

4. Графика.

4.1. Освещение.

Чтобы рассчитать затенение 3D-объекта, Unity необходимо знать интенсивность, направление и цвет падающего на него света.

Эти свойства предоставляются объектами Light в сцене. Базовый цвет и интенсивность устанавливаются одинаково для всех источников света, но направление зависит от того, какой тип источника света вы используете. Кроме того, свет может уменьшаться по мере удаления от источника. Четыре типа источников света, доступных в Unity, описаны ниже.

Точечный источник света располагается в точке пространства и одинаково излучает свет во всех направлениях. Направление света, падающего на поверхность, — это линия, идущая от точки контакта обратно к центру светового объекта. Интенсивность уменьшается с расстоянием от источника света, достигая нуля в определенном диапазоне. Точечные источники света полезны для имитации ламп и других локальных источников света в сцене. Вы также можете использовать их, чтобы заставить искру или взрыв убедительно осветить окружающую среду.

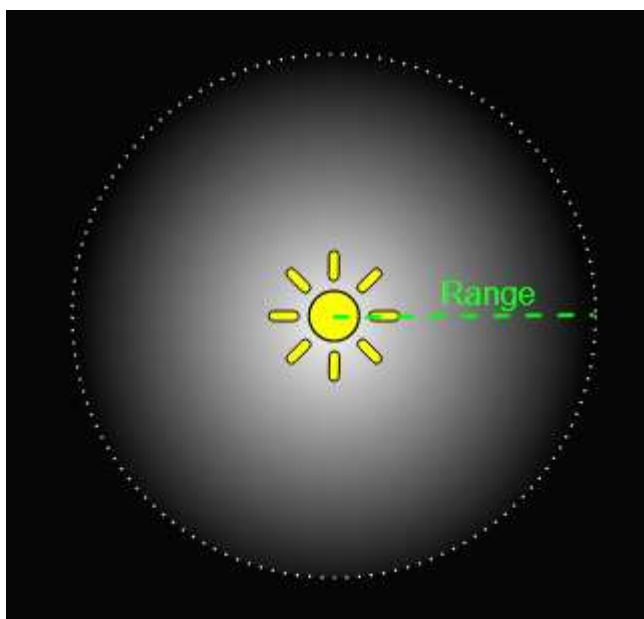


Рисунок 28 – Точечный источник света с радиусом

Подобно точечному источнику света, **прожектор** имеет определенное местоположение и диапазон, в котором свет падает. Однако прожектор ограничен углом, что приводит к конусообразной области освещения. Центр конуса указывает в прямом (Z) направлении светового объекта. Прожекторы обычно используются для источников искусственного света, таких как фонарики, автомобильные фары и прожекторы. С направлением,

управляемым сценарием или анимацией, движущийся прожектор освещает лишь небольшую область сцены и создает драматические световые эффекты.

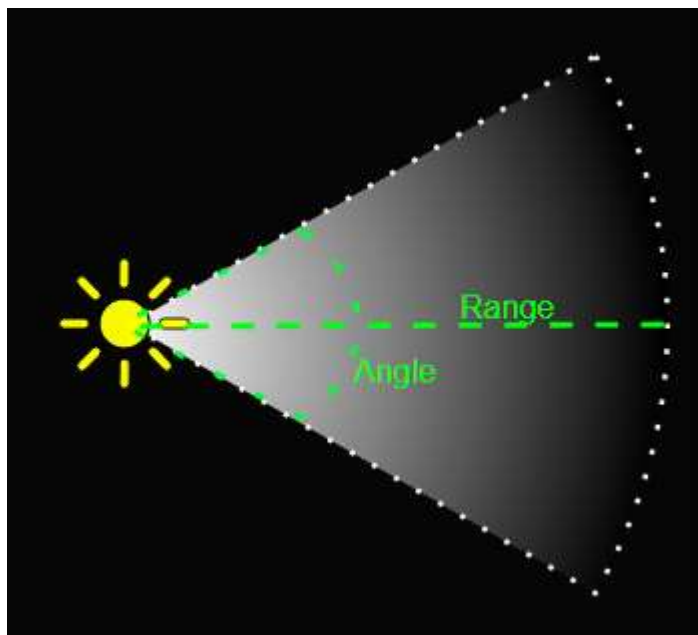


Рисунок 29 – Прожектор с дальностью и углом освещения

Направленный источник света не имеет идентифицируемой позиции источника, поэтому объект освещения обычно можно разместить в любом месте сцены. Все объекты в сцене освещены так, как будто свет всегда с одного направления. Расстояние света от целевого объекта не определено, поэтому свет не уменьшается. Направленные источники представляют собой большие удаленные источники, которые исходят из положения, находящегося за пределами игрового мира. В реалистичной сцене их можно использовать для имитации солнца или луны. В абстрактном игровом мире они могут быть полезным способом добавить убедительное затенение объектам без точного указания, откуда исходит свет. При проверке объекта в представлении сцены (например, чтобы увидеть, как выглядит его сетка, шейдер и материал) направленный свет часто является самым быстрым способом получить представление о том, как будет выглядеть его затенение. Для такого теста вас, как правило, не интересует, откуда исходит свет, а просто хотите увидеть, что объект выглядит «твердым», и найти сбои в модели.

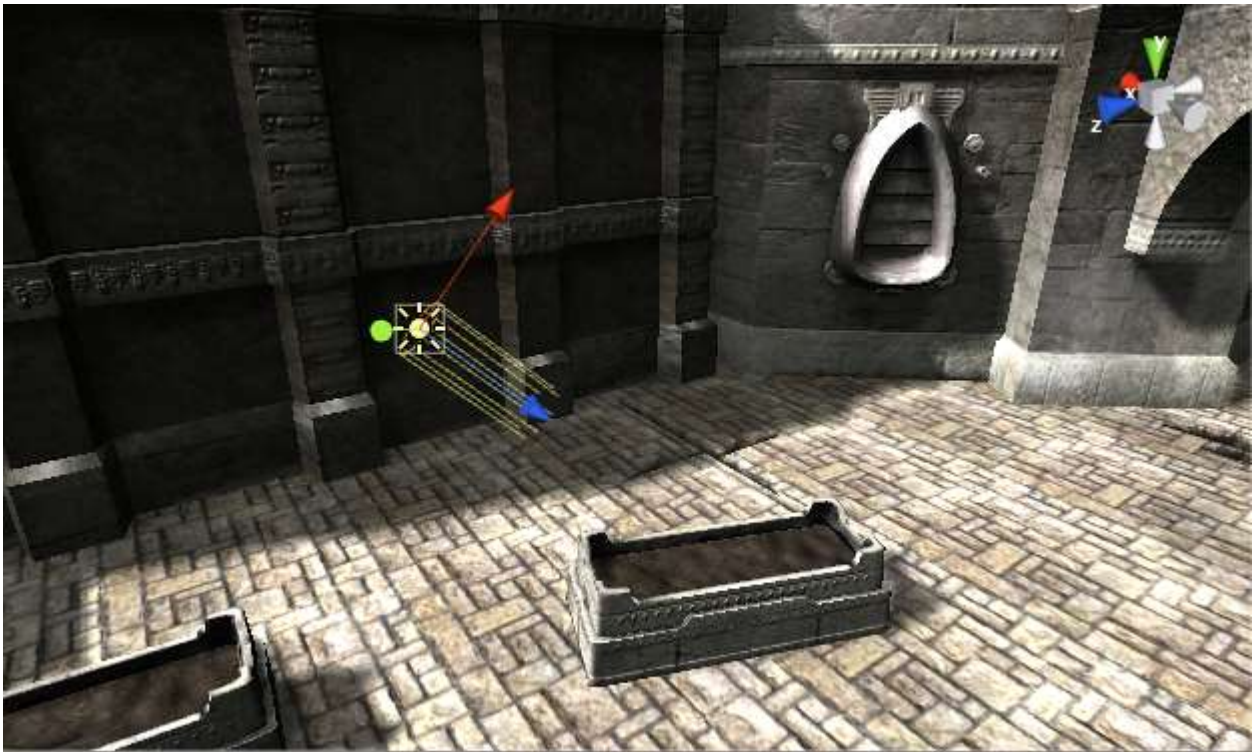


Рисунок 30 – Направленный источник в сцене

Зональный свет определяется прямоугольником в пространстве. Свет излучается во всех направлениях, но только с одной стороны прямоугольника. Свет падает в указанном диапазоне. Поскольку расчет освещения требует значительных ресурсов процессора, зональные источники света недоступны во время выполнения и могут быть только запечены в картах освещения.



Рисунок 31 – Пример зонального источника света

4.2. Камера.

Сцена Unity создается путем размещения и перемещения объектов в трехмерном пространстве. Поскольку экран зрителя двумерный, должен быть способ захватить вид и «сгладить» его для отображения. Это достигается с помощью камер.

Камера — это объект, определяющий вид в пространстве сцены. Положение объекта определяет точку обзора, а оси вперед (Z) и вверх (Y) объекта определяют направление взгляда и верхнюю часть экрана соответственно. Компонент камеры также определяет размер и форму области, попадающей в поле зрения. При настройке этих параметров камера может отображать на экране то, что она «видит» в данный момент. Когда объект камеры перемещается и вращается, отображаемый вид также будет перемещаться и вращаться соответственно.

Камера в реальном мире или даже человеческий глаз видят мир таким образом, что объекты кажутся меньше, чем дальше они находятся от точки обзора. Этот известный эффект перспективы широко используется в искусстве и компьютерной графике и важен для создания реалистичной сцены. Естественно, Unity поддерживает перспективные камеры, но для некоторых целей вы хотите визуализировать вид без этого эффекта. Например, вы можете захотеть создать карту или информационный дисплей, который не должен выглядеть точно так же, как реальный объект. Камера, которая не уменьшает размер объектов с расстоянием, называется орфографической, и у камер Unity также есть опция для этого. Перспективный и орфографический режимы просмотра сцены известны как проекции камеры.

Как перспективные, так и ортогональные камеры имеют ограничение на то, насколько далеко они могут «видеть» из своего текущего положения. Предел определяется плоскостью, перпендикулярной направлению камеры вперед (Z). Это известно, как дальняя плоскость отсечения, поскольку объекты, находящиеся на большем расстоянии от камеры, «отсекаются» (т. е. исключаются из рендеринга). Существует также соответствующая ближняя плоскость отсечения рядом с камерой — видимый диапазон расстояний равен расстоянию между двумя плоскостями.

4.3. Материалы, шейдеры и текстуры.

Рендеринг в Unity выполняется с помощью материалов, шейдеров и текстур. Между материалами, шейдерами и текстурами в Unity существует тесная связь.

Материалы — это определения того, как поверхность должна быть визуализирована, включая ссылки на используемые текстуры, информацию о тайлинге, цветовые оттенки и многое другое. Доступные параметры материала зависят от того, какой шейдер использует материал.

Шейдеры — это небольшие скрипты, которые содержат математические расчеты и алгоритмы для расчета цвета каждого визуализируемого пикселя на основе входных данных освещения и конфигурации материала.

Текстуры представляют собой растровые изображения. Материал может содержать ссылки на текстуры, чтобы шейдер Материала мог использовать текстуры при вычислении цвета поверхности объекта. В дополнение к основному цвету (альбедо) поверхности объекта текстуры могут отображать многие другие аспекты поверхности материала, такие как его отражательная способность или шероховатость.

Материал указывает один конкретный шейдер для использования, а используемый шейдер определяет, какие параметры доступны в материале. Шейдер указывает одну или несколько переменных текстуры, которые он предполагает использовать, а инспектор материалов в Unity позволяет вам назначать свои собственные ресурсы текстуры этим переменным текстуры.

Для большинства обычных рендеров, под которыми мы подразумеваем персонажей, пейзажи, окружение, твердые и прозрачные объекты, твердые и мягкие поверхности и т. д., стандартный шейдер обычно является лучшим выбором. Это настраиваемый шейдер, способный очень реалистично отображать многие типы поверхностей. Существуют и другие ситуации, когда может подойти другой встроенный шейдер или даже специально написанный шейдер, например, жидкости, листва, преломляющее стекло, эффекты частиц, мультяшные, иллюстративные или другие художественные эффекты или другие специальные эффекты, такие как ночное видение, тепловое или рентгеновское зрение и т.д.

4.4. Системы частиц.

Частицы — это маленькие простые изображения или сетки, которые отображаются и перемещаются в большом количестве системой частиц. Каждая частица представляет собой небольшую часть жидкого или аморфного объекта, и эффект всех частиц вместе создает впечатление полного объекта. Используя в качестве примера облако дыма, каждая частица будет иметь небольшую текстуру дыма, напоминающую крошечное облако.

Когда многие из этих мини-облаков располагаются вместе в области сцены, общий эффект создается большим, заполняющим объемом облаком.

Каждая частица имеет определённое время жизни (обычно несколько секунд), в течение которого частица может претерпеть различные изменения. Частица начинает свое существование, когда она создана, или испущена своей системой частиц. Система испускает частицы в случайных точках в пространстве, ограниченном регионом в форме сферы, полусферы, конусом, прямоугольным параллелепипедом (для простоты такую форму будем называть просто “коробкой”) или мешем произвольной формы. Частица отображается до тех пор, пока не закончится время её жизни, затем она удаляется из системы. Частота испускания частиц системы жёстко определяет количество испускаемых частиц в секунду, хотя точное время испускания содержит небольшой фактор случайности. Частота испускания в совокупности со средним временем жизни частицы определяют количество “стабильных” частиц (то есть, тех, чьё испускание и уничтожение происходят с одинаковой частотой) и время, необходимое системе для достижения такого состояния.

Настройки испускания и времени жизни влияют на общее поведение системы, однако отдельные частицы тоже могут изменяться со временем. Каждая частица имеет вектор скорости, определяющий направление и расстояние, которое проходит частица за один кадр. Скорость может быть изменена с помощью сил и гравитации, применяемых самой системой или когда частицы сдуваются в зонах ветра над Terrain (название объекта в Unity, представляющего земную поверхность). Цвет, размер и вращение каждой частицы также могут изменяться в течение времени жизни или пропорционально её текущей скорости движения. Цвет включает альфа-канал (для прозрачности), так что частица может плавно исчезать и появляться вместо резкого изменения видимости в таких случаях.