

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ  
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Институт №8 «Компьютерные науки и прикладная математика»  
Кафедра 806 «Вычислительная математика и программирование»

**Лабораторная работа №4  
по курсу “Компьютерная графика”**

*Основы 3D графики*

Выполнил: В. Н. Шиширин  
Группа: М8О-310Б-23  
Преподаватель: В. Д. Бахарев

Москва, 2025

## **Условие**

В этой лабораторной работе вам предстоит реализовать технику наложения теней. Предстоит работа с рендерингом вне кадра “от лица” направленного источника света в текстуру глубины с использованием расширения Vulkan 1.2 Dynamic Rendering, а также работа с использованием данных о глубине сцены, чтобы создать эффект тени на поверхностях моделей.

- Подготовить матрицу проекции для направленного источника света
- Подготовить текстуру глубины, куда будет производится рендеринг сцены “от лица” направленного источника освещения
- Использовать расширение Vulkan Dynamic Rendering (`vkCmdBeginRendering`, `vkCmdEndRendering`) для рисования сцены в текстуру глубины
- Приготовить объект сэмплера с поддержкой сравнения значений (или реализовать PCF фильтрацию самому в фрагментном шейдере без поддержки сэмплером сравнения референсных значений)
- Использование текстуры и специального сэмплера в фрагментном шейдере для получения информации о глубине для последующего затенения пикселей в модели Блинна-Фонга
- Используйте код предыдущей лабораторной работы, чтобы получилось показать в работе все изученные техники: освещение по модели Блинна-Фонга, наложение текстур и тени!

## **Метод решения**

Реализуется shadow mapping для направленного источника света: рендер сцены «от лица» источника в depth-texture (shadow map) с использованием Dynamic Rendering, затем в основном проходе используется полученная depth-текстура для сравнения глубин и затемнения фрагментов в расчёте Blinn-Phong.

### Математика (ключевые формулы):

- Световое пространство (light-space): для каждой модели вычисляется light-space позиция вершин:

`light_pos = L_proj * L_view * model * vec4(position, 1.0)`  
где `L_view` — видовая матрица «от лица» направленного света,  
`L_proj` — проекция света (обычно ортографическая для directional light).

- Проекция в текстурные координаты shadow map:  
`ndc = light_pos.xyz / light_pos.w`  
`tex = 0.5 * ndc + 0.5 // перевод из NDC в [0,1] для выборки текстуры`
- Сравнение глубин (основная логика):  
`sampledDepth = texture(shadowSampler, tex.xy).r`  
`currentDepthInLightSpace = tex.z`  
`shadow = (currentDepthInLightSpace - bias > sampledDepth) ? 1.0 : 0.0`  
итоговый цвет = ambient + (1.0 - shadow) \* (diffuse + specular)
- Depth bias и коррекция: `bias = c0 + c1 * max(0, 1 - (N · L))` — смещение глубины для уменьшения self-shadow acne.
- PCF (Percentage-Closer Filtering) — если нет аппаратной поддержки compare sampler, выполнять множественные выборки вокруг `tex.xy` и усреднять результаты:  
`shadow = average( texture(shadowMap, tex.xy + offset_i).r < tex.z - bias ? 1 : 0 )`

#### Проекция для направленного источника:

- Для directional light обычно используется ортографическая проекция:  
`L_proj = orthographic(left, right, bottom, top, near, far)`. Параметры подбираются так, чтобы покрыть область сцены, освещаемую солнцем (или можно вычислять AABB в пространстве света).
- Итоговая матрица для шейдера: `shadow_matrix = L_proj * L_view`

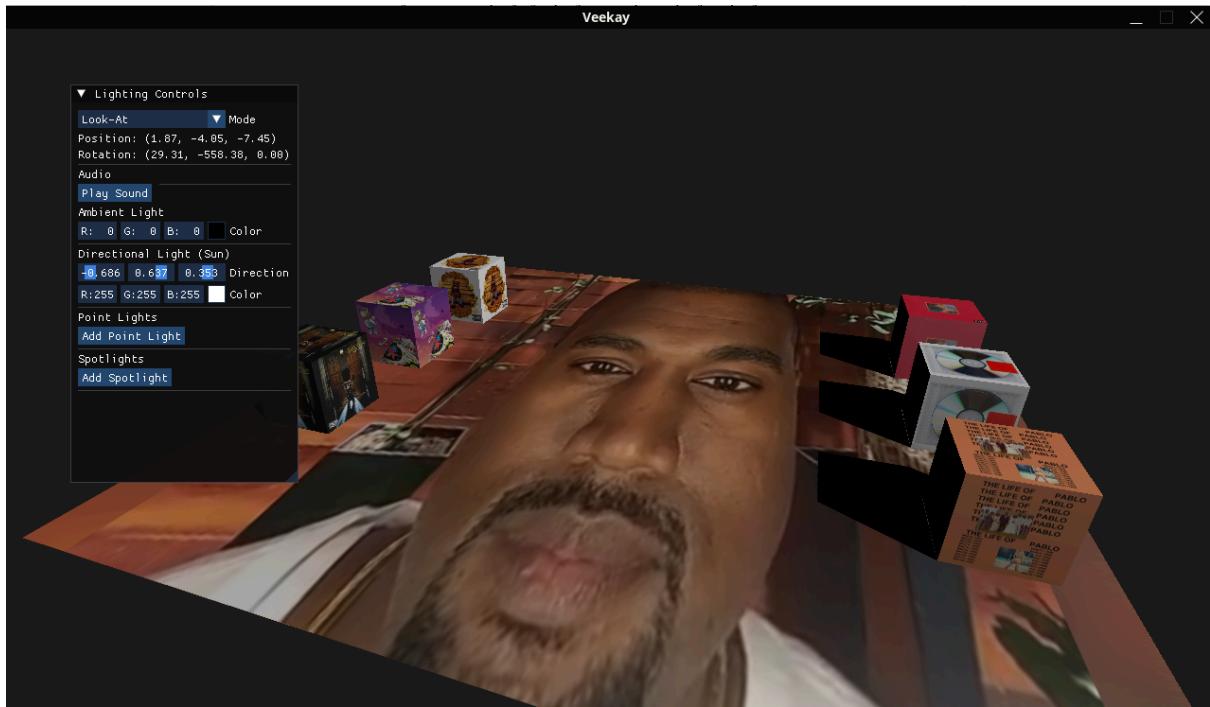
#### Основные шаги реализации:

1. Подготовить depth image (shadow map):
  - Создать `VkImage` с форматом глубины (например, `VK_FORMAT_D32_SFLOAT`), выделить память, создать `VkImageView`.
  - Размер shadow map (например, 2048 или 4096) выбирается в зависимости от качества.
2. Рендер в depth texture (shadow pass) с Dynamic Rendering:
  - Создать depth-only pipeline (вершинный шейдер для записи `gl_Position` по модели; фрагментный не нужен).

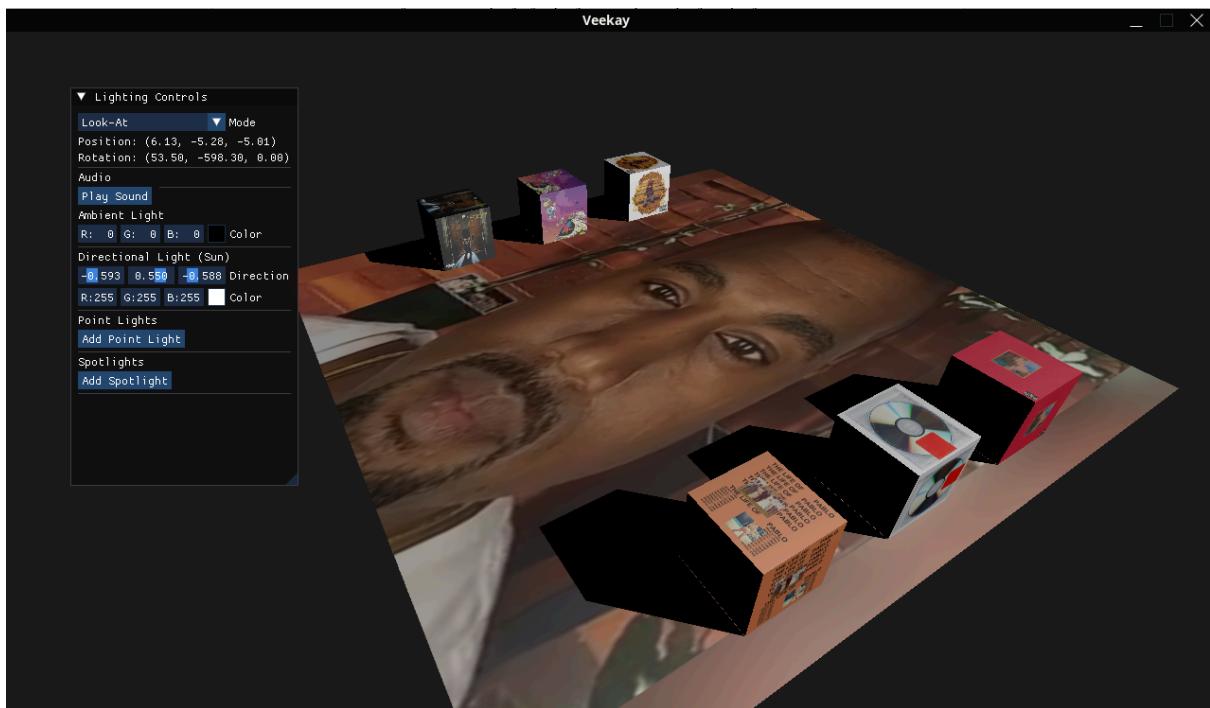
- Вызовы: vkCmdBeginRenderingKHR(...) с соответствующим VkRenderingAttachmentInfo для depth attachment → отрисовать сцены → vkCmdEndRenderingKHR(...).
  - При записи глубины включить depth bias (vkCmdSetDepthBias) и, при необходимости, изменить cull mode (например, VK\_CULL\_MODE\_FRONT\_BIT) чтобы уменьшить артефакты.
3. Создать и настроить sampler для shadow map:
    - Либо создать comparison sampler (VkSampler с compareEnable = VK\_TRUE и compareOp = VK\_COMPARE\_OP\_LESS\_OR\_EQUAL) и в шейдере использовать sampler2DShadow (hardware PCF доступен на некоторых устройствах);
    - Либо обычный sampler и реализовать PCF вручную в фрагментном шейдере (несколько выборок и усреднение).
  4. Передать shadow\_matrix и shadow map в основной проход:
    - Добавить mat4 shadow\_matrix или mat4 light\_space в UBO/scene uniforms (или в отдельный uniform).
    - Обновить VkDescriptorSet (биндинг VK\_DESCRIPTOR\_TYPE\_COMBINED\_IMAGE\_SAMPLER) с VkDescriptorImageInfo для shadow map + sampler.
  5. В основном фрагментном шейдере:
    - Преобразовать мировую позицию фрагмента в light-space: proj = shadow\_matrix \* vec4(worldPos, 1)
    - Получить tex = proj.xy / proj.w и depth = proj.z / proj.w → tex = 0.5\*tex + 0.5
    - Считать sampledDepth и выполнить сравнение с bias/PCF → получить коэффициент shadow ∈ [0,1]
    - Умножить diffuse/specular компоненты Blinn-Phong на (1.0 – shadow)
  6. Синхронизация и переходы layout:
    - Перед рендером в shadow map обеспечить layout VK\_IMAGE\_LAYOUT\_DEPTH\_STENCIL\_ATTACHMENT\_OPTIMAL для записи; перед чтением в основном проходе перейти в VK\_IMAGE\_LAYOUT\_SHADER\_READ\_ONLY\_OPTIMAL.
    - Используются image memory barriers (vkCmdPipelineBarrier) для корректной синхронизации между проходами.

## Результаты

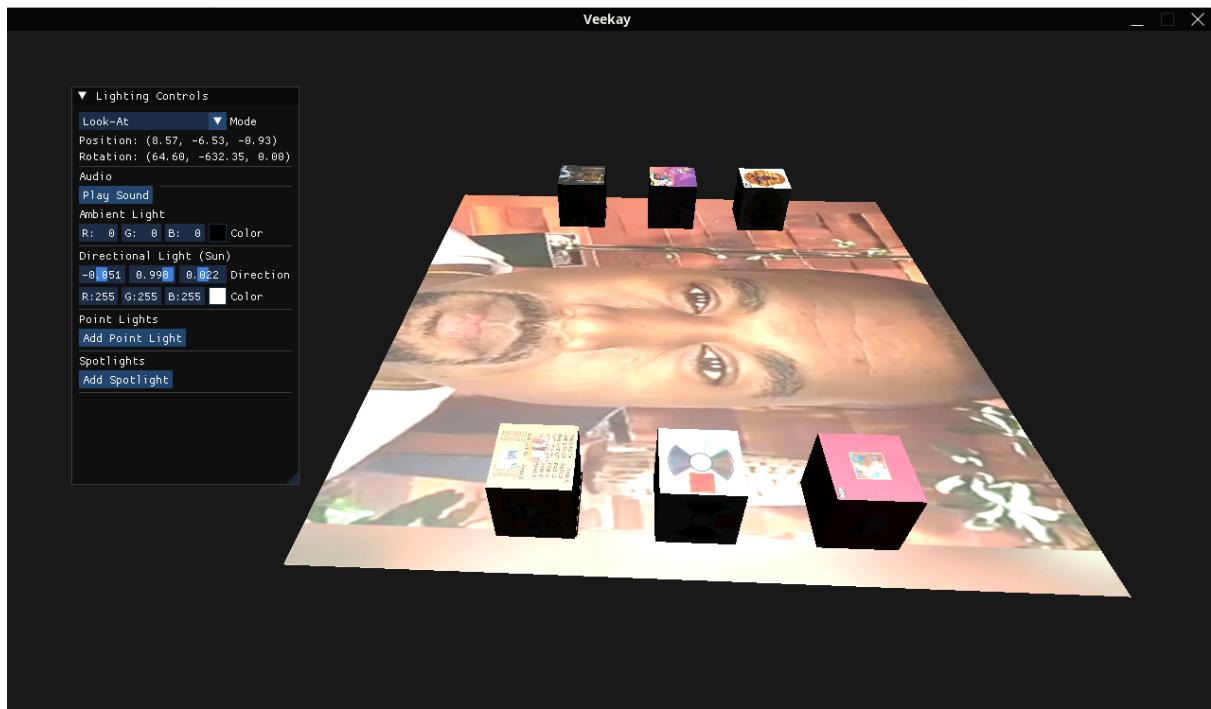
Тени от направленного света (солнца)



Тени от направленного света (солнца) под другим углом и положением направленного света



## Тени при направленном свете под прямым углом



## Выводы

Я научился(-ась) формировать матрицу проекции и вид из точки зрения направленного источника, рендерить сцену в depth-текстуру с использованием Dynamic Rendering, настраивать depth-image и comparison-sampler (или реализовывать PCF вручную), корректно сравнивать глубины в шейдере с учётом bias и применять результат для затемнения фрагментов в модели Блинн-Фонга; а также управлять этими ресурсами через UBO/descriptor sets и интегрировать тени с остальными техниками (освещение, текстурирование) в одном приложении.