



## ВИЗУАЛИЗАЦИЯ ЛИНИИ ВИДИМОСТИ НА ОСНОВЕ GPU

### 1. ПРЕДИСЛОВИЕ

Система прямой видимости визуализирует, какие части игрового мира можно увидеть, например, с точки зрения игрового персонажа от третьего лица. Область за пределами прямой видимости затемняется и остается скрытой от игрока.

Эта система была разработана как альтернатива визуализации линии видимости на основе ЦП. Системы на основе ЦП используют комбинацию raycasting и динамических сеток для вычисления линии обзора и часто интенсивно используют ЦП, что серьезно влияет на производительность.

Эта новая система прямой видимости на основе графического процессора использует метод, очень похожий на отображение теней, и работает значительно быстрее, чем любая система на базе ЦП, освобождая ценное время ЦП. Он был разработан с нуля с учетом производительности, простоты использования и масштабируемости. Эффект можно настроить для достижения любого желаемого визуального стиля. Пользователь может управлять внешним видом эффекта, выбирая, какие эффекты изображения применяются к области за пределами прямой видимости. Он идеально подходит для stealth или кшн-игр с видом сверху, но также работает с другими точками зрения и в играх другого типа.

Эта система совместима как с прямым, так и с отложенным конвейером рендеринга и поддерживает DX11. Поскольку в этой системе интенсивно используются текстуры рендеринга, в настоящее время она не поддерживается на мобильных платформах.

## 2. КАК

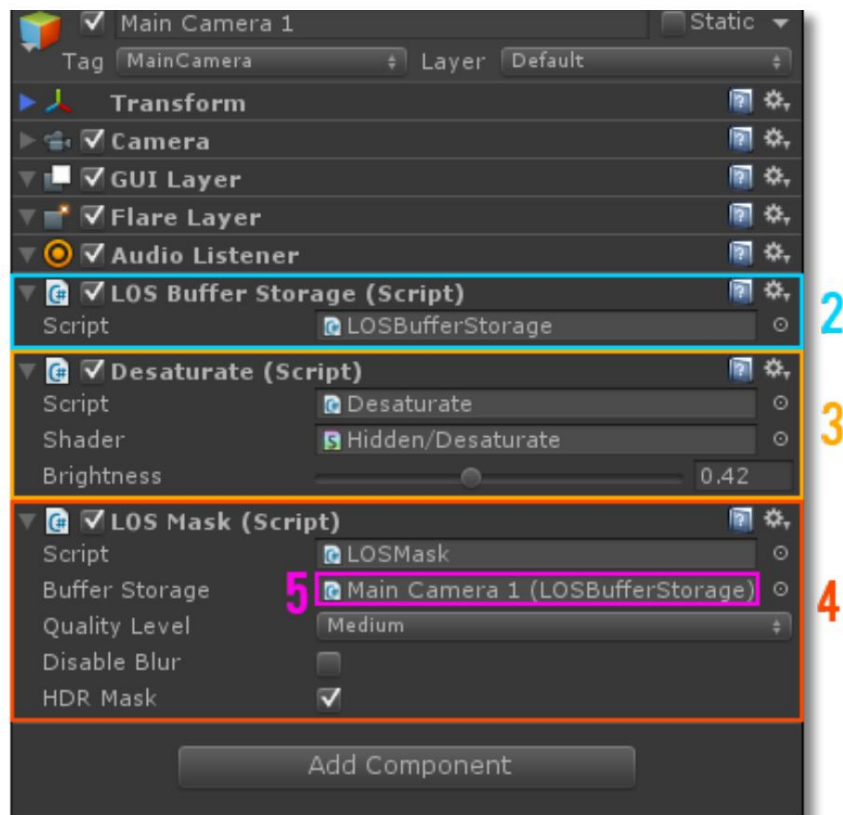
### 2.1 БЫСТРЫЙ СТАРТ

ТЛ; ДР? Нет времени читать всю инструкцию? Это пошаговое руководство научит вас основам настройки системы прямой видимости и поможет вам начать работу в кратчайшие сроки!

#### 2.1.1 НАСТРОЙКА ГЛАВНОЙ КАМЕРЫ

1. Выберите [основную камеру](#) в вашей сцене, она может быть либо перспективной, либо ортогональной.
2. Добавьте компонент [LOS Buffer Storage](#) в [GameObject](#) камеры (Add Component > Line of Sight > Buffer Storage).
3. Добавьте эффекты изображения, которые вы хотите применить к области за пределами прямой видимости (например, эффект затемнения).
4. Добавьте компонент «Маска прямой видимости» («Добавить компонент» > «Линия видимости» > «Маска прямой видимости»).
5. Назначьте ранее добавленный компонент [LOS Buffer Storage](#) с помощью [Buffer Storage](#) в [LOS](#).

Компонент маски



Порядок, в котором расположены компоненты LOS, очень важен. Компонент [буферного хранилища LOS](#) всегда должен располагаться перед (вверху в окне инспектора) эффектами изображения, которые вы хотите применить к области за пределами линии зрения. Компонент [LOS Mask](#) всегда должен располагаться после (ниже в окне инспектора) этих эффектов изображения.

Эффекты изображения, которые вы добавляете между компонентами [LOS Buffer Storage](#) и [LOS Mask](#), влияют только на область вне прямой видимости. Если вы хотите применить эффекты изображения ко всему экрану (например, bloom или SSAO), вы всегда можете добавить их после компонента [LOS Mask](#).

Компонент [LOS Buffer Storage](#) необязательно должен быть присоединен к тому же игровому объекту, что и маска [LOS](#), но он должен визуализироваться перед маской [LOS](#). Это дает вам больше свободы в настройке внешнего вида эффектов и не ограничивает вас использованием только эффектов изображения.

Например, вы можете визуализировать сцену с двумя отдельными камерами; первая камера (например, глубина 0) имеет подключенное [буферное хранилище LOS](#) и отображает подмножество сцены, вторая камера (например, глубина 1) отображает другое подмножество сцены. Сцену и использует маску [LOS](#) для объединения изображения, полученного обеими камерами.

## 2.1.2 ДОБАВЛЕНИЕ ИСТОЧНИКА ЛИНИИ ВИДИМОСТИ

Источник линии прямой видимости показывает части мира.

1. Создайте новый пустой [игровой объект](#) (Игровой объект > Создать пустой)
2. Добавьте [исходный компонент LOS](#) («Добавить компонент» > «Линия обзора» > «Источник LOS»).
3. Предыдущий шаг должен был добавить новый компонент [Camera](#) к тому же [GameObject](#).  
автоматически
4. Настройте эту камеру [на дальнюю плоскость отсечения](#) и [FOV](#), чтобы настроить область, охватываемую линией обзора.  
источник
5. Если исходная камера находится внутри объекта, обязательно включите этот объект с помощью источника.  
свойства маски отбраковки камер

Эффект теперь должен быть виден на вкладке игры или при запуске игры. Убедитесь, что камера, подключенная к исходному [игровому объекту LOS](#), видна основной камере. Для 360-градусной визуализации рекомендуется добавить компонент [LOS Source Cube](#) вместо компонента [LOS Source](#). Вместо этого вы также можете использовать один из встроенных источников LOS (расположенных в папке Line Of Sight/Prefabs).

## 2.2 ИСКЛЮЧЕНИЕ ОБЪЕКТОВ ИЗ МАСКИ ЛИНИИ ВИДИМОСТИ

Есть два способа исключать объекты из маски прямой видимости.

В первом методе используются компоненты сценария [LOS Stencil Mask](#) и [LOS Stencil Renderer](#). Он отображает исключенные объекты в буфере трафарета, который затем используется в качестве маски для эффекта изображения линии прямой видимости.

Второй метод использует компонент сценария [LOS Layer Excluder](#) и визуализирует исключенные объекты после эффекта изображения с помощью отдельной камеры.

У каждого из этих методов есть свои плюсы и минусы, и какой из них использовать, зависит от того, как настроен проект и сколько объектов необходимо исключать. Вот краткий обзор плюсов и минусов каждого метода:

### МЕТОД ТРАФАРНОЙ МАСКИ LOS

Поддерживает конвейеры прямой и отложенной рендеринга.

Требуется только одна камера.

Не поддерживает MSAA.

Дополнительный вызов отрисовки для каждого исключенного рендера.

### МЕТОД ИСКЛЮЧЕНИЯ СЛОЯ LOS

Поддерживает MSAA.

Объекты визуализируются только один раз.

Отложенный рендеринг не поддерживается.

Исключенные объекты должны быть организованы в слои.

### 2.2.1 ИСКЛЮЧЕНИЕ ОБЪЕКТОВ С ИСПОЛЬЗОВАНИЕМ КОМПОНЕНТА LOS STENCIL MASK

Исключение объектов с помощью компонентов [LOS Stencil Mask](#) и [LOS Stencil Renderer](#) очень просто.

- Если вы еще этого не сделали, настройте новую камеру (добавьте компоненты, необходимые для системы прямой видимости) для работы), следуя инструкции в кратком руководстве.
- Выберите новую камеру.
- Добавьте компонент сценария [LOS Stencil Mask](#) в [GameObject](#) новой камеры (Add Component > Line of Sight > Трафаретная маска LOS).
- Добавьте сценарий [LOS Stencil Renderer](#) к любому игровому объекту, который вы хотите исключить из прямой видимости (Add Component > Line of Sight > LOS Stencil Renderer).
- К этому [GameObject](#) уже должен быть прикреплен компонент [Mesh](#) или [Skinned Mesh Renderer](#).

### 2.2.2 ИСКЛЮЧЕНИЕ ОБЪЕКТОВ С ИСПОЛЬЗОВАНИЕМ КОМПОНЕНТА LOS LAYER EXCLUDER

Исключение набора объектов из маски линии обзора с помощью компонента [LOS Layer Excluder](#) требует использования нескольких камер, каждая из которых имеет разные свойства маски отрисовки. Первая камера будет отображать объекты, которые необходимо воздействовать на маску линии визирования, а затем вторая камера (с идентичными настройками) визуализирует оставшиеся объекты поверх.

Компонент сценария [LOS Layer Excluder](#) настроен для вас нескольких камер, поэтому вам не нужно делать это вручную.

Выполните следующие действия, чтобы настроить несколько камер с помощью сценария [исключения слоя LOS](#):

1. Если вы еще этого не делали, настройте основную камеру (добавив компоненты, необходимые для работы с темы [прямой видимости](#)), следуя инструкции в кратком руководстве.
2. Выберите основную камеру
3. Добавьте компонент скрипта [LOS Layer Excluder](#) в [GameObject](#) основной камеры (Add Component > Line of).  
Прицел > Исключение слоя прямой видимости
4. Задайте слои, которые вы хотите исключить из маски прямой видимости, используя свойство [Exclude Layers](#) в [LOS](#).  
Компонент сценария [Layer Excluder](#)
5. Это автоматически удалит выбранные слои из свойства [Culling Mask](#) основных камер (но если вы удалите слой из свойства [Exclude Layers](#), вам придется снова включить его вручную в [маску Culling Mask](#) основных камер).

Скрипт [LOS Layer Excluder](#) создаст дочерний объект с вторым компонентом камеры. Эта камера не будет удалена и может использоваться для применения эффектов изображения.

---

### 2.2.3 ПРЕДУПРЕЖДЕНИЕ ОБ ОШИБКЕ UNITY 1

При использовании нескольких камер вместе с прямой или отложенным рендерингом в единстве возникает ошибка, из-за которой вторая камера (например, глубина 1) иногда отображается неправильно, если для ее свойства «Очистить флаги» установлено значение «Не очищать» и если применен эффект изображения к первой камере (например, глубина 0).

Есть 2 способа решить эту проблему:

- Установите для свойства «Очистить флаги» первой камеры (глубина 0) значение «Только глубина».
- Включите сглаживание в настройках качества.

## 2.3 ПРЕДОТВРАЩЕНИЕ ОБЪЕКТОВ, ВЛИЯЮЩИХ НА ЛИНИЮ ВИДИМОСТИ

В некоторых случаях вы захотите, чтобы объекты закрывались линией обзора, но не влияли на саму линию обзора. Этого можно добиться, назначив объекту флажок `isOccluder` и включив этот слой из свойства [Culling Mask](#) в компоненте Camera, с выбранным [источником LOS](#).

Взгляните на «NPC» в примере сцены, чтобы понять, как это сделать.

## 2.4 КАК СКРЫТЬ ОБЪЕКТЫ, ЕСЛИ ОНИ НАХОДЯТСЯ ВНЕ ЛИНИИ ВИДИМОСТИ?

Несмотря на то, что объекты будут затемнены, когда они находятся за пределами прямой видимости (в зависимости от настроенных вами эффектов изображения прямой видимости), в некоторых случаях вы захотите скрыть их полностью (например, чтобы не выдавать их позиций своему игроку). Для этого просто добавьте компонент [LOS Object Hider](#) в [GameObject](#), содержащий объекты [Mesh](#) или [Skinned Mesh Renderer](#). Этот компонент будет включать или отключать тот рендеринг объектов в зависимости от того, виден ли объект или нет для любого из источников LOS в сцене.

## 3. ОБЗОР КОМПОНЕНТОВ

### 3.1 ЛОС МАСКА

Этот компонент является сердцем системы прямой видимости. Он визуализирует маску линии обзора и использует ее для создания окончательного изображения. Этот компонент реализован как эффект изображения и поэтому может быть добавлен только к [GameObject](#), содержащий компонент камеры.

#### 3.1.1 НАСТРОЙКИ

- Буферное хранилище: содержит ссылку на компонент [буферного хранилища LOS](#), используемый этой маской LOS.
- Уровень качества: устанавливает уровень качества
  - Высокое: удваивает разрешение всех исходных текстур рендеринга LOS.
  - Средний: по умолчанию без изменений
  - Низкое: вдвое уменьшает разрешение буферов маски прямой видимости.
- Маска HDR: включает отображение этой маски в HDR. При включенном HDR маска будет преобразована в Полуформатная визуализация текстуры ARGB вместо ARGB32. Для достижения наиболее точных результатов рекомендую использовать линейное [цветовое пространство](#) вместо гаммы для вашего проекта.

На изображении ниже показана разница между включенной и выключенной маской HDR.



## 3.2 ИСТОЧНИК ЛОС

Этот компонент должен быть добавлен к каждой камере, которая используется в качестве источника прямой видимости. Компонент источника LOS имеет несколько настроек, которые позволяют настроить внешний вид конуса прямой видимости.

### 3.2.1 НАСТРОЙКИ

- Цвет маски: устанавливает цвет
- Интенсивность маски: устанавливает интенсивность (наиболее заметна в HDR-режиме)
- Инверсия маски: инвертирует маску
- Render Target Width: управляет шириной этих источников.  
визуализировать цель
- Render Target Height: контролирует высоту этих источников  
визуализировать цель
- Затухание на расстоянии: определяет, насколько сильно конус прямой видимости исчезает на расстоянии
- Edge Fade: управляет мягкостью левого и правого краев изображения конуса
- Исключить задние грани: исключает из маски лицевые стороны, обращенные назад

#### В маске

- Минимальная дисперсия: помогает уменьшить артефакты в маске, лучше всего держать их как можно ниже.
- За пределами области: определяет, как обрабатываются пиксели выше и ниже усеченной пирамиды камеры.  
Пожалуйста, обратитесь к разделу 5.1 для более подробной информации.
  - Clamp: использует зафиксированное значение глубины для расчета видимости
  - Включить: область всегда видна
  - Исключить: область никогда не отображается

Другие настройки, такие как длина и ширина конуса прямой видимости, контролируются камерой напрямую. [Дальняя плоскость отсечения](#) камеры управляет длиной конуса, а [Глубе обзора](#) (вместе с настройками разрешения источника LOS) управляет шириной конуса.

Имейте в виду, что в Unity FOV управляет высотой камеры, а не шириной. Этот компонент имеет настраиваемый гизмо, который отображает правильную усеченную пирамиду камеры тем же цветом, что и настройка [цвета маски](#). Усеченная пирамида камеры, отображаемая в редакторе (компонентом камеры), неверна, потому что предполагается, что ее цель рендеринга имеет то же разрешение, что и разрешение, установленное в игровом режиме.

### 3.2.2 ПРЕДУПРЕЖДЕНИЕ ОХАРАКТЕРИСТИКАХ

Наличие нескольких различных разрешений для ваших компонентов [LOS Source](#) увеличит объем видеопамяти. Попробуйте использовать одно и то же разрешение для максимально возможного количества источников LOS, чтобы они могли использовать одну и ту же текстуру рендеринга в качестве цели рендеринга.

### 3.3 БУФЕРНОЕ ХРАНЕНИЕ LOS

Этот компонент используется для хранения буфера краев перед рендерингом эффектов изображения линии прямой видимости. Буфер краев, созданный этим компонентом, используется компонентом [маски LOS](#) для создания окончательного изображения. Этот компонент не требует никаких настроек.

### 3.4 ТРАФАРНАЯ МАСКА LOS

Этот компонент сценария служит заменой компонента сценария [LOS Layer Excluder](#). Он не требует нескольких камер и совместим с отложенным рендерингом. Он копирует Stencil Buffer (в который рисуют компоненты [LOS Stencil Renderer](#)) в Render Texture, которая считывается компонентом [LOS Mask](#) для включения объектов из линии обзора.

Этот компонент не требует никаких настроек.

#### 3.4.1 ПРЕДУПРЕЖДЕНИЕ

Этот компонент не поддерживает MSAA. Если вы используете упреждающий рендеринг и вам требуется MSAA, используйте вместо этого компонент [LOS Layer Excluder](#).

### 3.5 LOS STENCIL RENDERER

Этот компонент скрипта используется вместе с компонентом [LOS Stencil Mask](#) для включения [Mesh](#) или [Skinned Mesh Renderer](#) из линии видимости. Он создает командный буфер, который рисует [визуализатор](#) в буфере трафарета.

#### 3.5.1 НАСТРОЙКИ

- Предотвратить динамическую пакетную обработку: при динамической пакетной обработке [визуализаторов](#) возникает некоторая потеря точности, вероятно, вызванная слиянием нескольких мешей вместе и их матрицами преобразования.

комбинированный.

Поскольку компонент [LOS Stencil Renderer](#) снова визуализирует объект поверх их одного, эта потеря точности может привести к мерцанию объекта из-за небольших различий в значениях хранения в глубине буфера.

Включение этого параметра предотвратит динамическую пакетную обработку [Renderer](#), сделав его материалом уникальным. Этот параметр следует использовать с осторожностью только после возникновения проблемы. Когда вы пытаетесь включить этот параметр для [GameObject](#), который помечен как статический в редакторе, будет отображаться предупреждение, и параметр вернется к значению `false`.

Еще один способ решить эту проблему — полностью отключить [динамическую пакетную обработку](#) в настройках [проигрывателя Unity](#), но это может оказаться приемлемым решением не для каждого проекта.

#### 3.5.2 ПРЕДУПРЕЖДЕНИЕ

Этот компонент должен быть прикреплен к [GameObject](#), содержащему [Mesh](#) или [Skinned Mesh Renderer](#).

## 3.6 ИСКЛЮЧИТЕЛЬНО СЛОЯ LOS

Этот компонент используется для исключения слоев из поля зрения. Пожалуйста, обратитесь к разделу 2.2 для получения дополнительной информации о том, как его настроить.

### 3.6.1 НАСТРОЙКИ

- Исключить слои: выберите, какие слои исключить из поля зрения.

## 3.7 ЛОС КУЛЛЕР

Этот компонент проверяет, виден ли [GameObject](#), к которому он присоединен, любому из источников LOS. Вы можете проверить, виден ли этот объект, получив свойство [Visible](#). Этот компонент используется компонентами [LOS Object Hider](#) и [LOS Object Revealer](#) для скрытия объекта за пределами зоны прямой видимости.

Этот компонент сначала проверит, попадают ли границы сетки присоединенного компонента [Renderer](#) внутрь одной из усеченных камер источников LOS. Если границы находятся внутри усеченной камеры, он проверит, не блокирует ли другой объект линию обзора, используя [raycasts](#). Поскольку этот компонент использует [raycasts](#), объекты, которые должны его блокировать, требуют коллайдера.

### 3.7.1 НАСТРОЙКИ

- Маска слоя Raycast: выберите, какие слои блокируют [raycasts](#), используемые для расчета видимости.

### 3.7.2 ПРЕДУПРЕЖДЕНИЕ

Этот компонент должен быть присоединен к [GameObject](#), содержащему [Mesh](#) или [Skinned Mesh Renderer](#), потому что он использует границы сетки для вычисления своей видимости.

## 3.8 ИНФОРМАЦИЯ О ВИДИМОСТИ

Этот компонент предоставляет список всех источников LOS, к которым виден [GameObject](#) (к которому он прикреплен). Вы можете проверить, для каких источников LOS виден объект, извлекая свойство [VisibleSources](#).

Этот компонент также предоставляет несколько событий C#, на которые вы можете подписаться и которые запускаются, когда [GameObject](#) входит, выходит или остается в поле зрения источников LOS.

Этот компонент использует те же функции для расчета своей видимости, что и компонент [LOS Culler](#).

### 3.8.1 НАСТРОЙКИ

- Маска слоя Raycast: выберите, какие слои блокируют [raycasts](#), используемые для расчета видимости.

### 3.8.2 СОБЫТИЯ

- OnLineOfSightEnter: срабатывает, когда объект входит в их общее поле зрения LOS.

- `OnLineOfSightStay`: срабатывает, когда объект остается в поле зрения источника LOS
- `OnLineOfSightExit`: срабатывает, когда объект выходит из поля зрения источника LOS.

---

### 3.8.3 ПРЕДУПРЕЖДЕНИЕ

Этот компонент должен быть присоединен к `GameObject`, содержащему `Mesh` или `Skinned Mesh Renderer`, потому что он использует границы сетки для вычисления своей видимости.

---

### 3.8.4 ПРЕДУПРЕЖДЕНИЕ ОПЕРАТИВНЫХ ХАРАКТЕРИСТИКАХ

Этот компонент вычисляет видимость объектов по отношению к каждому источнику LOS в сцене, поэтому он может быть очень ресурсоемким, если в вашей сцене много источников LOS.

## 3.9 СКРЫТИЕ ОБЪЕКТОВ ПОТЕРИ

Этот компонент используется для скрытия объектов за пределами прямой видимости. Он приостанавливает визуализатор `GameObjects`, когда для свойства `Visibility` компонентов `LOS Culler` установлено значение `false`.

Этот компонент требует, чтобы компонент `LOS Culler` был прикреплен к тому же `GameObject`. Когда вы добавляете этот компонент, он будет добавлен автоматически, если он еще не существует.

---

### 3.9.1 ПРЕДУПРЕЖДЕНИЕ

Этот компонент должен быть прикреплен к `GameObject`, содержащему `Mesh` или `Skinned Mesh Renderer`.

## 3.10 ПОИСК ОБЪЕКТОВ ЛОС

Этот компонент используется для скрытия объектов до тех пор, пока они не появятся на линии обзора. После обнаружения объекты становятся видимыми.

---

### 3.10.1 ПРЕДУПРЕЖДЕНИЕ

Этот компонент должен быть прикреплен к `GameObject`, содержащему `Mesh` или `Skinned Mesh Renderer`.

## 4. УСТАНОВКИ КОМПОНЕНТЫ

### 4.1 LOS CAMERA SYNC (УСТАНОВКИ ЛОБ ВЕРСИИ 1.1)

Устарело в версии 1.1 в пользу нового скрипта [LOS Layer Excluder](#). Удалено в версии 1.2.

Этот компонент следует использовать только при использовании нескольких камер для исключения набора объектов из маски прямой видимости (с.м. Исключение объектов из маски прямой видимости). Он синхронизирует свойства основной камеры с вспомогательной камерой, содержащей этот компонент скрипта, так что обе камеры отображают сцену одинаково.

Этот компонент также устанавливает для свойства «Очистить флаги» значение «Не очищать» (второстепенная камера не должна очищать то, что визуализировалось первой камерой, и должна использовать тот же буфер глубины) и обеспечивает правильную настройку глубины второй камеры, чтобы она рендерилась после основной камеры.

### 4.2 LOS SOURCE CUBE (УСТАНОВКИ ЛОБ ВЕРСИИ 1.2.2)

Устарело в версии 1.2.2 из-за проблем с совместимостью на платформах Mac.

Этот компонент разработан специально для быстрой визуализации линии прямой видимости на 360 градусов.

Этот источник выполняет рендеринг в текстуру кубической карты (вместо обычной текстуры рендеринга), уменьшая количество проходов шейдера, необходимых для полного обзора на 360 градусов, с 6 до 1, что значительно повышает производительность. Есть пара недостатков; этот источник прямой видимости менее точен, чем обычный компонент [источника LOS](#) (он использует закодированные 16-битные плавающие вместо 32bit), а также не поддерживает размытие.

Большинство настроек этого компонента идентичны настройкам компонента [LOS Source](#), за некоторыми исключениями. Изменение [поля зрения](#) камеры не имеет никакого эффекта, так как она всегда будет отображать весь окружающий мир. Через свойство [Cube Map Resolution](#) можно указать только одно разрешение, поскольку высота и ширина кубических карт должны быть одинаковыми.

Это разрешение также должно быть степенью двойки (64, 128, 256, 512, 1024 и т.д.). Если вы введете число, не равное степени двойки, оно будет округлено до ближайшего числа двойки (например, 300 станет 512).

#### 4.2.1 НАСТРОЙКИ

- Цвет маски: устанавливает цвет
- Интенсивность маски: устанавливает интенсивность (наиболее заметно в режиме HDR).
- Инверсия маски: инвертирует маску
- Разрешение карты куба: управляет шириной и высотой цели рендеринга из одной карты куба.
- Затухание на расстоянии: определяет степень затухания линии обзора на расстоянии.
- Минимальная дисперсия: помогает уменьшить артефакты в маске, лучше всего держать их как можно ниже.

Радиус линии визирования контролируется камерами в [дальней плоскости отсечения](#) напрямую. Сохранение этого как можно меньшего значения приведет к максимальной точности.

Этот компонент имеет настраиваемый [гизмо](#). Сфера гизмо предоставляет собой область, на которую влияет линия обзора. Цвет этой сферы соответствует цвету, установленному в свойстве [Mask Color](#).

#### 4.2.2 ПРЕДУПРЕЖДЕНИЕ ОХАРАКТЕРИСТИКАХ

Наличие нескольких различных разрешений для компонентов [LOS Source Cube](#) увеличит необходимую видеопамять.

Старайтесь использовать одинаковое разрешение для максимально возможного количества компонентов [исходного куба LOS](#), чтобы они могли использовать одну и ту же текстуру рендеринга карты куба в качестве цели рендеринга.

#### 4.3 LOS FINAL RESOLVE (УСТАРЕЛО В ВЕРСИИ 1.4.0)

Устарело в версии 1.4.0 в пользу нового компонента [LOS Stencil Mask](#).

Этот компонент следует использовать только при использовании отложенного рендеринга и нескольких камер для исключения набора объектов из маски прямой видимости (см. раздел 2.2. Исключение объектов из маски прямой видимости). Он используется для исправления ошибки в Unity при использовании конвейера отложенного рендеринга в сочетании с несколькими камерами (рендеринг в одну и ту же цель рендеринга) и эффектами изображения. Пожалуйста, обратитесь к этому сообщению на форуме Unity для получения более подробной информации: <http://forum.unity3d.com/threads/deferred-renderer-multiple-cameras-posts-ssao-bugged-ufps-onrenderimage.198632/>

## 5. ИЗВЕСТНЫЕ ПРОБЛЕМЫ

### 5.1 УГЛУБЛЕННАЯ КАМЕРА

На рисунках ниже показана «проблема», возникающая, когда источник прямой видимости приближается к стене. Верхняя часть стены и часть пола не будут видны камере, прикрепленной к этому источнику. В реальности человек, стоящий близко к стене и смотрящий прямо перед собой, тоже будет видеть только часть стены.

Несмотря на то, что эта система точно визуализирует то, что можно увидеть с определенной позиции, это не всегда может быть желаемым результатом. Свойство [Out of Bound Area](#) компонента [LOS Source](#) позволяет вам управлять тем, что происходит с пикселями выше и ниже усеченной пирамиды камеры (на изображении окрашены в **красный цвет**).

Значение по умолчанию — `Clamp`, при котором для расчета видимости будет использоваться фиксированное значение глубины. Эта настройка хорошо работает в большинстве случаев, но начинает создавать артефакты, когда вы находитесь близко к стене. Один из способов уменьшить эти артефакты — увеличить высоту целевой текстуры рендеринга [LOS Source](#) по отношению к ширине.

Параметр `Include` замаскирует область за пределами границы как видимую. Это вызовет некоторые артефакты, когда, например, стена на изображении выше видна с правой стороны.

Настройка `Include` является наиболее точной, она замаскирует область за пределами границы как невидимую.

## 5.2 СОХРАНЕНИЕ СЦЕН

Поскольку некоторые сценарии в этом ресурсе используют атрибут `ExecuteInEditMode` (чтобы разрешить предварительный просмотр изменений в окне игры редактора), сцены, в которых используются эти компоненты, будут помечены как несохраненные, даже если не было внесено никаких изменений.

## 6. ПРИМЕР СЦЕН

В примере сцены используются все доступные компоненты LOS, и она может служить справочной информацией о том, как правильно их настроить. Вы можете безопасно удалить папку примера сцены (или вообще не импортировать ее), если она вам не нужна.

Сцена не использует никаких пользовательских слоев или тегов, чтобы не мешать настройкам вашего проекта, поэтому имейте в виду, что слои, которые используются в этой сцене, могут выглядеть нелогичными. Слой `TransparentFX` используется для объектов, которые не должны влиять на линию обзора. Слой `Water` используется для объектов, которые влияют на линию обзора, но должны быть скрыты маской.

Синие капсулы представляют собой очень простых NPC, управляемых ИИ. Они используют сценарий `LOSTObjectHider`, чтобы отключить свой рендерер (фактически скрывать их), когда они находятся за пределами прямой видимости.

Когда вы запускаете сцену в игровом режиме, вся мебель будет скрыта до тех пор, пока она не появится в поле зрения.

После этого она станет видимой даже за пределами прямой видимости. Этот эффект достигается за счет добавления к объекту скрипта `LOSTObjectRevealer`.

### 6.1.1 ОРГАНЫ УПРАВЛЕНИЯ

- Движение персонажа: клавиши со стрелками или WASD.
- Оглядываться вокруг: Мышь
- Поместите камеру с источником LOS: пробел
- Уровень сброса: R



## 7. ЖУРНАЛ ИЗМЕНЕНИЙ

### 7.1 ВЕРСИЯ 1.4

- Ресурс сделан совместимым с Unity версии 5.2 и выше.
- Добавлен новый метод иключения объектов при использовании конвейера отложенного рендеринга.
- Добавлен компонент LOS Stencil Mask.
- Добавлен компонент LOS Stencil Renderer.
- Исправлена ошибка, из-за которой шейдер LOS Mask пытался скомпилироваться в SM 2.0.

### 7.2 ВЕРСИЯ 1.3

- Добавлены настройки обратной стороны в компонент LOS Source для удаления общих артефактов.
- Добавлены заголовки в компонент LOS Source.
- Удален компонент LOSSourceCube (устарел в версии 1.2.2).
- Удалено размытие (экспериментальная функция).

### 7.3 ВЕРСИЯ 1.2.3

- Добавлен компонент LOS Visibility Info.
- Повышена точность компонента LOS Culler.
- Исправлены проблемы с Unity версии 5.3.2f.

### 7.4 ВЕРСИЯ 1.2.2

- Устаревший LOSSourceCube.

### 7.5 ВЕРСИЯ 1.2.1

- Исправление кубической карты для LOSSourceCube.

### 7.6 ВЕРСИЯ 1.2

- Добавлен компонент LOSSourceCube для более быстрого кругового обзора.
- Рефакторинг и оптимизация существующей кодовой базы.
- Исправлены проблемы реконструкции мирового пространства с орфографическими камерами.
- Добавлены гизмо.
- Добавлены всплывающие подсказки.
- Обновлен пример сцены.
- Улучшена обработка ошибок.

## 7.7 ВЕРСИЯ 1.1

- В маску LOS добавлен режим HDR.
- Добавлено свойство Color Mask для источника LOS
- Добавлено свойство Intensity для источника LOS
- Добавлено свойство Out of Bound Area для источника LOS •

Добавлена опция Invert Mask для источника LOS •

Добавлено свойство Layer Mask для LOS Culler •

Добавлены префабы-примеры •

Добавлено исключение слоя LOS компонент сценария •

Упрощенный процесс настройки для нескольких камер

- Улучшенная производительность

## 8. ПОДДЕРЖКА

Отзывы, предложения или вопросы?

Пожалуйста, заполните нашу форму поддержки [здесь](http://www.entropi-games.com/unity-asset-store-support/): <http://www.entropi-games.com/unity-asset-store-support/>