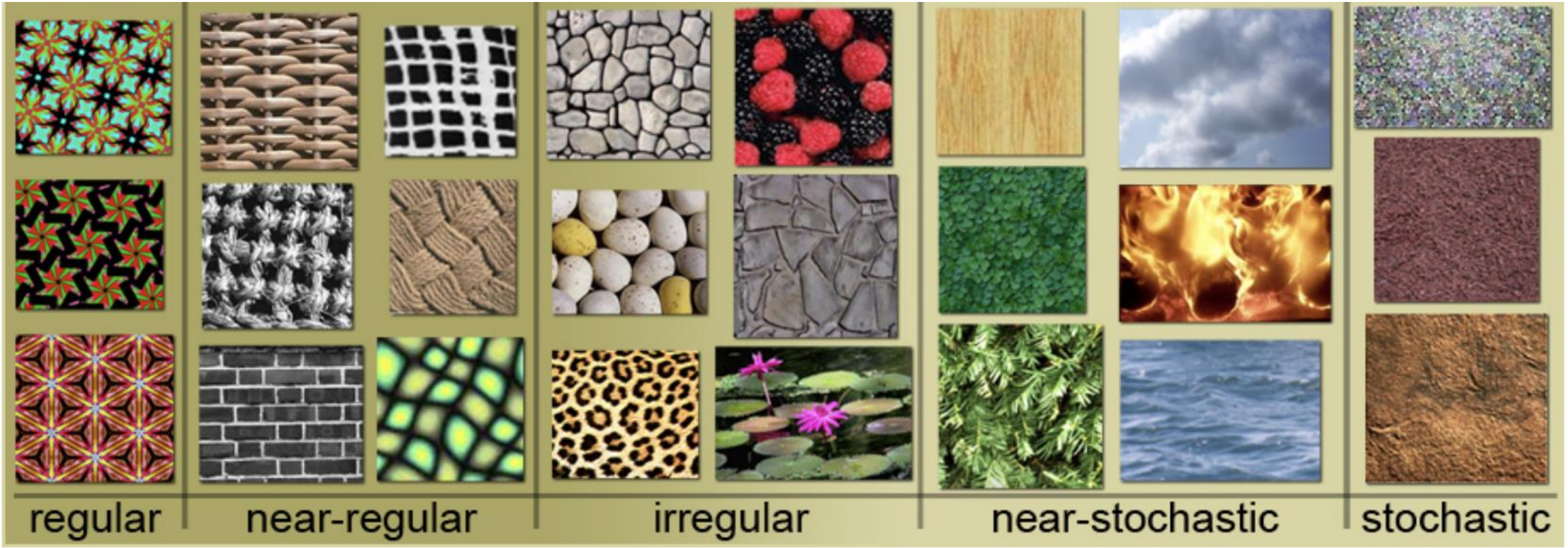
Растеризация. Освещение. Текстурирование

Компьютерная графика

План

- Растеризация
- Освещение
- Шейдинг
- Текстурирование

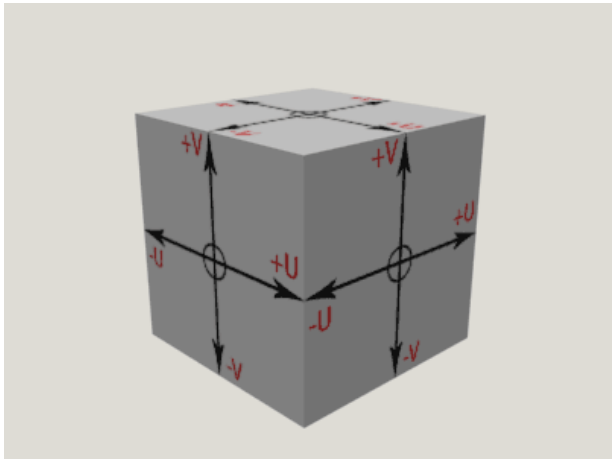
Разнообразие текстур

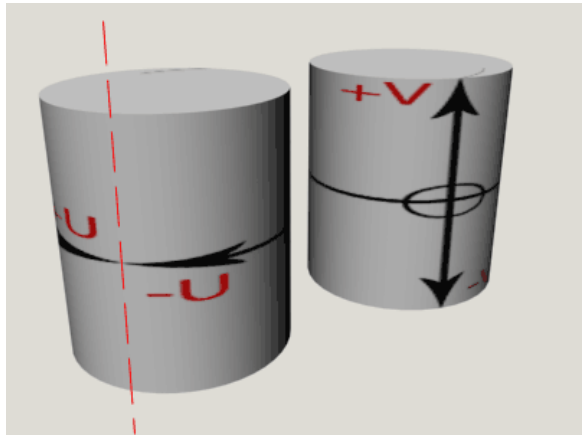


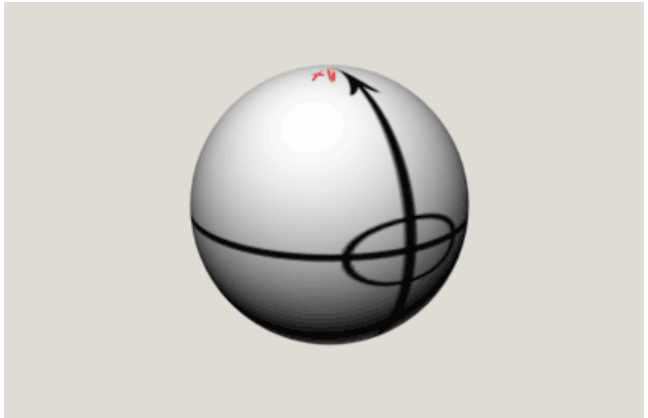
Типы текстур

- Diffuse / Albedo Map: Основной цвет поверхности
- Normal Map: Карта нормалей. Имитирует мелкий рельеф, не меняя геометрию
- Specular / Roughness / Metallic Map (PBR): Управление отраженными свойствами
- Displacement / Height Map: Настоящее изменение геометрии (тесселяция)
- Emissive (свечение)
- Ambient Occlusion (затенение трещин)
- Opacity (прозрачность)

Параметрическое представление

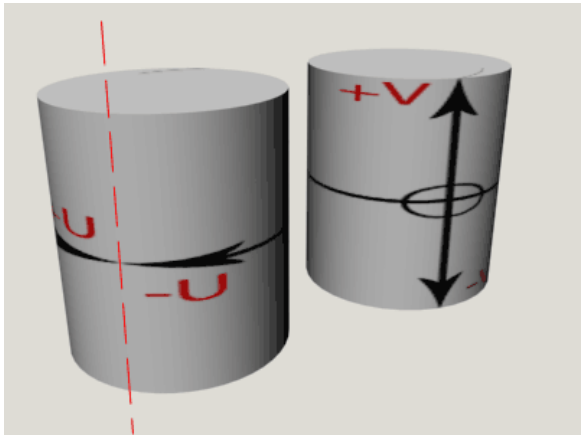


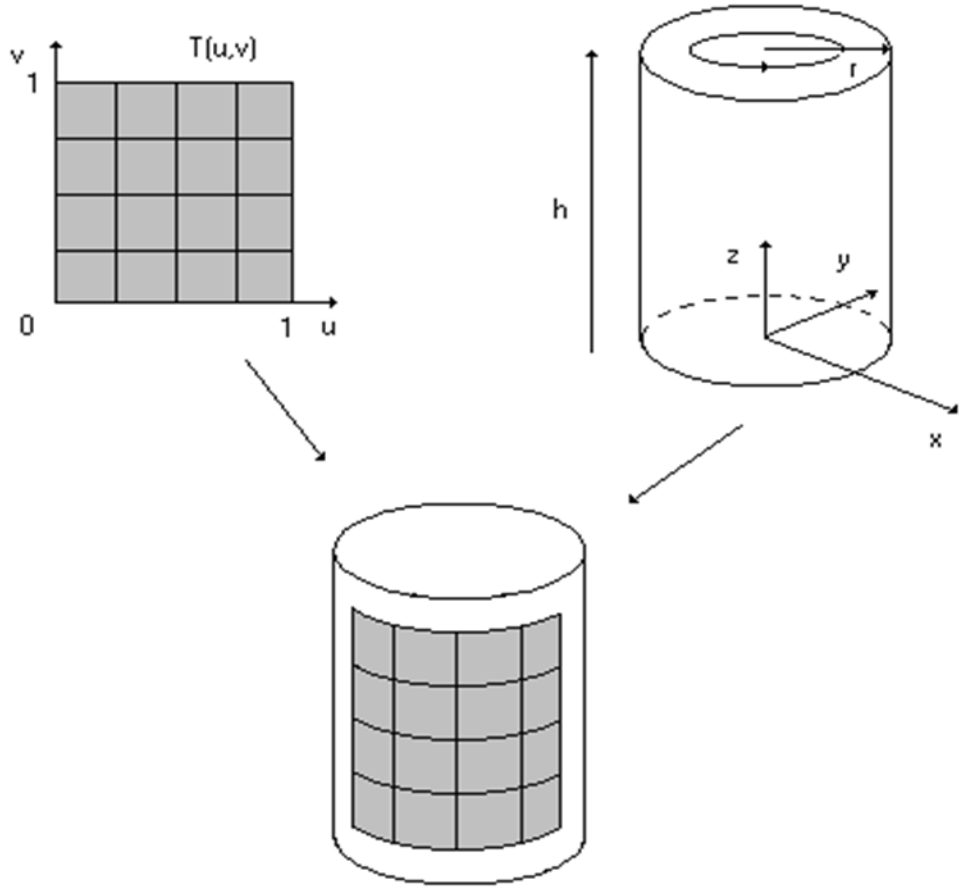




Цилиндрическая поверхность

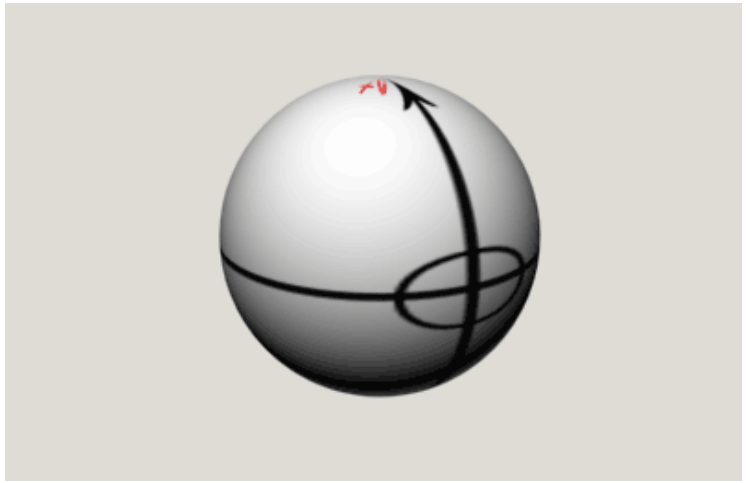
$$\begin{cases} x &= \rho\cos\varphi \\ y &= \rho\sin\varphi \\ z &= h \end{cases}, \quad \varphi \in [0, 2\pi)$$

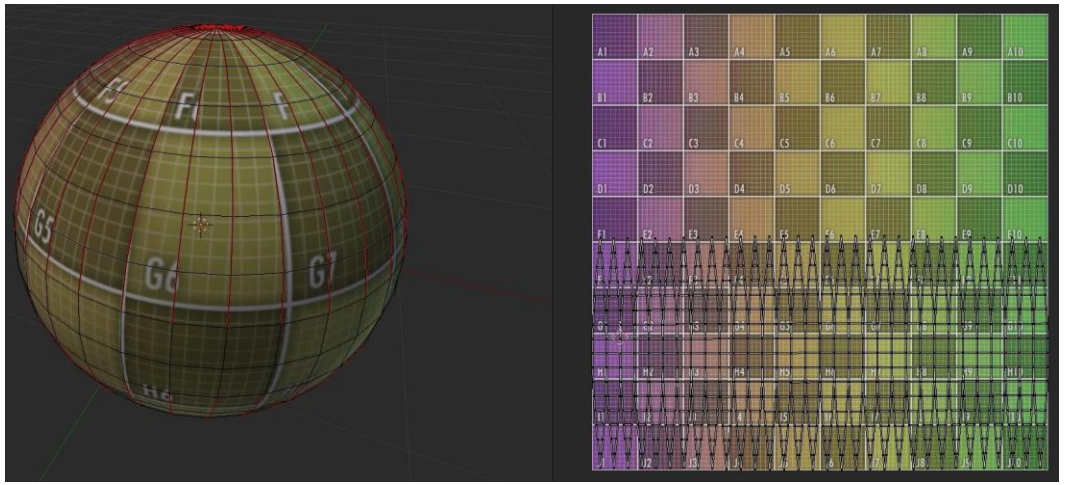




Сфера

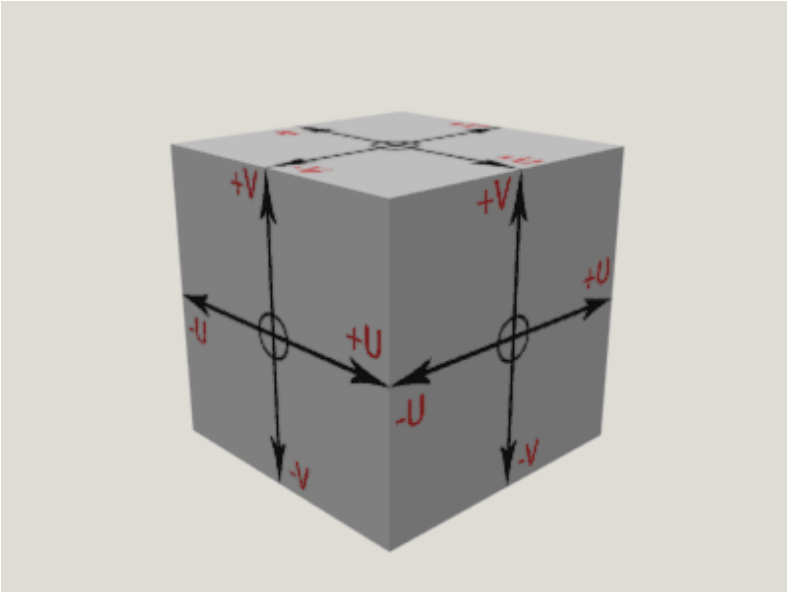
$$\begin{cases} x &= \rho\cos\varphi\cos\theta \\ y &= \rho\cos\varphi\sin\theta \\ z &= \rho\sin\varphi \end{cases}, \quad \varphi \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], \ \theta \in [0, 2\pi)$$

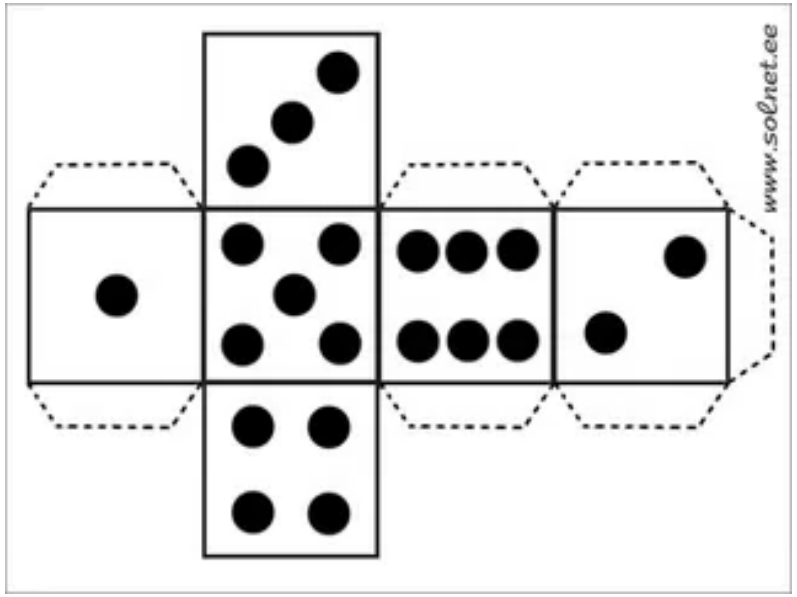




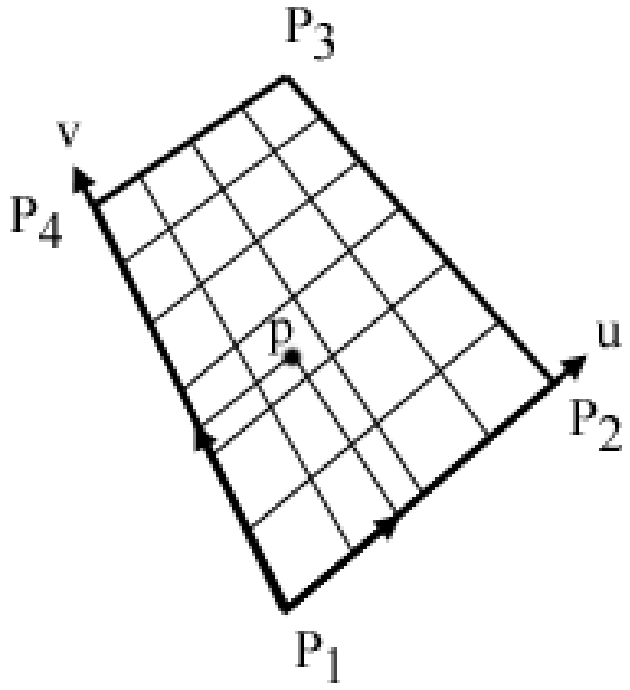
Куб - грань

$$p = a + ue_1 + ve_2$$





Текстурирование граней

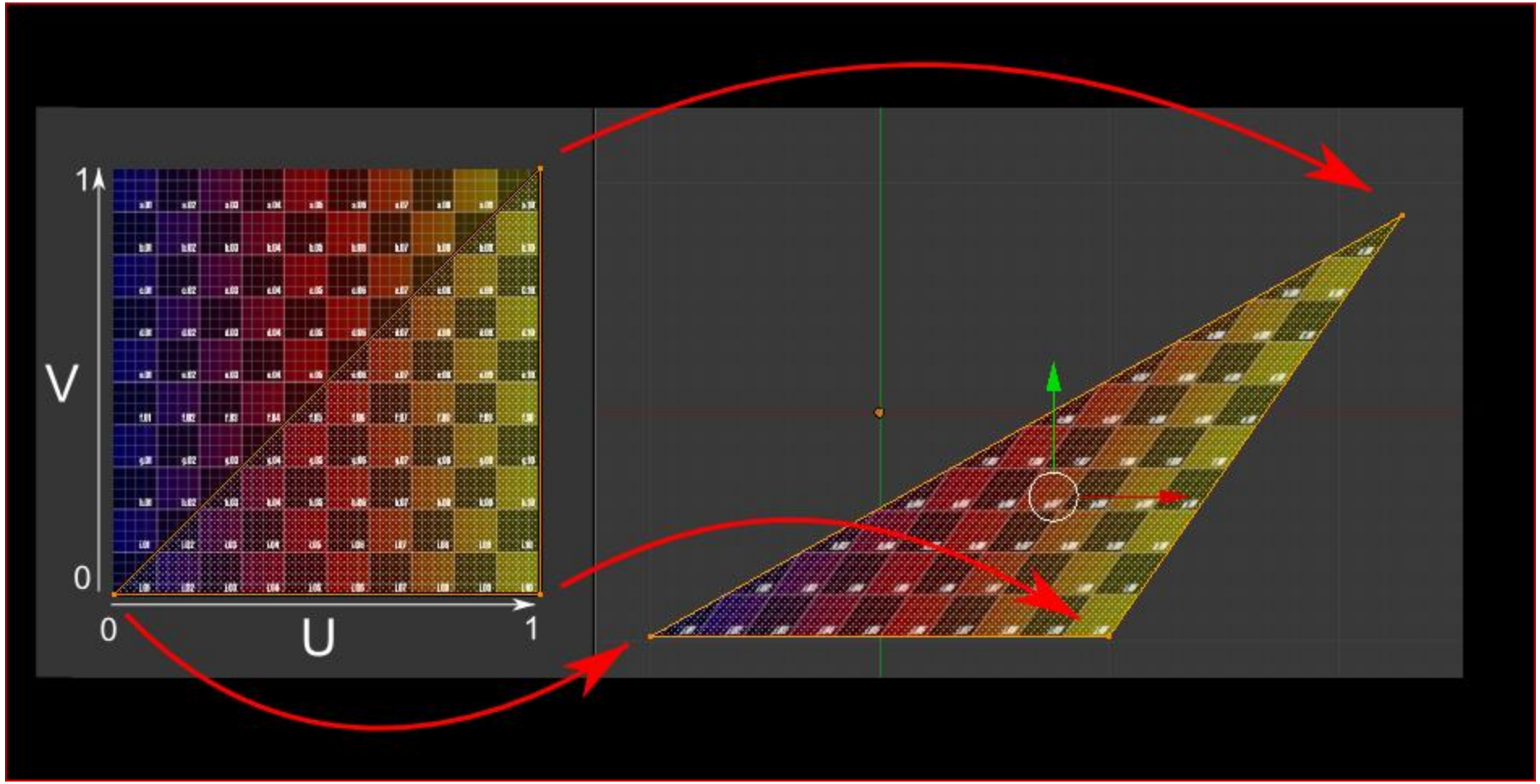


$a = P_1$;
$e_1 = P_2 - P_1$;
$e_2 = P_4 - P_1$;

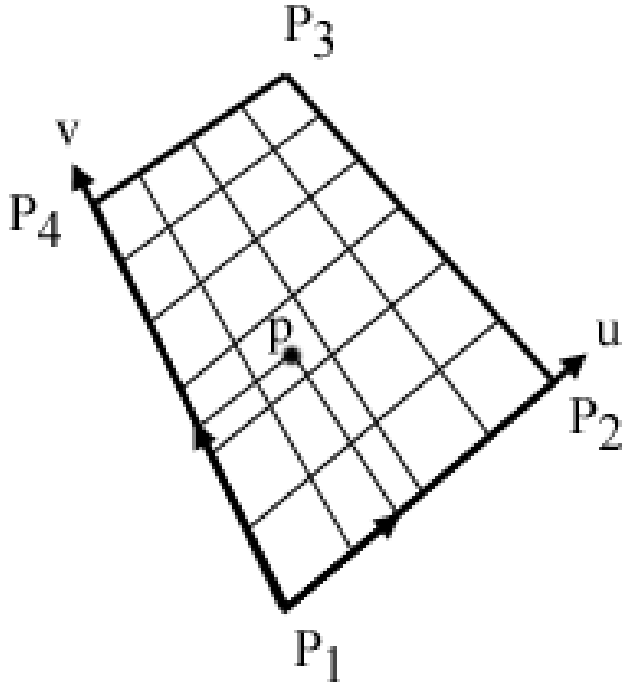
$p = a + ue_1 + ve_2$

$n = [e_1, e_2]$

Искажения при проективной проекции



Текстурирование граней при перспективной проекции



X = x / z, Y = y / z

$p = a + ue_1 + ve_2$

$$X = \frac{a_x + ue_{1_x} + ve_{2_x}}{a_z + ue_{1_z} + ve_{2_z}}$$

$$Y = \frac{a_y + ue_{1_y} + ve_{2_y}}{a_z + ue_{1_z} + ve_{2_z}}$$

$$\begin{cases} u(Xe_{1_z} - e_{1_x}) + v(Xe_{2_z} - e_{2_x}) = a_x - a_z X \\ u(Ye_{1_z} - e_{1_y}) + v(Ye_{2_z} - e_{2_y}) = a_y - a_z Y \end{cases}$$

Решение системы - Правило Крамера

$$\begin{cases} u(Xe_{1_z} - e_{1_x}) + v(Xe_{2_z} - e_{2_x}) = a_x - a_z X \\ u(Ye_{1_z} - e_{1_y}) + v(Ye_{2_z} - e_{2_y}) = a_y - a_z Y \end{cases}$$

$$u = \frac{\Delta u}{\Delta}, \quad v = \frac{\Delta v}{\Delta}$$

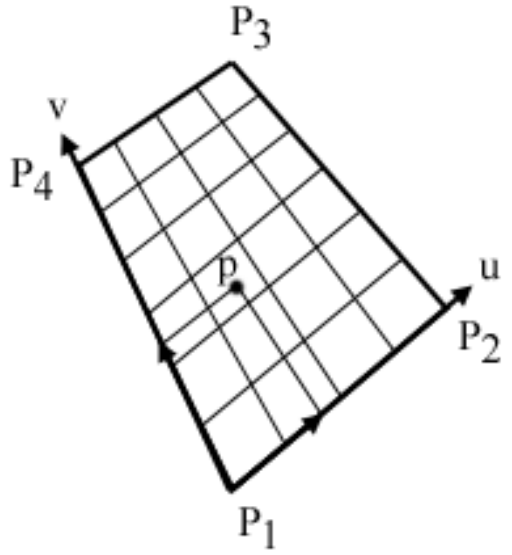
$$\Delta = Xn_x + Yn_y + n_z$$

$$\Delta u = Xm_x + Ym_y + m_z$$

$$\Delta v = Xl_x + Yl_y + l_z$$

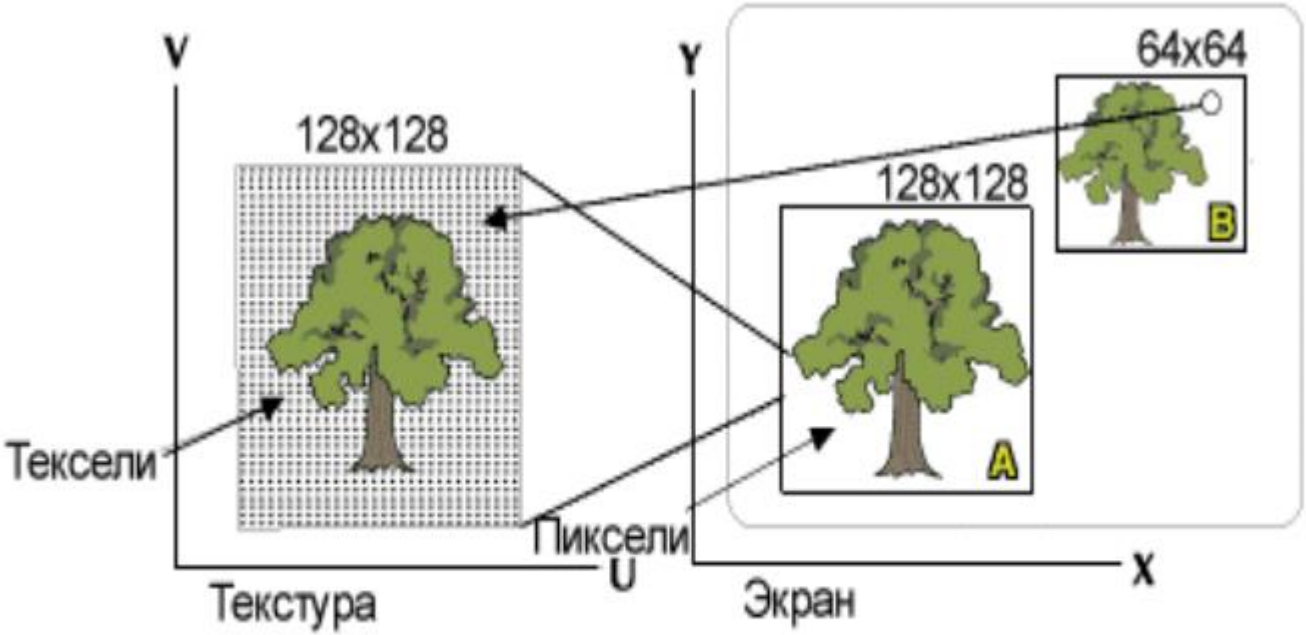
n = [e_1, e_2] ,
m = [e_2, a],
l = [a, e_1]

a = P_1;
e_1 = P_2 - P_1;
e_2 = P_4 - P_1;



$$\begin{pmatrix} \Delta u \\ \Delta v \\ \Delta \end{pmatrix} = \begin{pmatrix} m_x & m_y & m_z \\ l_x & l_y & l_z \\ n_x & n_y & n_z \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

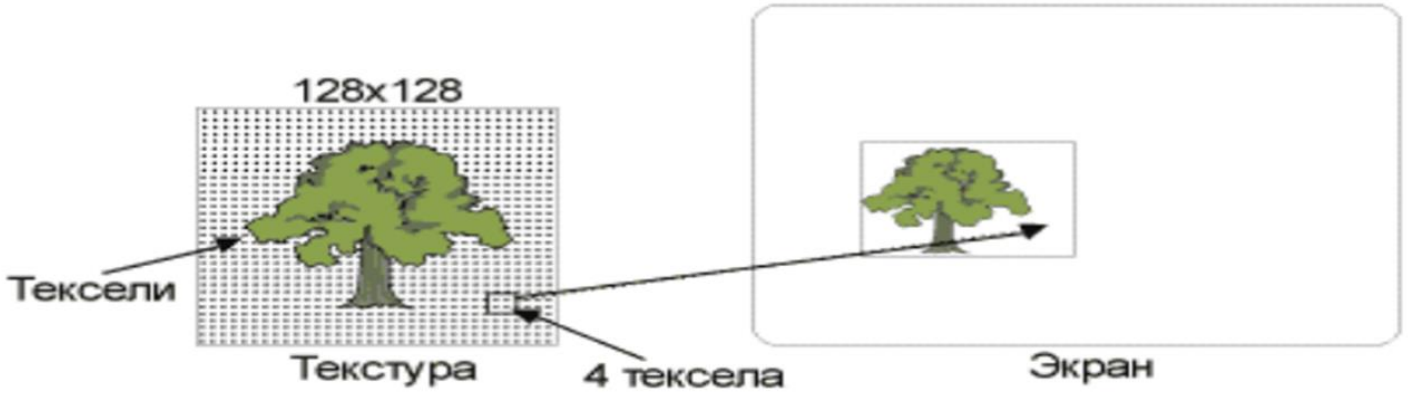
Текстурирование. Сэмплинг

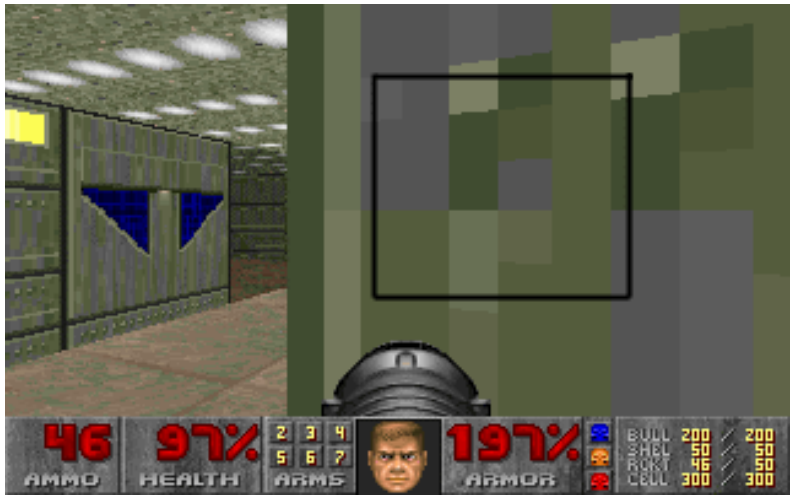


A: однозначное соответствие между текселами и пикселами
B: pixel 0:texel 0; pixel 1:texel 2; pixel 2:texel 4...

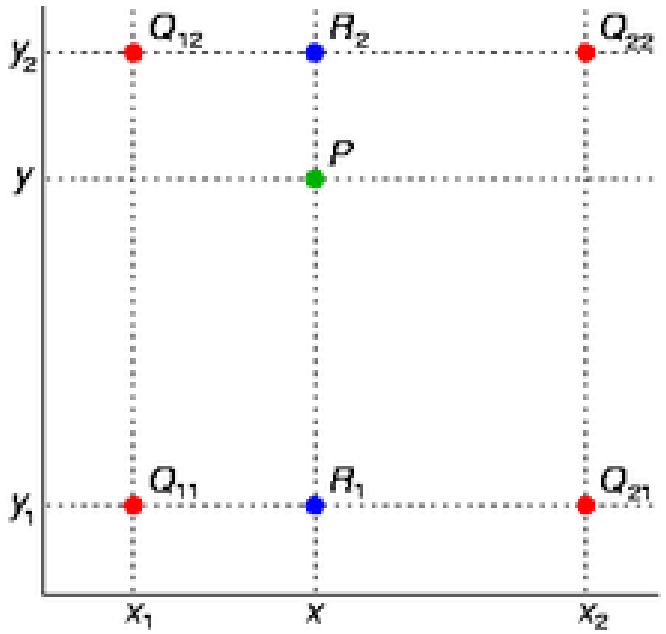
Масштабирование: прямая и обратная проблемы

- несколько текселей в один пиксель
- одному текселю соответствует несколько пикселей экрана





Устранение этого недостатка - билинейная интерполяция



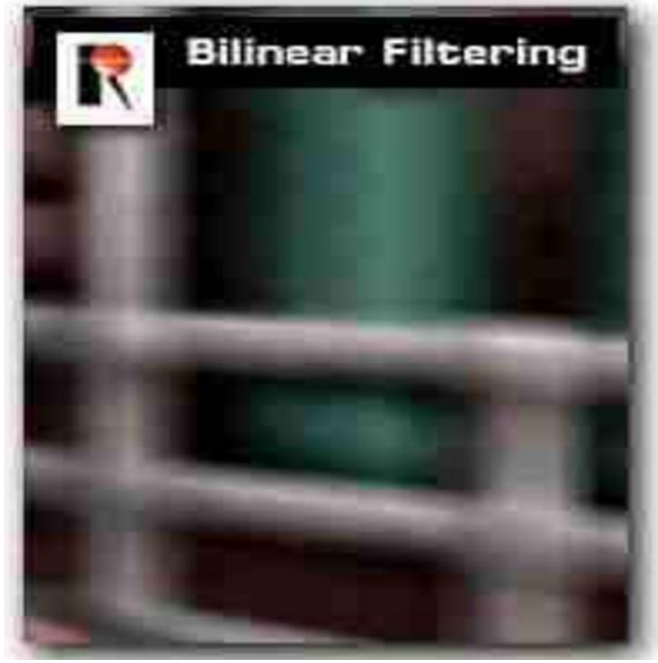
Четыре красные точки представляют собой известные значения функции.

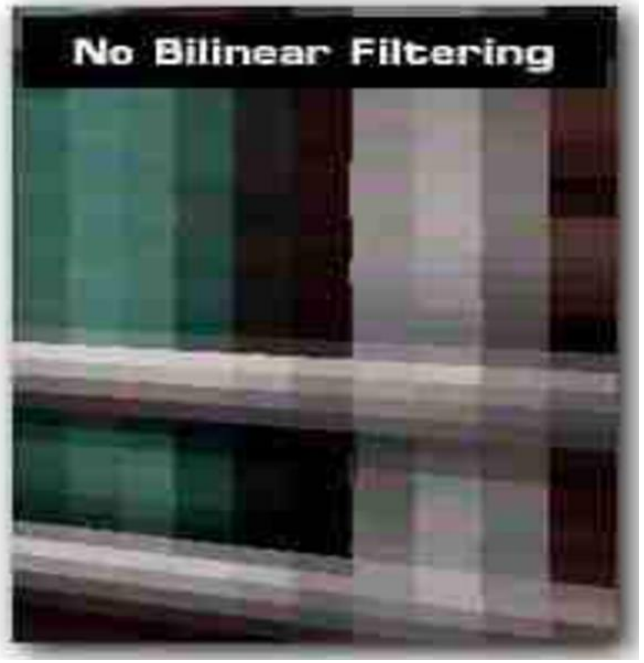
Значение в зеленой точке должно быть интерполировано.

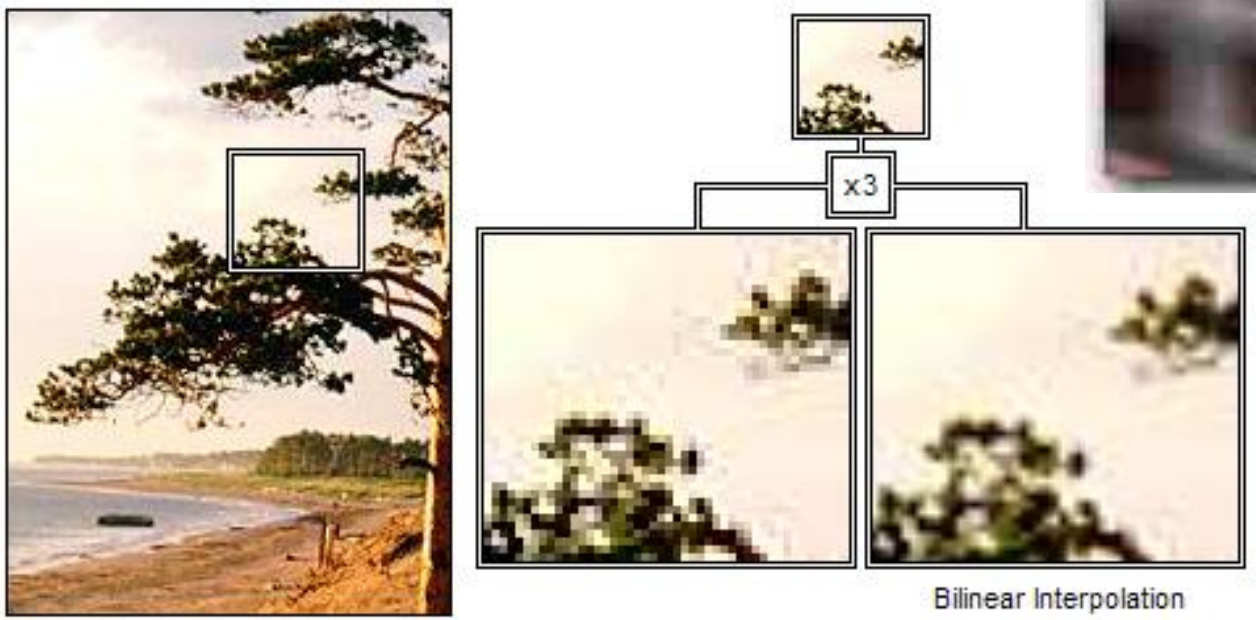
Пример увеличения части изображения

простым масштабированием

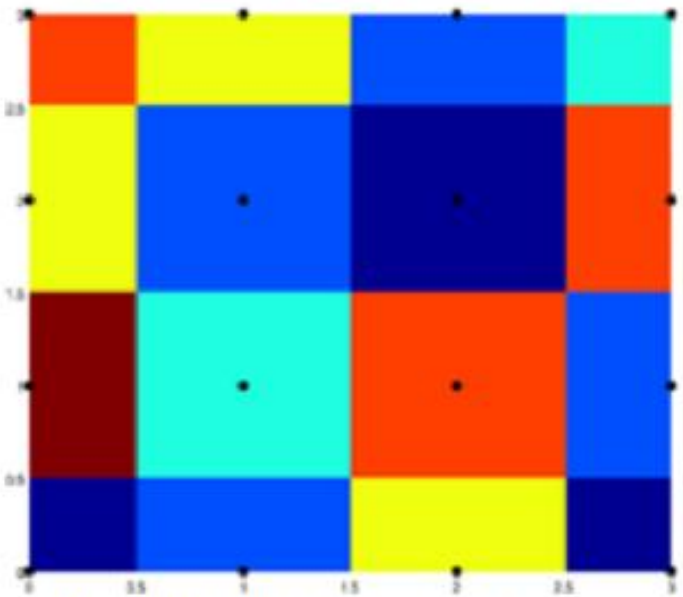
с применением билинейной интерполяции



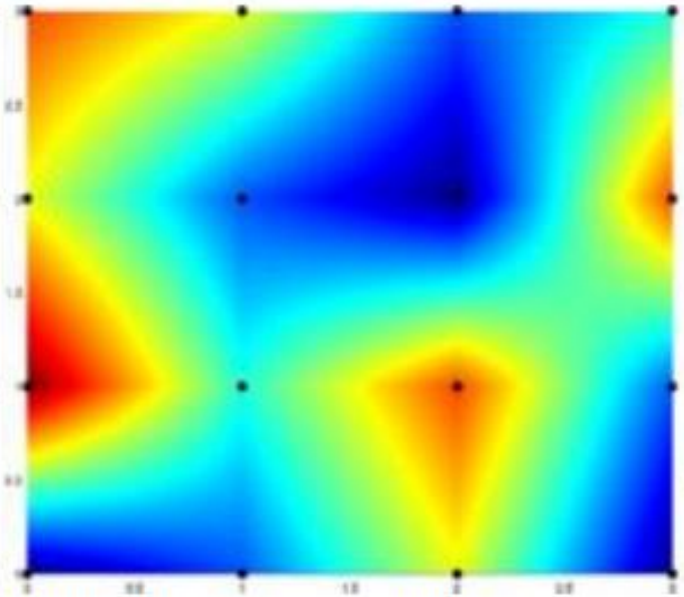




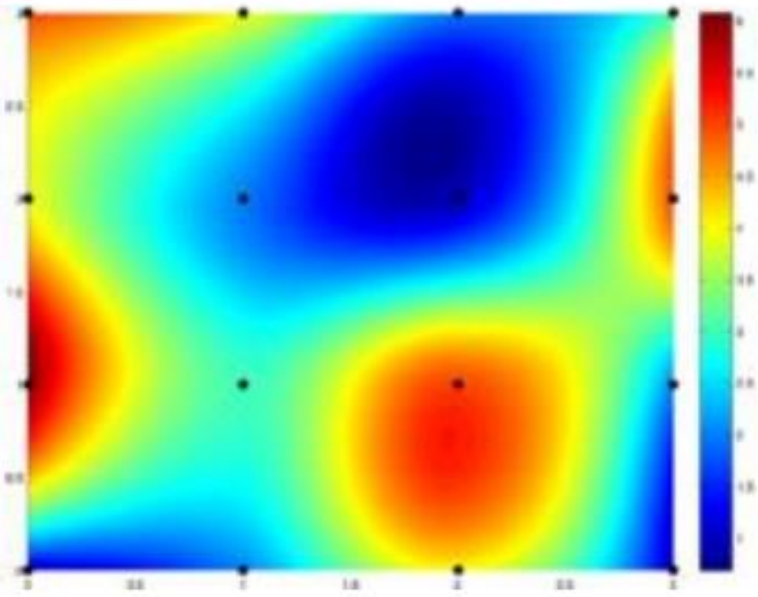
Интерполяция



Интерполяция методом ближайшего соседа

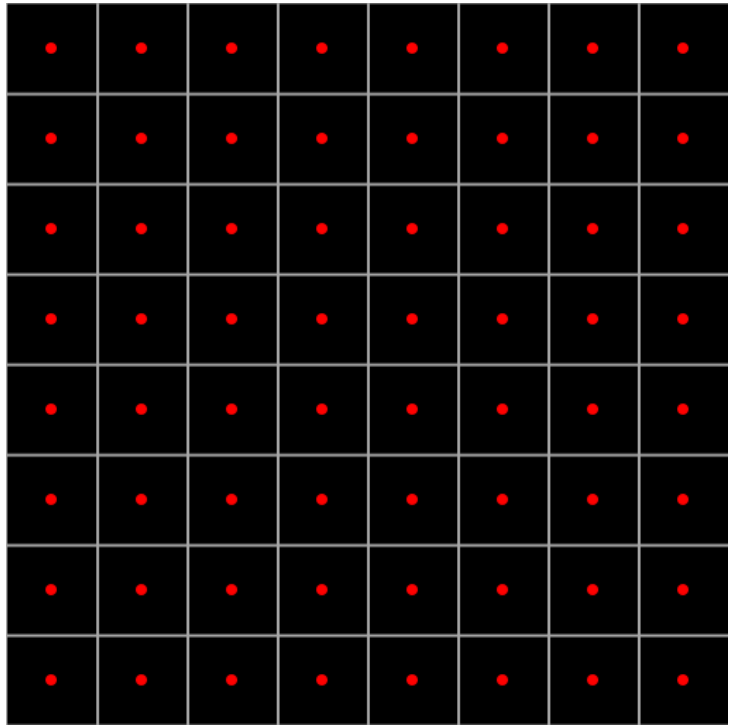


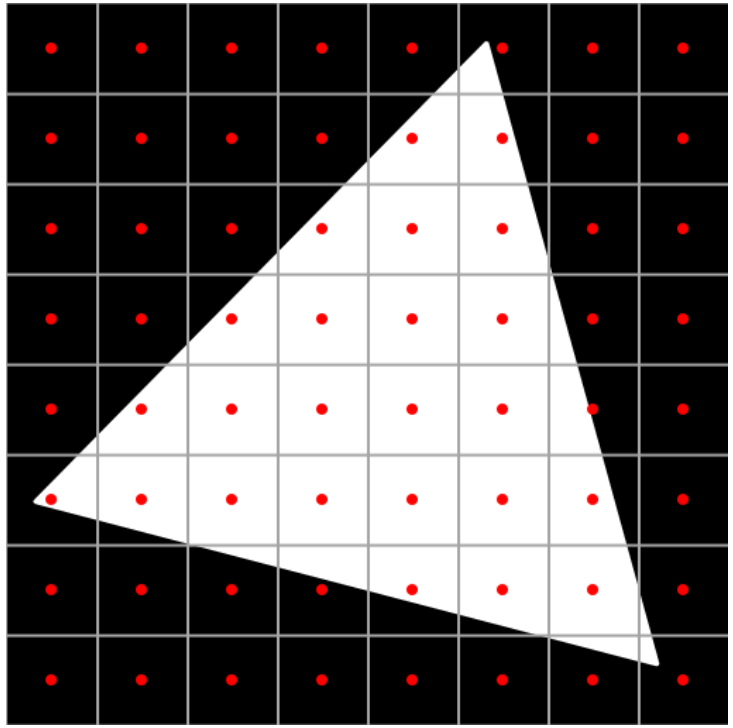
Билинейная интерполяция. Частные производные терпят разрыв на границах квадратов

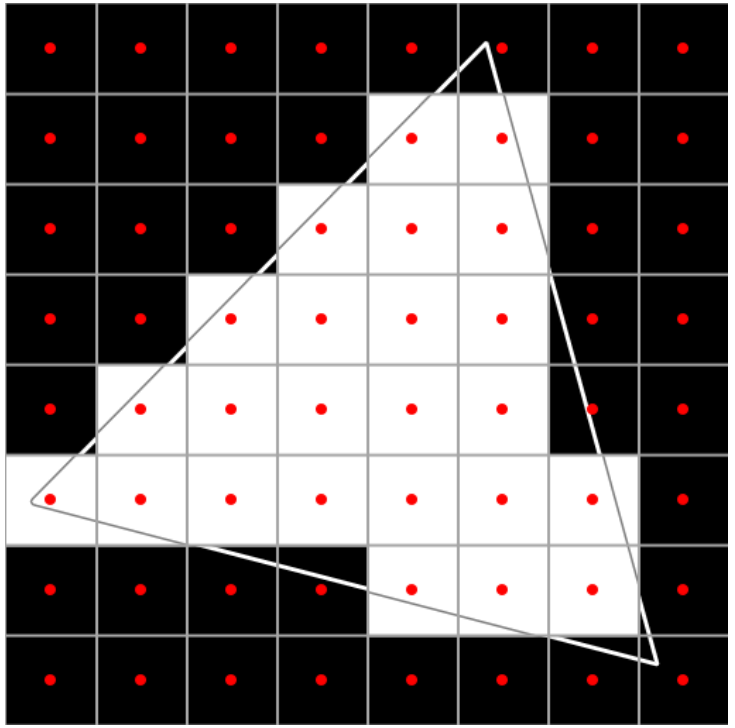


Бикубическая интерполяция (ru.wikipedia.org)

Простейший алиасинг







Типы алиасинга

- Единственный выровненный относительно экрана прямоугольник с частично прозрачной текстурой.
- «Мельница», состоящая из выровненных относительно экрана переменных белых и чёрных треугольников.
- Несколько чёрных линий различной ширины, начиная с 1 пикселя сверху до 0,4 пикселя снизу, и белая линия толщиной 0,5, отображающая синусоиду.
- Куб, состоящий из шести плоских закрашенных прямоугольников
- Наклонная плоскость, текстурированная высокочастотной текстурой травы.
- Выровненный относительно экрана прямоугольник с пиксельным шейдером, определяющим цвет каждого пикселя на основе функции синуса.

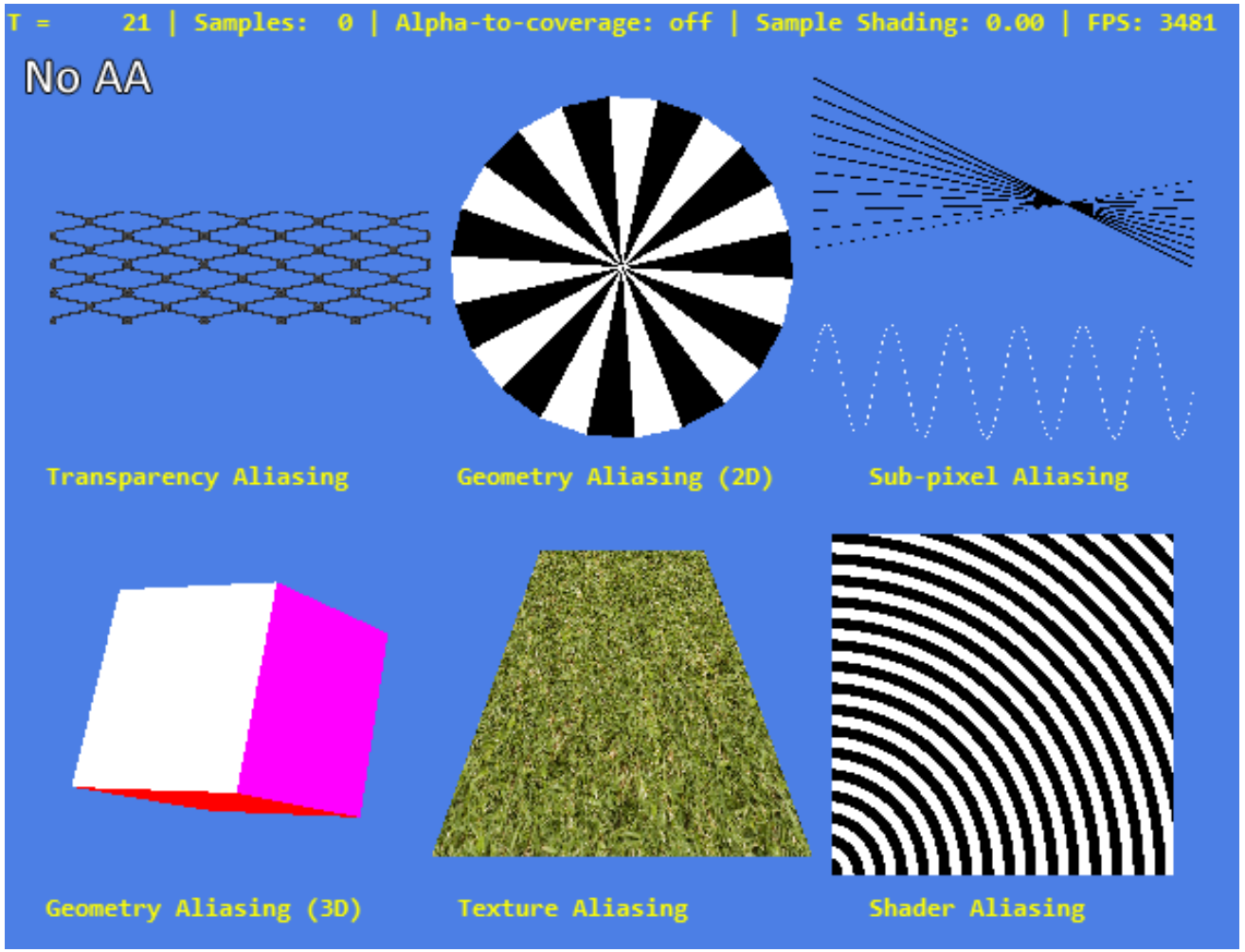
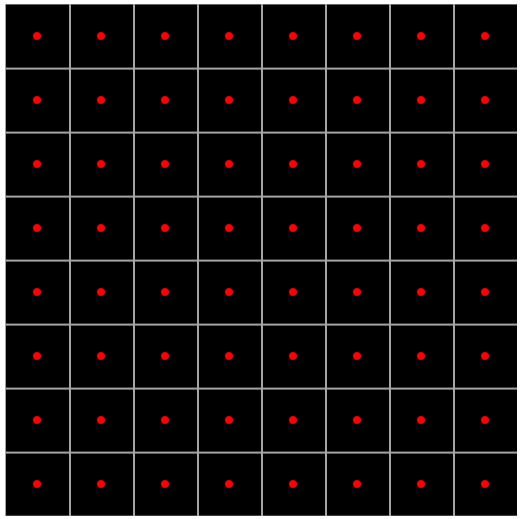
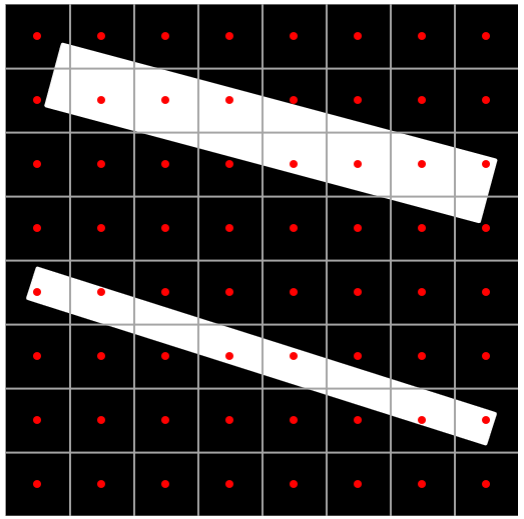
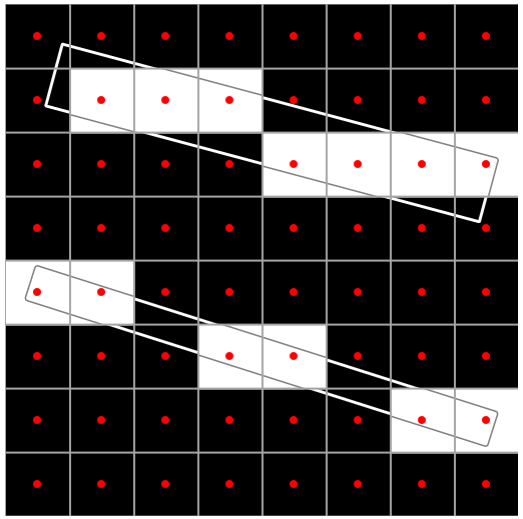
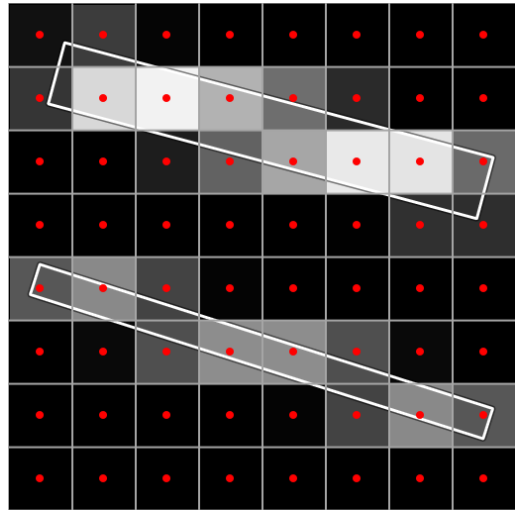


Иллюстрация подпиксельного алиасинга









Текстурный алиасинг



Обычно артефакты, создаваемые таким типом алиасинга, не очевидны на неподвижных скриншотах, но проявляются в движении как мерцание и неустойчивость пикселей.

Сэмплинг и его артефакты

Приближение

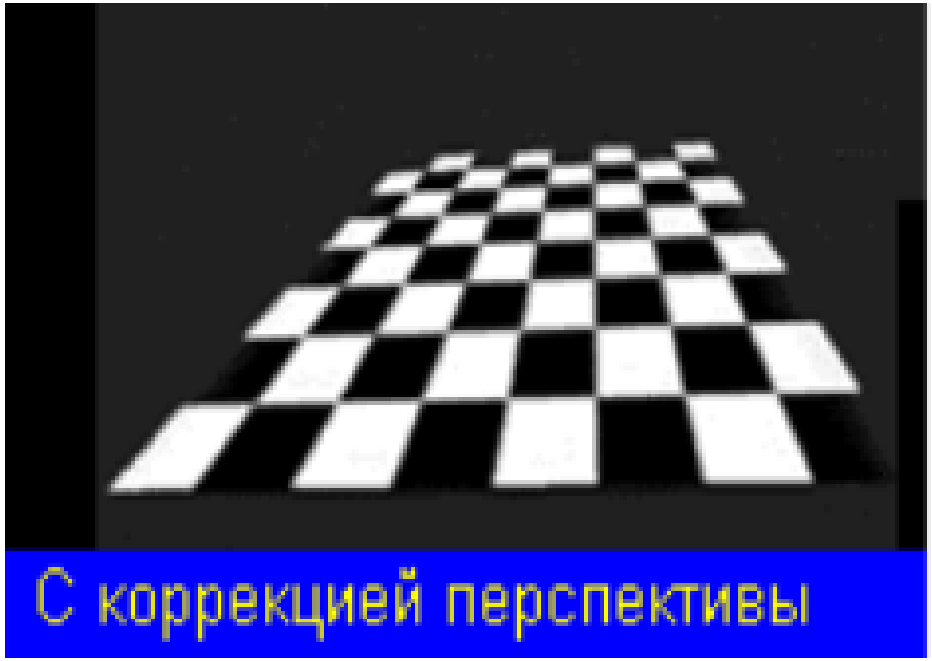


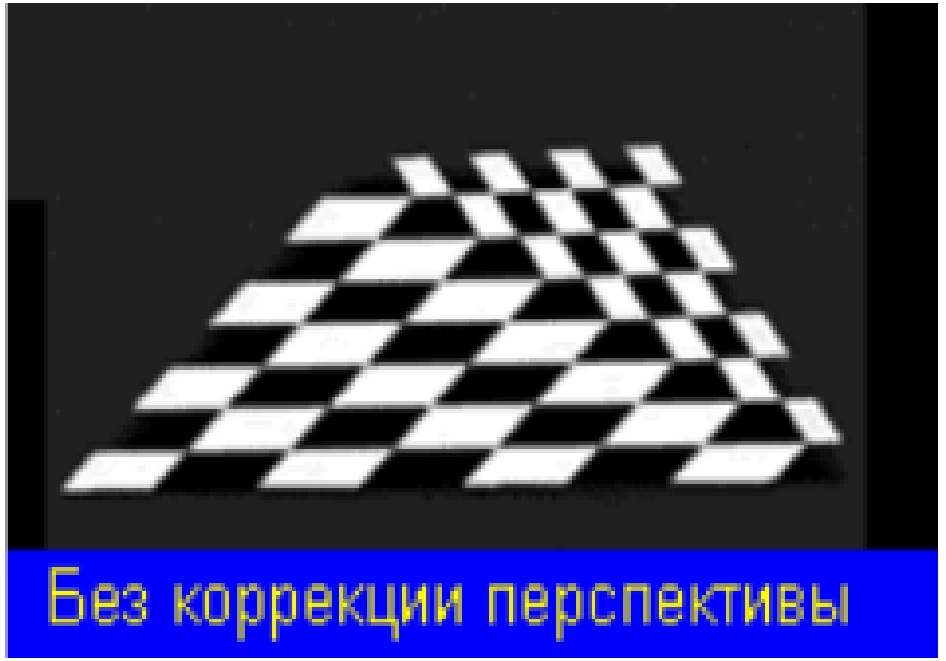
Пикселизация



Ещё один артефакт сэмплинга — блочность. При медленном вращении или приближении/удалении объекта могут быть заметны «перескакивания» пикселов с одного места на другое.

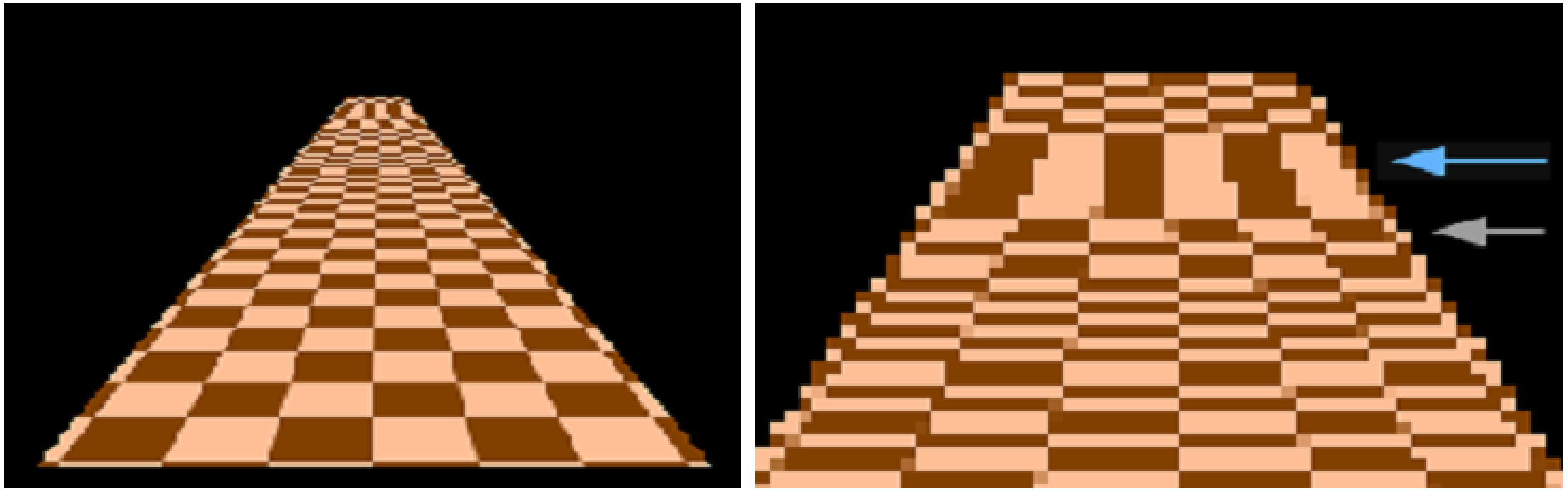
Артефакты depth aliasing и перспективная коррекция





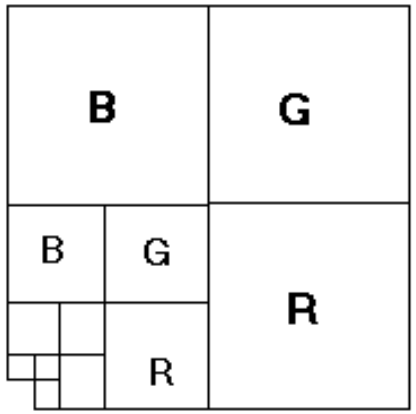
Перспективная коррекция – ресурсоемкая процедура (одна операция деления на каждый пиксел), поэтому 3D-ускорители реализуют ее аппаратно.

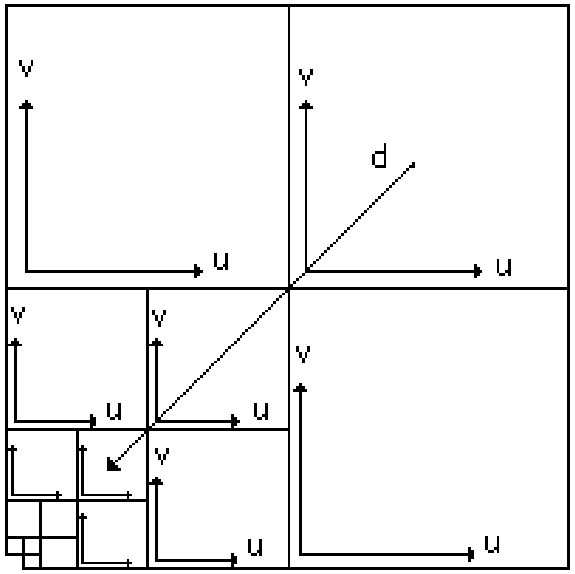
Артефакты depth aliasing



"depth aliasing" (ошибки глубины сцены или Z-aliasing), от которых билинейная фильтрация не избавляет и не может избавить (например, несколько квадратов сливаются в один).

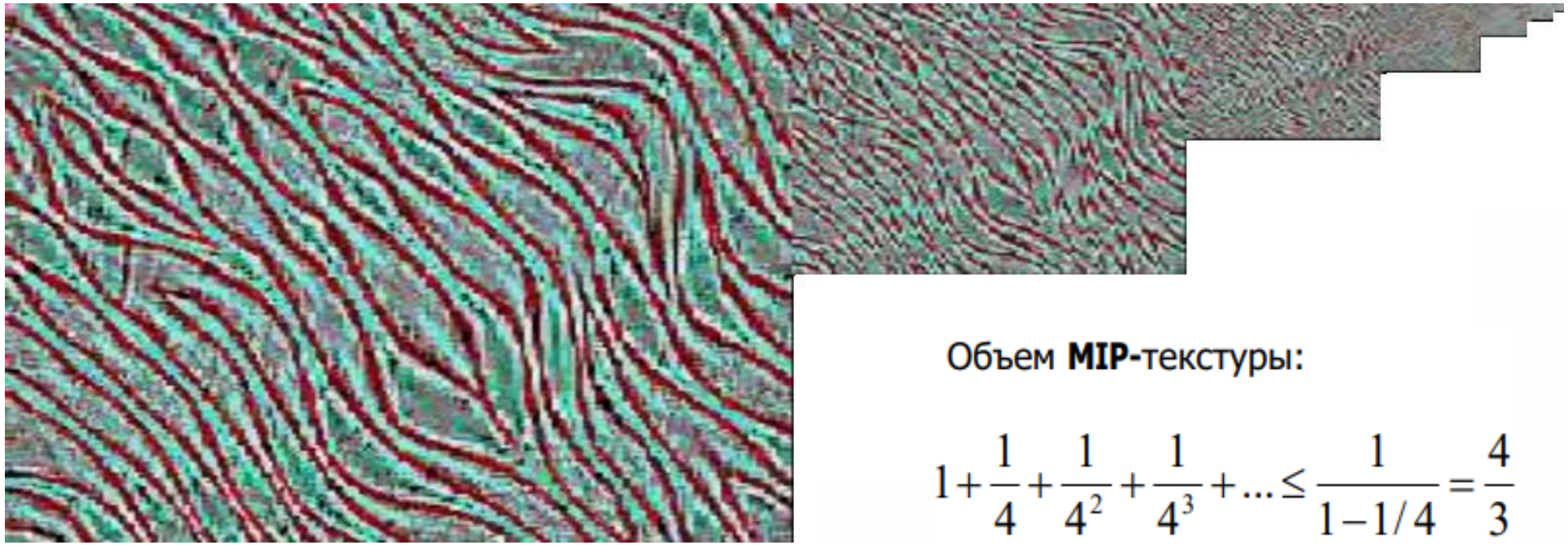
Пирамидальное фильтрование — Mip-mapping (лат. multum in parvo — много в малом)





Доступ к пирамиде осуществляется с помощью трех координат:
u, v, d, где u и v – координаты текстуры, а d (compression) – мера компрессии текстуры, и может быть рассмотрена как вертикальная координата пирамиды.

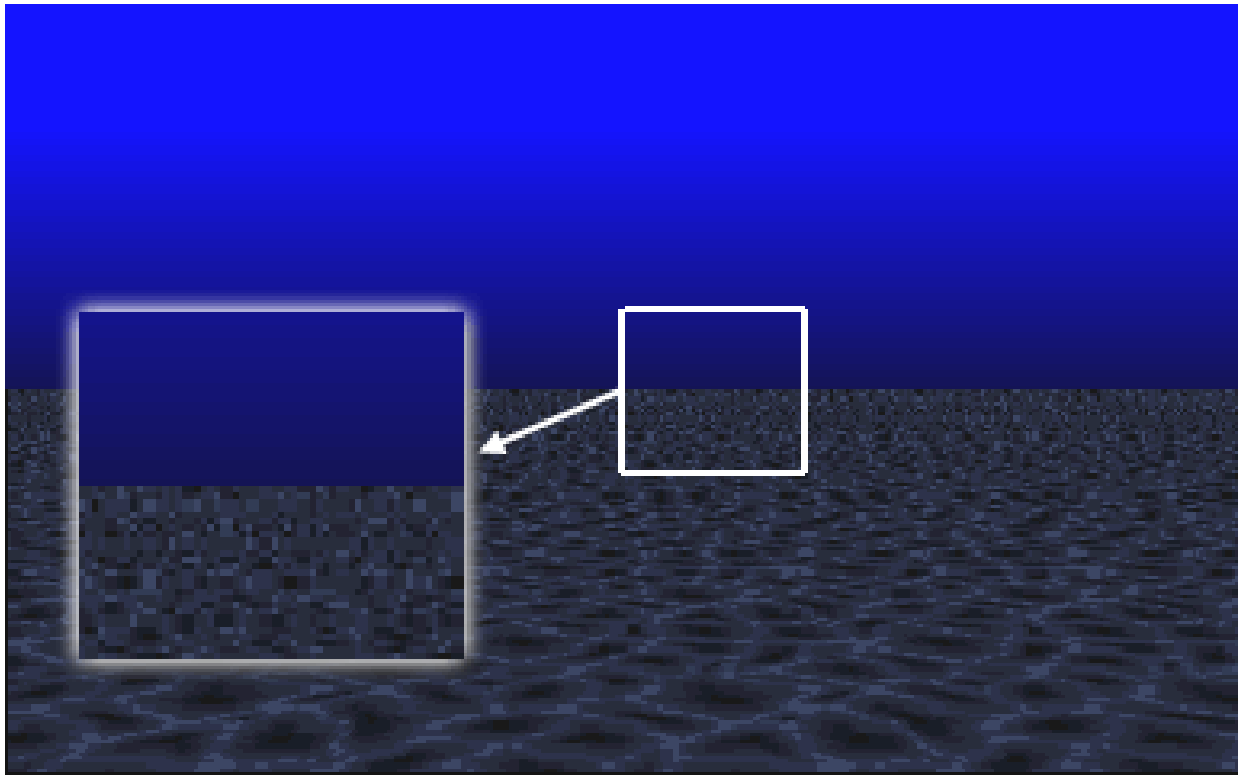
Объём Mip-текстуры

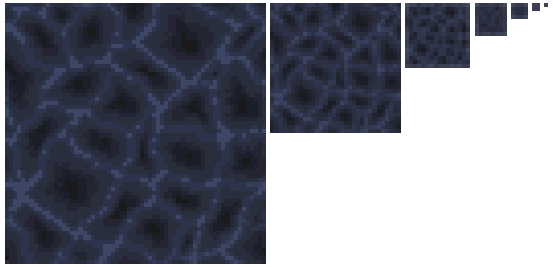


Объем **MIP**-текстуры:

$$1+\frac{1}{4}+\frac{1}{4^2}+\frac{1}{4^3}+\ldots\leq\frac{1}{1-1/4}=\frac{4}{3}$$

Mip-mapping и уровни детализации (LOD)

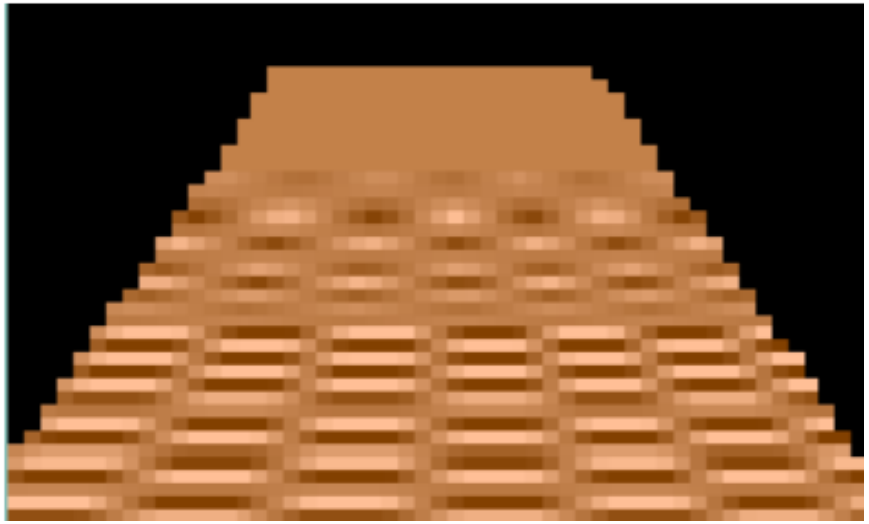




Степень или уровень детализации - Level of Detail (LOD), используются для определения, какой mip-map уровень (степень детализации) следует выбрать для наложения текстуры на объект.

Mip-текстурирование по каждому полигону

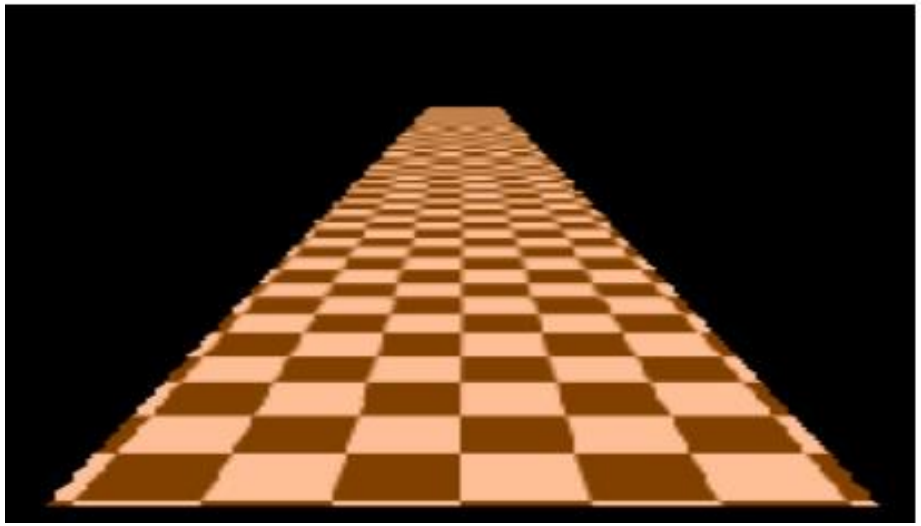
Степень или уровень детализации - Level of Detail или просто LOD, используются для определения, какой mip-map уровень (или какую степень детализации) следует выбрать для наложения текстуры на объект.



LOD вычисляется лишь раз для всего треугольника, следствием использования этого значения для всех пикселов треугольника становится эффект растрескивания, когда некоторые треугольники, из которых состоит анимированный объект, вдруг внезапно становятся чрезмерно размытыми или с неровностями.

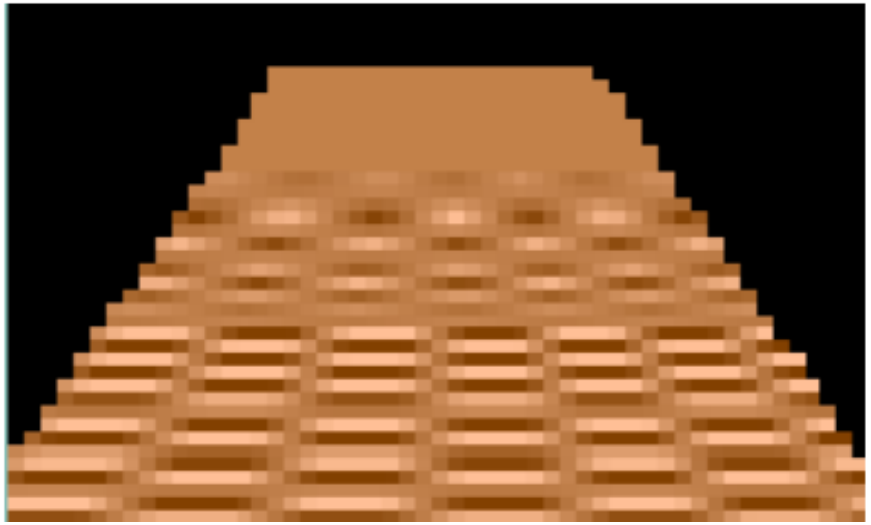
Per-pixel mip-mapping

Попиксельное mip-текстурирование

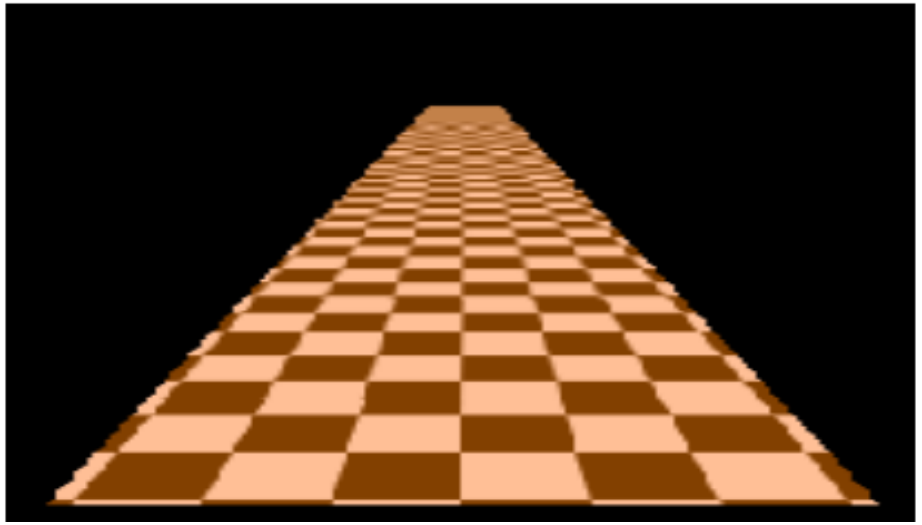


Значение LOD вычисляется для каждого пиксела. В результате предотвращается появление ошибок визуализации и излишней размытости.

MIP-mapping и уровни детализации (LOD)

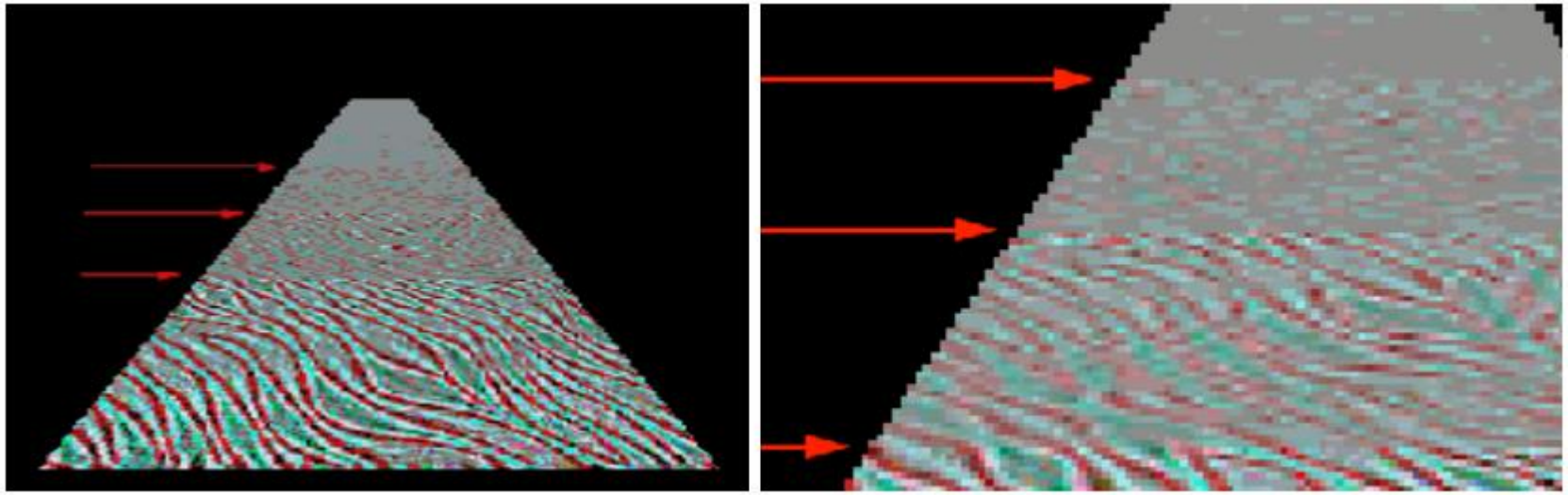


Per-polygon mip-mapping



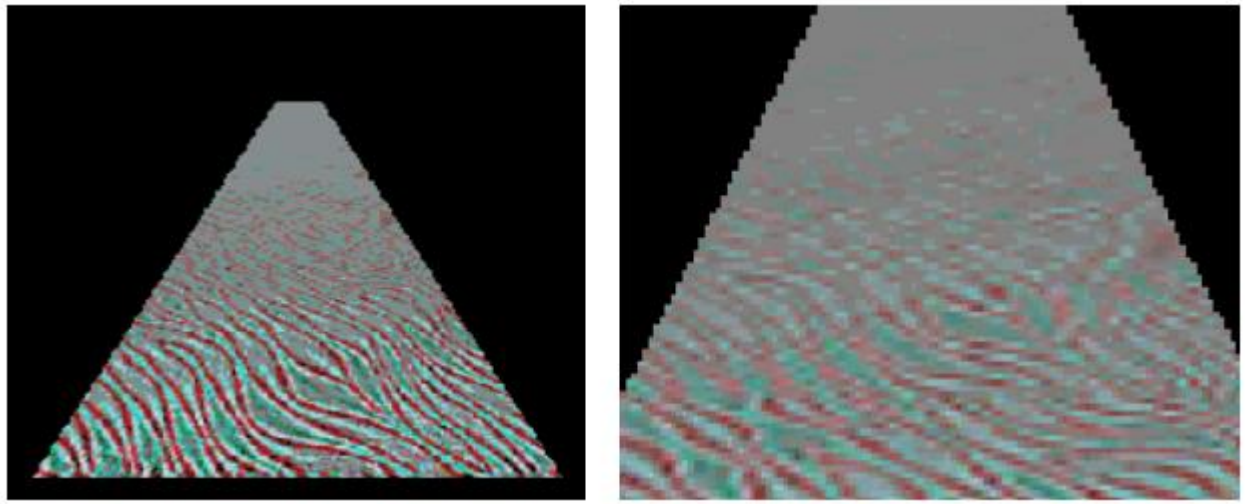
Per-pixel mip-mapping

MIP-текстурирование и mip-banding (полосатость)



В переходном состоянии от одного mip-map уровня к другому, появляется особый тип ошибок визуализации, известных под названием "mip-banding" (мип-бендинг) – полосатость или слоёность, т.е. явно различимые границы перехода от одного mip-map уровня к другому

MIP-banding и трилинейная фильтрация



При трилинейной фильтрации берется блок из четырех текселей и находится интерполированное значение цвета, как при билинейной, затем берется такой же четырехтексельный блок из соседнего mip-map уровня и так же усредняется.
И после всего этого находится среднее значение между двумя полученными, которое и будет цветом обрабатываемого пикселя.

Анизотропная фильтрация (Anisotropic filtering)



из-за проекционных искажений необходимо использовать анизотропный тип фильтрации

Анизотропная фильтрация обычно оперирует с не менее чем 8 текселами, для получения одного пиксела обрабатывает до 32 текселов

Антиалиасинг: краевой и полный

Алиасинг – результат преобразования непрерывного изображения в дискретное.
Алиасинг ухудшает качество изображения, вызывая разнообразные артефакты: такие, как лестничный эффект.

Антиалиасинг – процедура сглаживания может помочь улучшить (или, как минимум, не ухудшить) качество графического изображения.

По предназначению антиалиасинг делится на краевой и полный.

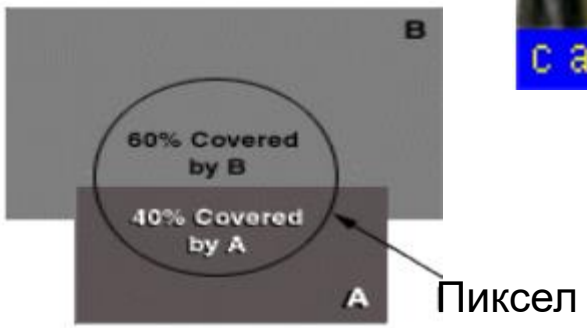
Алиасинг возникает при движении (1280x720)

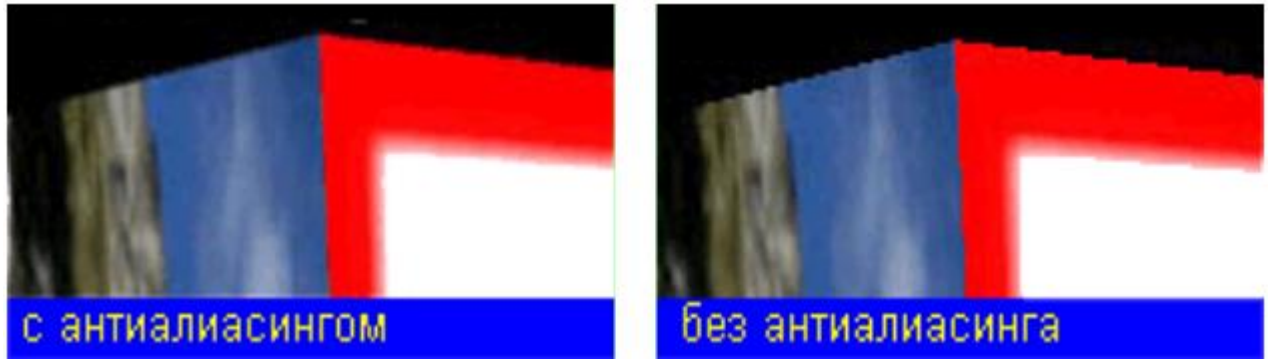


Та же сцена (3840x2160)



Сглаживание (anti-aliasing). Краевой антиалиасинг





Краевой антиалиасинг (КАА) – механизм борьбы с лестничным эффектом. КАА сглаживает края полигонов и диагональные линии.
Для реализации краевого антиалиасинга чаще всего используют технику усреднения по площади (area averaging).

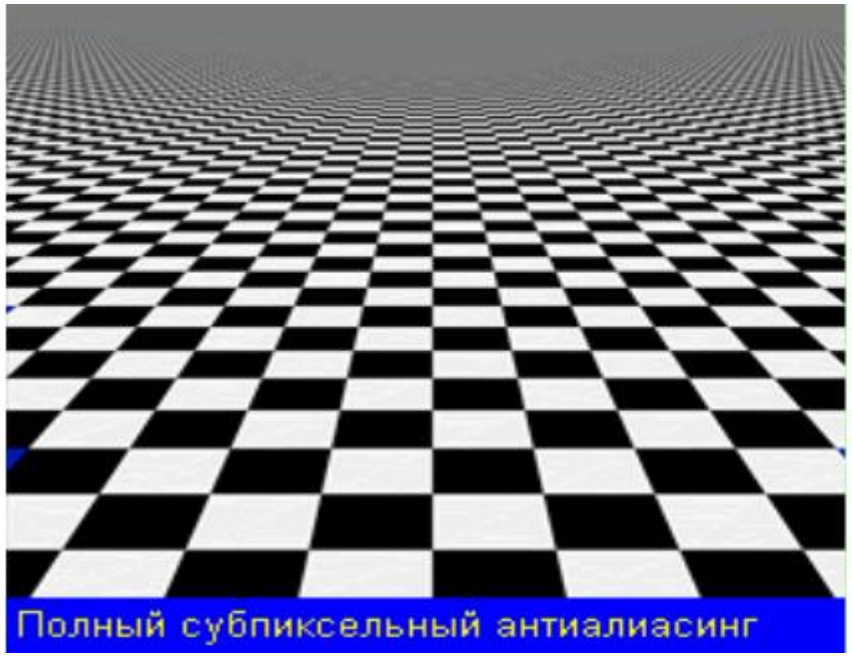
Anti-aliasing. Полный антиалиасинг

Полный антиалиасинг, в отличие от краевого, направлен на полную нейтрализацию алиасинга, в том числе bleeding-а (просачивания цветов фона).

Представителем полного антиалиасинга является субпиксельный антиалиасинг, который реализован в 3D-ускорителях.

Субпиксельный антиалиасинг базируется на технике суперсэмплинга.

Суперсэмплинг





Суперсэмплинг означает, что вся сцена рендерится в каком-то большом виртуальном разрешении, а затем сжимается до фактического разрешения.

В общем случае виртуальное и фактическое разрешения могут быть некратными.