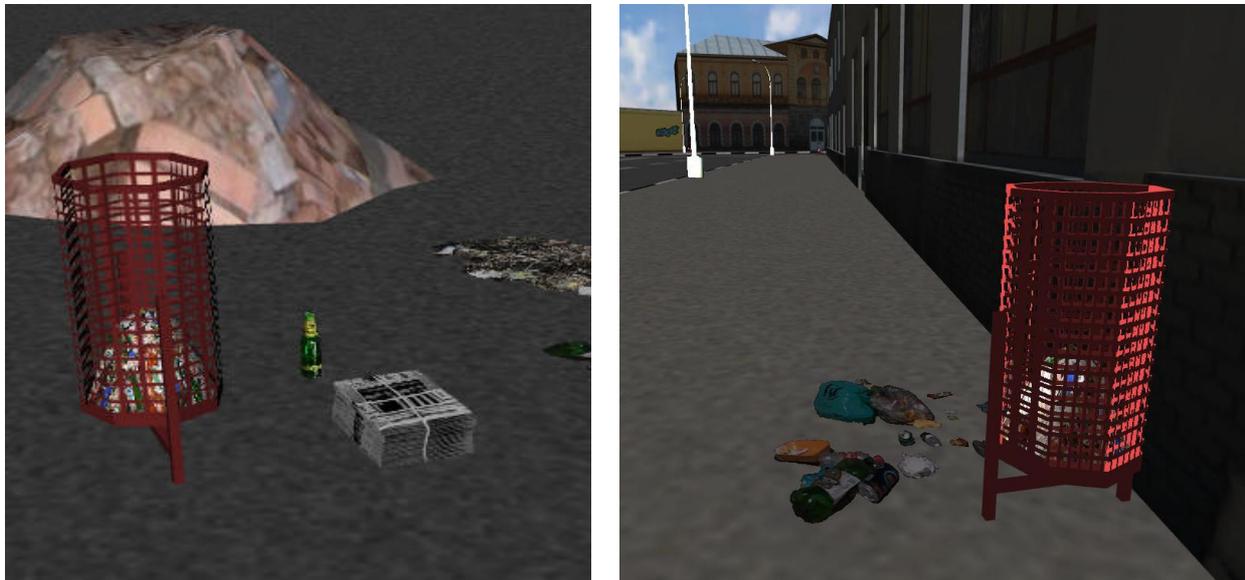


Рис. 1.7. Добавление в сцену мусора

Данные атрибуты являются не основными элементами городского пейзажа, а всего лишь «фоном». Поэтому сцена не должна быть перенасыщена этими элементами (рис. 1.8).



Рис. 1.8. Фотореалистичное изображение модели фасада здания

В реальном мире предметы практически всегда имеют скошенные края. Поскольку степень освещения поверхности зависит от направления нормали в точке, не сглаженные углы дают резкий переход уровня освещенности (рис. 1.9). Этот фактор влияет на правдоподобность изображения. Поэтому очень важно при моделировании объектов производить закругление краев.

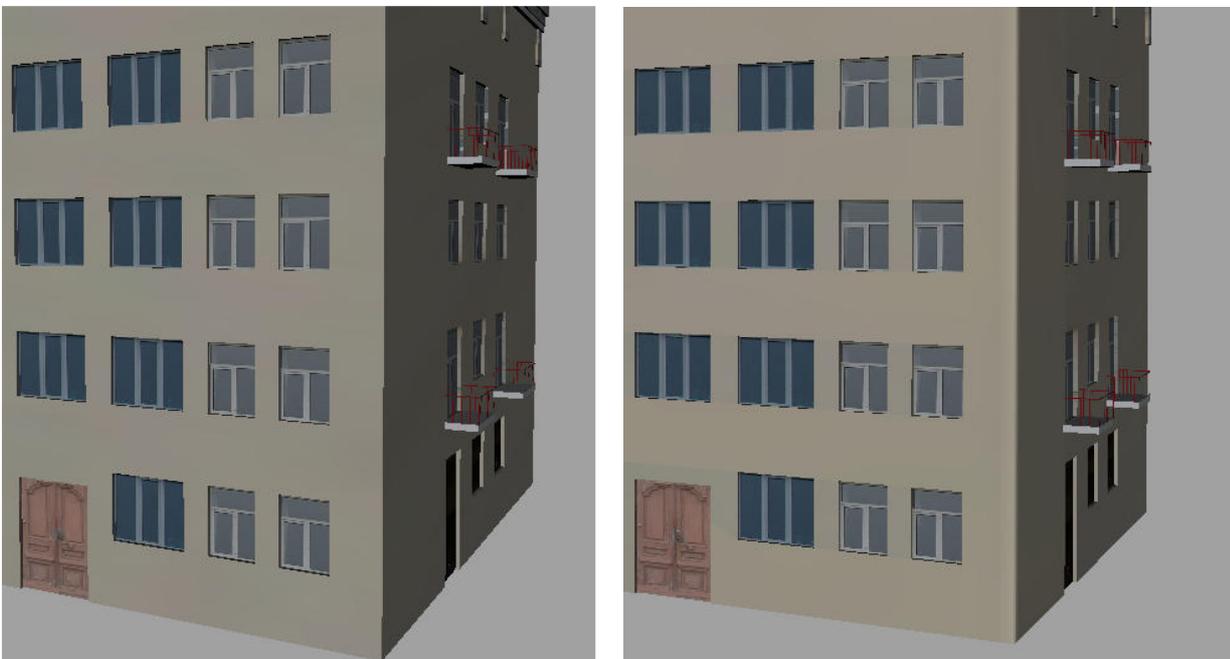


Рис. 1.9. Примеры модели здания с нескошенными и скошенными углами

Все объекты имеют толщину. Часто при моделировании тонких предметов этим атрибутом пренебрегают. В результате моделируемые предметы выглядят неестественно (рис. 1.10).



Рис. 1.10. Модели крыши здания с ярко выраженной толщиной и без неё

Свет, попадающий на поверхность объекта, частично отражается от неё. Если лучи света, падающие на поверхность, идут после отражения также по близким направлениям, такое отражение называется зеркальным. Зеркальные поверхности полностью отражают окружающее пространство. Городская среда содержит большое количество зеркальных поверхностей. Поэтому правильная имитация зеркального отражения является важнейшим элементом создания реалистичной поверхности (рис. 1.11).

Если отраженные от объекта лучи расходятся в разные стороны, отражение будет являться рассеянным. Человеческий глаз, воспринимая излучаемый свет от поверхности, определяет цвет этой поверхности. Грязь, попавшая на объект, либо трещины на поверхности объекта снижают коэффициент рассеивания. Поэтому очень важно имитировать цветовую неоднородность поверхности (рис 1.11).

Микроструктура поверхности объекта влияет на уровень зеркального и рассеянного отражения света. Поэтому степень реалистичности 3D сцены зависит от качества имитации оптических свойств материалов объектов, расположенных на этой сцене.



1.11. Имитация зеркального и рассеянного отражения света

2. 3D визуализация в режиме реального времени

Узнаваемость конкретной городской среды по её 3D модели предполагает создание высокодетализированной модели, учитывающей все характерные элементы моделируемой среды.

Довольно часто высокодетализированная модель не может быть визуализирована на современном компьютере в режиме реального времени. Поэтому всегда приходится искать компромисс между фотореалистичностью сцены и ее сложностью. Существуют несколько техник достижения этого компромисса:

1. Имитация материала.
2. Использование текстур.
3. Оптимизация геометрии модели.
4. Оптимизация процесса визуализации 3D сцены.
5. Имитация освещения. Тени.
6. Постобработка. Шейдерные эффекты.

2.1 Материалы. Имитация фактуры. Текстурные карты

Правильная имитация оптических свойств материалов поверхностей является важнейшим элементом создания реалистичных объектов. Моделирование всех деталей микроструктуры объекта весьма трудоемкий вычислительный процесс. Поэтому при визуализации в режиме реального времени имитация оптических свойств материала происходит с помощью заранее подготовленных текстурных карт. Текстурная карта (или текстура) – это обычное изображение, предназначенное для использования вместе с формой для представления объекта. Для задания свойства материала в определенной точке поверхности используется соответствующий элемент текстурной карты, называемый текселем (англ. *Texel* - аббревиатура от двух слов «*TEXTure*» и «*ELement*» - текстура и элемент).

Способность поверхности объекта к поглощению и отражению света имитируется с помощью карт диффузного отражения. Текстурные карты имитируют также такие свойства материала, как отражение (используется карта отражений), прозрачность (карта прозрачности), блики (карта бликов), яркость (карта яркости), неровности (карта неровностей) и т.п.

Приведем описание наиболее распространенных текстурных карт. Укажем области их применения и проиллюстрируем результат их использования.

2.1.1 Карта цвета

Карта цвета (англ. *Color map*) это текстура, содержащая все основные цвета, которыми обладает реальная поверхность. Цвет текселя карты определяет цвет материала в соответствующей точке (рис. 2.1). Карта цвета является ос-

новным типом текстурных карт, поскольку она является основой для других карт, имитирующих оптические свойства материала.

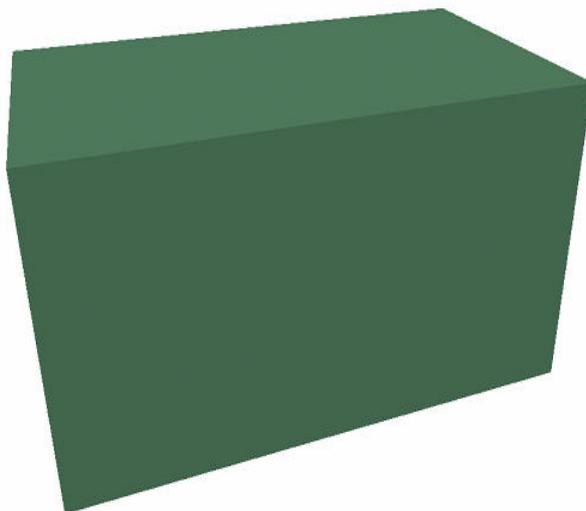


Рис. 2.1. Пример применения карты цвета

2.1.2 Карта рассеивания

Карты рассеивания (англ. *Diffuse map*) регулируют количества света, рассеивающегося на поверхности [3]. От уровня рассеивания света поверхностью зависит цвет этой поверхности, воспринимаемый наблюдателем. Различные материалы могут по-разному рассеивать свет. Например, глянцевые или пластиковые поверхности имеют высокий коэффициент рассеивания, а у металлических или зеркальных поверхностей он стремится к нулю. Также необходимо помнить, что даже на поверхности одного материала различные области могут иметь разный коэффициент рассеивания. Например, грязь, попавшая на объект, либо трещины на поверхности объекта снижают коэффициент рассеивания.



Рис. 2.2. Карта рассеивания и результат её применения к примитиву (карта рассеивания взята с сайта <http://walnuss.deviantart.com>)

Для того чтобы задать уровень рассеивания света на различных участках поверхности используют карты рассеивания. Карта рассеивания представляет собой текстуру, каждый тексель которой задает степень рассеивания в соответствующей точке поверхности (рис. 2.2) с помощью градаций серого цвета. Более темные тексели карты соответствуют низкому коэффициенту рассеивания, а более светлые – высокому. Таким образом, с помощью карт рассеивания имитируется неоднородность поверхности. Часто такие карты используются для создания эффекта наличия загрязнения на поверхности объекта. На рисунке 2.2 показан результат наложения карты рассеивания на примитив из рисунка 2.1.

2.1.3 Карты зеркального отражения

При зеркальном отражении света отраженные лучи не рассеиваются, а собираются в одну точку. Таким образом, свет, испускаемый от окружающих зеркальную поверхность предметов, отражается в глаз наблюдателю, передавая информацию об их цвете. Карты зеркального отражения (англ. *Reflection map*) позволяют имитировать отражение окружающих объектов на поверхности материала (рис. 2.3). Они представляют собой текстуру, содержащую изображение отражаемых предметов. Поскольку отражение в этом случае остается неизменным, такой подход приемлем только для статичных объектов.

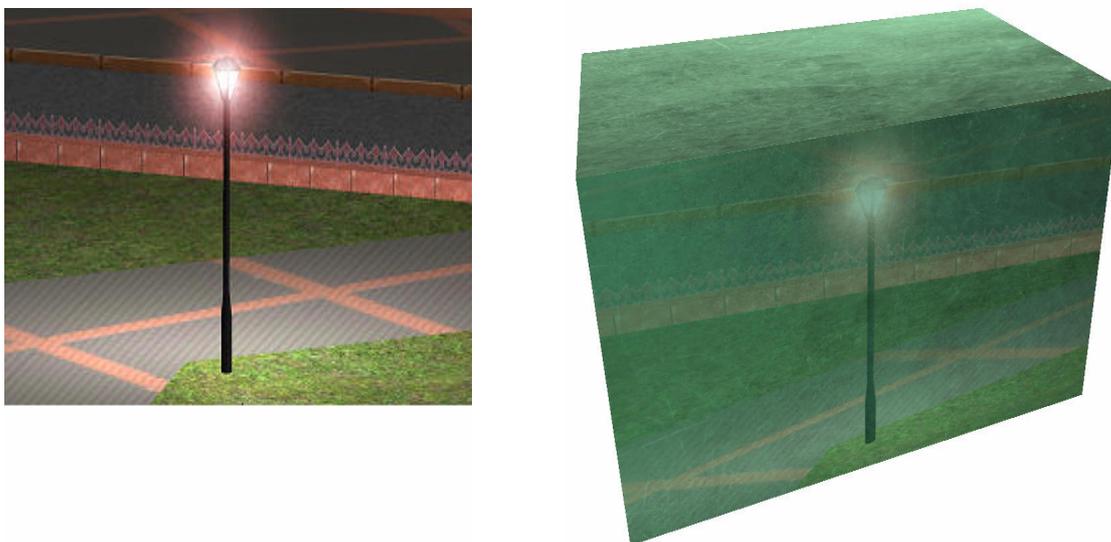


Рис. 2.3. Карта зеркального отражения и результат её применения к примитиву

2.1.4 Карта отражения

Степень отражения поверхности зависит от типа материала поверхности. Абсолютное отражение имеют зеркала или хромированные материалы. Такие материалы как стекло, пластик, окрашенные металлы, мрамор, керамика имеют высокий уровень отражения. К не отражающим материалам можно отнести

ткани, дерево, бетон, резина и т.д. Поэтому части одной поверхности, покрытые разными материалами, будут иметь разную отражательную способность (рис. 2.4).

Карта отражения (англ. *Specular Map*) представляет собой текстуру, каждый тексель которой задает степень отражения в соответствующей точке поверхности с помощью градаций серого цвета (рис. 2.4). Более темные тексели карты соответствуют низкому коэффициенту отражения, а более светлые – высокому.

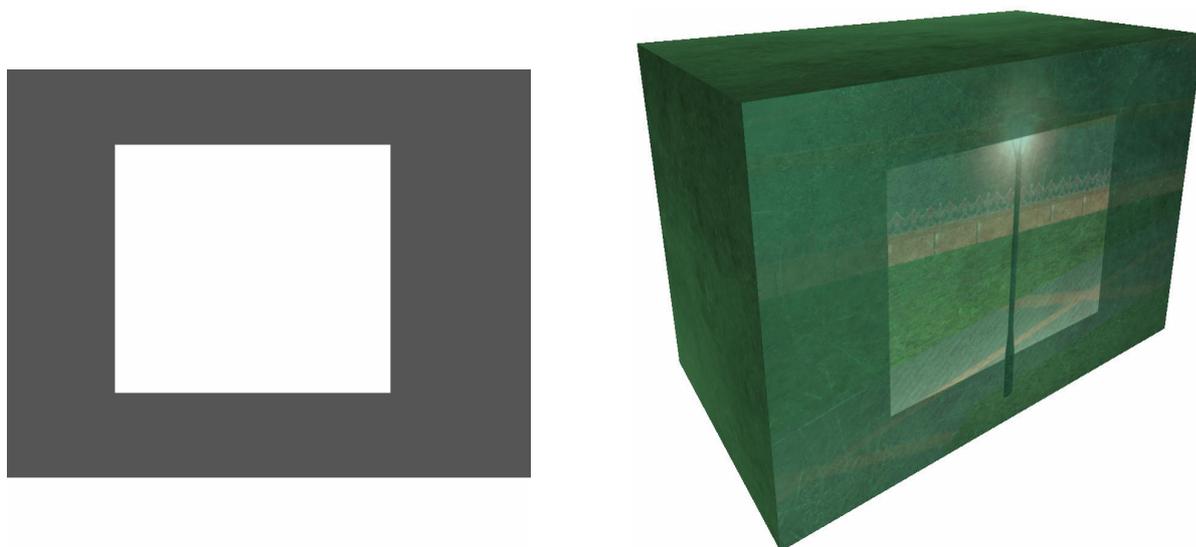


Рис. 2.4. Карта отражения и результат её применения к примитиву

Следует помнить, что в реальной жизни отражение от поверхности, покрытой одним материалом, выглядит не однородно. На поверхности всегда найдутся такие элементы, как пыль, грязь, ржавчина, которые влияют на её отражательную способность. Поэтому при имитации отражения следует немного изменять уровни отражательной способности в карте отражения вдоль всей поверхности.

2.1.5 Карта свечения

Карта свечения (англ. *Self illumination map*) позволяет задавать уровень свечения материала поверхности. Таким образом, с помощью такого эффекта можно имитировать, например освещённость окон в ночное время. Карта свечения представляет собой текстуру, каждый тексель которой задает плотность светового потока испускающегося из соответствующей точки поверхности. Уровень свечения задается с помощью градаций серого цвета. На рисунке 2.5 показан результат наложения карты свечения на примитив из рисунка 2.2.

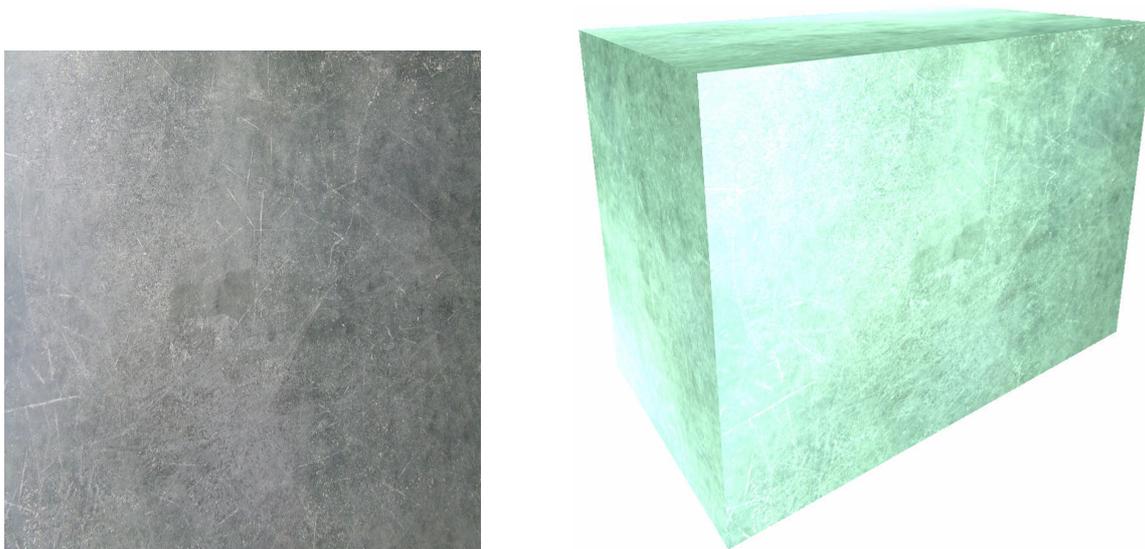


Рис. 2.5. Карта свечения и результат её применения к примитиву

2.1.6 Карта прозрачности

Карта прозрачности (англ. *Transparency/Opacity Map*) определяет количество света, проходящего сквозь поверхность [3]. Карта прозрачности представляет собой текстуру, каждый тексель которой задает прозрачность соответствующей точки поверхности. Степень прозрачности задается с помощью градаций серого цвета. На рисунке 2.6 показан результат наложения карты прозрачности на примитив из рисунка 2.2.

Карта прозрачности используется для имитации полупрозрачных элементов.

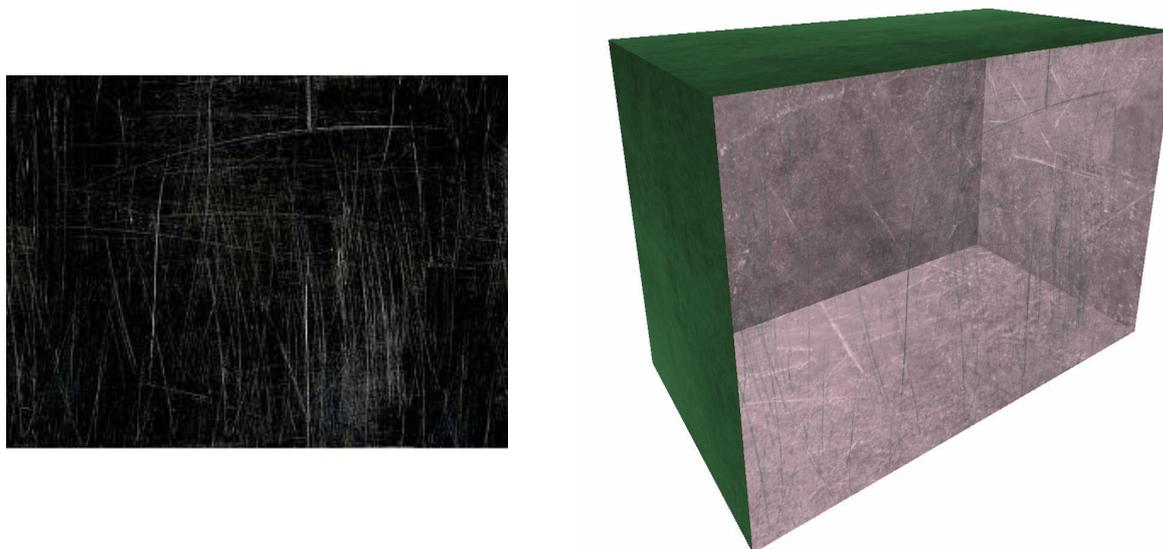


Рис. 2.6. Карта прозрачности и результат её применения к примитиву (карта прозрачности взята с сайта <http://walnuss.deviantart.com>)

2.1.7 Карты усечения

Карта усечения (англ. *Clip image maps*) определяют области полной прозрачности и непрозрачности объектов. Такие карты состоят из двухцветного черно-белого изображения, белые области которого представляют собой полностью непрозрачные зоны, а чёрные – полностью прозрачные.

Точки поверхности, соответствующие черным текселям в карте усечения не отображаются на изображении (рис.2.7).

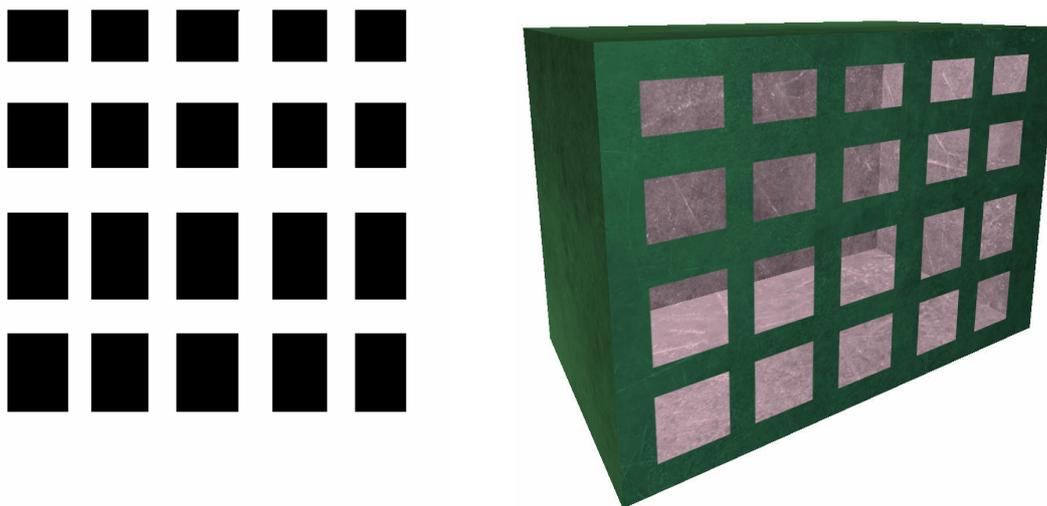


Рис. 2.7. Карта усечения и результат её применения к примитиву

При моделировании городской среды карты усечения обычно используют для имитации сложного контура у плоских объектов. Например, элементы кустарника, листва деревьев, решетки, заборы и т.д.

2.1.8 Карты неровностей

Количество треугольников, обрабатываемых в режиме реального времени, ограничено. Поэтому существуют подходы, позволяющие использовать карты неровностей для снижения геометрии в сложных сценах.

Наиболее известные методы симуляции неровностей:

- 1) *Displacement Mapping*
- 2) *Bump Mapping*
- 3) *Normal Mapping*
- 4) *Parallax Mapping*

2.1.8.1 *Displacement Mapping*

Данный метод изменяет геометрию поверхности, добавляя деталей низкополигональным объектам. Алгоритм работает следующим образом. Низкопо-

лигональная поверхность разбивается на регулярную сеть точек. Каждая точка поверхности сдвигается по нормали на величину, исходя из значений в картах смещения. Карта смещения (англ. *displacement map*) представляет собой текстуру, каждый тексель которой задает величину смещения соответствующей точки поверхности с помощью градаций серого цвета (рис. 2.8). Наиболее часто карты смещения используются для создания рельефа земной поверхности.

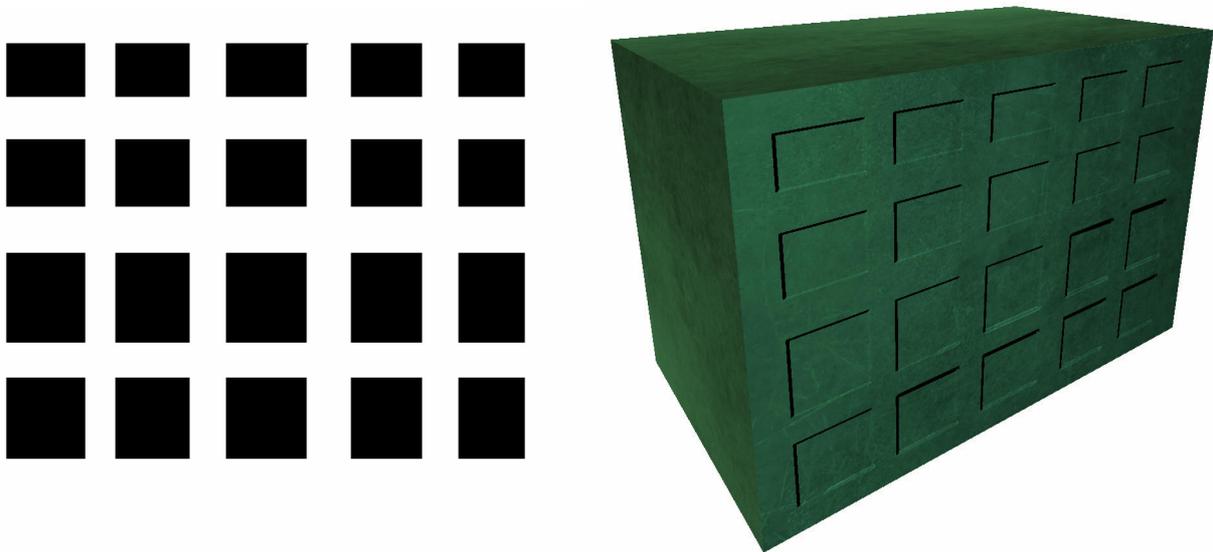


Рис. 2.8. Карта смещения и результат её применения к примитиву

2.1.8.2 *Bump Mapping*

В отличие от *Displacement Mapping* подход, использующий *Bump Mapping* (бампмаппинг), имитирует неровности не добавляя геометрии в сцену.

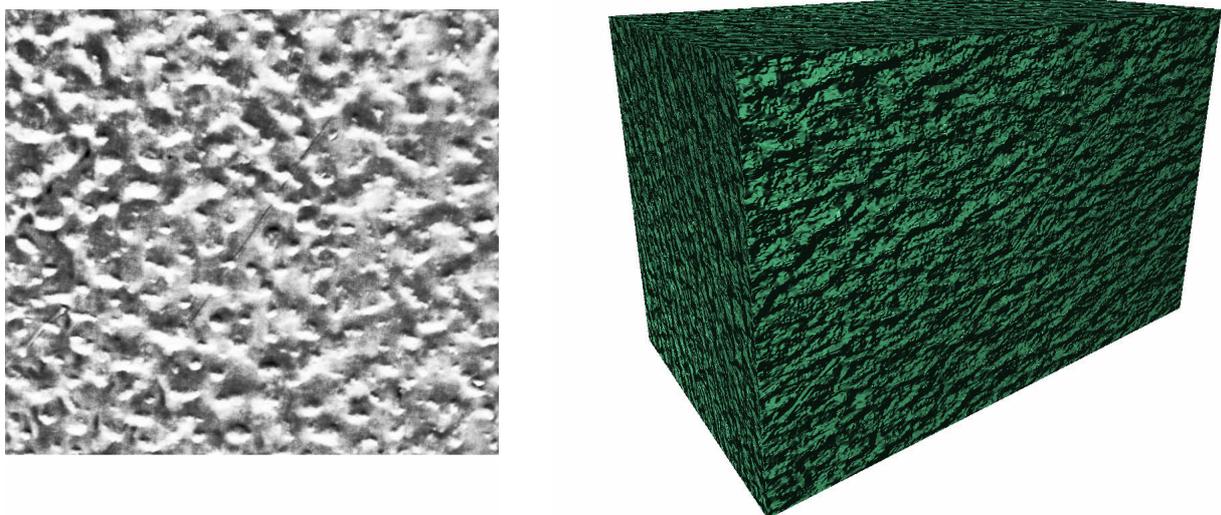


Рис. 2.9. Карта высот и результат её применения к примитиву

Степень освещенности поверхности зависит от угла падения света. На неровной поверхности степень освещенности в каждой точке будет разной. Именно по признаку освещенности человеческий мозг определяет степень неровности поверхности. Поэтому, устанавливая каждой точке поверхности уровень её освещенности, можно обмануть человеческий мозг и создать иллюзию неровности.

Вычисление освещения точки поверхности выполняется исходя из значений в специальной карте высот, называемой *bump map*. Данная карта высот представляет собой текстуру, каждый тексель которой задает высоту соответствующей точке поверхности с помощью градаций серого цвета (рис. 2.9).

С помощью техники *Bump Mapping* можно имитировать мелкие неровности и шероховатости поверхности без увеличения геометрии модели. Например, каменные и кирпичные стены, кафельную плитку, поры на коже и т.д.

2.1.8.3 Normal Mapping

Существует более усовершенствованная разновидность техники *Bump Mapping*. В этом случае имитация неровности также производится с помощью установки различных значений освещенности. Для вычисления освещенности в точке необходимо знать угол падения света в эту точку. В свою очередь, чтобы вычислить такой угол необходимо знать направление нормали поверхности в точке. Значение нормали содержится в специальной карте нормалей (англ. *Normal map*). Эта карта представляет собой цветную текстуру, каждый тексель которой хранит нормаль в соответствующей точке поверхности (рис. 2.10). Тексель хранит цветовые составляющие $[r, g, b]$, интерпретирующиеся в оси вектора $[x, y, z]$, на основе которого вычисляется нормаль.

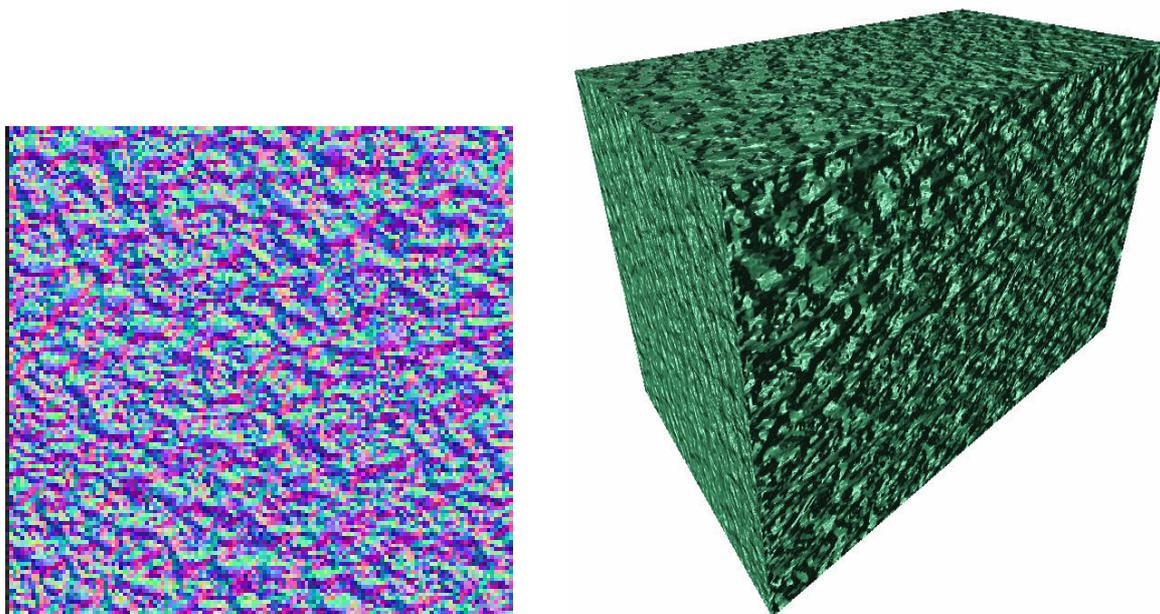


Рис. 2.10. Карта нормалей и результат её применения к примитиву

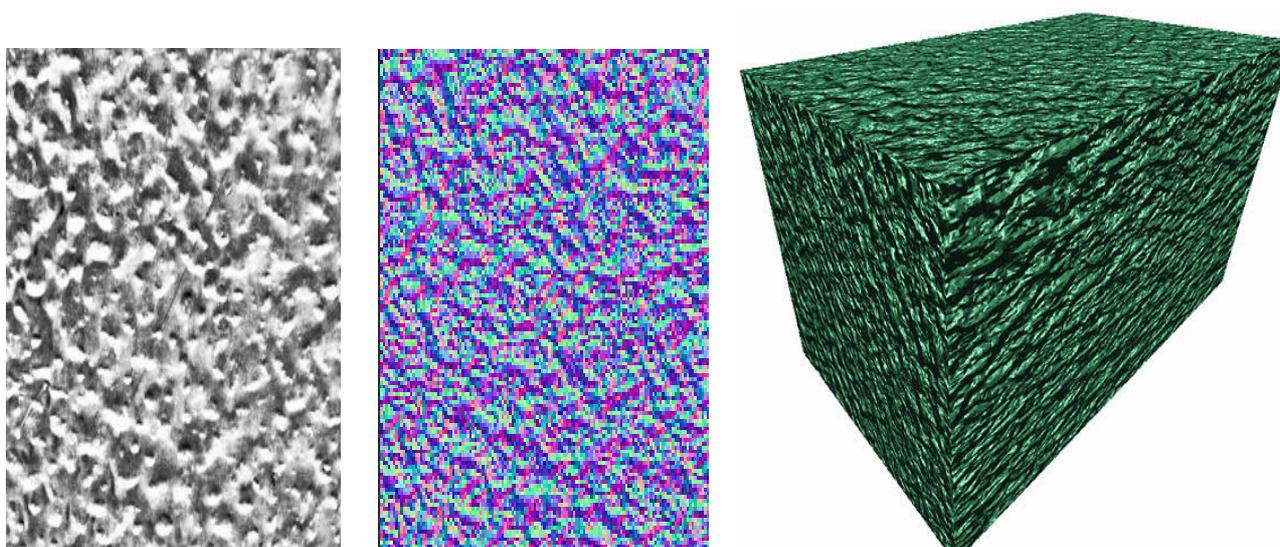
Техника, использующая для имитации неровностей карту нормалей, называется *Normal Mapping* (нормалмаппинг). Такой подход существенно точнее классического *Bump Mapping*, поскольку в нем задействованы сразу три канала текстуры вместо одного (рис. 2.10).

Карты нормалей обычно создаются на основе разницы между низкополигональной и высокополигональной версий модели поверхности.

2.1.8.4 *Parallax Mapping*

Такие технологии как *Bump Mapping* и *Normal Mapping* не могут корректно имитировать неровности на крайних полигонах объекта и больших углах наклона поверхности. Этот недостаток хорошо заметен на рисунке 2.9 на верхней и боковой сторонах куба.

Технология *Parallax Mapping* (параллакспаппинг) для имитации неровностей использует не только операции с освещением, но и смещением текстурных координат поверхности. В этом случае используются и карта нормалей и карта высот (рис. 2.11). Идея метода состоит в том, чтобы возвращать текстурные координаты той точки, где видовой вектор пересекает поверхность [4]. Тем самым положение текстуры меняется в зависимости от угла зрения, имитируя неровность поверхности.



2.11. Технология *Parallax Mapping* для имитации неровностей

Использование карт неровностей позволяет либо оптимизировать визуализируемую модель, уменьшая количество полигонов поверхности, либо добавить реализма существующей модели, имитируя шероховатость и неровность поверхности. Главный недостаток такого подхода – существенные требования к видеопамяти и её пропускной способности.

2.2 Использование текстур

Выполнение принципов фотореализма резко увеличивает детализацию моделируемого объекта. В тоже время, как следует из предыдущего раздела, определенные детали объекта можно передать не с помощью моделируемой геометрии, а текстурированием посредством текстурных карт.

Наложение текстуры или текстурирование (англ. *texture mapping*) – это метод, посредством которого на поверхность объекта накладывается некоторое изображение, называемое изображением текстуры [5].

2.2.1 Фототекстуры

Для задания цвета поверхности моделируемого объекта используется карта цвета. В качестве такой карты могут выступать как текстура, нарисованная вручную, так и текстура, изготовленная из фотографии этого объекта. Такая текстура, называемая фототекстурой, способна добавить реализма моделируемому объекту, поскольку кроме характерных деталей поверхности она передает признаки старения (пыль грязь, гниль, ржавчина) и дефекты поверхности объектов, составляющих сцену (трещины, прорехи, щели). Важность использования фототекстур при моделировании наглядно продемонстрирована на рисунке 2.12.

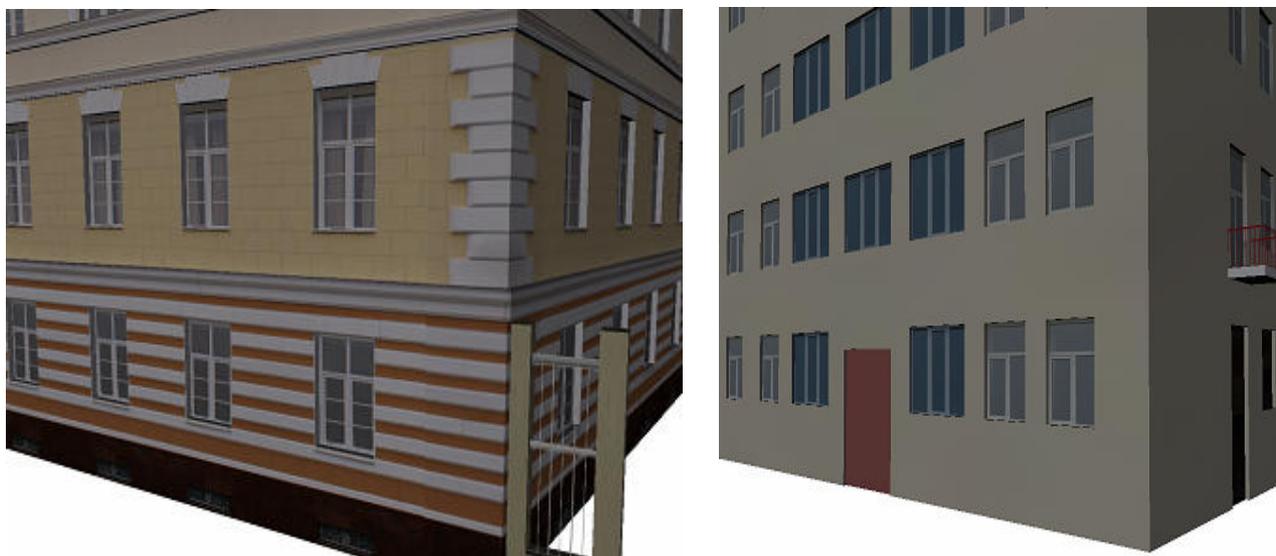


Рис. 2.12. Модель фасада здания с использованием фототекстур и без них

При подготовке фототекстуры исходные фотографии не должны содержать информации о направлении света и не иметь затененных участков. Поэтому процесс съемки городского пейзажа должен производиться в пасмурную погоду, после чего необходимо произвести цветокоррекцию полученного изображения.

Поскольку объем памяти (графической или оперативной) для хранения текстур ограничен, необходимо выработать единый подход к разрешению текстур. Оптимальным вариантом при моделировании городской среды является подход, когда на 1 см² поверхности приходится 1 пиксель покрывающей ее текстуры. Такой принцип позволяет не только добиться оптимального разрешения текстур, но и соблюсти единый размер изображения на всех объектах сцены. Это важный элемент достижения правдоподобности изображений.

2.2.2 Повторяющиеся элементы

Городской пейзаж изобилует повторяющимися элементами: зеркальная симметрия фасадов зданий, повторяющиеся окна, колонны, кирпичная кладка, тротуарная плитка и т.д. При моделировании таких объектов создается только один элемент повторения, который впоследствии дублируется для остальных повторяющихся частей объекта (рис. 2.13).

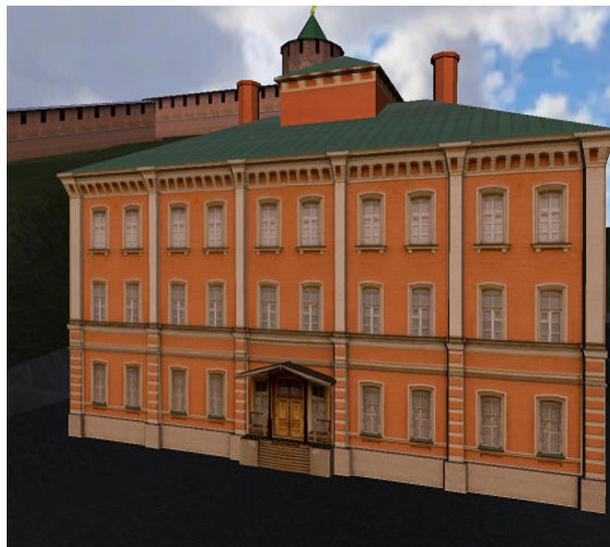


Рис. 2.13. Использование зеркальной симметрии в моделях фасадов зданий

Если в качестве повторяющегося элемента поверхности используется текстура – такой процесс называется *тайлингом* (англ. *tiling*). В этом случае повторяющаяся текстура должна быть бесшовной, т.е. не содержать в себе элементов образующих стыки в момент *тайлинга* (рис. 2.14). Такой подход позволяет уменьшить трудоемкость создания 3D модели и существенно сократить объем используемой памяти для её хранения и визуализации [6].



Рис. 2.14. Бесшовная текстура и пример её использования на модели фасада здания

2.2.3 MIP-текстурирование

Качество изображения текстуры 3D объекта находится в прямой зависимости от её разрешения, габаритов объекта, разрешения экрана и расположения объекта на сцене. Если текстура имеет высокое разрешение, а размеры объекта по сравнению с областью видимости малы, на экране могут появиться визуальные искажения. Данные искажения проявляются в виде мерцания и дополнительных узоров на поверхности текстуры. Вызвано это тем, что при отдалении объекта от точки обзора или его уменьшении одному пикселю на экране может соответствовать несколько пикселей текстуры. Убрать визуальные искажения можно путем усреднения близлежащих пикселей. Однако реализация этой процедуры потребует больших вычислительных затрат, что препятствует ее выполнению непосредственно в процессе визуализации. Проблему можно решить путем создания и хранения дополнительных текстур, в которых значения будут усреднены заранее. В этом случае значение пикселя на этапе визуализации определяется путем выбора из соответствующей текстуры.

Метод текстурирования, использующий несколько копий одной текстуры с разной детализацией называется *MIP-текстурированием* [7]. Термин *MIP-текстурирование* (*MIP mapping*) был предложен Ланцем Вильямсом (*Lance Williams*) в своем докладе «*Pyramidal Parametrics*» на конференции *SIGGRAPH* в 1983 году. Буквы *MIP* являются аббревиатурой латинской фразы «*multum in parvo*», означающего «множество в малом».

Каждая текстура из текстурного набора, используемого при *MIP-текстурировании*, называется *MIP-уровнем* (от англ. *MIP level*) и вместе они составляют *MIP-пирамиду*. *MIP-пирамида* заносится в память видеоадаптера. Основу *MIP-пирамиды* составляет исходное изображение. Каждый следующий

MIP-уровень этой пирамиды представляет собой изображение вдвое меньшее, чем предыдущее (рис. 2.15). В зависимости от положения объекта и разрешения экрана на объект накладывается текстура одного из уровней этой пирамиды. То есть разрешение текстуры, накладываемой на объект, находится в прямой зависимости от расстояния до этого объекта и разрешения экрана. Таким образом, изображение остается качественным при приближении к объекту и при удалении от него, используя минимальное количество вычислений для этого.

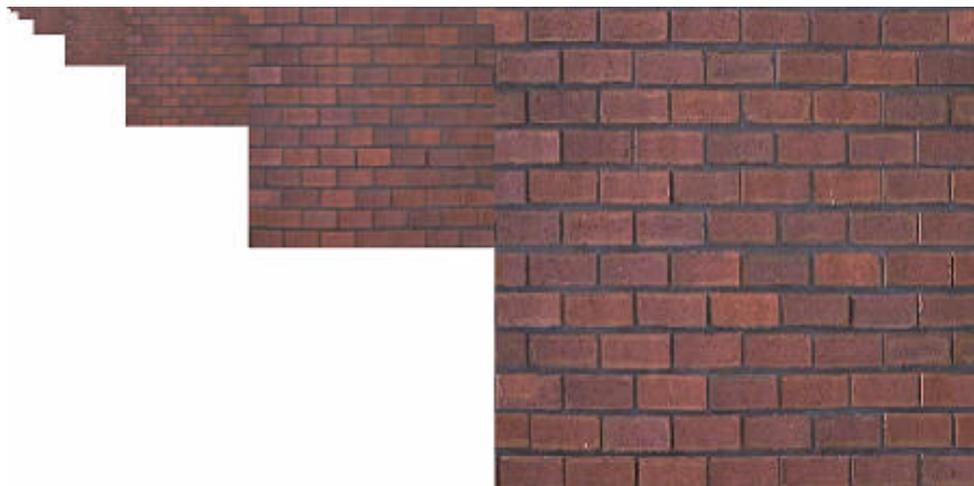


Рис. 2.15. *MIP*-пирамида

Создание *MIP-уровней* может производиться несколькими способами. Один из способов предполагает создание *MIP-уровней* вручную. Качество такого подхода полностью зависит от мастерства художника готовившего текстуры. Другой вариант полностью автоматический и предполагает генерацию *MIP-уровней* с помощью графических акселераторов в момент, когда исходные текстуры загружаются в видеопамять. Последний способ предполагает создание *MIP-уровней* с помощью специальных программных комплексов для редактирования изображений. Современные редакторы изображений имеют наборы различных фильтров улучшающих качество изображения. Использование этих фильтров в зависимости от типа изображения и их комбинация может дать наилучший результат [8].

Размеры *MIP-уровней* составляющих *MIP-пирамиду* представляют собой убывающую геометрическую прогрессию со знаменателем $1/4$. Таким образом, общий объем видеопамати для хранения *MIP-пирамиды* увеличивается на треть по сравнению с объемом, занимаемым исходной текстурой.

Существуют два подхода *MIP-текстурирования*: *per-polygon mip-mapping* (*MIP-текстурирование* по каждому полигону) и *per-pixel mip-mapping* (попиксельное *MIP-текстурирование*). В методе *per-pixel mip-mapping* *MIP-уровень* вычисляться для каждого пикселя в полигоне. Это наиболее точный вид *MIP-текстурирования*, но он отличается большой трудоемкостью. При использовании *per-polygon mip-mapping* *MIP-уровень* вычисляется один раз для всего полигона. Такой подход обладает значительно более высокой скоростью, но качество изображения может быть ухудшено появлением размытых полигонов либо

разрывов по краям полигонов. Это вызвано тем, что к соседним полигонам могут быть применены различные *MIP-уровни* [9].

Для того чтобы при *MIP-текстурировании* сделать незаметным момент перехода между различными разрешениями текстуры используют фильтрацию текстур.

2.2.4 Методы фильтрации текстур

В процессе визуализации 3D сцены для каждого пикселя экрана определяются все области сцены, которые будут спроектированы в него. Таким образом, определяются все тексели текстуры, попавшие в данный пиксель экрана. Для получения окончательного цвета пикселя экрана происходит выбор и комбинация найденных текселов текстуры. Самый простой метод выбора называется *сэмплингом* (англ. *point-sampling*). Здесь цвет конечного пикселя определяется путем выбора цвета ближайшего текселя. Серьезным недостатком такого метода является появление эффекта «блочности». Вызвано это тем, что в момент приближения наблюдателя к текстуре размер текселов станет больше размера пикселей, в результате чего возникает пикселизация (образование «квадратов») изображения (рис. 2.16).



Рис. 2.16. Формирование изображения без фильтрации текстур



Рис. 2.17. Формирование изображения с использованием билинейной фильтрации

Также при перемещении или повороте текстурированных объектов могут возникнуть искажения сформированного на экране изображения в виде «перескакивания» пикселей. Фильтрация текстур призвана устранить эти искажения.

Билинейная фильтрация (англ. *Bilinear Filtering*) – процесс извлечения нескольких текселей исходной текстуры с последующим усреднением их значений для получения окончательного значения пикселя [10]. Этот вариант фильтрации является по сути усовершенствованным *сэмплингом* с той лишь разницей, что для окончательного определения цвета пикселя используется усредненное значение четырех близлежащих текселей вместо одного (рис. 2.17).

Однако использование билинейной фильтрации не может избавить от артефактов, возникающих в результате ошибок определения глубины сцены. Класс этих артефактов визуализации называется *depth-алиасинг* (англ. *depth aliasing* или *Z-aliasing*). В случае, когда объект отдаляется от наблюдателя, его изображение уменьшается в размерах. Соответственно увеличивается область текстурного изображения объекта, которая будет спроецирована в один пиксель экрана. Простое усреднение четырех близлежащих текселей может дать неправильную интерполяцию данной области и вызвать мерцание результирующего изображения на экране. Для избежания ошибок *depth-алиасинга* используется описанная ранее техника *MIP-текстурирования*. В свою очередь использование *MIP-текстурирования* может привести к появлению других артефактов под названием *MIP-бендинг* (англ. *MIP-banding*). Возникают они в момент перехода от одного *MIP*-уровня к другому и выражаются в ярко выраженных границах перехода между ними. Особенно этот артефакт заметен на протяженных объектах устремленных вглубь сцены (рис. 2.18).

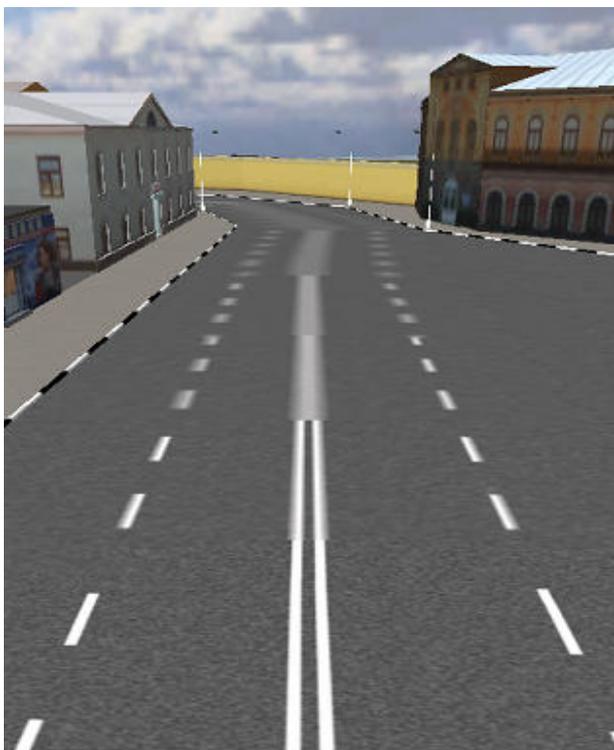


Рис. 2.18. Появление ошибок визуализации “*mip-бендинг*” на протяженных объектах



Рис. 2.19. Формирование изображения с использованием трилинейной фильтрации

Трилинейная фильтрация (англ. *Trilinear filtering*) позволяет исправить этот недостаток *MIP-текстурирования* с помощью некоторого снижения резко-

сти текстур. Трилинейная фильтрация представляет собой усовершенствованный вариант билинейной фильтрации. В этом случае для определения цвета пикселя выбираются два соседних *MIP*-уровня в зависимости от удаленности объекта. На каждом уровне к области, которая будет спроектирована в этот пиксель, применяется билинейная фильтрация. И дополнительно происходит фильтрация между соседними *MIP*-уровнями (рис. 2.19).

Трилинейная фильтрация дает качественные результаты и широко используется на практике. Недостаток такой фильтрации состоит в том, что она корректно работает только с плоскими текстурированными поверхностями. Если текстурированная поверхность находится под острым углом к наблюдателю, будет заметна размытость текстуры (рис. 2.19). Вызвано это неправильным выбором текселей для обработки. Во-первых, на наклонных поверхностях существенно увеличивается область текстурного изображения объекта, которая будет спроецирована в один пиксель экрана. А при трилинейной фильтрации количество текселей остается неизменным. Во-вторых, выбранные тексели всегда образуют квадрат, а форма области текстурной поверхности спроецированной в один пиксель экрана меняется в зависимости от угла наклона поверхности. Тем самым может случиться ситуация, когда степень фильтрации в одном направлении существенно выше, чем в другом. Решить эту проблему призвана анизотропная фильтрация (рис 2.20).

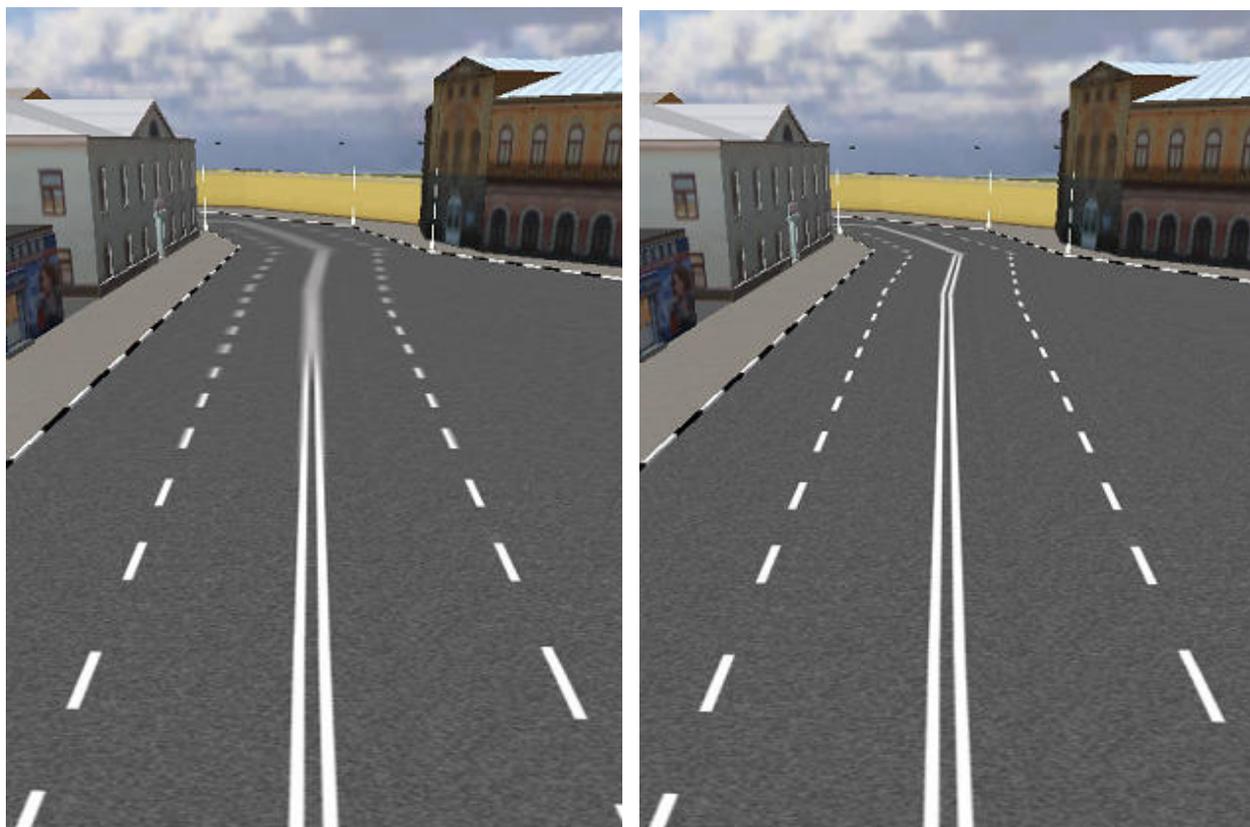


Рис. 2.20. Формирование изображения с использованием анизотропной фильтрации с уровнями фильтрации $2x$ и $8x$

Анизотропная фильтрация (англ. *anisotropic filtering*) – метод повышения качества изображения, при котором форма области из текстелей, используемых для получения окончательного цвета пикселя, меняется в зависимости от обстоятельств. В частности, при определении формы анализируемой области текстелей принимается во внимание расположение текстурируемого полигона в трехмерном пространстве относительно наблюдателя. Цвет пикселя в этом случае определяется наиболее корректно. Количество текстелей, используемых для определения цвета пикселя, тоже может варьироваться. Это значение называется уровнем фильтрации. На современных графических картах уровни фильтрации могут выставляться вручную. Распространены следующие варианты выбора уровня фильтрации: *1x* (используется 8 текстелей), *2x* (16 текстелей), *4x* (32 текстеля), *8x* (64 текстеля) и *16x* (128 текстелей) (рис. 2.20). Таким образом, анизотропная фильтрация позволяет сохранить большую детальность изображения. Однако это преимущество требует больших вычислительных затрат и может сильно замедлить работу графического процессора.

2.2.5 Текстурный атлас

Текстурный атлас – это большое изображение, включающее в себя множество изображений меньшего размера, каждое из которых является текстурой для некоторой части 3D объекта [11] (рис. 2.21). Формирование текстурных атласов обусловлено особенностью архитектуры графического процессора. Такой подход позволяет оптимизировать процесс формирования 3D изображения на экране пользователя.



Рис. 2.21. Текстурный атлас

Использование текстурного атласа дает следующие преимущества:

- значительное сокращение количества обращений к видеокарте в момент помещения текстур в видеопамять;
- возможность размещения в себе текстур, размеры которых не кратны степени двойки;
- использование одного текстурного слота, количество которых ограничено;
- минимизация степени фрагментации видеопамати.

Несмотря на эффективность использования текстурного атласа, работа с ним вызывает ряд проблем, к числу которых относятся:

- трудоемкость ручного создания текстурных атласов;
- не оптимальное размещение текстур в текстурном атласе, что может привести к лишнему расходу видеопамати;
- сложность в использовании *MIP*-текстурирования;
- отсутствие отступов между текстурами, что может привести к их смешиванию при использовании различных алгоритмов фильтрации;
- осложнение процесса совмещения краев текстур, содержащихся в текстурном атласе.

2.2.6 Формат хранения графических данных *DDS*

Наиболее часто для хранения текстур в 3D приложениях используется формат *DDS (Direct Draw Surface)*, разработанный компанией *Microsoft*.

Особенностью формата *DDS* являются:

- возможность хранения *MIP-уровней* текстуры;
- возможность хранения альфа-канала в изображении;
- поддержка, как обычных двумерных изображений, так и трехмерных, и кубических текстур;
- поддержка *DXT* сжатия (также известного как сжатие *S3TC*).

Хранение *MIP-уровней* текстуры в файле имеет следующие преимущества:

- *MIP-уровни* создаются заранее, а не генерируются автоматически в процессе работы 3D приложения, что существенно ускоряет работу 3D приложения.
- *MIP-уровни* могут создаваться вручную, что повышает качество изображения.

Альфа-канал это дополнительный канал, который может быть добавлен в изображение, чтобы хранить информацию о прозрачности в каждом пикселе. По своему назначению и использованию альфа-канал аналогичен картам усечения описанным ранее.

Формат *DDS* может содержать как сжатые, так и несжатые текстуры. Сжатие текстур происходит в формате *S3TC (DXT1, DXT2, DXT3, DXT4 и DXT5)*. Такой формат сжатия поддерживается современными видеокартами на

аппаратном уровне. Тем самым экономится время, необходимое на декомпрессию сжатых текстур. Сжатие в формате *DXT1* производится для изображений с однобитным альфа-каналом или без него. Коэффициент сжатия при использовании такого формата фиксирован и равен 8:1. Форматы *DXT3* и *DXT5* используются для изображений, в которых значение альфа-канала не однородно. *DXT5* обычно гораздо точнее сохраняет альфа-канал после сжатия и рекомендуется вместо *DXT3*. Коэффициент сжатия в этом случае равен 4:1. Форматы *DXT2* и *DXT4* не используются. Их принцип сжатия во многом аналогичен форматам *DXT3* и *DXT5* соответственно. Основным недостатком форматов *S3TC* заключается в том, что сжатие происходит с потерями. Но потери качества на большинстве изображений не столь значительны и ими можно пренебречь.

2.3 Низкополигональное моделирование

Геометрия 3D модели не может сравниться с геометрией и микроструктурой объектов из реального мира. Существует два вида представления модели объекта: низкополигональное и высокополигональное. Высокополигональная модель содержит такое количество полигонов, которое позволяет сделать модель максимально приближенной к оригиналу. В тоже время обработка большого количества полигонов в сцене требует максимальной мощности процессора, видеокарты и прочих ресурсов компьютера. Поэтому необходим подход, при котором приемлемое качество визуализации достигалось меньшими затратами, путем использования методов компьютерной графики.

Чётких критериев определения низкополигональности не существует. Поэтому низкополигональная модель содержит такое количество полигонов, которое достаточно для визуального восприятия объекта. Иными словами низкополигональная модель содержит минимальное количество полигонов при сохранении требуемой геометрической сложности (рис. 2.22). Использование низкополигонального моделирования резко снижает количество полигонов в моделируемой сцене, тем самым являясь идеальным решением при 3D визуализации в режиме реального времени. Главным недостатком такого подхода являются ослабленные требования к степени реализма моделируемого объекта. Определенные детали объекта в этом случае передаются не с помощью моделируемой геометрии, а имитируются при помощи текстурных карт.

Низкополигональное моделирование идеально подходит для объектов, состоящих преимущественно из простых форм, или объектов, не требующих высокой детализации. Урбанистическая сцена характерна обилием прямых линий в контурах объектов. Такое свойство позволяет при создании 3D сцены городского пейзажа руководствоваться принципами низкополигонального моделирования.

На рисунке 2.22 изображена низкополигональная модель фасада здания. В данном случае моделируются только крупные формы частей здания (выделено зелеными линиями). Мелкие детали фасада здания зачастую имеют сложную

замысловатую форму и требуют слишком много полигонов для их моделирования. Поэтому они имитируются с помощью текстурных карт: карты цвета, карт нормалей, прозрачности, неровности и т.д.

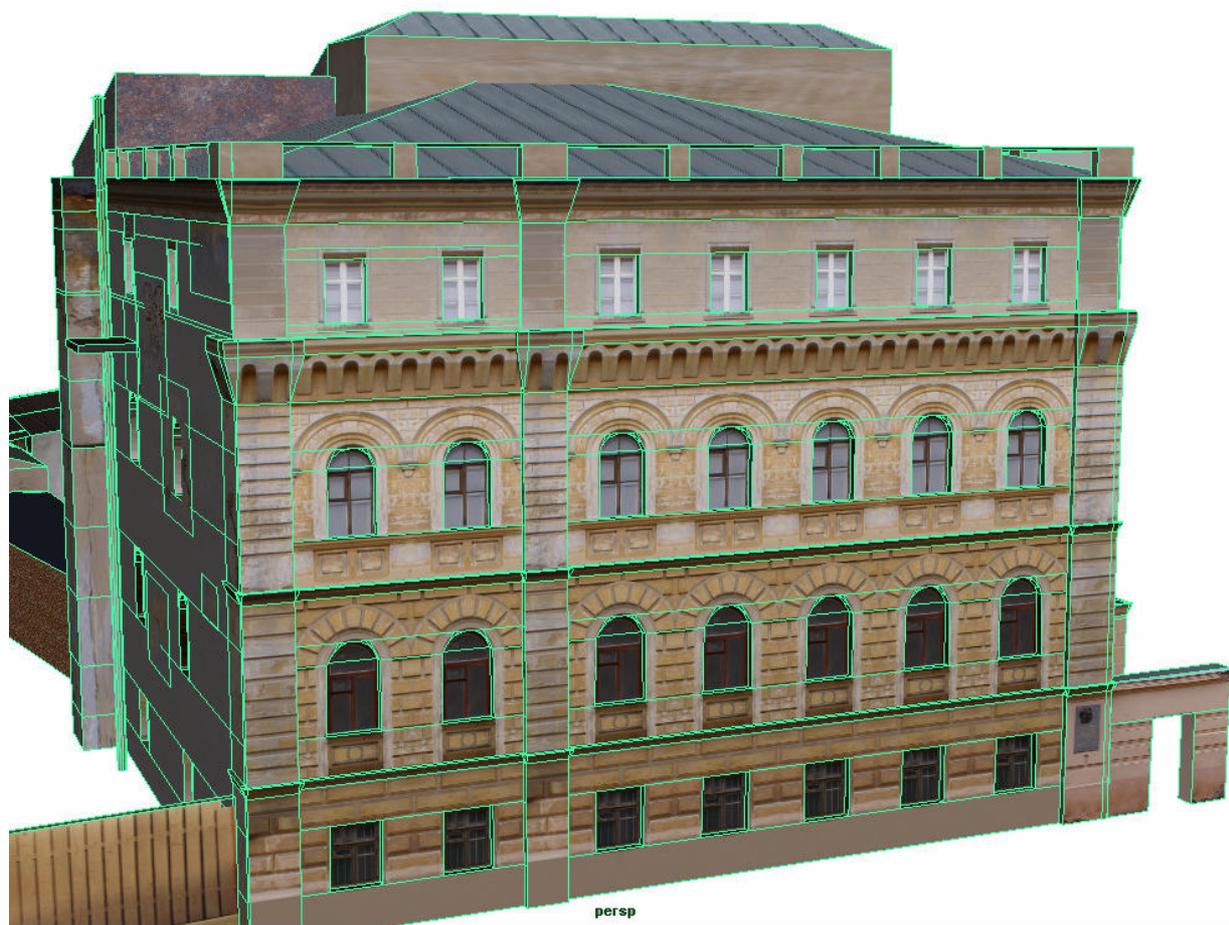


Рис. 2.22. Низкополигональная модель фасада здания

2.4 Оптимизация процесса визуализации

2.4.1 Библиотека моделей и текстур

Человек воспринимает окружающий мир объектно. Человеческий мозг анализирует объекты реального мира во всем многообразии их причинно-следственных связей и сопоставляет их с известными ему понятиями (объектами). Таким образом, человек с легкостью сопоставит реальный объект с его унифицированной 3D моделью, что позволяет типизировать объекты городской среды. В этом случае создается модель не самого объекта, а используется заранее созданная модель шаблона, к типу которого отнесен рассматриваемый объект (рис. 2.23, 2.24).

Характерной чертой урбанистического пространства является большое количество повторяющихся типовых объектов. К таким объектам относятся: светильники наружного освещения, лестницы, фонтаны, ограды, стенды для

афиш и реклам, парковые скамьи, детские игровые комплексы, урны, киоски, павильоны, торговые лотки, торговые автоматы и т. д.



Рис. 2.23. Шаблоны типовых объектов городской среды

Создание и классификация пополняемой библиотеки 3D моделей и текстур типовых объектов позволяет существенно автоматизировать процесс построения 3D моделей городской среды.

Использование нескольких копий 3D моделей и текстур позволяет существенно сократить объем данных загружаемых в память для последующей визуализации. Как следствие такой подход сокращает общее время загрузки 3D сцены. Экономия памяти позволяет увеличить качество визуализируемых данных (повысить разрешение текстур и детализацию геометрии).



Рис. 2.24. Шаблоны типовых объектов городской среды

2.4.2 Затуманивание и область видимости

Одним из способов оптимизации процесса визуализации больших, протяженных сцен является ограничение области видимости.



Рис. 2.25. Эффект затуманивания

Существенно сократить область видимости при визуализации городской среды можно путем использования эффекта затуманивания сцены (англ. *fogging*) (рис. 2.25).

Данный эффект образуется за счет смешивания цветов изображения с цветом тумана, который меняется в зависимости от глубины затуманивания.

2.4.3 Уровни детализации

В тех случаях, когда объект находится далеко от камеры или он не является центром ее внимания, не имеет смысла визуализировать большое количество полигонов этого объекта. Тогда применяются различные уровни детализации (англ. *Levels Of Detail* или *LOD*). Различают два типа *LOD*: статический и динамический.

Статический *LOD* это приём в программировании трёхмерной графики, заключающийся в создании нескольких вариантов одного объекта с различными степенями детализации, которые переключаются в зависимости от удаления объекта от виртуальной камеры [12] (рис. 2.26). Объекты различной детализации создаются заранее до этапа визуализации.

Преимущества статического *LOD*:

- простая реализация. Несколько заранее созданных вариантов объекта просто подменяются друг другом в зависимости от расстояния до камеры;
- малое использование центрального процессора при визуализации.

Недостатки статического *LOD*:

- возможно появление эффекта “скачкообразного” перехода в момент смены детализации;
- сложность изготовления модели, поскольку для каждого объекта создаются несколько вариантов его 3D модели.



Рис. 2.26. Различные уровни детализации модели

Динамический *LOD* использует одну модель объекта и различные алгоритмы, позволяющие плавно регулировать число полигонов в объекте в зависимости от расстояния до виртуальной камеры (рис. 2.27).

Преимущества динамического *LOD*:

- плавный переход к различным детализациям;
- эффективное использование ресурсов. Изменение детализации происходит только там где оно действительно необходимо.
- возможность устанавливать детализацию в реальном времени. Детализируются только те части объекта, точность отображения которых важна в данный момент.

Недостатки динамического *LOD*:

- сложная реализация;
- высокая вычислительная сложность.

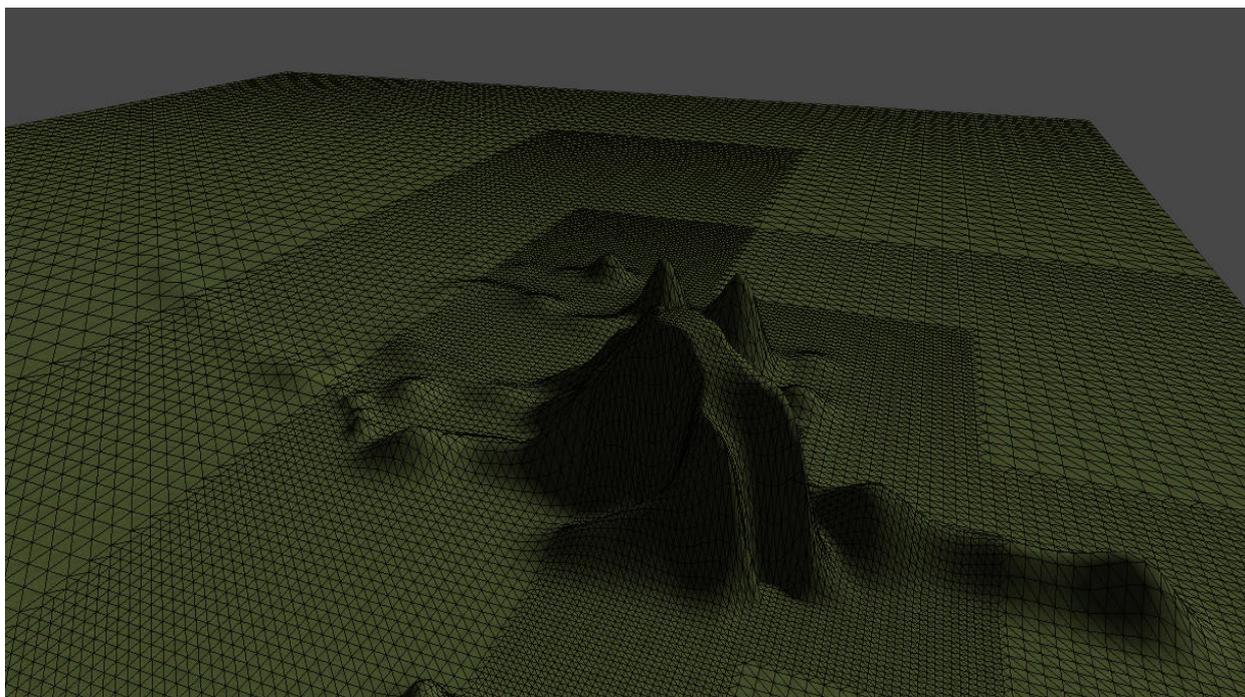


Рис. 2.27. Пример динамического уровня детализации

Статический *LOD* в силу своей простоты и малой загрузки центрального процессора используется для визуализации в режиме реального времени объектов городской среды. Динамический *LOD* используется для сложных объектов, имеющих большое количество мелких деталей. Он идеально подходит для визуализации рельефа местности.

2.5 Реалистичное освещение

Свет является ключевой составляющей эстетического восприятия сцены. Чтобы сцена выглядела правдоподобно, очень важно правильно имитировать освещение.

В реальном мире ослабление пучка света при распространении в поглощающей среде определяется согласно закону Бугера - Ламберта - Бера [13]. Согласно этому физическому закону интенсивность света уменьшается с расстоянием по экспоненциальному закону. К сожалению, высокая вычислительная сложность не позволяет использовать физически точную модель освещения при визуализации в режиме реального времени. Поэтому такую вычислительно сложную зависимость ослабляют до квадратичной либо линейной. Современные системы визуализации используют физически корректные расчеты для построения траектории луча света. Такой принцип позволяет имитировать освещение, основываясь на физических законах реального мира.

В зависимости от соотношения размера источника света и расстояния от него до приемника света различают точечные и протяженные источники света. Если размер источника света значительно меньше расстояния от него до объекта-приемника света, то такой источник называется точечным источником света.

В ином случае источник света называется протяженным. Например, для объекта находящегося под фонарем уличного освещения в парке, свет от фонаря будет являться протяженным источником света. А для объекта, находящегося за пределами парка, свет от фонаря будет являться точечным источником света.

В трехмерной графике для имитации освещения используются два подхода: прямое и глобальное освещение [14]. Модель прямого освещения, используемая при визуализации в режиме реального времени, предполагает, что луч света останавливается сразу после того, как встречает первое препятствие. В реальном же мире световой луч при встрече с отражающей поверхностью меняет свое направление и распространяется дальше. Глобальное освещение учитывает отраженный свет от различных поверхностей сцены. Такой подход позволяет создавать более реалистичные изображения, но требует больших вычислительных ресурсов. Поэтому для визуализации в режиме реального времени отраженный свет имитируют с помощью установки дополнительных световых источников. Это очень важный прием, позволяющий добавить дополнительного реализма в сцену без существенного увеличения её вычислительной сложности [15].

Существуют четыре основных типа источников света, используемых в компьютерной графике: рассеянный источник (англ. *ambient light*), всенаправленный источник (англ. *omni light*), направленный параллельный свет (англ. *directional light*) и прожектор (англ. *spot light*) [16].

Рассеянный источник света применяется для имитации рассеянного света, существующего в реальном мире. Он равномерно испускает лучи во всех направлениях и освещает всю сцену. С помощью этого источника можно изменить яркость и общий цвет сцены. Рассеянный источник света должен использоваться только вместе с другими типами источников света, так как в ином случае визуализированная сцена будет выглядеть плоско и не реалистично.



Рис. 2.28. Всенаправленный источник света

Всенаправленный источник света равномерно испускает лучи во всех направлениях. Таким образом, освещается область вокруг него. Такой тип освещения используется для создания заполняющего света. Часто применяется для имитации горения электрической лампочки (рис. 2.28).

Направленный параллельный свет наиболее схож с солнечным светом. Затухания при таком типе освещения не предусмотрено. Лучи направленного света распространяются параллельно и бесконечно. Поэтому расположение источника света не играет большой роли. Главный параметр здесь – направление движения лучей. Тени в сцене, освещаемой направленным источником света, создаются путем проецирования на поверхность параллельных лучей (Рис. 2.29) [15].



Рис. 2.29. Направленный свет

Принцип действия источника света типа «*прожектор*» можно понять из его названия. Область распространения лучей света из такого источника имеет форму конуса. Таким образом, область освещения сцены источником увеличивается по мере отдаления от него (рис. 2.30). Главные параметры распространения света типа «*прожектор*» – угол конуса, устанавливающий величину области распространения света, и угол полутени, устанавливающий степень размытости границ светового пятна. Таким образом, максимальная мощность освещения от «*прожектора*» будет в точке, в которую он направлен, а к краям его влияния, заданным углом полутени, освещенность экспоненциально падает. С помощью этих параметров можно создать как яркий резкий луч, так и незаметное рассеянное световое пятно. Данный тип освещения может быть использован для имитации света от прожекторов, ламп, автомобильных фар и т.д.

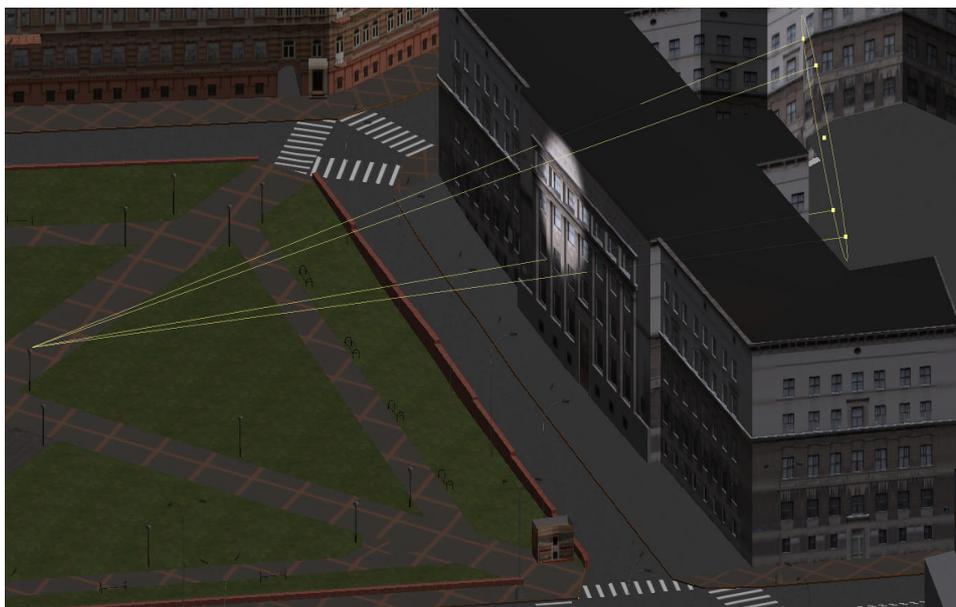


Рис. 2.30. Источник света типа «прожектор»

2.6 Методы формирования теней в режиме реального времени

Одним из важных факторов, влияющих на реализм генерируемого изображения, является наличие тени. С помощью теней можно оценить глубину сцены и взаимное расположение объектов. Тень позволяет закрепить объект с окружающей его обстановкой.

Существуют два типа теней:

- жесткие тени (англ. *hard shadows*)
- мягкие тени (англ. *soft shadows*)

Жесткие тени имеют четкую границу (рис. 2.31). Согласно геометрической оптике Френеля такие тени образуются от точечных или направленных параллельных источников света.

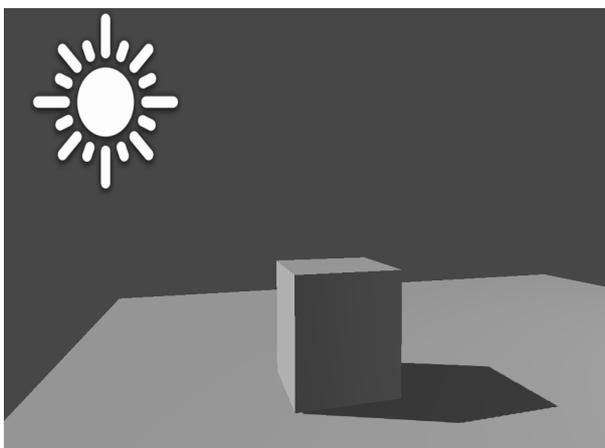


Рис. 2.31. Жесткие тени

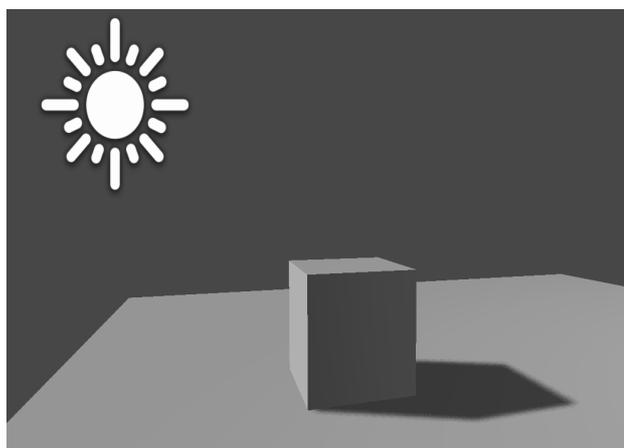


Рис. 2.32. Мягкие тени

Протяженный источник света можно представить как набор большого числа точечных источников. Поэтому при наличии такого источника света от

объекта получается не одна тень, а целая серия теней, которые накладываются друг на друга. В этом случае ближе к границе образуются менее затемненные области, образующие полутени. Такие тени называют мягкими (рис. 2.32). Самый простой алгоритм реализации мягких теней – это размытие границ тени на некоторую фиксированную величину.

Существуют разные подходы генерации теней в реальном времени [17]:

- проектируемая геометрия
- карты освещения
- вершинные тени
- теневые объемы
- теневые карты
- теневые буферы

2.6.1 Проектируемая геометрия

Проектируемая геометрия (англ. *Projected geometry*) использует геометрическое взаимоотношение источника света и объекта. Основная идея состоит в проектировании объекта из точки положения источника света на плоский участок, на котором формируется тень. Количество вычислений таких «*теневых проекций*» зависит от количества источников света в сцене.

Достоинства подхода:

- простота реализации;
- хорошее качество получаемой тени.

Недостатки:

- высокая вычислительная сложность;
- тень генерируется только на абсолютно плоской поверхности.

Проектируемая геометрия наиболее часто используется при генерации теней в режиме реального времени. Однако ввиду высокой вычислительной сложности применяется такой подход только для формирования наиболее заметных и необходимых теней.

2.6.2 Карты освещения

Карты освещения (англ. *lightmaps*) наиболее часто используются в 3D приложениях для статического освещения сцены. Карта освещения представляет собой текстуру, генерируемую перед процессом визуализации, которая определяет степень освещенности определенного полигона сцены. Размеры карты определяются размерами минимального, ограничивающего полигон прямоугольника. Процесс формирования карты освещения происходит следующим образом. Карта освещения выравнивается с текстурой полигона, таким образом, чтобы одному текселю карты соответствовало 4-32 текселя текстуры [17].

В результате каждому текселю карты будет соответствовать точка на полигоне. Из этой точки проводится луч на каждый источник света в сцене. Если луч пересекается с одним из полигонов сцены, то соответствующий источник света не освещает эту точку. В ином случае происходит увеличение значения тексела карты освещения на величину, зависящую от используемой модели освещения и положения источника света относительно точки. Окончательное значение этого тексела определяет уровень освещенности соответствующей точки полигона.

На этапе визуализации карта освещения накладывается на полигон вместе с основной текстурой меняя ее яркость. Количество карт освещения зависит от количества полигонов в освещаемой сцене. Поэтому для больших сцен следует объединять карты освещения в текстурный атлас.

Достоинства подхода:

- высокое качество получаемого освещения;
- низкая загрузка процессора.

Недостатки подхода:

- работает только со статической геометрией;
- занимает большой объем видеопамати при визуализации.

2.6.3 Вершинное освещение

При вершинном освещении (англ. *VertexLight*) освещение рассчитывается в каждой вершине полигональной сетки, а затем результирующие значения этого расчёта интерполируются для вычисления цвета каждого пикселя, который находится внутри полигона [18]. Алгоритм расчета освещенности вершин такой же, как и алгоритм нахождения яркостей текселов карт освещения. Но поскольку освещенность определяется только для вершин, данный метод может быть реализован в режиме реального времени. Вершинное освещение применяется для сцен, состоящих из большого количества маленьких полигонов.

Достоинства подхода:

- простота реализации;
- не загружает видеопамать.

Недостатки подхода:

- низкое качество освещения низкополигональных объектов.

2.6.4 Теневые объемы

Теневой объем (англ. *Shadow Volumes*) – это область пространства, закрываемая от заданного источника света каким-либо объектом (объектами). Данная область определяется путем проекции минимального силуэта объекта вдаль от источника света. Точка, попадающая в этот теневой объем, находится в тени, а точка, лежащая вне его, освещается данным источником света [19]. Форма те-

невого объема может быть довольно сложной, что существенно усложняет процесс проверки на попадание точки внутрь него. Поэтому существует простой прием упрощающий эту проверку. Из проверяемой точки испускается луч и производится подсчет пересечений этого луча с лицевыми и нелицевыми гранями, ограничивающими теневой объем. Данная точка лежит вне теневого объема при равном количестве пересечений луча с лицевыми и нелицевыми гранями. Поскольку направление луча не имеет большого значения данную задачу можно еще сильнее упростить для максимального использования аппаратных возможностей современных видеокарт. Если луч из проверяемой точки будет направлен на наблюдателя, то для хранения количества пересечений может быть использован *stencil* буфер (буфер трафарета) [20].

Достоинства подхода:

- генерирует совершенно точные тени на любом расстоянии;
- производительность не зависит от сложности и количества приемников тени.

Недостатки подхода:

- вычислительно сложен для высокополигональных объектов;
- отсутствует возможность сгенерировать тени от полупрозрачных объектов либо объектов, имеющих поверхности, точки которых отбрасываются при проверке на прозрачность;
- невыпуклые объекты не могут отбрасывать тени сами на себя.

Подход, использующий теневые объемы, является одним из наиболее распространенных приемов построения теней в реальном времени для небольшого количества низкополигональных объектов сцены, не использующих элементы прозрачности.

2.6.5 Теневые карты

Идея подхода, использующего теневые карты (англ. *shadow maps*), состоит в предположении, что фрагменты, освещенные источником света, являются участками поверхности, видимыми из положения источника света. Таким образом, если переместить камеру (наблюдателя) в положение источника света, то тень отбрасываемая объектом при освещении этим источником будет полностью загорожена самим объектом.

Метод разделен на две стадии. Сначала для каждой пары «источник света - объект» создается текстура, в которую рисуется объект, как он виден из положения источника света. Объект рисуется только черным цветом на белом фоне. Полученная текстура называется теневой картой, хранящей поперечный срез конуса тени объекта. Вторая стадия состоит в наложении карты затенения на затененные объекты. Для этого используется технология проективного наложения текстур [21].

Преимущества метода:

- простота реализации.

Недостатки метода:

- карта затенения формируется для каждой пары «источник света – объект», что значительно усложняет работу алгоритма при использовании нескольких источников света;
- невыпуклые объекты не могут отбрасывать тени сами на себя.

2.6.6 Теневые буферы

Идея метода, использующего теневые буферы (англ. *Shadow buffers*), также состоит в предположении, что затенёнными являются те точки, которые не видны из источника света. Сначала отрисовывается вся сцена с точки зрения источника света в буфер глубины. В буфере записаны глубины всех ближайших к источнику света точек. Карта глубин, сохраненная в теневом буфере, будет использоваться для теста на принадлежность точек тени. То есть точка находится в тени, если её расстояние до источника света больше чем значение соответствующей точки в теневом буфере.

Преимущества метода:

- простота реализации;
- производительность не зависит от сложности сцены;
- объекты могут отбрасывать тень сами на себя;
- корректная работа с полупрозрачными объектами и объектами, имеющими поверхности, точки которых отбрасываются при проверке на прозрачность.

Недостатки метода:

- для каждого источника света требуется отдельная отрисовка всей сцены;
- плохо работает для точечных источников света. В этом случае потребуется несколько раз осуществлять отрисовку сцены для различных направлений от источника света.

Ни один из предложенных методов не может быстро генерировать тени в реальном времени для сложных высокополигональных сцен. Каждый из них имеет свои достоинства и недостатки. Тем не менее, эти методы можно комбинировать, чтобы сочетать сильные стороны каждого.

2.7 Воздушная перспектива

Важным фактором достижения реализма при формировании изображения трехмерного пространства является правильная имитация глубины сцены. При создании эффекта глубины отображаемого пространства необходимо помнить о таком природном явлении, как воздушная перспектива.

Воздушная перспектива - (лат. *perspicere* – смотреть сквозь, проникать взором) - размытость очертаний отдалённых объектов, возникающая из-за ис-

чезновения цвета и контрастности между фоном и этими объектами. Является монокулярным сигналом глубины пространства [22]. Данное явление связано с наличием в атмосфере мельчайших частиц пыли, влаги, дыма и т.д., вследствие чего свет меняет свою яркость, а контраст светотени смягчается (рис. 2.33).



Рис. 2.33. Имитация эффекта воздушной перспективы

Для имитации такого природного явления при формировании изображения 3D сцены необходимо уменьшать четкость и ясность очертаний предметов по мере их удаления от глаз наблюдателя. Также дальний план должен выглядеть более светлым, чем передний план.

2.8 Постобработка

Постобработка это дополнительная обработка изображения, полученного после визуализации сцены. При визуализации в режиме реального времени постобработка производится с помощью пиксельных шейдеров [23].

Изображение является фотореалистичным, если его трудно отличить от фотографии. Поэтому для повышения реалистичности изображения необходимо имитировать эффекты, присущие фотографии. Наиболее часто применяют следующие эффекты:

- *Motion Blur* – имитация размытия изображения при воспроизведении сцен движения или быстро движущихся объектов.
- *Bloom* – имитация передержки камеры при съёмке, имитация размытости света на ярких гранях сцены.

- *Depth Of Field* – имитация размывания объектов в зависимости от их положения относительно фокуса камеры.

В разделе произведено подробное описание представленных эффектов.

2.8.1 *Motion Blur*

В реальной жизни, быстро движущиеся объекты кажутся нам немного смазанными. Вызвано это тем, что сетчатка глаза фиксирует не мгновенную освещенность, а световой поток за определенный промежуток времени [24]. По такому же принципу работает фото и видео техника. Поэтому просмотр сцен с движением без эффекта смазанности вызывает чувство неестественности.

Изображение 3D сцен при визуализации формируется заново каждый кадр. Добавление к полученному изображению эффекта размытия на движущихся объектах увеличивает степень реализма. В компьютерной графике такой эффект называется *Motion Blur* (рис. 2.34).



Рис. 2.34. Применение эффекта *Motion Blur*

Самым простым алгоритмом данного эффекта является полноэкранный смазывание, когда для рендеринга текущего кадра используются данные из предыдущих кадров. Алгоритм обрабатывает кадр целиком независимо от того, что попадает в поле зрения. Более современные методы *Motion Blur* не используют информацию о предыдущих кадрах, а основываются на векторах движения объектов в кадре. Таким образом эффект может применяться как ко всему кадру, так и к отдельным движущимся объектам.

2.8.2 Bloom

Часто при фото или видео съемке изображение получается засвеченным. Данный эффект возникает в результате передержки камеры при съемке и обусловлен некоторыми видами aberrаций линз объектива камеры и фотооптическими свойствами пленки. В этом случае появляется видимость свечения вокруг ярких поверхностей, причем свет от них частично воздействует и на более темные области, соседствующие с яркими поверхностями.



Рис. 2.35. Изображение 3D модели городской среды без эффекта постобработки



Рис. 2.36. Изображение 3D модели городской среды с эффектом постобработки *Bloom*

Для имитации такого кинематографического эффекта используется *Bloom* фильтр, который делает яркие участки изображения (рис. 2.35) ещё более яркими, и добавляет свечение вокруг этих участков (рис. 2.36).

2.8.3 Depth Of Field

Особенность строения человеческого глаза выражается в том, что мы видим одинаково четко не все объекты, попавшие в поле нашего зрения. Есть определенная область пространства, на которую сфокусировано наше зрение в данный момент. Эта область выглядит наиболее четко. Все остальные области в поле нашего зрения выглядят размыто. Точно по такому же принципу работает фото и видео камеры.

В компьютерной графике каждый объект на изображении 3D сцены выглядит идеально четко (рис. 2.37). Поэтому, для того чтобы приблизить наше изображение к особенностям человеческого восприятия, объекты не попавшие в резко отображаемое пространство размываются (рис. 2.38). Данный эффект постобработки получил название *Depth of field* (глубина резкости). Существует несколько вариантов реализации этого эффекта. Но на практике чаще всего применяют метод, при котором в режиме реального времени происходит смешивание оригинального изображения с его размытой версией. Размытая версия

изображения формируется на основе *blur* фильтра, который определяет интенсивность размытия в зависимости от дальности к резко отображаемому пространству.



Рис. 2.37. Изображение 3D модели городской среды без эффекта постобработки



2.38. Изображение 3D модели с эффектом постобработки *Depth of field*

Список литературы

1. Билл Флеминг Фотореализм. Профессиональные приемы работы. – М.: ДМК, 2000. – 376 с.
2. Наталья Валукевич Грязь. – Режим доступа: <http://dlight.ru/forums/showthread.php?t=27>, свободный.
3. Билл Флеминг Создание фотореалистичных изображений. М.: ДМК, 376 с.
4. Материал из википедии (свободной энциклопедии). Режим доступа: http://ru.wikipedia.org/wiki/Parallax_mapping, свободный.
5. Виктор Коваленко Текстура в задачах трехмерной визуализации. – М.: «Открытые системы», 1996. – № 06. Режим доступа: <http://www.osp.ru/os/1996/06/179044/>, свободный.
6. Юрий Рогач Tiles & Maya. Режим доступа: <http://dlight.ru/forums/showthread.php?t=31>, свободный.
7. Материал из википедии (свободной энциклопедии). Режим доступа: <http://ru.wikipedia.org/wiki/MIP-текстурирование>, свободный.
8. Мипмап (MipMap) и Мипмаппинг (MipMapping). Режим доступа: <http://www.gamedev.ru/terms/mipmap>, свободный.
9. Трилинейная фильтрация: техника улучшения качества изображений. Режим доступа: <http://www.ixbt.com/video/3linear.html>, свободный.
10. Материал из википедии (свободной энциклопедии). Режим доступа: http://ru.wikipedia.org/wiki/Билинейная_фильтрация, свободный.
11. Материал из википедии (свободной энциклопедии). Режим доступа: http://en.wikipedia.org/wiki/Texture_atlas, свободный.
12. Материал из википедии (свободной энциклопедии). Режим доступа: http://ru.wikipedia.org/wiki/Level_of_Detail, свободный.
13. Материал из википедии (свободной энциклопедии). Режим доступа: http://ru.wikipedia.org/wiki/Закон_Бугера_—_Ламберта_—_Бера, свободный.
14. Philip Dutre, Global Illumination Compendium: A Collection of the Important Formulae, September 2003.
15. Постановка света в 3D графике. Режим доступа: <http://3d-uroki.org/tutorial/raznoe/170>, свободный.
16. Марк Джамбруно Трехмерная (3D) графика и анимация. М.: Вильямс, 2002. – 640 с.
17. Андрей Цыганков Рендеринг теней в реальном времени. Режим доступа: <http://dtf.ru/articles/read.php?id=19>, свободный.
18. Вершинное освещение совместно с LightMaps. Режим доступа: <http://www.gamedev.ru/articles/?id=30120>, свободный.
19. Создание теней при помощи теневых объемов. Режим доступа: <http://steps3d.narod.ru/tutorials/shadow-volumes-tutorial.html>, свободный.
20. Стенсил буфер (Stencil Buffer). Режим доступа: <http://www.gamedev.ru/terms/StencilBuffer>, свободный.

21. Проективные текстуры (projective texture). Режим доступа: <http://www.ixbt.com/video/projectivetexture.shtml>, свободный.
22. Жмуров В. А. Большая энциклопедия по психиатрии. – М.: Джангар, 2010. – 864 с.
23. Алексей Берилло Современная терминология 3D графики. Режим доступа: <http://www.ixbt.com/video2/terms2k5.shtml>, свободный.
24. Добавляем эффект Motion Blur в WPF-приложения. Режим доступа: <http://habrahabr.ru/post/194734/>, свободный.

ФОТОРЕАЛИСТИЧНОЕ МОДЕЛИРОВАНИЕ И ВИЗУАЛИЗАЦИЯ РАЙОНОВ ГОРОДСКОЙ СРЕДЫ

Составитель

Михаил Павлович Осипов

Учебно-методическое пособие

Государственное образовательное учреждение высшего
Профессионального образования «Нижегородский государственный
университет им. Н.И. Лобачевского»
603950, Нижний Новгород, пр. Гагарина, 23.

Подписано в печать __.__.____г. Формат 60x84 1/16.
Бумага офсетная. Печать офсетная. Гарнитура Таймс.
Усл. печ. л. _____. Уч.-изд. л. ____
Заказ № _____. Тираж 100 экз.

Отпечатано в типографии Нижегородского госуниверситета
им. Н.И. Лобачевского
603600, г. Нижний Новгород, ул. Большая Покровская, 37
Лицензия ПД № 18-0099 от 14.05.01