

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРОГРАМНИХ СИСТЕМ ТА ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп'ютерних систем

«До захисту допущено»

Завідувач кафедри

_____ Віталій РОМАНКЕВИЧ

“_____” червня 2026 р.

Дипломний проєкт

на здобуття ступеня бакалавра

за освітньо-професійною програмою

«Системне програмування та спеціалізовані комп'ютерні системи»

Спеціальність 123 «Комп'ютерна інженерія»

на тему: «Комп'ютерні засоби реалізації атласу рей-трейсингових тіней для динамічних джерел світла»

Виконав : студент IV курсу,

групи КВ-21

Самойленко Денис Артемович

Керівник

доц. каф. СПСКС, к.т.н., Клятченко Ярослав Михайлович

Консультант з нормоконтролю

доц. каф. СПСКС, к.т.н., Клятченко Ярослав Михайлович

Рецензент

доцент кафедри ПЗКС, к. т. н. Заболотня Тетяна Миколаївна

Засвідчую, що у цьому дипломному проєкті
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____

(підпис)

Київ – 2026 року

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРОГРАМНИХ СИСТЕМ ТА ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп'ютерних систем

Рівень вищої освіти – перший (бакалаврський)

Спеціальність 123 «Комп'ютерна інженерія»

Освітньо-професійна програма «Системне програмування та спеціалізовані комп'ютерні системи»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Віталій РОМАНКЕВИЧ
(підпис) (ініціали, прізвище)

«__» червня 2026 р.

ЗАВДАННЯ

на дипломний проєкт студента
Самойленка Дениса Артемовича

1. Тема проєкту «Комп'ютерні засоби реалізації атласу рей-трейсингових тіней для динамічних джерел світла», керівник проєкту Клятченко Я.М., кандидата технічних наук, доцента кафедри системного програмування і спеціалізованих комп'ютерних систем, затверджені наказом по університету від «__» _____ 2026 р. № _____
2. Термін подання студентом проєкту _____
3. Вихідні дані до проєкту: див. Технічне завдання
4. Зміст пояснювальної записки
 - Аналіз існуючих рішень та актуальність теми дипломного проєкту
 - Програмне та апаратне забезпечення
 - Опис розробленої системи
 - Тестування розробленої комп'ютерної системи
5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо)

- Схема структурна
- Схема алгоритму
- Схема алгоритму
- Схема структурна

6. Консультанти розділів проєкту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Клятченко Я.М., к.т.н доцент каф. СПіСКС		

7. Дата видачі завдання 6 грудня 2025 року

Календарний план

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту	Примітка
	Вивчення літератури за тематикою проєкту	07.12.2025	
	Аналіз існуючих рішень	10.01.2026	
	Розробка технічного завдання	05.02.2026	
	Підготовка матеріалів першого розділу дипломного проєкту	15.02.2026	
	Підготовка матеріалів другого розділу дипломного проєкту	12.03.2026	
	Підготовка матеріалів третього розділу дипломного проєкту	14.04.2026	
	Підготовка матеріалів четвертого розділу дипломного проєкту	05.05.2026	
	Підготовка макету для наочної демонстрації	17.05.2026	
	Оформлення документації дипломного проєкту	22.05.2026	
	Попередній огляд матеріалів диплому на кафедрі	01.06.2026	

Студент

(підпис)

Самойленко Д.А.

Керівник проєкту

(підпис)

Клятченко Я.М.

АНОТАЦІЯ

Кваліфікаційна робота включає пояснювальну записку (85 с., 23 рис. 5 табл., 4 додатки).

Об'єкт розробки – процес формування тіней у графічних системах реального часу для сцен із динамічними джерелами світла.

Предмет розробки – програмні засоби реалізації атласу рей-трейсингових тіней, генерації карти видимості та застосування тіньового фактора у гібридному графічному конвеєрі.

Розроблена система дозволяє формувати спільний атлас тіней для множини активних джерел світла, адаптивно виділяти області атласу для окремих джерел, будувати карту відповідності texel-to-light, генерувати первинні значення видимості методом трасування променів та застосовувати тіньовий фактор під час lighting pass. Для підвищення якості результату в системі реалізовано декілька режимів апскейлу та фільтрації, зокрема Point, Bilinear, PCF, Joint Bilateral, Poisson-based та комбіновані режими.

В ході розробки:

- проведено аналіз методів побудови тіней у реальному часі;
- обґрунтовано вибір гібридної архітектури графічної підсистеми;
- розроблено структуру спільного атласу рей-трейсингових тіней;
- реалізовано механізм формування метаданих активних джерел світла;
- реалізовано карту відповідності texel-to-light;
- реалізовано генерацію первинної карти видимості за допомогою DirectX Raytracing;
- реалізовано інтеграцію тіньового фактора у кластерне відкладене освітлення.
- реалізовано режими апскейлу та фільтрації тіньового фактора.
- Проведено функціональне тестування, дослідження візуальної якості та оцінювання продуктивності системи.

Отримані результати показали, що запропонована система може використовуватися як складова частина графічного рушія для побудови тіней від великої кількості динамічних джерел світла з керованим балансом між якістю зображення та продуктивністю.

Ключові слова:

АТЛАС ТІНЕЙ, РЕЙ-ТРЕЙСИНГОВІ ТІНІ, ДИНАМІЧНІ ДЖЕРЕЛА СВІТЛА, DIRECTX RAYTRACING, DIRECT3D, SHADOW FACTOR, UPSCALE, ФІЛЬТРАЦІЯ ТІНЕЙ, CLUSTRED DEFERRED LIGHTING.

ABSTRACT

The qualification work includes an explanatory note (85 pages, 23 figures, 5 tables, 4 appendices).

The object of development is the process of shadow generation in real-time graphics systems for scenes with dynamic light sources.

The subject of development is software tools for implementing a ray-traced shadow atlas, generating a visibility map, and applying the shadow factor in a hybrid graphics pipeline.

The developed system makes it possible to generate a shared shadow atlas for a set of active light sources, adaptively allocate atlas regions for individual lights, build a texel-to-light correspondence map, generate primary visibility values using ray tracing, and apply the shadow factor during the lighting pass. To improve the visual quality of the result, several upscaling and filtering modes were implemented in the system, including Point, Bilinear, PCF, Joint Bilateral, Poisson-based, and combined modes.

During the development:

- methods for real-time shadow generation were analyzed;
- the choice of a hybrid graphics subsystem architecture was justified;
- the structure of a shared ray-traced shadow atlas was developed;
- a mechanism for generating metadata of active light sources was implemented;
- the texel-to-light correspondence map was implemented;
- primary visibility map generation using DirectX Raytracing was implemented;
- integration of the shadow factor into clustered deferred lighting was implemented;
- upscaling and filtering modes for the shadow factor were implemented;
- functional testing, visual quality analysis, and performance evaluation of the system were carried out.

The results obtained showed that the proposed system can be used as a component of a graphics engine for generating shadows from a large number of dynamic light sources with a controllable balance between image quality and performance.

Keywords:

SHADOW ATLAS, RAY-TRACED SHADOWS, DYNAMIC LIGHT SOURCES, DIRECTX RAYTRACING, DIRECT3D, SHADOW FACTOR, UPSCALE, SHADOW FILTERING, CLUSTERED DEFERRED LIGHTING.

[illegible]

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
	A4	ІАЛЦ.467800.005 Д1	Загальна архітектура	1		
			системи атласу			
			рей-трейсингових тіней			
			Схема структурна			
	A4	ІАЛЦ.467800.006 Д2	Алгоритм формування	1		
			атласу рей-трейсингових			
			тіней та карти texel-to-light			
			Схема алгоритму			
	A4	ІАЛЦ.467800.007 Д3	Застосування атласу			
			рей-трейсингових тіней у	1		
			кластерному відкладеному			
			освітленні			
			Схема алгоритму			
	A4	ІАЛЦ.467800.008 Д4	Багатопотокова підготовка	1		
			та виконання GPU-команд			
			у графічному рушії			
			Схема структурна			
		Диск CD-ROM	Відеодемонстрація роботи	1		
			проєкту			
			ІАЛЦ.467800.001 ОА			Арк.
Змін.	Арк.	№ докум.				2
		Підпис	Дата			

ЗМІСТ

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ.	2
2. ПІДСТАВА ДЛЯ РОЗРОБКИ.	2
3. ЦІЛЬ І ПРИЗНАЧЕННЯ РОБОТИ.	2
4. ДЖЕРЕЛА РОЗРОБКИ.	2
5. ТЕХНІЧНІ ВИМОГИ.	2
5.1. Вимоги до програмного продукту, що розробляється.	2
5.2. Вимоги до апаратного забезпечення.	3
5.3. Вимоги до програмного та апаратного забезпечення користувача. .	3
6. ЕТАПИ РОЗРОБКИ.	4

					ІАЛЦ.467800.002 ТЗ				
Змін	Арк.	№ докум.	Підпис	Дата	Комп'ютерні засоби реалізації атласу рей-трейсингових тіней для динамічних джерел світла Технічне завдання	Літ.	Аркуш	Аркушів	
Розробив		Самойленко Д.А.							
Перевірів		Клятченко Я.М.					1	4	
						КПІ ім. Ігоря Сікорського, ФПСІМ КВ-21			
Н. контроль		Клятченко Я.М.							
Затвердив		Романкевич В.О							

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ

Назва розробки: «Комп'ютерні засоби реалізації атласу рей-трейсингових тіней для динамічних джерел світла».

Галузь застосування: системи комп'ютерної графіки реального часу, ігрові рушії, інтерактивні системи візуалізації, тренажери та інші програмні засоби, у яких необхідне формування тіней від великої кількості динамічних джерел світла.

2. ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки є завдання на дипломне проектування на здобуття першого (бакалаврського) рівня вищої освіти, затверджене кафедрою системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України «Київський Політехнічний Інститут імені Ігоря Сікорського».

3. МЕТА І ПРИЗНАЧЕННЯ РОБОТИ

Метою даної роботи є розроблення програмних засобів реалізації атласу рей-трейсингових тіней для багатьох динамічних джерел світла в системі комп'ютерної графіки реального часу.

4. ДЖЕРЕЛА РОЗРОБКИ

Джерелом інформації є технічна та науково-технічна література, технічна документація, публікації у періодичних виданнях та електронні статті у мережі Інтернет.

5. ТЕХНІЧНІ ВИМОГИ

					ІАЛЦ.467800.002 ТЗ	Арк.
						2
Змін.	Арк.	№ докум.	Підпис	Дата		

5.1. Вимоги до програмного продукту, що розробляється

- формування тіней для багатьох динамічних джерел світла в реальному часі;
- підтримка напрямлених, точкових і прожекторних джерел світла;
- використання трасування променів для визначення видимості джерела світла;
- формування спільного атласу рей-трейсингових тіней;
- адаптивний вибір розміру області тіні залежно від важливості джерела світла;
- використання даних буферів для відновлення параметрів поверхні під час побудови тіней;
- інтеграція підсистеми тіней у deferred shading та clustered lighting;
- підтримка кількох режимів фільтрації та апскейлу тіней;
- можливість параметричного налаштування якості тіней;
- забезпечення прийнятної швидкодії та візуальної стабільності в динамічних сценах.

5.2. Вимоги до апаратного забезпечення

- процесор: не менше 4 фізичних ядер;
- оперативна пам'ять: не менше 16 ГБ;
- графічний адаптер з підтримкою DirectX 12 і DirectX Raytracing;
- відеопам'ять: не менше 6 ГБ;
- наявність накопичувача для зберігання програмних компонентів і тестових сцен.

5.3. Вимоги до програмного та апаратного забезпечення користувача

- операційна система Windows 10 або Windows 11;
- підтримка DirectX 12;
- графічний адаптер з підтримкою DXR;
- оперативна пам'ять не менше 8 ГБ;

					ІАЛЦ.467800.002 ТЗ	Арк.
						3
Змін.	Арк.	№ докум.	Підпис	Дата		

- наявність встановлених драйверів відеоадаптера, сумісних із DirectX 12 або DXR.

6. ЕТАПИ РОЗРОБКИ

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту
1	Вивчення літератури за тематикою проєкту	07.12.2025
2	Аналіз існуючих рішень	10.01.2026
3	Розробка технічного завдання	05.02.2026
4	Підготовка матеріалів першого розділу дипломного проєкту	15.02.2026
5	Підготовка матеріалів другого розділу дипломного проєкту	12.03.2026
6	Підготовка матеріалів третього розділу дипломного проєкту	14.04.2026
7	Підготовка матеріалів четвертого розділу дипломного проєкту	05.05.2026
8	Підготовка макету для наочної демонстрації	17.05.2026
9	Оформлення документації дипломного проєкту	22.05.2026
10	Попередній огляд матеріалів диплому на кафедрі	01.06.2026

[illegible]

[illegible]

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту

на тему: Комп'ютерні засоби реалізації атласу рей-трейсингових тіней
для динамічних джерел світла

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ.	4
ВСТУП.	8
1. ПОРІВНЯЛЬНІ ХАРАКТЕРИСТИКИ СУЧАСНИХ СИСТЕМ КОМП'ЮТЕРНОЇ ГРАФІКИ ДЛЯ ФОРМУВАННЯ ТІНЕЙ.	10
1.1 Особливості формування тіней у сучасних системах комп'ютерної графіки. . .	10
1.2 Аналіз існуючих методів побудови тіней у реальному часі	12
1.2.1 Shadow Mapping	12
1.2.2 Variance Shadow Maps.	14
1.2.3 Moment Shadow Mapping.	15
1.2.4 Cascaded Shadow Maps.	16
1.2.5 Screen-Space методи для формування тіней.	18
1.2.6 Ray Traced Shadows.	19
1.2.7 Методи фільтрації та апскейлу тіньових даних у системах реального часу. .	20
1.3 Порівняльний аналіз існуючих підходів до побудови тіней.	23
1.4 Аналіз проблем реалізації тіней для багатьох динамічних джерел світла.	25
1.5 Обґрунтування актуальності теми дипломного проєкту.	27
ВИСНОВКИ ДО РОЗДІЛУ 1.	29
2. ЗАСОБИ РЕАЛІЗАЦІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.	30
2.1 Обґрунтування вибору архітектури графічної підсистеми.	30
2.1.1 Використання deferred shading.	31
2.1.2 Використання єдиного атласу тіней для множини джерел світла	31
2.1.3 Використання кластерного освітлення.	32
2.1.4 Використання багатопотокової обробки в рушії.	34
2.2 Обґрунтування вибору технологій розробки.	35
2.2.1 Вибір мови програмування C++ для реалізації рушія.	35

					ІАЛЦ.467800.004 ПЗ			
Змін	Арк.	№ докум.	Підпис	Дата				
Розробив		Самойленко Д.А.			Комп'ютерні засоби реалізації атласу рей-трейсингових тіней для динамічних джерел світла Пояснювальна записка	Літ.	Аркуш	Аркушів
Перевірив		Клятченко Я.М.					1	84
Н. контроль		Клятченко Я.М.						
Затвердив		Романкевич В.О.				КПІ ім. Ігоря Сікорського, ФПСІМ КВ-21		

2.2.2 Використання HLSL для реалізації шейдерних модулів.	35
2.2.3 Використання Direct3D 12.	36
2.2.4 Використання DirectX Raytracing (DXR)	36
2.3 Опис використовуваних структур даних та графічних ресурсів.	37
ВИСНОВКИ ДО РОЗДІЛУ 2.	39
3. ОСОБЛИВОСТІ СТРУКТУРНОЇ ОРГАНІЗАЦІЇ РОЗРОБЛЕНОЇ СИСТЕМИ.	40
3.1 Загальна організація реалізації системи.	40
3.2 Формування атласу тіней для динамічних джерел світла.	42
3.2.1 Формування метаданих активних джерел світла.	42
3.2.2 Адаптивний вибір розміру тайла.	43
3.2.3 Розміщення тайлів у спільному атласі тіней.	44
3.2.4 Побудова карти відповідності texel-to-light.	45
3.3 Генерація первинної карти видимості методом трасування променів.	47
3.3.1 Відновлення параметрів точки поверхні.	48
3.3.2 Використання normal-dependent bias.	49
3.3.3 Формування тіньових променів.	50
3.3.4 Трасування променя в TLAS з раннім завершенням.	50
3.3.5 Запис первинного значення видимості в атлас.	52
3.4 Обробка різних типів джерел світла.	52
3.5 Методи апскейлу та фільтрації тіньового фактора.	54
3.5.1 Point sampling та Bilinear filtering.	54
3.5.2 PCF-фільтрація.	54
3.5.3 Joint Bilateral фільтрація.	55
3.5.4 Poisson-based фільтрація.	56
3.5.5 Комбіновані режими фільтрації.	56
3.6 Параметричне налаштування якості тіней.	57
3.7 Інтеграція атласу тіней у кластерне відкладене освітлення.	59
3.8 Особливості програмної реалізації.	60
3.8.1 Основні програмні модулі реалізації.	60
3.8.2 Організація виконання GPU-проходів у межах кадру.	61
3.8.3 Багатопотокова підготовка даних та виконання команд.	62
3.8.4 Реалізація темпорального згладжування кадру.	63
3.9 Обмеження реалізованої системи та напрями подальшого розвитку.	64

ВИСНОВКИ ДО РОЗДІЛУ 3.	66
4. ТЕСТУВАННЯ РОЗРОБЛЕНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ.	67
4.1 Мета та завдання тестування системи.	67
4.2 Методика тестування.	67
4.3 Функціональне тестування системи.	69
4.3.1 Перевірка коректності формування атласу тіней і карти texel-to-light. ...	69
4.3.2 Перевірка коректності генерації тіней для різних типів джерел світла. ...	70
4.4 Дослідження візуальної якості тіней.	72
4.4.1 Порівняння режимів фільтрації та апскейлу.	72
4.4.2 Аналіз впливу параметрів фільтрації та корекції тіні.	74
4.5 Дослідження продуктивності системи.	76
4.5.1 Вплив кількості динамічних джерел світла на час кадру.	76
4.5.2 Вплив режиму фільтрації.	77
4.6 Узагальнення результатів тестування.	78
ВИСНОВКИ ДО РОЗДІЛУ 4.	81
ВИСНОВКИ.	82
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	84
ДОДАТКИ	

ДОДАТОК 1. ІАЛЦ.467800.005 Д1 ЗАГАЛЬНА АРХІТЕКТУРА СИСТЕМИ АТЛАСУ РЕЙ-ТРЕЙСИНГОВИХ ТІНЕЙ. СХЕМА СТРУКТУРНА.

ДОДАТОК 2. ІАЛЦ.467800.006 Д2 АЛГОРИТМ ФОРМУВАННЯ АТЛАСУ РЕЙ-ТРЕЙСИНГОВИХ ТІНЕЙ ТА КАРТИ TEXEL-TO-LIGHT. СХЕМА АЛГОРИТМУ.

ДОДАТОК 3. ІАЛЦ.467800.007 Д3 ЗАСТОСУВАННЯ АТЛАСУ РЕЙ-ТРЕЙСИНГОВИХ ТІНЕЙ У КЛАСТЕРНОМУ ВІДКЛАДЕНОМУ ОСВІТЛЕННІ. СХЕМА АЛГОРИТМУ.

ДОДАТОК 4. ІАЛЦ.467800.008 Д4 БАГАТОПОТОКОВА ПІДГОТОВКА ТА ВИКОНАННЯ GPU-КОМАНД У ГРАФІЧНОМУ РУШІІ. СХЕМА СТРУКТУРНА.

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ

Атлас тіней – набір текстурних областей, у яких зберігаються тіньові дані для декількох джерел світла.

Апскейл – процес підвищення ефективної роздільної здатності тіньових даних під час їх застосування у фінальному освітленні.

Альbedo – базовий колір поверхні без урахування світлових ефектів.

Дизоклюзія – ситуація, коли в поточному кадрі стає видимою ділянка сцени, яка в попередньому була перекрита іншим об'єктом або перебувала поза видимістю.

Джерело світла – елемент сцени, що задає напрям, інтенсивність, колір та інші параметри освітлення.

Карта відповідності texel-to-light – службовий буфер, що визначає, якому джерелу світла відповідає кожен тексель атласу тіней.

Кластеризоване освітлення – метод організації освітлення, за якого простір камери поділяється на кластери, де для кожного кластера формується список релевантних джерел світла

Оклюдер – частина геометрії сцени, що перекриває шлях світла від джерела до точки поверхні та спричиняє утворення тіні

ПЗ – програмне забезпечення.

Растрезація – етап графічного конвеєра, на якому геометричні примітиви перетворюються у фрагменти екранного простору.

Скайбокс – елемент графічної сцени, що використовується для відображення віддаленого оточення.

Структура прискорення – ієрархічна структура даних, що використовується для прискорення пошуку перетинів променів із геометрією сцени.

Тайл – прямокутна область у тіньовому атласі, виділена для зберігання тіньових даних джерел світла.

					ІАЛЦ. 467800.004 ПЗ	Арк.
						4
Змін.	Арк.	№ докум.	Підпис	Дата		

Тексель – елемент текстурного зображення, що відповідає одній дискретній комірці текстури.

Текстура глибини – текстурний ресурс, у якому зберігаються значення фрагментів глибини сцени.

Тіньовий фактор – коефіцієнт видимості джерела світла, що використовується для модифікації його внеску у фінальне освітлення.

Трасування променів – метод побудови зображення шляхом випускання променів у сцену та перевірки їх перетину з геометрією.

Фільтрація тіней – обробка тіньових даних для зменшення шуму, згладжування меж тіні та збереження геометричних контурів.

Фрустум – видимий об’єм камери у тривимірному просторі, обмежений ближньою та дальньою площинами відсікання, а також боковими площинами поля зору.

API – Application Programming Interface, програмний інтерфейс прикладного програмування.

CPU – Central Processing Unit, центральний процесор.

CSM – Cascaded Shadow Maps, метод побудови тіней.

D3D12 – Direct3D 12, низькорівневий графічний API для роботи з графічним процесором.

DXR – DirectX Raytracing, розширення Direct3D 12 для апаратно-прискореного трасування променів.

GBuffer – Geometry Buffer, набір текстурних ресурсів, що містять дані про поверхні сцени, зокрема геометрії, нормалей, кольору та фізичних властивостей.

GPU – Graphics Processing Unit, графічний процесор.

HLSL – High-Level Shader Language, мова програмування шейдерів для Direct3D 12.

IBL – Image-Based Lighting, метод освітлення сцени на основі зображення оточення.

LTC – Linearly Transformed Cosines, метод апроксимації освітлення від площинних джерел світла.

					ІАЛЦ.467800.004 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		5

MSM – Moment Shadow Mapping, метод побудови тіней на основі збереження кількох моментів розподілу глибини.

PBR – Physically Based Rendering, фізично коректна модель рендерингу матеріалів.

PCF – Percentage-Closer Filtering, метод згладжування тіней шляхом усереднення результатів кількох порівнянь глибини.

RT – Ray Tracing, трасування променів.

TLAS – Top-Level Acceleration Structure, верхній рівень структури прискорення для трасування променів.

VSM – Variance Shadow Maps, метод побудови тіней на основі збереження першого та другого моментів глибини.

Light bleeding – артефакт побудови тіней, за якого освітлення помилково просочується в затінену область.

Lighting pass – етап графічного конвеєра, на якому обчислюється фінальне освітлення сцени.

Peter penning – артефакт побудови тіней, за якого тінь візуально відривається від об'єкта, що її створює.

Shadow acne – артефакт побудови тіней, що проявляється у виді самозатінення, смуг або плям на поверхні об'єкта.

Motion vector buffer – буфер, у якому зберігається зміщення пікселя між поточним і попереднім кадрами.

TAA – Temporal Anti-Aliasing, метод темпорального згладжування, який використовує інформацію з попередніх кадрів, motion vector buffer, jitter проекції та history buffer.

Tonemapping – етап постобробки зображення, під час якого колірні значення з розширеного динамічного діапазону перетворюються у діапазон, придатний до відображення на екрані.

History buffer – буфер попереднього кадру, що використовується в TAA для повторного використання накопиченої інформації під час формування поточного зображення.

					ІАЛЦ.467800.004 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		6

Jitter – невелике субпіксельне зміщення проекції камери між кадрами, яке використовується для накопичення додаткової просторової інформації.

Deferred Shading – відкладене освітлення, техніка організації графічного конвеєра, за якого на першому етапі формуються GBuffer з параметрами поверхонь сцени, а обчислення освітлення виконується окремим проходом після завершення рендерингу геометрії.

LTC – (Linearly Transformed Cosines, метод апроксимації освітлення від площинних джерел світла, за допомогою попередньо обчислених таблиць і перетворених косинусних розподілів.

					ІАЛЦ.467800.004 ПЗ	Арк.
						7
Змін.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

У сучасних умовах розвитку систем комп'ютерної графіки реального часу все більшого набуває значення формування реалістичного освітлення у складних динамічних сценах. Одним із важливих елементів такого освітлення є тіні, оскільки вони дозволяють коректно сприймати просторове розташування об'єктів. Особливо актуальною є задача побудови тіней у сценах із великою кількістю динамічних джерел світла, де необхідно одночасно забезпечити достатню якість зображення, часову стабільність і прийнятну продуктивність.

Існує багато підходів до формування тіней у реальному часі. Вони мають порівняно високу швидкість та добре інтегруються у растеризаційний графічний конвеєр, однак мають низку обмежень. До них належать артефакти дискретизації, недостатня роздільна здатність тіньових карт, shadow acne, peter panning, light bleeding, складність масштабування для багатьох джерел світла та необхідність додаткової фільтрації результату. Screen-space підходи можуть бути швидкими, але залежать від інформації, доступної лише в межах поточного кадру, тому не враховують геометрію, яка перебуває поза екраном або прихована іншими об'єктами.

Більш геометрично коректним підходом є використання трасування променів, оскільки в цьому випадку видимість джерела світла визначається шляхом безпосередньої перевірки перетину променя з геометрією сцени. Проте повноекранне застосування ray tracing для великої кількості динамічних джерел світла є обчислювально дорогим.

Метою даного дипломного проєкту є розроблення програмних засобів реалізації атласу рей-трейсингових тіней для динамічних джерел світла у системі комп'ютерної графіки реального часу.

У межах даного проєкту передбачено розробку підсистеми, яка формує спільний атлас тіней для активних джерел світла, адаптивно визначає розмір області атласу для кожного джерела, будує карту відповідності texel-to-light, генерує карту видимості методом трасування променів та застосовує отриманий

результат під час фінального освітлення сцени. Для кожного текселя атласу визначається відповідне джерело світла, після чого за даними GBuffer відновлюється світова позиція точки поверхні та виконується трасування променя у напрямку джерела світла. Якщо промінь перетинає геометрію сцени, точка вважається затіненою; якщо перетину немає, вона вважається освітленою.

Отримана карта видимості використовується під час lighting pass для обчислення тіньового фактора. У подальшому він застосовується для модифікації внеску відповідного джерела світла у фінальне освітлення. Для підвищення якості результату в системі передбачено декілька режимів вибірки, апскейлу та фільтрації тіньових даних. Це дозволяє керувати компромісом між продуктивністю, м'якістю меж тіні, рівнем шуму та збереженням геометричних контурів.

Об'єктом розробки є процес формування тіней у графічних системах реального часу для сцен із динамічними джерелами світла. Предметом розробки є програмні засоби побудови спільного атласу рей-трейсингових тіней, генерації сирової карти видимості та застосування тіньового фактора у гібридному графічному конвеєрі.

У даному дипломному проєкті розглянуто існуючі методи побудови тіней у реальному часі, обґрунтовано вибір програмного та апаратного забезпечення, описано архітектуру розробленої системи, особливості формування атласу тіней, генерацію карти видимості, інтеграцію результату в кластерне відкладене освітлення, а також методи фільтрації та параметричного налаштування якості тіней. Запропонована система може бути використана як складова частина графічного рушія або інтерактивної системи візуалізації, де необхідно формувати тіні від великої кількості динамічних джерел світла з керованим балансом між якістю та продуктивністю.

1. ПОРІВНЯЛЬНІ ХАРАКТЕРИСТИКИ СУЧАСНИХ СИСТЕМ КОМП'ЮТЕРНОЇ ГРАФІКИ ДЛЯ ФОРМУВАННЯ ТІНЕЙ

1.1 Особливості формування тіней у сучасних системах комп'ютерної графіки

Формування тіней є ключовою складовою сучасних систем комп'ютерної графіки. Саме тіні забезпечують візуальне сприйняття взаємного розташування об'єктів, їхнього зв'язку та рівня просторової правдоподібності зображення. За відсутності коректного затінення деталізовані геометричні моделі та складні моделі освітлення сприймаються штучно, оскільки користувач втрачає важливі зорові ознаки, що характеризують положення джерела світла, форму об'єкта та його взаємодії із середовищем. Саме тому система тіней є повноцінним компонентом освітлення сцени.

Особливістю сучасних систем візуалізації є необхідність формування тіней у реальному часі. На відміну від офлайн-рендерингу, де допустимими є значні витрати обчислювальних ресурсів на один кадр, у інтерактивних системах, таких як ігрові рушії, тренажери, візуалізаційні комплекси та системи віртуальної реальності, побудова тіней повинна поєднувати високу швидкодію з візуальною якістю. У зв'язку з цим основною проблемою стає пошук компромісу між точністю моделювання поширення світла, стабільністю зображення під час руху камери та кількістю обчислень, які може дозволити собі графічний конвеєр.

Ще однією важливою особливістю є динамічний характер сучасних сцен. У більшості практичних застосувань змінюється не лише положення камери, а й положення, орієнтація, параметри джерел світла та самих об'єктів. Це означає, що тіні повинні перебудовуватись безперервно, коректно реагуючи на всі зміни сцени. У таких умовах особливого значення набуває не лише якість статичного зображення, а й часова стабільність тіней.

Складність задачі додатково зумовлюється різноманітністю типів джерел світла, що використовуються у сучасних системах комп'ютерної графіки.

Напрявлені, точкові, прожекторні та площинні джерела мають різні геометричні та фотометричні властивості, а отже, породжують різні вимоги до алгоритмів затінення. Для напрямлених джерел характерним є формування тіней на великих ділянках сцени, для точкових - необхідність урахування відстані до джерела та сферичної природи випромінювання, для прожекторів - додаткова перевірка належності точки до світлового конуса, а для площинних джерел - потреба моделювання м'яких тіней із вираженою напівтіннію. Унаслідок цього універсальна система тіней повинна бути достатньо гнучкою, щоб працювати з різними типами освітлення в межах єдиного конвеєра рендерингу.

Для сучасних графічних систем також характерна наявність великої кількості динамічних джерел світла в межах однієї сцени. Класичні підходи, в яких для кожного джерела формується окрема повнорозмірна карта тіней, стають дуже витратними як за обсягом відеопам'яті, так і за часом побудови. Тому актуальними є методи, що забезпечують здатність підтримувати значну кількість джерел без пропорційного зростання обчислювальної вартості. Досягнення такої масштабованості зазвичай пов'язане з адаптивним представленням тіньових даних, вибіркоким виділенням роздільної здатності для окремих джерел, кластеризованим освітленням, а також різноманітними механізмами пріоритезації, що дозволяють виділяти більше ресурсів візуально важливим джерелам і менше — другорядним.

Суттєвою особливістю побудови тіней у реальному часі є наявність характерних артефактів, які необхідно мінімізувати в межах обмеженого часу кадру. До них належать зубчастість меж тіней, нестача просторової роздільної здатності, самозатінення поверхонь, відрив тіні від об'єкта, витік освітлення через межі геометрії, шум і мерехтіння [2-7],[15]. Причиною таких дефектів є дискретна природа представлення сцени в буферах і обмеженість кількості вибірок, доступних у межах одного кадру. Тому подальший аналіз методів побудови тіней має враховувати не лише спосіб визначення видимості світла, а й механізми підвищення візуальної якості результату.

1.2 Аналіз існуючих методів побудови тіней у реальному часі.

Протягом розвитку графічних технологій було запропоновано значну кількість підходів до формування тіней, кожен з яких має власні переваги. Частина методів орієнтована на мінімізацію витрат пам'яті та часу обчислення, інші забезпечують кращу геометричну точність або вищу якість меж тіней. Аналіз існуючих методів є необхідним для визначення підходу, який найбільш повно відповідає вимогам систем, що повинні підтримувати значну кількість динамічних джерел світла та забезпечувати якість затінення в реальному часі.

1.2.1 Shadow Mapping

Одним із найпоширеніших і найбільш вживаних методів побудови тіней у реальному часі є Shadow Mapping. Схема роботи даного методу зображена на рисунку 1.1.

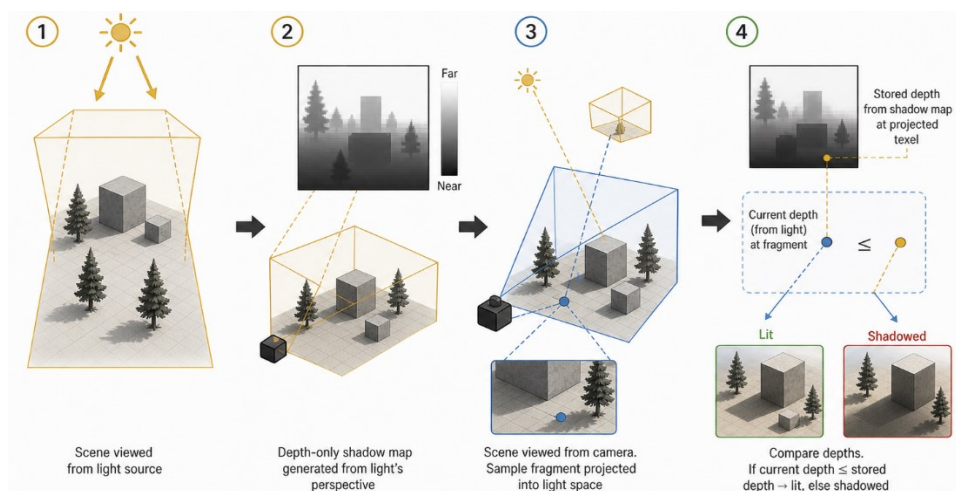


Рисунок 1.1 – Схема роботи методу Shadow Mapping

Основна ідея цього підходу полягає у попередньому рендерингу сцени з

точки зору джерела світла для побудови карти глибин, яка використовується для визначення, чи є точка сцени видимою для світла [2]. Якщо глибина точки, що освітлюється, перевищує значення, записане у shadow map, така точка вважається затіненою. В іншому випадку вона розглядається як освітлена.

Перевагою Shadow Mapping є його простота реалізації, хороша сумісність із традиційним растеризаційним конвеєром і порівняно невисока вартість обчислення для невеликої кількості джерел світла. Саме завдяки цим властивостям цей метод став фактичним стандартом для ігрових рушіїв та інтерактивних візуалізаційних систем протягом тривалого часу.

Водночас базовий варіант Shadow Mapping має низку суттєвих недоліків, наведених на рисунку 1.2.

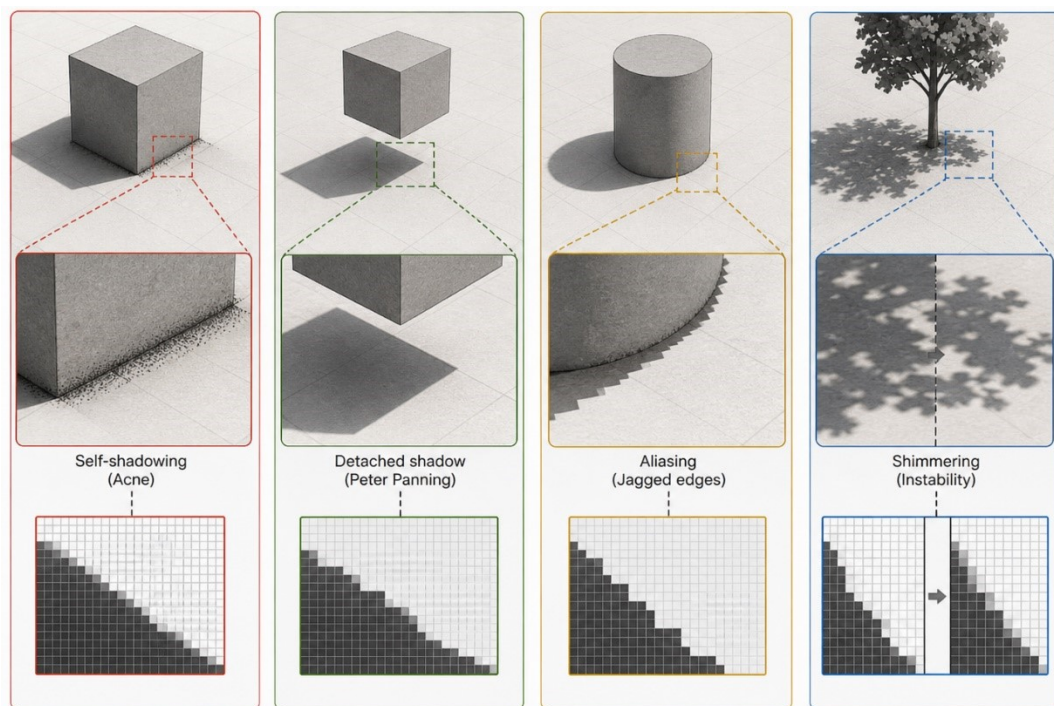


Рисунок 1.2 – зображення артефактів методу Shadow Mapping

Насамперед це обмежена роздільна здатність карти тіней, яка призводить до зубчастості меж та появи артефактів дискретизації. Крім того, метод чутливий до проблем самозатінення, зміщення тіні від об'єкта та нестачі точності при великих масштабах сцени. Значною проблемою є й те, що для кожного

динамічного джерела світла, як правило, потрібна окрема карта тіней, що ускладнює масштабування системи на велику кількість світлових джерел.

1.2.2 Variance Shadow Maps

Одним із важливих етапів розвитку методів побудови тіней у реальному часі стали Variance Shadow Maps. На відміну від класичного Shadow Mapping, у VSM використовується збереження статистичних моментів розподілу глибини, що дає змогу виконувати лінійну фільтрацію тіньової карти [5],[6]. Саме ця властивість робить можливим застосування до таких карт звичайної білінійної фільтрації, mipmaping, а також більш складних схем. У практичному сенсі це дозволяє будувати м'якші межі тіней і зменшувати ефект аліасингу без необхідності виконання великої кількості індивідуальних вибірок порівнянь глибини. Загальна схема роботи зображена на рисунку 1.3

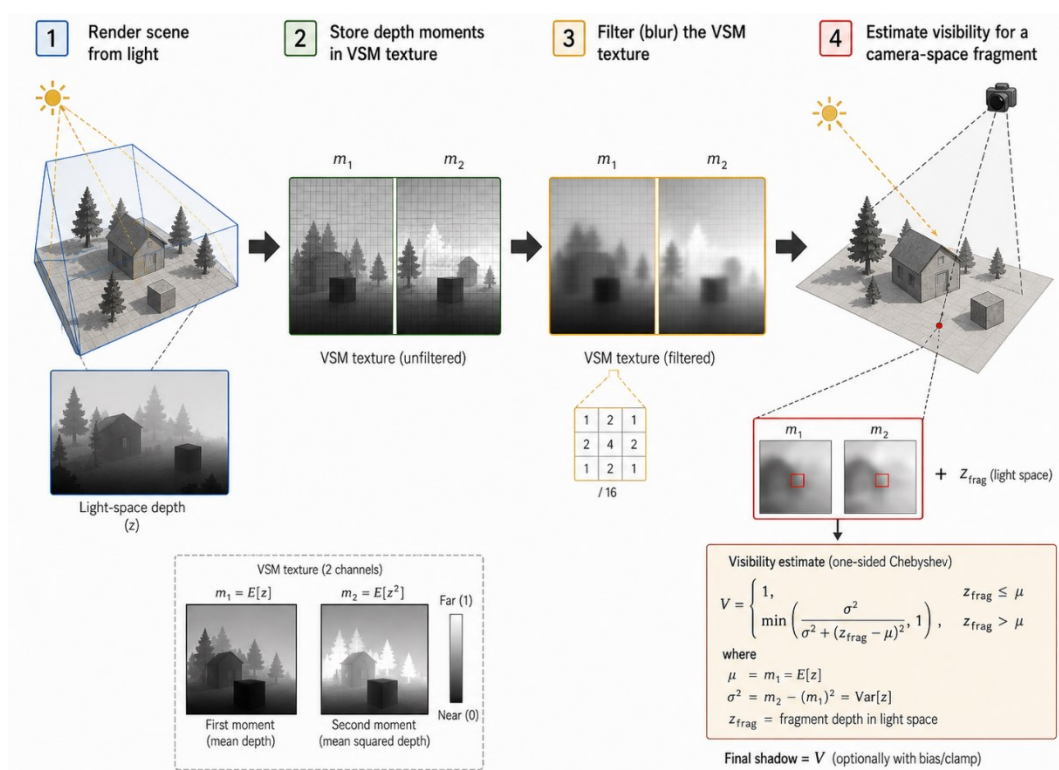


Рисунок 1.3 – Схема роботи методу Variance Shadow Maps

Перевагою Variance Shadow Maps є висока швидкодія при фільтрації та добра сумісність із графічним конвеєром реального часу. Крім того, цей підхід є особливо привабливим для задач, де потрібна змінна ширина фільтра або швидке наближене формування м'яких тіней. Разом із тим VSM мають і суттєве обмеження: через статистичну природу оцінювання видимості їм властивий артефакт light bleeding – коли освітлення частково просочується у зону тіні [5],[6]. Попри це, VSM залишаються важливим етапом еволюції підходів тіньових карт, оскільки вперше зробили фільтрацію тіней по-справжньому ефективною та придатною для використання в реальному часі.

1.2.3 Moment Shadow Mapping

Подальшим розвитком ідей фільтрованого представлення тіней став метод Moment Shadow Mapping, схема роботи якого зображена на рисунку 1.4.

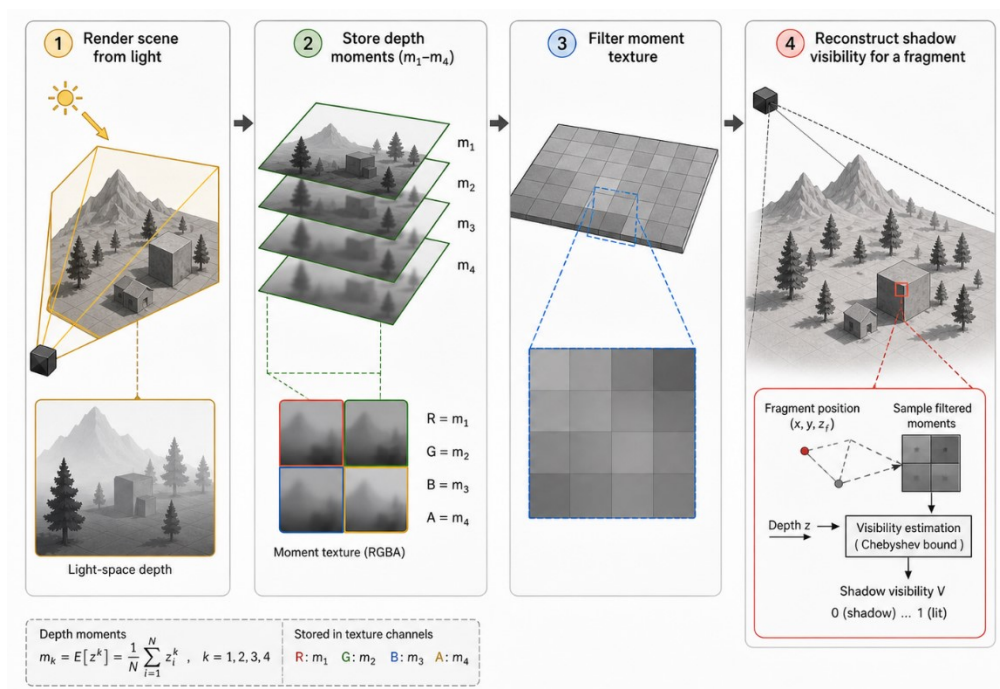


Рисунок 1.4 – Схема роботи методу Moment Shadow Mapping

У техніці, як і у VSM, тіньова карта подається у формі придатній до прямої

фільтрації. Однак замість обмеження двома моментами використовуються розширені моментні представлення, що дозволяє точніше описувати розподіл глибини в межах області фільтрації [7]. Завдяки цьому MSM забезпечує швидке формування згладжених жорстких тіней і водночас значно краще контролює проблему light leaking. Автори методу прямо позиціонують MSM як швидку техніку для фільтрованих жорстких тіней, що зберігає переваги префільтрації, але підвищує надійність результату.

Подальший розвиток цього напрямку пов'язаний із нелінійно квантизованими MSM, де моментне представлення стискається у форму, зручнішу для практичного використання у графічних системах із жорсткими вимогами до пам'яті. Такі варіанти дозволяють отримувати жорсткі тіні зі згладженими межами та мінімальним проникненням світла де вартість обчислень залишається близькою до VSM. Це робить MSM значущою сучасною технікою, яку доцільно розглядати як один із найрозвиненіших представників фільтрованих карт тіней [7].

1.2.4 Cascaded Shadow Maps

Для напрямлених джерел світла, які впливають на значні просторові області, було запропоновано метод Cascaded Shadow Maps, схему роботи якого зображено на рисунку 1.5.

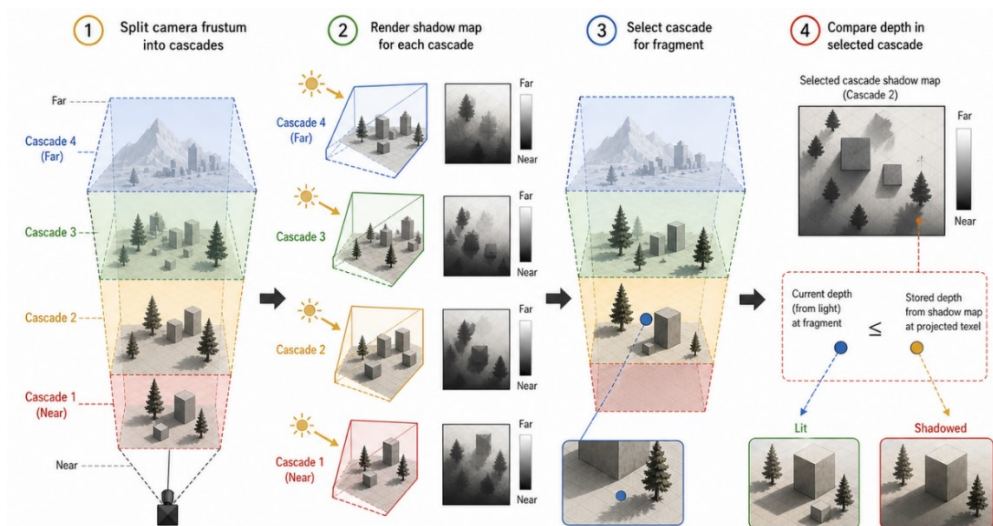


Рисунок 1.5 – Схема роботи методу Cascaded Shadow Maps

Суть методу полягає у розбитті видимого об'єму сцени на кілька каскадів за глибиною відносно камери, для кожного з яких формується окрема карта тіней із власним масштабом проєкції [8]. Такий підхід дозволяє зосередити більшу роздільну здатність поблизу камери, де дефекти затінення найбільш помітні, та зменшити деталізацію на віддалених ділянках сцени.

Перевага цього методу полягає у покращенні якості тіней у ближній до спостерігача області без необхідності глобального збільшення розміру карти тіней. Завдяки цьому Cascaded Shadow Maps стали поширеним рішенням для візуалізації сонячного освітлення у великих відкритих сценах. Однак цей підхід має й обмеження. По-перше, він орієнтований переважно на напрямлене освітлення і не є універсальним для довільних типів джерел світла. По-друге, каскадне розбиття може спричиняти артефакти на межах каскадів. Крім того, зі зростанням кількості джерел світла каскадний підхід стає обчислювально дорогим, оскільки для кожного джерела або каскаду необхідні окремі проходи рендерингу.

1.2.5 Screen-Space методи для формування тіней

Альтернативним напрямом розвитку систем тіней стали screen-space методи, у яких затінення обчислюється безпосередньо у просторі екрана на основі вже сформованих буферів атрибутів поверхні. Головна перевага таких методів полягає в тому, що вони не потребують окремого рендерингу сцени з точки зору джерела світла, а отже, дозволяють зменшити кількість додаткових графічних проходів [9], [11]. Це робить їх доцільними для інтерактивних застосувань, де критично важлива економія часу кадру. Загальна схема роботи screen-space методів зображена на рисунку 1.6.

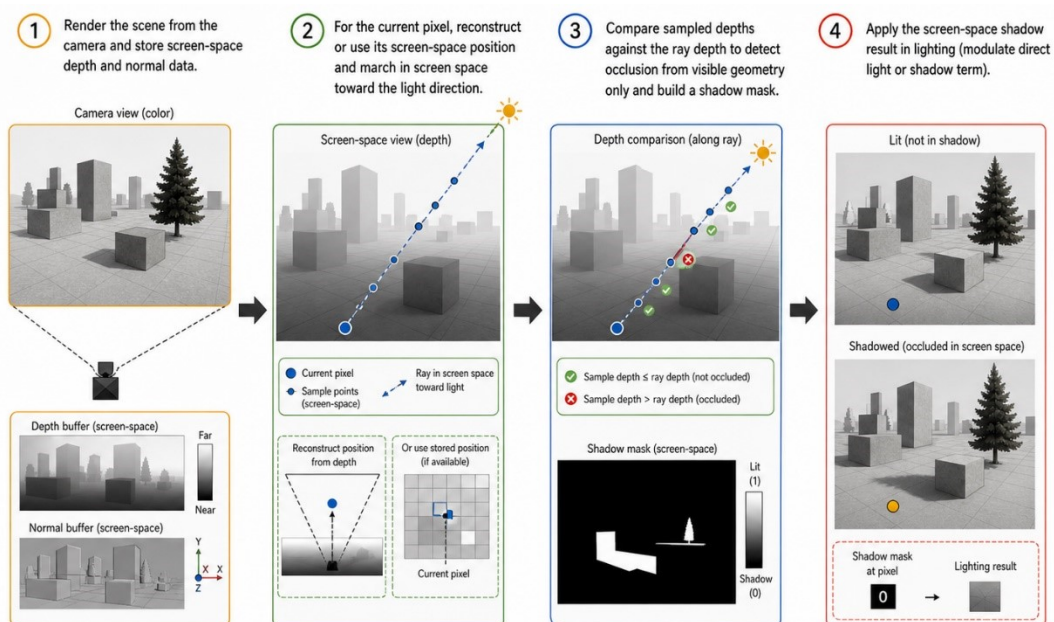


Рисунок 1.6 – Схема роботи Screen-space методів.

Але Screen-space підходи мають фундаментальні обмеження, пов'язані з тим, що вони працюють лише з тією частиною геометрії, яка вже потрапила у видимий об'єм камери. Геометрія, що знаходиться поза екраном або прихована іншими об'єктами, у таких методах не враховується, що призводить до втрати повноти інформації про сцену [1], [9]. У результаті можливі помилки в затіненні, витік освітлення та нестабільність результату при зміні ракурсу.

Змін.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467800.004 ПЗ

Арк.

18

Тому зазвичай screen-space тіні використовують як допоміжні тіні для не пріоритетних джерел світла.

1.2.6 Ray Traced Shadows

З розвитком апаратної підтримки трасування променів особливого поширення набули методи Ray Traced Shadows, що базуються на безпосередньому трасуванні променя від точки поверхні до джерела світла або у протилежному напрямку променя освітлення, зображений на рисунку 1.7.

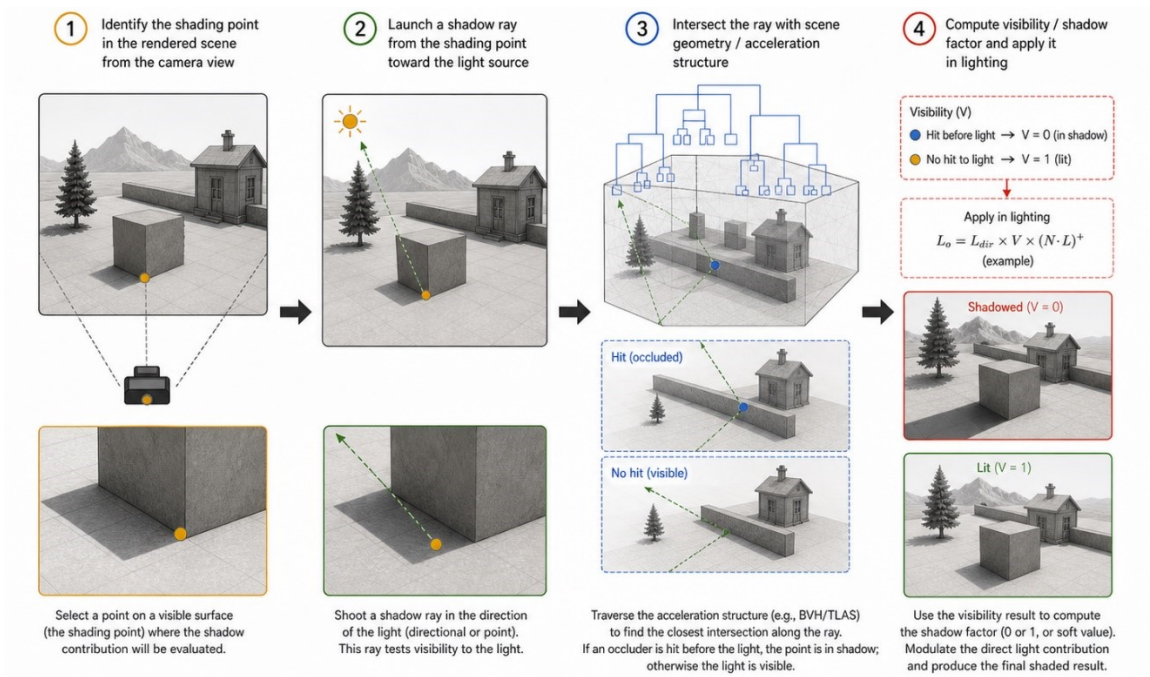


Рисунок 1.7 – Схема роботи Ray Traced Shadows.

На відміну від shadow mapping, де видимість світла визначається через опосередковане порівняння глибин у карті тіней, трасування променів дозволяє перевіряти реальне геометричне перекриття у просторі сцени [10], [14]. Це забезпечує вищу точність, зменшення кількості типових артефактів shadow map та кращу узгодженість тіней із реальною геометрією.

Перевагою Ray Traced Shadows є також природна підтримка складних випадків затінення, зокрема для різних типів джерел світла, складної геометрії та

детальних сцен. Такий підхід дозволяє отримувати фізично більш правдоподібний результат і краще адаптується до динамічних змін положення джерел світла та об'єктів. Проте пряме застосування трасування променів для кожного джерела світла та кожного пікселя екрана є вкрай витратним, особливо у сценах із великою кількістю динамічних джерел.

1.2.7 Методи фільтрації та апскейлу тіньових даних у системах реального часу

У разі коректного визначення видимості світла важливу роль у сучасних системах побудови карт тіней відіграють методи обробки тіньових даних. Це пов'язано з тим, що карти тіней часто мають обмежену роздільну здатність, а сам результат затінення може містити шум, дискретні переходи та локальні артефакти. У зв'язку з цим у системах реального часу застосовуються методи апскейлу, інтерполяції та фільтрації, які дозволяють покращити сприйняття меж тіней без пропорційного збільшення вартості вихідної побудови.

Найпростішими методами вибірки тіней є point sampling та bilinear interpolation. Point підхід передбачає використання найближчого доступного значення без будь-якого згладжування, тому характеризується мінімальною вартістю, але створює виражену зубчастість меж, зображену на рисунку 1.8.

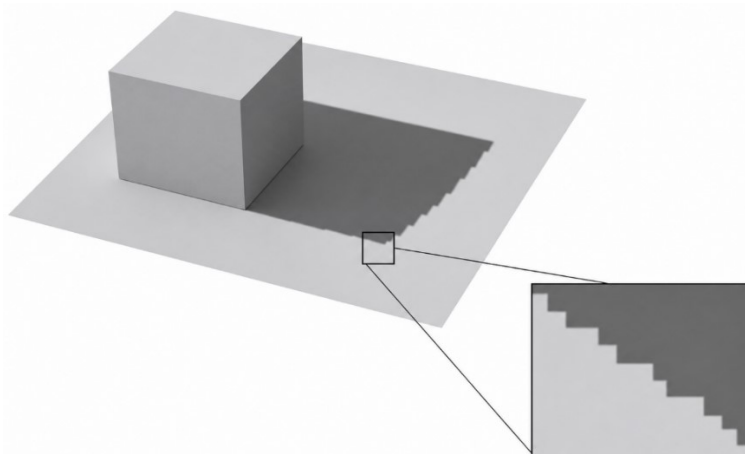


Рисунок 1.8 – Зубчастість межі тіні в результаті point sampling

					ІАЛЦ.467800.004 ПЗ	Арк.
						20
Змін.	Арк.	№ докум.	Підпис	Дата		

Bilinear підхід забезпечує плавніший результат завдяки лінійній інтерполяції між сусідніми вибірками, проте не враховує реальну геометрію сцени й може призводити до небажаного проникнення тіні або освітлення через межі об'єктів. Такі методи доцільно розглядати як базові або проміжні, але їхньої якості зазвичай недостатньо для складних сучасних сцен.

Метод Percentage Closer Filtering є одним із найпоширеніших способів пом'якшення меж тіней, зображеного на рисунку 1.9.

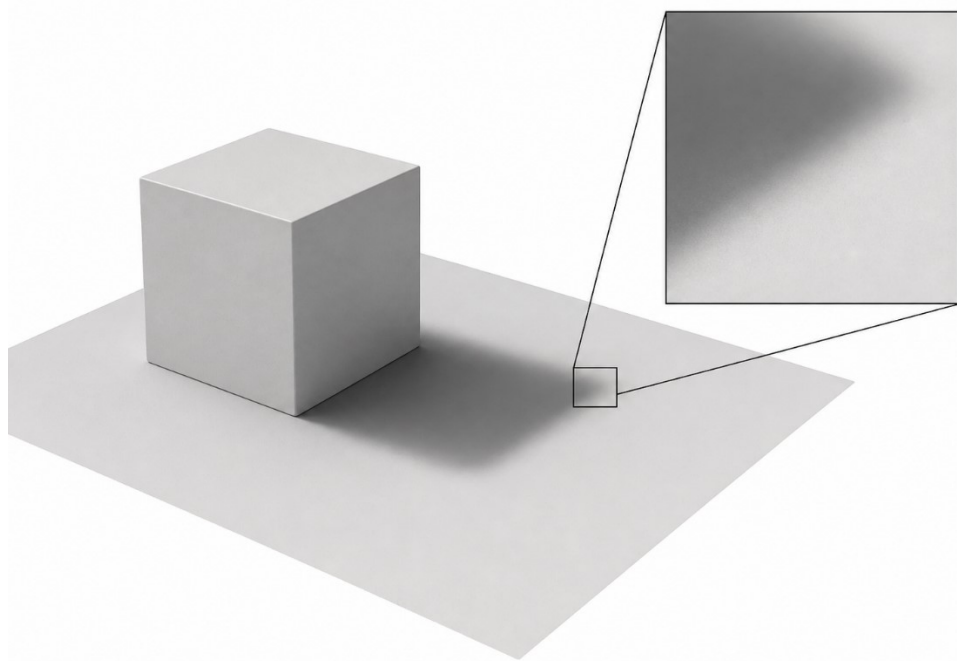


Рисунок 1.9 – Приклад результату роботи PCF.

Його суть полягає у виконанні декількох порівнянь у локальному okolí та усередненні результатів групу складають методи [4]. Це дозволяє отримати візуально м'якші переходи між освітленими та затіненими ділянками. Перевагою PCF є передбачуваний результат. Водночас зі збільшенням розміру ядра різко зростає кількість вибірок і вартість обчислення. Крім того фільтрація цього типу не враховує інформацію про геометричні межі сцени, що може спричиняти розмивання контурів і втрату чіткості деталей.

Для зменшення небажаного розмивання меж були розроблені bilateral та joint bilateral методи. Їхня ідея полягає у зважуванні сусідніх вибірок не лише за просторовою відстанню, а й за подібністю глибини та нормалі поверхні. Це дозволяє зберігати геометричні межі, зменшуючи витік освітлення через контури об'єктів. Дані підходи ефективні в гібридних системах, де дані GBuffer вже доступні після етапу растеризації. Основним недоліком bilateral-фільтрації є вища обчислювальна складність порівняно з простими методами інтерполяції, а також ризик появи зернистості та нестабільності при надто агресивних параметрах відсікання.

Існують методи, що використовують стохастичні або квазірівномірні шаблони вибірки, зокрема Poisson Disk Sampling, результат роботи якого зображено на рисунку 1.10. Їх застосування дозволяє формувати візуально м'якші та природніші переходи у напівтіні. У поєднанні з квазівипадковим обертанням шаблону вибірки ці методи добре підходять для побудови м'яких тіней та пригнічення просторових артефактів. Недоліком є можливість появи шуму або мерехтіння у разі недостатньої кількості вибірок, що часто вимагає додаткових механізмів стабілізації.

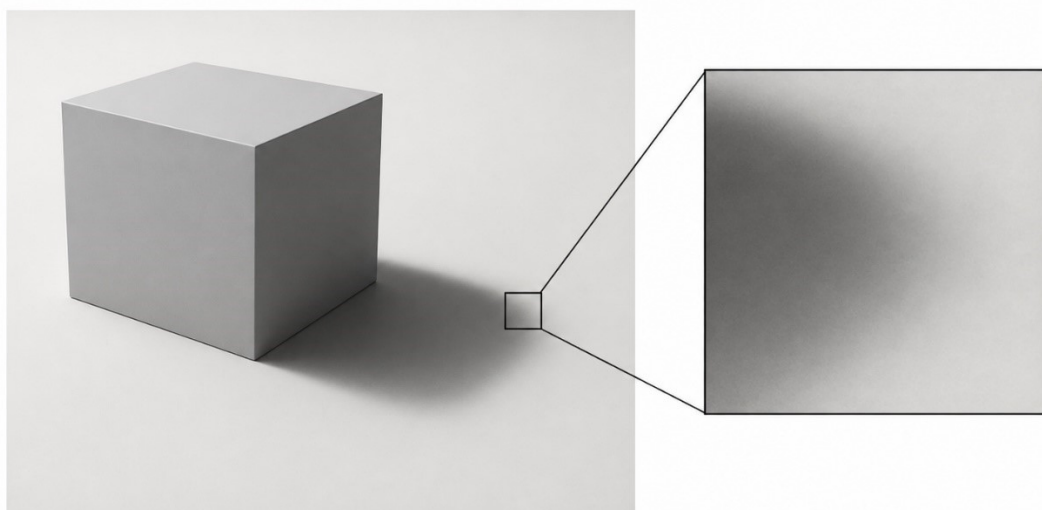


Рисунок 1.10 – Приклад результату роботи Poisson Disk Sampling.

Порівняння методів фільтрації свідчить, що універсального рішення, оптимального за всіма критеріями, не існує. Найпростіші методи вибірки забезпечують мінімальні витрати ресурсів, але не дають достатньої якості. PCF покращує візуальне сприйняття тіней, однак має тенденцію до розмивання меж і швидко дорожчає зі збільшенням розмірів ядра. Bilateral та Joint-Bilateral підходи краще зберігають геометричну структуру сцени, проте вимагають доступу до допоміжних буферів і чутливі до вибору параметрів. Poisson-based схеми здатні зменшити регулярність артефактів і сформувати більш природне затінення, однак потребують ретельного контролю шуму та часової стабільності. Тому в масштабованих системах тіней найдоцільнішим є використання кількох режимів фільтрації з можливістю адаптивного вибору або комбінування залежно від поточних вимог до якості та продуктивності.

1.3 Порівняльний аналіз існуючих підходів до побудови тіней

Доцільно виконати порівняльний аналіз методів побудови карт тіней за критеріями, які мають практичне значення для сучасних графічних систем. До таких критеріїв належать візуальна якість сформованих тіней, обчислювальна продуктивність, вимоги до пам'яті, стабільність результату та масштабованість у сценах із великою кількістю динамічних джерел світла.

Базовий метод Shadow Mapping забезпечують прийнятний результат у відносно простих сценах і залишаються поширеним рішенням завдяки простоті реалізації та високій швидкодії. Більшість обчислень у цьому випадку виконується засобами традиційної растеризації, яка добре оптимізована сучасним графічним обладнанням. Водночас якість таких методів істотно залежить від роздільної здатності карти тіней, правильного налаштування зміщення глибини та способу фільтрації. Для цього класу підходів типовими є артефакти shadow acne, peter panning та зубчастість меж тіні. Крім того, зі збільшенням кількості джерел світла зростає кількість необхідних тіньових карт,

					ІАЛЦ.467800.004 ПЗ	Арк.
						23
Змін.	Арк.	№ докум.	Підпис	Дата		

проходів рендерингу та обсяг використовуваної відеопам'яті, що обмежує масштабованість даного методу у багатоджерельних сценах.

Метод CSM частково розв'язує проблему нестачі роздільної здатності для напрямленого освітлення шляхом поділу області видимості камери на декілька каскадів. Це дозволяє підвищити деталізацію тіней у ближній до камери частині сцени, де дефекти найбільш помітні для спостерігача. Проте такий підхід є дорожчим за звичайне використання однієї карти тіней, оскільки для одного джерела світла необхідно формувати кілька тіньових представлень. Крім того, можливими залишаються артефакти на межах переходу між каскадами, неоднорідність якості на різних відстанях і залежність результату від налаштувань розбиття області видимості. Через це використання даного методу є доцільним лише для головних напрямлених джерел світла.

VSM та MSM покращують насамперед можливості фільтрації тіньових карт. VSM зберігає статистичні характеристики глибини та допускає ефективну лінійну фільтрацію, завдяки чому може формувати плавніші межі тіней із меншою вартістю, ніж схеми з великою кількістю depth-comparison вибірок. Однак цей метод схильний до артефакту light bleeding, за якого світло помилково потрапляє в затінені області. MSM використовує складніше представлення і краще контролює такі артефакти, але потребує складнішого у обчисленнях відновлення тіньового фактора. Обидва підходи покращують якість і вартість фільтрації вже побудованих тіньових карт, проте не усувають необхідності створення окремого тіньового представлення для активних джерел світла. Тому їхня перевага стосується переважно якості обробки тіньових даних, а не архітектурної масштабованості всієї системи.

Screen-Space методи мають низьку обчислювальну вартість, оскільки використовують уже наявні дані поточного кадру, зокрема глибину, нормалі та інші атрибути поверхонь. Вони не потребують окремого рендерингу сцени з позиції джерела світла та можуть добре працювати як допоміжний ефект для локального покращення затінення. Проте головним обмеженням таких підходів

					ІАЛЦ.467800.004 ПЗ	Арк.
						24
Змін.	Арк.	№ докум.	Підпис	Дата		

є неповнота інформації про сцену: метод працює лише з геометрією, яка вже присутня в екранному просторі. Через це не враховуються позакадрові та приховані об'єкти, можливі помилки на краях екрана, зникнення тіней при зміні ракурсу та нестабільність при русі камери. Тому хоч і screen-space підходи можуть бути швидкими, але поступаються іншим методам за геометричною достовірністю.

Найвищу геометричну точність забезпечують Ray Traced Shadows, оскільки вони ґрунтуються на безпосередній перевірці перекриття шляху світла реальною геометрією сцени. Такий підхід краще узгоджується з просторовими взаємозв'язками між об'єктами, дозволяє коректніше обробляти складну геометрію та зменшує кількість артефактів, пов'язаних із дискретизацією карт тіней. Водночас прямолінійне повноекранне трасування для багатьох джерел світла є обчислювально дорогим, оскільки кожна перевірка видимості потребує трасування променя у структурі прискорення сцени.

З погляду масштабованості для великої кількості динамічних джерел світла найбільш перспективними є не окремі класичні методи, а гібридні архітектури, що поєднують просторове групування джерел, адаптивний розподіл обчислювальних ресурсів і вибіркове підвищення якості для візуально значущих освітлювачів. Кластерне освітлення дозволяє обмежити перелік джерел, які враховуються для просторового кластера, а адаптивне представлення тіньових даних дає змогу не витратити однаковий обсяг ресурсів на всі джерела світла. У результаті формується архітектура, здатна підтримувати значну кількість динамічних джерел без пропорційного зростання витрат на кожне з них.

1.4 Аналіз проблем реалізації тіней для багатьох динамічних джерел світла

Реалізація тіней для багатьох динамічних джерел світла є комплексною задачею. На відміну від сцен з одним основним джерелом світла, багатоджерельні сцени потребують механізмів відбору, пріоритезації та

					ІАЛЦ.467800.004 ПЗ	Арк.
						25
Змін.	Арк.	№ докум.	Підпис	Дата		

адаптивного розподілу ресурсів між джерелами світла [1].

Однією з головних проблем є організація обчислень. Збільшення кількості джерел світла не повинно призводити до неконтрольованого зростання часу кадру. У реальній системі вартість формується не лише самим алгоритмом побудови тіні, а й кількістю службових ресурсів, числом звернень до тіньових даних, кількістю вибірок під час фільтрації тіньового фактора та витратами на підготовку даних. Тому система тіней для багатьох джерел повинна розглядатися як частина загальної архітектури рендерингу, а не як окремий незалежний ефект [1].

У сучасних графічних системах камера, об'єкти та джерела світла можуть змінюватися на кожному кадрі, що вимагає постійного оновлення тіньових даних. Якщо система не встигає адаптуватися до цих змін або використовує надто грубі наближення, виникають мерехтіння меж тіні, стрибкоподібні зміни деталізації, шум і часова нестабільність результату.

Окремою проблемою є обмеженість роздільної здатності тіньового представлення. У системах реального часу неможливо безкінечно збільшувати деталізацію кожної тіні, тому алгоритм повинен поєднувати знижене або адаптивне представлення видимості з подальшим відновленням візуальної якості. Для цього застосовуються апскейл і фільтрація, які мають зменшувати дискретність та шум, не руйнуючи геометричні межі тіні. Водночас залишаються актуальними класичні артефакти затінення, зокрема самозатінення, відрив тіні від об'єкта, помилки на дотичних кутах і витік світла через межі геометрії.

Важливою умовою масштабованості є також інтеграція тіней у загальний конвеєр освітлення. У сценах із великою кількістю джерел повний перебір усіх освітлювачів для кожного пікселя є надмірно дорогим. Тому система тіней повинна узгоджуватися з механізмами просторового відбору, зокрема з кластерним освітленням, який обмежує перелік релевантних джерел для конкретної області екрана. Така інтеграція потребує узгодження структур даних освітлення, GBuffer-ресурсів та параметрів фільтрації в межах одного конвеєра.

Тому основні проблеми реалізації тіней для багатьох динамічних джерел світла полягають у необхідності одночасного забезпечення масштабованості, високої візуальної якості, прийнятної швидкодії, часової стабільності та ефективного використання пам'яті. Класичні підходи не дають повного розв'язання цієї сукупності вимог, оскільки або погано масштабуються, або не забезпечують достатньої геометричної коректності, або є надто дорогими в обчислювальному сенсі. Саме тому доцільним є розроблення гібридних систем, у яких трасування променів поєднується з адаптивним представленням тіньових даних, пріоритезацією джерел світла, кластеризованим освітленням та фільтрацією тіньового фактора під час застосування тіней [1].

1.5 Обґрунтування актуальності теми дипломного проєкту

Актуальність теми дипломного проєкту зумовлена зростанням складності інтерактивних тривимірних сцен, у яких одночасно використовуються численні динамічні джерела світла, рухома камера та змінна геометрія об'єктів. У таких умовах традиційні підходи до побудови тіней не завжди забезпечують необхідний баланс між геометричною коректністю, часовою стабільністю та продуктивністю. Особливо ця проблема проявляється у сценах з великою кількістю джерел світла, де побудова окремого тіньового представлення для кожного освітлювача стає надмірно витратною.

Одним із напрямів розв'язання цієї проблеми є використання гібридної архітектури, у якій трасування променів відповідає за геометрично коректне визначення видимості, а адаптивне представлення тіньових даних і кластерне освітлення обмежують обсяг необхідних обчислень. У межах даної роботи такий підхід реалізується через спільний атлас рей-трейсингових тіней, де різним джерелам світла виділяються області адаптивного розміру, а вибірка, апскейл і фільтрація тіньового фактору виконуються під час застосування тіней у lighting pass.

					ІАЛЦ.467800.004 ПЗ	Арк.
						27
Змін.	Арк.	№ докум.	Підпис	Дата		

Практична цінність обраної теми полягає в тому, що запропонований підхід орієнтований не лише на отримання факту затінення, а й на забезпечення керованої якості результату. Система підтримує різні режими фільтрації, параметричне налаштування візуального вигляду тіней і інтеграцію з кластерним освітленням. Це дозволяє розглядати розроблення засобів реалізації атласу рейтрейсингових тіней для динамічних джерел світла як актуальний напрям розвитку систем комп'ютерної графіки реального часу [1].

					ІАЛЦ.467800.004 ПЗ	Арк.
						28
Змін.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ ДО РОЗДІЛУ 1

У першому розділі було проаналізовано особливості формування тіней у сучасних системах комп'ютерної графіки та встановлено, що підсистема тіней є важливим компонентом реалістичної візуалізації, оскільки впливає на сприйняття глибини сцени, просторового розташування об'єктів і правдоподібності освітлення. Розглянуто основні підходи до побудови тіней у реальному часі, зокрема Shadow Mapping, Variance Shadow Maps, Moment Shadow Mapping, Cascaded Shadow Maps, Screen-Space методи та Ray Traced Shadows. Показано, що методи на основі карт тіней мають прийнятну продуктивність у простіших сценаріях, але обмежені за масштабованістю та схильні до артефактів дискретизації, тоді як трасування променів забезпечує вищу геометричну коректність, але потребує спеціальних засобів зменшення обчислювальної вартості.

У результаті порівняльного аналізу встановлено, що для сцен із великою кількістю динамічних джерел світла доцільним є застосування гібридного підходу, який поєднує трасування променів, адаптивне представлення тіньових даних, пріоритезацію джерел світла, кластерне освітлення та апскейл з фільтрацією тіньового фактора під час застосування тіней у lighting pass. У межах даної роботи такий підхід реалізується на основі спільного атласу рейтрейсингових тіней для динамічних джерел світла, що створює основу для подальшого опису архітектури та програмної реалізації системи.

2. ЗАСОБИ РЕАЛІЗАЦІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Обґрунтування вибору архітектури графічної підсистеми

Архітектура графічної підсистеми для реалізації атласу рей-трейсингових тіней повинна забезпечувати підтримку великої кількості динамічних джерел світла, можливість побудови тіней у реальному часі, ефективне використання даних сцени та баланс між візуальною якістю і швидкодією. З огляду на це в межах роботи було обрано гібридну архітектуру, що поєднує відкладене та кластерне освітлення, трасування променів та адаптивне представлення тіньових даних у спільному атласі. Такий підхід дає змогу відокремити формування геометричних і матеріальних параметрів поверхні від етапу освітлення та зменшити кількість перевірок джерел світла для кожного пікселя.

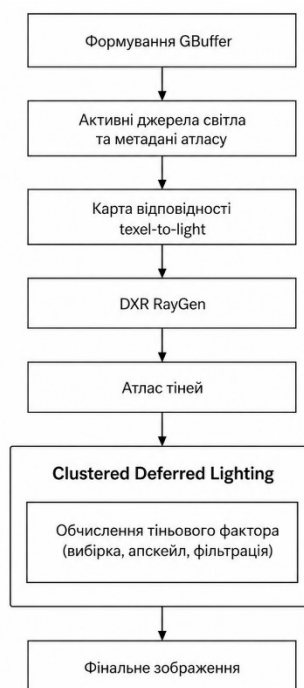


Рисунок 2.1 – Загальна архітектура графічної підсистеми з інтеграцією атласу тіней

На рисунку 2.1 показано загальну послідовність роботи графічної підсистеми з інтеграцією атласу рей-трейсингових тіней. Спочатку формується

GBuffer, який містить дані, необхідні для подальшого відновлення параметрів поверхні. Після цього визначаються активні джерела світла, формуються метадані атласу та карта відповідності texel-to-light. Далі DXR RayGen використовує ці дані для запису значень видимості в атлас тіней. На етапі Clustered Deferred Lighting для кожного релевантного джерела виконується обчислення тіньового фактора, що включає вибірку з атласу, апскейл і фільтрацію.

2.1.1 Використання deferred shading

Вибір deferred shading як базової архітектури освітлення зумовлений необхідністю роботи з великою кількістю динамічних джерел світла. У forward-підході обчислення освітлення виконується безпосередньо під час рендерингу геометрії, що призводить до зростання вартості обробки зі збільшенням числа джерел світла. Натомість deferred shading дозволяє спочатку сформувати набір буферів, у яких зберігаються основні параметри поверхні, після чого освітлення обчислюється окремим проходом [10], [11]. Це особливо важливо для задачі, у якій тіньовий фактор повинен застосовуватися до великої кількості джерел світла в межах одного кадру.

Перевагою такого підходу є можливість повторного використання даних буферів для кількох незалежних етапів. У розробленій системі саме depth- та normal- компоненти GBuffer використовуються для відновлення параметрів точки поверхні, що надалі дає змогу виконувати трасування променя до джерела світла без побудови окремих геометричних представлень для кожного пікселя екрану.

2.1.2 Використання спільного атласу тіней для множини джерел світла

Окреме представлення тіней для кожного джерела світла є малоефективним

у сценах де їх багато. У зв'язку з цим у роботі було обрано підхід, за якого первинні значення видимості, отримані трасуванням променів для різних джерел світла, розміщуються в межах одного спільного атласу тіней. Кожному джерелу виділяється окрема прямокутна область, розмір якої визначається адаптивно залежно від його важливості для поточного кадру. Така організація дозволяє гнучко розподіляти доступну роздільну здатність між джерелами, концентруючи більше ресурсів на тих із них, які сильніше впливають на сприйняття сцени [1].

Перевагою спільного атласу є скорочення кількості окремих графічних ресурсів, зменшення витрат на перемикання текстур і спрощення інтеграції тіней у lighting pass. Додатково така схема забезпечує можливість побудови сирої карти видимості єдиним проходом, у межах якого для кожного текселя атласу визначається відповідне джерело світла через службову карту відповідності. Це дозволяє організувати більш компактне та кероване представлення тіньових даних.

Використання атласу також добре узгоджується з подальшим отриманням тіньового фактору під час lighting pass. Оскільки тіньовий фактор зберігається в межах локальних областей різного розміру, під час освітлення система може для кожного джерела виконати перехід від екранних координат фрагмента до відповідної області атласу, після чого застосувати потрібний режим апскейлу або фільтрації. Тобто фільтрація не є окремим глобальним проходом над усім атласом, а виконується безпосередньо під час застосування тіні до внеску конкретного джерела світла. У результаті спільний атлас тіней виступає центральним проміжним представленням первинної видимості, яке використовується на етапі clustered deferred lighting.

2.1.3 Використання кластерного освітлення

Навіть за наявності ефективної підсистеми побудови тіней обчислення освітлення може залишатися надто дорогим, якщо для кожного пікселя

					ІАЛЦ.467800.004 ПЗ	Арк.
						32
Змін.	Арк.	№ докум.	Підпис	Дата		

перевіряти внесок усіх джерел світла сцени. Для розв'язання цієї проблеми в роботі використовується кластерне освітлення. Його суть полягає в тому, що видимий об'єм камери поділяється не лише за координатами екрана, а й за глибиною, утворюючи тривимірну сітку кластерів [12]. Кожний кластер відповідає певній області простору, у якій можуть знаходитися фрагменти сцени та діяти лише частина джерел світла.

Принцип роботи кластерного освітлення наведено на рисунку 2.2.

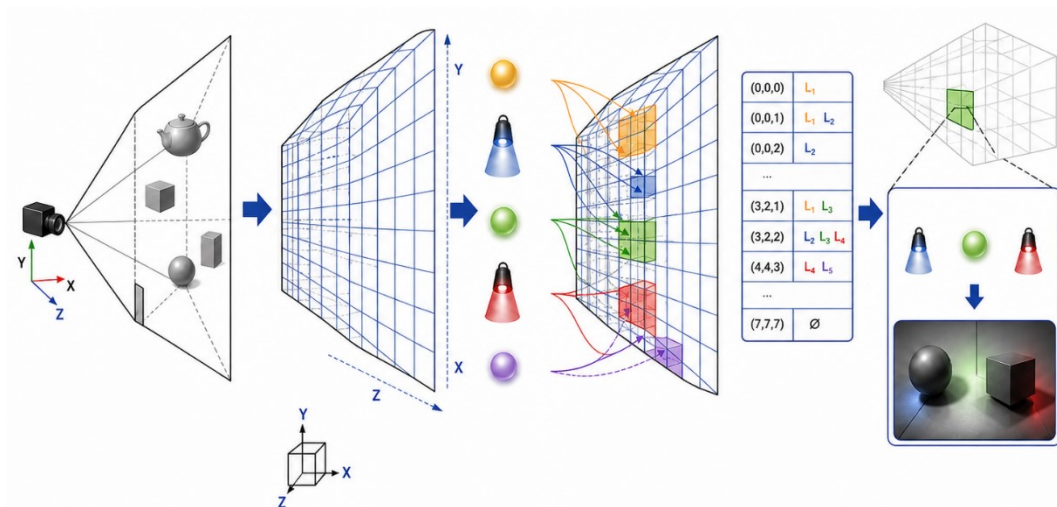


Рисунок 2.2 – Принцип роботи кластерного освітлення

Спочатку формується фрустум камери, який визначає видиму частину сцени. Далі цей об'єм поділяється на комірки за координатами екрана та глибиною. Для кожного кластера виконується перевірка перетину його просторової області із зонами впливу джерел світла. Якщо джерело світла може впливати на фрагменти всередині певного кластера, його індекс додається до списку джерел цього кластера. У результаті замість одного глобального списку всіх освітлювачів формується набір локальних списків, кожен із яких містить лише релевантні для відповідної області джерела.

Під час lighting pass для пікселя визначається кластер, до якого він належить. Після цього система перебирає не всі джерела світла сцени, а лише список джерел, прив'язаний до цього кластера. Це суттєво зменшує кількість

перевірок на піксель [12]. Такий підхід добре узгоджується з deferred shading, оскільки після формування GBuffer освітлення обчислюється окремим проходом і може використовувати готові дані.

У контексті даної роботи кластерне освітлення має додаткове значення, оскільки тіньовий фактор обчислюється окремо для конкретного джерела світла. Зменшення кількості джерел, які перевіряються для кожного пікселя, автоматично зменшує і кількість звернень до атласу тіней. Для кожного релевантного джерела під час lighting pass виконується вибірка з відповідної області атласу, апскейл і фільтрація тіньового фактора, тоді як джерела поза списком поточного кластера взагалі не розглядаються. Це особливо важливо для локальних точкових і прожекторних джерел з обмеженою областю впливу, характерних для складних інтерактивних сцен. Отже, кластерне освітлення в розробленій системі є важливою частиною масштабованої архітектури атласу рей-трейсингових тіней.

2.1.4 Використання багатопотокової обробки в рушії

Побудова системи тіней для великої кількості джерел світла включає значний обсяг підготовчих операцій на стороні центрального процесора: відбір активних освітлювачів, оцінювання їхньої важливості, формування метаданих атласу, підготовку списків для кластерного освітлення, оновлення службових буферів та запис графічних команд. Якщо всі ці дії виконувати послідовно в одному потоці, витрати процесорного часу можуть стати помітним обмеженням загальної продуктивності. Саме тому в архітектурі рушія доцільно використовувати багатопотокову обробку, яка дозволяє розподілити підготовчі етапи між кількома потоками виконання.

Паралельна підготовка даних кадру дає змогу частково винести з критичного шляху обчислення ті операції, які не потребують негайної взаємодії з графічним процесором, зокрема сортування джерел світла, побудову

					ІАЛЦ.467800.004 ПЗ	Арк.
						34
Змін.	Арк.	№ докум.	Підпис	Дата		

службових структур відповідності та підготовку оновлень для ресурсів атласу. Це дозволяє краще використовувати доступні ресурси центрального процесора та зменшувати затримки в межах одного кадру.

2.2 Обґрунтування вибору технологій розробки

Реалізація системи атласу рей-трейсингових тіней вимагає використання технологій, які дозволяють безпосередньо керувати графічними ресурсами, програмувати різні типи шейдерних проходів, будувати структури прискорення трасування променів і інтегрувати всі етапи в межах власного графічного рушія. Саме тому в роботі було обрано мову програмування C++, мову шейдерів HLSL, графічний API Direct3D 12 та технологію DirectX Raytracing.

2.2.1 Вибір мови програмування C++ для реалізації рушія

Мова C++ є доцільним вибором для реалізації низькорівневих підсистем рендерингу завдяки високій продуктивності, можливості безпосередньої роботи з системними і графічними API та підтримці об'єктно-орієнтованої та узагальненої парадигм програмування. Для задачі, що розв'язується в даній роботі, важливими є контроль над пам'яттю, ефективне керування життєвим циклом ресурсів та підтримка багатопотокового виконання. Усі ці властивості роблять C++ найкращим вибором для розроблення системи, в якій поєднуються ресурсоємні графічні алгоритми та складна архітектура рушія.

2.2.2 Використання HLSL для реалізації шейдерних модулів

Оскільки значна частина логіки рендерингу та обробки тіней виконується безпосередньо на GPU, необхідним є використання спеціалізованої мови шейдерного програмування. У роботі для цього застосовується HLSL, яка є

стандартною мовою для Direct3D та забезпечує реалізацію шейдерних модулів у межах єдиного технологічного стеку [13], [14]. Використання HLSL дозволяє програмувати всі основні GPU-етапи системи: формування даних GBuffer, ray generation для побудови первинної карти видимості, вибірку та фільтрацію тіньового фактору під час lighting pass, а також обчислення фінального освітлення. Це забезпечує цілісність архітектури та спрощує інтеграцію окремих компонентів у єдину систему.

2.2.3 Використання Direct3D 12

Вибір Direct3D 12 зумовлений потребою в низькорівневому керуванні графічним конвеєром і ресурсами GPU [13]. На відміну від більш високорівневих API, Direct3D 12 надає широкі можливості для керування станами ресурсів, дескрипторними купами, командними списками, синхронізацією між CPU та GPU та ефективного запису команд у багатопотоковому режимі. Для задачі побудови атласу рей-трейсингових тіней це особливо важливо, оскільки система оперує великою кількістю текстурних ресурсів, службових буферів, структур прискорення та різних типів шейдерних проходів, які повинні бути узгоджені в межах одного кадру.

2.2.4 Використання DirectX Raytracing (DXR)

Технологія DXR є ключовою складовою реалізації запропонованого методу, бо саме вона забезпечує можливість побудови тіней на основі безпосереднього трасування променів у геометрії сцени. Вибір DXR обумовлений його інтеграцією з Direct3D 12, підтримкою апаратно прискорених BLAS та TLAS-структур, можливістю реалізації шейдерів генерації променів, промаху та найближчого влучання, а також сумісністю з сучасними GPU. Для даної роботи DXR є інструментом побудови карти видимості, на основі якої надалі

					ІАЛЦ.467800.004 ПЗ	Арк.
						36
Змін.	Арк.	№ докум.	Підпис	Дата		

формується остаточний тінювий фактор. Це робить використання DirectX Raytracing технологічно обґрунтованим і безпосередньо пов'язаним із предметом дослідження.

2.3 Опис використовуваних структур даних та графічних ресурсів

Ефективність підсистеми атласу рей-трейсингових тіней значною мірою залежить від правильної організації графічних ресурсів і службових структур даних. У розробленій системі одночасно використовуються GBuffer, параметри джерел світла, метадані атласу, карта відповідності texel-to-light, спільний атлас тіней, кластерні буфери та структура прискорення трасування променів. Кожен із цих ресурсів виконує окрему функцію в загальному графічному конвеєрі та забезпечує зв'язок між етапами растеризації, трасування променів і фінального освітлення.

GBuffer є набором текстурних ресурсів містить основні параметри поверхонь сцени. До них належать глибина, нормалі, альбеда, матеріальні характеристики та інші допоміжні атрибути, необхідні для подальшого освітлення. Для підсистеми тіней особливо важливими є depth- та normal-компоненти GBuffer, оскільки вони дозволяють використовувати вже сформовану інформацію про поверхню без повторного рендерингу геометрії з позиції кожного джерела світла.

Окрему групу становлять структури, що описують джерела світла та їхні метадані в атласі. Параметри джерела світла містять його тип, положення або напрям, колір, інтенсивність, радіус впливу та інші характеристики, необхідні для обчислення освітлення. Метадані атласу задають зв'язок між конкретним джерелом світла і його областю в атласі: координати тайла, його розмір та параметри, потрібні для переходу між екранним простором і простором тінювого атласу. Завдяки такому розділенню система може незалежно працювати з фізичними характеристиками джерела та з його тінювим

представленням.

Центральним текстурним ресурсом підсистеми є спільний атлас тіней. У ньому для активних джерел світла виділяються окремі прямокутні області, у яких зберігаються значення видимості. Така організація дозволяє працювати з багатьма джерелами в межах одного графічного ресурсу та зменшує потребу у створенні окремих карт тіней для кожного освітлювача. Для зв'язку між текселями атласу та джерелами світла використовується карта відповідності texel-to-light. Вона визначає, якому джерелу належить тексель, і дає змогу використовувати атлас як єдиний керований простір тіньових даних [1].

Для виконання трасування променів використовується верхньорівнева структура прискорення TLAS, яка забезпечує ефективний доступ до геометрії сцени під час перевірки видимості джерела світла. Наявність такої структури є необхідною умовою практичного застосування рей трейсингу у реальному часі. У поєднанні з атласом тіней TLAS дозволяє формувати значення видимості без побудови окремого геометричного представлення для кожного джерела.

					ІАЛЦ.467800.004 ПЗ	Арк.
						38
Змін.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ ДО РОЗДІЛУ 2

У другому розділі було обґрунтовано вибір архітектури графічної підсистеми, технологій розробки та основних структур даних, необхідних для реалізації атласу рей-трейсингових тіней для динамічних джерел світла. Показано, що доцільною основою системи є гібридний підхід, який поєднує deferred shading, кластерне освітлення, трасування променів і адаптивне представлення тіньових даних у спільному атласі. Така архітектура дозволяє ефективно використовувати дані GBuffer, зменшувати кількість перевірок джерел світла під час lighting pass і застосовувати тіньовий фактор лише для релевантних джерел освітлення.

Також у розділі розглянуто роль спільного атласу тіней, карти відповідності texel-to-light, метаданих атласу та кластерних структур даних. Обґрунтовано, що використання цих ресурсів дає змогу організувати компактне представлення тіньових даних для багатьох джерел світла та забезпечити їх подальше застосування в межах кластерного відкладеного освітлення. Окремо визначено роль TLAS як структури прискорення, необхідної для виконання трасування променів у реальному часі.

Для програмної реалізації системи обґрунтовано використання C++, HLSL, Direct3D 12 та DirectX Raytracing. C++ забезпечує ефективну реалізацію рушія та керування ресурсами, HLSL використовується для програмування GPU-проходів, Direct3D 12 надає низькорівневий контроль над графічним конвеєром, а DXR забезпечує апаратно прискорене трасування променів для побудови карти видимості. Таким чином, розглянуті архітектурні та технологічні рішення формують основу для подальшого опису програмної реалізації розробленої системи.

3. ОСОБЛИВОСТІ СТРУКТУРНОЇ ОРГАНІЗАЦІЇ РОЗРОБЛЕНОЇ СИСТЕМИ

3.1 Загальна організація реалізації системи.

Розроблена система призначена для формування тіней від великої кількості динамічних джерел світла у складі гібридного графічного конвеєра реального часу. На відміну від класичних підходів, у яких для кожного джерела світла може створюватися окрема карта тіней, у даній роботі використовується спільний атлас тіней. У межах цього атласу кожному активному джерелу світла виділяється окрема область, у якій зберігається первинне значення видимості, отримане шляхом трасування променів.

Загальна ідея реалізації полягає в поєднанні растеризаційного рендерингу, відкладеного освітлення, кластерного відбору джерел світла та DirectX Raytracing. Растеризаційні проходи використовуються для формування GBuffer та інших службових ресурсів кадру, тоді як трасування променів застосовується лише для перевірки видимості джерел світла. Такий підхід дозволяє не виконувати повноекранне трасування для кожного джерела, а обмежити ray tracing-прохід побудовою компактного атласного представлення тіньових даних.

У межах одного кадру система спочатку оновлює стан сцени, камери, анімацій, джерел світла та службових структур рендерингу. Після цього виконується формування GBuffer, що містить дані, необхідні для подальшого відновлення параметрів видимих поверхонь. Окремо на стороні CPU формується інформація про активні джерела світла, їхні метадані в атласі, розміри тайлів та карта відповідності texel-to-light. Ця карта визначає, якому джерелу світла належить кожний тексель атласу, і надалі використовується в DXR RayGen як службовий вхідний ресурс.

Після підготовки метаданих і службових ресурсів запускається DXR-прохід генерації атласу тіней. Для кожного текселя активної області атласу визначається відповідне джерело світла, відновлюється точка поверхні за даними GBuffer,

					ІАЛЦ.467800.004 ПЗ	Арк.
						40
Змін.	Арк.	№ докум.	Підпис	Дата		

формується тіньовий промінь і виконується перевірка його перетину з геометрією сцени через TLAS. Результатом цього етапу є запис первинного значення видимості в атлас тіней: значення, близьке до одиниці, відповідає видимому джерелу світла, а значення, близьке до нуля, — перекритому.

Застосування сформованого атласу відбувається не окремим постпроцесом, а безпосередньо під час lighting pass. Для пікселя визначається кластер, до якого він належить, після чого система обробляє лише список релевантних джерел світла цього кластера. Для кожного такого джерела виконується звернення до відповідної області атласу, апскейл і фільтрація тіньового фактора. Отримане значення використовується як коефіцієнт видимості світла і множиться на внесок конкретного джерела у фінальне освітлення сцени.

Важливою особливістю реалізації є те, що фільтрація тіней не виконується як окремий глобальний прохід над усім атласом. Натомість вибірка, апскейл і фільтрація тіньового фактора здійснюються в момент застосування тіні до конкретного джерела світла у clustered deferred lighting. Це дозволяє уникнути зайвої обробки тих областей атласу або тих джерел, які не беруть участі в освітленні поточного фрагмента.

Організація виконання кадру в рушії побудована як набір взаємопов'язаних GPU-проходів і CPU-задач із явними залежностями між ними. У програмній реалізації це проявляється через послідовне створення задач для оновлення камери, джерел світла, анімації, побудови BLAS/TLAS, формування кластерів, рендерингу GBuffer, генерації атласу тіней, виконання lighting pass, TAA, bloom, postprocess і презентації кадру.

Після обчислення освітлення система може виконувати додаткові проходи фінальної обробки зображення. Зокрема, у рушії передбачено темпоральне згладжування кадру, яке використовує дані поточного та попереднього кадрів для зменшення субпіксельного мерехтіння, шуму та нестабільності зображення. Для системи атласу рей-трейсингових тіней це важливо, оскільки тіньовий фактор формується в атласі зі адаптивною роздільною здатністю, а його

подальше застосування може супроводжуватися просторовими та часовими артефактами.

3.2 Формування атласу тіней для динамічних джерел світла

Формування атласу тіней є одним із центральних етапів розробленої системи, оскільки саме на цьому рівні визначається, які джерела світла отримують тіньове представлення, яку роздільну здатність має кожна область атласу та яким чином окремі тайли зв'язуються з відповідними джерелами. На відміну від класичного підходу з окремою картою тіней для кожного джерела, у розробленій системі використовується спільний текстурний ресурс — атлас тіней. У ньому для активних джерел світла виділяються прямокутні області різного розміру, що дозволяє компактно зберігати первинні значення видимості та надалі використовувати їх під час lighting pass.

Загальна послідовність формування атласу складається з декількох етапів. Спочатку система формує набір метаданих для активних джерел світла, після чого для кожного джерела визначається розмір тайла. Далі ці тайли розміщуються у спільному атласі, а для кожного текселя службової карти texel-to-light записується ідентифікатор джерела, якому належить відповідна область. У результаті DXR RayGen може обробляти весь атлас як єдиний ресурс, визначаючи джерело світла без запуску окремого ray tracing-проходу для кожного освітлювача.

3.2.1 Формування метаданих активних джерел світла

На першому етапі система отримує дані про джерела світла, що беруть участь у поточному кадрі, і формує для них службові метадані атласу. До таких метаданих належать індекс джерела світла, координати його області в атласі, ширина й висота тайла, а також екранна область, з якою пов'язується відповідне

					ІАЛЦ.467800.004 ПЗ	Арк.
						42
Змін.	Арк.	№ докум.	Підпис	Дата		

тіньове представлення. Ці дані зберігаються у структурі LightAtlasMeta і надалі використовуються як під час генерації атласу, так і під час застосування тіньового фактора в lighting pass.

У програмній реалізації формування метаданих виконується на стороні CPU. Система аналізує поточний набір джерел світла, визначає для них бажані розміри тайлів і передає ці прямокутні області до пакувальника атласу. Після розміщення тайлів у спільному текстурному просторі для кожного джерела заповнюються поля AtlasOffsetX, AtlasOffsetY, AtlasWidth, AtlasHeight та інші параметри, необхідні для переходу між простором атласу, екранним простором і конкретним джерелом світла.

Важливо, що метадані оновлюються не безумовно на кожному кадрі, а відповідно до змін у сцені. Якщо змінюється набір джерел світла, їхні параметри або стан камери, система позначає службові дані як такі, що потребують оновлення. Такий підхід дозволяє уникати зайвого перерахунку атласу в ситуаціях, коли структура освітлення залишається незмінною.

Метадані виконують роль зв'язувального шару між логікою освітлення та фізичним розміщенням тіньових даних у текстурі. Джерело світла в системі описується власними фізичними параметрами — положенням, напрямом, кольором, інтенсивністю та радіусом впливу, тоді як LightAtlasMeta описує його представлення в атласі. Завдяки такому розділенню одна частина системи відповідає за обчислення освітлення, а інша — за організацію тіньових даних.

3.2.2 Адаптивний вибір розміру тайла

Після формування переліку джерел світла система визначає розмір тайла, який буде виділений кожному джерелу в атласі. Використання однакового розміру для всіх джерел є неефективним, оскільки різні освітлювачі мають різну візуальну значущість у межах поточного кадру. Джерела, розташовані ближче до камери або такі, що потенційно формують більш помітні тіні, потребують

					ІАЛЦ.467800.004 ПЗ	Арк.
						43
Змін.	Арк.	№ докум.	Підпис	Дата		

більшої роздільної здатності. Натомість для віддалених або менш помітних джерел достатньо меншого тайла.

У поточній реалізації значущість джерела для вибору розміру тайла апроксимується переважно відстанню від джерела світла до камери. Чим ближче джерело розташоване до камери, тим більший розмір тайла воно може отримати. Для цього обчислюється нормалізований коефіцієнт, залежний від відстані до камери та дальньої площини видимості. На його основі вибирається розмір між мінімальним і максимальним допустимими значеннями.

Після обчислення розмір тайла квантується до степеня двійки та обмежується заданими межами. Така квантизація спрощує подальше пакування тайлів у спільному атласі й дозволяє працювати з обмеженим набором типових розмірів, наприклад 64, 128 або 256 текселів. Це зменшує складність розміщення областей і робить структуру атласу більш передбачуваною. Дана робота показана у фрагменті коду:

```
int AtlasRTShadowSystem::ComputeShadowSize(const LightComponentData& L) {
    auto camPos = m_pOwner->GetCurrentCameraPosition();
    auto zFar = m_pOwner->GetCameraFrustum().GetZFar();

    auto dist = FD3DW::hsl::length(L.Position - camPos);
    auto lod = FD3DW::hsl::saturate(1.0f - dist / zFar);

    auto MinSize = RT_SHADOW_ATLAS_MIN_TILE_RESOLUTION_WIDTH;
    auto MaxSize = RT_SHADOW_ATLAS_MAX_TILE_RESOLUTION_WIDTH;

    auto sizeF = FD3DW::hsl::lerp((float)MinSize, (float)MaxSize, lod);
    auto size = 1 << (int)std::round(std::log2(sizeF));
    size = std::clamp(size, MinSize, MaxSize);

    return size;
}
```

3.2.3 Розміщення тайлів у спільному атласі тіней

Після визначення розмірів тайлів виконується їх розміщення в межах спільного атласу тіней. Для цього система формує набір прямокутників, кожен із яких відповідає одному джерелу світла, і передає їх до пакувальника атласу. Пакувальник визначає координати областей таким чином, щоб розмістити доступні тайли в межах фіксованого текстурного ресурсу.

Результатом цього етапу є набір прямокутних областей атласу, кожна з яких має власне положення та розмір, зображеного на рисунку 3.1.

					ІАЛЦ.467800.004 ПЗ	Арк.
						44
Змін.	Арк.	№ докум.	Підпис	Дата		

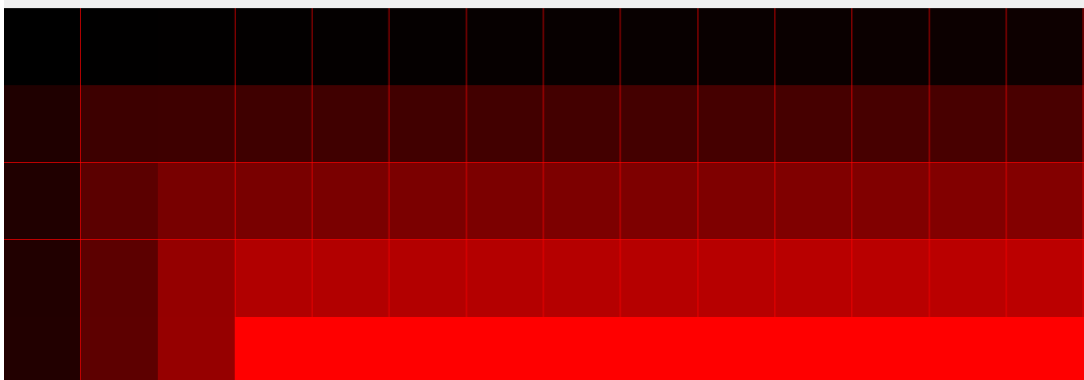


Рисунок 3.1 – Приклад розміщення тайлів у спільному атласі тіней

Для кожного джерела світла ці параметри записуються в метадані.

Особливе значення має обробка змін у структурі атласу. Якщо джерело світла зникає, змінюється його розмір або змінюється розміщення тайла, відповідні області позначаються як змінені. Такі області додаються до списку DirtyRegions, який надалі використовується для часткового оновлення службової карти texel-to-light. Завдяки цьому система не повинна щоразу повністю перезаписувати всю карту відповідності, а може оновлювати лише ті частини атласу які змінилися.

Такий підхід є важливим для продуктивності, оскільки розмір атласу може бути значним, а повне оновлення всієї службової текстури на кожному кадрі призводило б до зайвих передач даних між CPU та GPU. Використання списку змінених областей дозволяє зменшити обсяг оновлень і краще узгодити роботу CPU-підготовки з GPU-проходами.

3.2.4 Побудова карти відповідності texel-to-light

Для того щоб DXR міг обробляти весь атлас тіней одним проходом, необхідно визначити, якому джерелу світла відповідає кожний тексель атласу. Для цього в системі використовується службова карта відповідності texel-to-light. Вона є текстурним ресурсом, у якому для кожного текселя зберігається індекс джерела світла або спеціальне службове значення, що позначає неактивну

область.

Карта texel-to-light формується до запуску DXR і використовується ним як вхідний ресурс. Під час обробки конкретного текселя ray generation шейдер спочатку зчитує індекс джерела світла з цієї карти. Якщо текстель належить активному тайлу, шейдер використовує отриманий індекс для вибору відповідних метаданих джерела та побудови тіньового променя. Якщо ж текстель не належить жодному активному тайлу, для нього записується нейтральне значення видимості, і подальше трасування не виконується.

На CPU карта відповідності представлена масивом `m_vTexelMap`. Під час оновлення атласу система проходить по доданих або видалених прямокутних областях і записує в цей масив або індекс відповідного джерела, або службове значення неактивної області. Після цього змінені прямокутні ділянки завантажуються в GPU-ресурс через часткове оновлення текстури. Така схема дозволяє підтримувати актуальність карти відповідності без повного перезапису всього атласу. Оновлення карти відповідності texel-to-light для доданого тайла можна побачити у фрагменті коду:

```
for (int yy = 0; yy < r.H; ++yy) {
    int ay = r.Y + yy;
    for (int xx = 0; xx < r.W; ++xx) {
        int ax = r.X + xx;
        auto idx = ay * RT_SHADOW_ATLAS_MAX_WIDTH + ax;
        m_vTexelMap[idx] = isVisible
            ? (RT_SHADOW_ATLAS_LIGHT_IDX_FORMAT_CPU)r.ID
            : std::numeric_limits<RT_SHADOW_ATLAS_LIGHT_IDX_FORMAT_CPU>::max();
    }
}

m_DirtyRegions.push_back(r);
```

У цьому фрагменті кожному текселю доданого тайла призначається ідентифікатор джерела світла. Якщо область не повинна брати участь у побудові тіні, у карту записується максимальне службове значення, яке надалі інтерпретується як неактивна область. Додавання прямокутника до `DirtyRegions` забезпечує подальше часткове оновлення відповідної області GPU-текстури.

					ІАЛЦ.467800.004 ПЗ	Арк.
						46
Змін.	Арк.	№ докум.	Підпис	Дата		

Фрагмент програмного коду:

```
D3D12_BOX region = {};  
region.left = r.X;  
region.top = r.Y;  
region.right = r.X + r.W;  
region.bottom = r.Y + r.H;  
region.front = 0;  
region.back = 1;  
  
m_pTexelToLight->UploadTextureRegion(  
    device,  
    list,  
    m_vTexelMap.data(),  
    0,  
    region,  
    D3D12_RESOURCE_STATE_NON_PIXEL_SHADER_RESOURCE  
);
```

демонструє, що в GPU-ресурс передається не вся карта відповідності, а лише прямокутна область, яка змінилася після перепакуння атласу. Це зменшує обсяг передачі даних і дозволяє ефективніше підтримувати актуальний стан службових ресурсів.

Під час виконання DXR карта texel-to-light використовується для швидкого визначення активності поточного текселя. Якщо значення в карті відповідає неактивній області, шейдер записує нейтральне значення видимості та завершує обробку без трасування променя. Дана поведінка описана у фрагменті програмного коду:

```
uint texelLookup = TexelToLight.Load(int3(pixel, 0));  
if (texelLookup >= 0xFFu) {  
    ShadowAtlasUAV[pixel] = 1.0f;  
    return;  
}
```

3.3 Генерація первинної карти видимості методом трасування променів

Після формування метаданих атласу та карти відповідності texel-to-light система переходить до етапу генерації первинної карти видимості. На цьому етапі використовується DXR RayGen, який обробляє текселі атласу тіней і для кожного активного текселя визначає, чи є відповідне джерело світла видимим з точки поверхні. На відміну від класичних shadow map-підходів, система не рендерить сцену з позиції кожного джерела світла. Замість цього вона використовує вже сформовані дані GBuffer і виконує трасування тіньового

					ІАЛЦ.467800.004 ПЗ	Арк.
						47
Змін.	Арк.	№ докум.	Підпис	Дата		

променя в геометрії сцени.

Вхідними даними для ray tracing-проходу є структура прискорення TLAS, буфер джерел світла, буфер метаданих LightAtlasMeta, карта texel-to-light, depth-компонент GBuffer, normal-компонент GBuffer та текстура атласу тіней. Спочатку RayGen отримує координати поточного текселя атласу, перевіряє його належність активному тайлу, визначає відповідне джерело світла, після чого відновлює координати точки поверхні в просторі сцени. Результатом проходу є запис значення видимості в атлас: 1.0 відповідає відсутності перешкод між точкою поверхні та джерелом світла, а 0.0 — наявності перекриття.

3.3.1 Відновлення параметрів точки поверхні

Оскільки атлас тіней формується не в екранному просторі, а в просторі власних тайлів, перед трасуванням необхідно встановити відповідність між текселем атласу та точкою поверхні на екрані. Для цього RayGen спочатку зчитує з карти texel-to-light індекс джерела світла, після чого отримує метадані відповідного тайла. На основі зміщення тайла в атласі та його розміру обчислюються локальні координати текселя в межах області джерела.

Локальні координати тайла перетворюються в екранні координати за допомогою полів ScreenMinU, ScreenMinV, ScreenMaxU та ScreenMaxV, які зберігаються в LightAtlasMeta. Далі система визначає відповідний піксель GBuffer, зчитує з нього глибину та нормаль і відновлює світову позицію точки поверхні за допомогою оберненої матриці view-projection. Такий підхід дозволяє використовувати результати растеризаційного проходу як основу для трасування променів і не виконувати повторний рендеринг геометрії для кожного джерела світла.

```

uint2 localPixel = uint2(meta.AtlasOffsetX, meta.AtlasOffsetY);
uint2 dtPixel = pixel - localPixel;
float2 uvLocal = (float2(dtPixel) + 0.5f) /
float2(meta.AtlasWidth, meta.AtlasHeight);

float screenU = lerp(meta.ScreenMinU, meta.ScreenMaxU, uvLocal.x);
float screenV = lerp(meta.ScreenMinV, meta.ScreenMaxV, uvLocal.y);

float2 screenPos = float2(screenU * AtlasData.ScreenWidth,
screenV * AtlasData.ScreenHeight);

uint2 gBuffersPixel = uint2(floor(screenPos));
gBuffersPixel.x = min(gBuffersPixel.x, AtlasData.ScreenWidth - 1);
gBuffersPixel.y = min(gBuffersPixel.y, AtlasData.ScreenHeight - 1);

float2 uv = (float2(gBuffersPixel) + 0.5f) /
float2(AtlasData.ScreenWidth, AtlasData.ScreenHeight);

float3 worldPos = ReconstructWorldPosition(
    uv,
    DepthBuffer.Load(int3(gBuffersPixel, 0)).x,
    AtlasData.InverseViewProjectionMatrix
);

float3 normal = normalize(GBufferNormal.Load(int3(gBuffersPixel, 0)).xyz);

```

У наведеному фрагменті видно, що тексель атласу спочатку переводиться в локальні координати тайла, потім — в екранний простір, а вже після цього використовується для доступу до GBuffer. Це забезпечує зв'язок між атласним представленням тіней і реальними поверхнями, видимими у поточному кадрі.

3.3.2 Використання normal-dependent bias

Під час трасування тіньового променя однією з основних проблем є самоперетин променя з поверхнею, від якої він починається. Такий ефект може спричинити помилкове самозатінення – коли точка поверхні вважається перекритою власною геометрією. Для зменшення цього артефакту в системі використовується зміщення початку променя вздовж нормалі поверхні. У розробленій реалізації величина зміщення залежить від кута між нормаллю поверхні та напрямом на джерело світла. Якщо промінь іде майже вздовж поверхні, ризик самоперетину зростає, тому bias збільшується. Якщо ж напрям променя ближчий до нормалі, достатньо меншого зміщення. Такий підхід є компромісом між зменшенням self-shadow артефактів і збереженням геометричної точності тіні. Обчислення normal-dependent bias наведений у

фрагменті:

```
float NdotL = dot(normal, dir);  
  
float bias = clamp(  
    0.0002f * sqrt(1.0f - NdotL * NdotL) / max(NdotL, 1e-5f),  
    0.002f,  
    0.05f  
);  
ray.Origin = origin + normal * bias;  
ray.Direction = dir;  
ray.TMin = 0.0f;  
ray.TMax = dist - bias;
```

3.3.3 Формування тіньових променів

Після відновлення світової позиції точки поверхні та її нормалі система формує тіньовий промінь відповідно до типу джерела світла. Для напрямленого джерела світла використовується фіксований напрям, протилежний напрямку світла, а максимальна довжина променя задається великим значенням. Це відповідає моделі джерела, розташованого умовно нескінченно далеко від сцени.

Для точкового джерела світла напрям променя обчислюється як нормалізований вектор від точки поверхні до позиції джерела. Довжина променя в цьому випадку дорівнює відстані до джерела світла. Додатково враховується радіус впливу джерела: якщо точка розташована за межами цього радіуса, джерело не повинно впливати на неї, тому трасування можна не виконувати.

Для прожекторного джерела використовується такий самий алгоритм трасування, але додатково перевіряється належність точки поверхні до світлового конуса. Якщо точка знаходиться поза зовнішнім кутом прожектора або за межами радіуса впливу, значення видимості встановлюється без виконання трасування.

3.3.4 Трасування променя в TLAS з раннім завершенням

Після формування тіньового променя система виконує його трасування у верхньорівневій структурі прискорення TLAS. Ця структура містить посилання на геометрію сцени та дозволяє апаратно прискорити пошук перетину променя з

					ІАЛЦ.467800.004 ПЗ	Арк.
						50
Змін.	Арк.	№ докум.	Підпис	Дата		

об'єктами. Для задачі побудови тіней не потрібно знаходити всі перетини. Достатньо лише встановити факт наявності будь-якого оклюдера між точкою поверхні та джерелом світла.

Саме тому в системі використовується режим раннього завершення трасування після першого знайденого перетину. У виклику `TraceRay()` застосовується прапор `RAY_FLAG_ACCEPT_FIRST_HIT_AND_END_SEARCH`, який дозволяє завершити пошук одразу після виявлення першої геометрії на шляху променя:

```
ShadowPayload payload;
payload.Visible = 1.0f;

TraceRay(
    SceneBVH,
    RAY_FLAG_CULL_BACK_FACING_TRIANGLES |
    RAY_FLAG_ACCEPT_FIRST_HIT_AND_END_SEARCH,
    0xFF,
    0,
    1,
    0,
    ray,
    payload
);

return payload.Visible;
```

Це зменшує кількість обчислень і є важливою оптимізацією для роботи в реальному часі.

Початкове значення `payload.Visible` дорівнює 1, тобто промінь вважається неперекритим. Якщо під час трасування знаходиться перетин з геометрією, викликається `closest hit shader`, який змінює це значення на 0. Якщо перетину не знайдено, виконується `miss shader`, і значення видимості залишається рівним одиниці. Це можна побачити на фрагментах шейдерного коду:

```
[shader("closesthit")]
void ClosestHit(inout ShadowPayload payload, Attributes attr)
{
    payload.Visible = 0.0f;
}

[shader("miss")]
void Miss(inout ShadowPayload payload)
{
    payload.Visible = 1.0f;
}
```

Отже, логіка трасування зведена до визначення бінарної видимості джерела світла. Така постановка задачі добре відповідає потребам тіньового проходу, бо на цьому етапі система повинна встановити лише наявність або відсутність

					ІАЛЦ.467800.004 ПЗ	Арк.
						51
Змін.	Арк.	№ докум.	Підпис	Дата		

перекриття.

3.3.5 Запис первинного значення видимості в атлас

Фінальним кроком рей-трейсингового проходу є запис обчисленого значення видимості у відповідний тексель атласу тіней. На даному етапі формується саме первинна карта видимості, приклад якої зображений на рисунку 3.2.



Рисунок 3.2 – Результат запису обчисленого значення видимості у атлас тіней для 1 джерела світла

3.4 Обробка різних типів джерел світла

Розроблена система атласу рей-трейсингових тіней орієнтована на роботу з різними типами динамічних джерел світла. У межах єдиного атласу можуть зберігатися тіньові дані для напрямлених, точкових і прожекторних джерел, але інтерпретація цих даних під час освітлення залежить від фізичної моделі конкретного типу джерела. Тому система використовує спільний механізм збереження первинної видимості в атласі, але різні функції обчислення світлового внеску в lighting pass.

					ІАЛЦ.467800.004 ПЗ	Арк.
						52
Змін.	Арк.	№ докум.	Підпис	Дата		

Для напрямленого джерела світла тіньовий фактор використовується як коефіцієнт видимості глобального освітлення з фіксованим напрямом. Оскільки таке джерело моделює світло, що надходить з умовно нескінченної відстані, його внесок не залежить від відстані до точки поверхні та не використовує просторове затухання. У lighting pass тіньовий фактор лише масштабує результат обчислення напрямленого освітлення, зберігаючи загальну модель взаємодії світла з матеріалом поверхні.

Для точкового джерела світла тіньовий фактор застосовується разом із локальною моделлю освітлення, у якій враховуються положення джерела, відстань до поверхні, радіус впливу та затухання. На відміну від напрямленого світла, внесок точкового джерела є просторово обмеженим, тому навіть за повної видимості джерела його вплив поступово зменшується з відстанню.

Для прожекторного джерела світла додатково враховується кутова область впливу. Його внесок залежить не лише від відстані до джерела, а й від положення точки відносно світлового конуса. У lighting pass використовується спад інтенсивності між внутрішнім і зовнішнім кутами конуса, після чого результат масштабується тіньовим фактором.

Окремий випадок становлять прямокутні джерела світла. У розробленому русії вони можуть використовуватися на етапі обчислення освітлення, зокрема через підхід LTC для моделювання площинного освітлення. Проте повноцінна рей-трейсингова оклюзія для прямокутних джерел у межах поточної реалізації не розглядається. Це пов'язано з тим, що для площинного джерела недостатньо одного променя до фіксованої точки: необхідно враховувати розподілену площу випромінювання та формування напівтіні. Реалістична тінь від такого джерела потребує або кількох вибірок по площі джерела, або окремої апроксимації м'якої тіні, що збільшує обчислювальну вартість.

Тому напрямлені, точкові та прожекторні джерела світла підтримуються в межах єдиного атласного представлення тіней, але застосовуються в різних моделях освітлення. Атлас зберігає для них первинні значення видимості, а

					ІАЛЦ.467800.004 ПЗ	Арк.
						53
Змін.	Арк.	№ докум.	Підпис	Дата		

lighting pass інтерпретує ці значення відповідно до типу джерела, його просторового впливу та параметрів освітлення.

3.5 Методи апскейлу та фільтрації тіньового фактора

У розробленій системі апскейл і фільтрація тіньового фактора виконуються не як окремий глобальний прохід над усім атласом, а безпосередньо під час lighting pass для конкретного джерела світла. Така організація пов'язана з тим, що атлас містить тайли різного розміру, а отже, первинне значення видимості зберігається в локальній області атласу, роздільна здатність якої є нижчою за роздільну здатність екрана. Під час освітлення система переходить від екранних координат фрагмента до відповідної області атласу, виконує вибірку з цієї області та формує кінцевий тіньовий фактор.

Вибір конкретного режиму отримання тіньового фактора визначається параметрами системи. У реалізації передбачено декілька режимів: point sampling, bilinear filtering, PCF, joint bilateral, Poisson-based та комбіновані варіанти.

3.5.1 Point Sampling та Bilateral filtering

У режимі Point Sampling система використовує одну вибірку з відповідної області атласу. Практична роль режиму у системі полягає не лише в мінімальній вартості, а й у можливості перевірити коректність попередніх етапів: розміщення тайлів, формування метаданих, побудови карти texel-to-light та відповідності між екранними координатами й областю джерела в атласі.

Bilinear filtering у системі використовується як базовий режим згладжування при вибірці з атласу. На відміну від point sampling, він враховує сусідні значення всередині тайла, що дозволяє зменшити дискретність результату при апскейлі.

3.5.2 PCF-фільтрація

PCF-фільтрація в розробленій системі реалізована як режим локального усереднення значень тіньового фактора в околі вибірки. На відміну від класичного використання PCF для тіньової карти, у даній системі фільтрація виконується над значеннями видимості, що вже записані в атлас після DXR-проходу. Тобто на цьому етапі не виконується повторне порівняння глибини, а усереднюються готові значення з локальної області тайла, що видно у фрагменті коду:

```
float sum = 0.0f;
float weightSum = 0.0f;

for (int y = -kernelRadius; y <= kernelRadius; ++y)
{
    for (int x = -kernelRadius; x <= kernelRadius; ++x)
    {
        float2 offset = float2(x, y) * texelSize * filterRadius;
        float2 sampleUV = ClampToLightTile(atlasUV + offset, meta);

        float w = ComputeSpatialWeight(x, y);
        sum += SampleShadowAtlas(sampleUV) * w;
        weightSum += w;
    }
}

float shadowFactor = sum / max(weightSum, 1e-5f);
```

Розмір ядра задається параметром системи й безпосередньо впливає на кількість звернень до атласу. Оскільки фільтрація виконується під час lighting pass, її вартість залежить від кількості джерел у поточному кластері.

3.5.3 Joint Bilateral фільтрація

Joint bilateral фільтрація використовується в системі як механізм збереження геометричних меж під час згладжування тіньового фактора. На відміну від звичайного усереднення значень атласу, цей режим додатково використовує дані про глибину та нормалі поверхонь.

У реалізації для кожної сусідньої вибірки обчислюється вага, що залежить від різниці глибини та кута між нормальми центральної точки й точки вибірки. Якщо сусідня вибірка відповідає іншій поверхні або іншому шару сцени, її внесок зменшується. Таким чином, система намагається згладжувати тінь

					ІАЛЦ.467800.004 ПЗ	Арк.
						55
Змін.	Арк.	№ докум.	Підпис	Дата		

переважно в межах однієї поверхні, не переносячи тіньовий фактор через різкі геометричні контури.

3.5.4 Poisson-based фільтрація

Poisson-based фільтрація використовується як режим, у якому вибірки беруться за нерегулярним шаблоном. У межах реалізації це дозволяє зменшити помітність повторюваних структур, які можуть виникати при використанні регулярного квадратного ядра. Для атласу рей-трейсингових тіней це важливо, оскільки тайли можуть мати невелику роздільну здатність, а регулярна фільтрація в таких умовах може створювати помітні просторові патерни.

У системі шаблон вибірок обертається за допомогою шумової компоненти. Це дозволяє декорелювати вибірки між сусідніми пікселями та зменшити регулярність артефактів. Водночас такий режим потребує обережного налаштування, оскільки надмірне використання шуму може призвести до часової нестабільності. Тому Poisson-based фільтрація розглядається як режим підвищеної якості, який має використовуватися разом із параметрами контролю шуму та, за потреби, темпоральною стабілізацією.

3.5.5 Комбіновані режими фільтрації

Комбіновані режими фільтрації поєднують просторову або стохастичну вибірку з joint bilateral зважуванням. У системі це використовується для того, щоб одночасно отримати м'якший тіньовий фактор і зменшити небажане змішування значень між різними поверхнями. До таких режимів належать поєднання Joint Bilateral з PCF та Joint Bilateral з Poisson-based фільтрацією.

У режимі Joint Bilateral з PCF локальне ядро усереднення доповнюється вагами, обчисленими за глибиною та нормаллю. У режимі Joint Bilateral з Poisson-based фільтрацією аналогічне зважування застосовується до

					ІАЛЦ.467800.004 ПЗ	Арк.
						56
Змін.	Арк.	№ докум.	Підпис	Дата		

нерегулярних вибірок. Обидва режими мають вищу вартість, оскільки для кожної вибірки потрібно не лише звернутися до атласу, а й отримати або відновити відповідні екранні координати, зчитати дані GBuffer і обчислити вагу.

3.6 Параметричне налаштування якості тіней

Розроблена система атласу рей-трейсингових тіней передбачає параметричне керування якістю зображення та обчислювальною вартістю фільтрації. Це необхідно через те, що різні сцени можуть мати різні вимоги до вигляду тіней: в одних випадках важливішою є максимальна швидкодія, в інших – м'якість меж, стабільність при русі камери або збереження геометричних контурів. Тому параметри фільтрації не задаються жорстко в коді, а передаються до шейдерів через структуру налаштувань атласу і можуть змінюватися без перебудови всієї архітектури системи.

До основних параметрів належать радіус фільтрації, розмір ядра PCF, коефіцієнти відсікання за глибиною та нормаллю, масштаб шуму, рівень затемнення та контраст тіні.

Параметр радіусу фільтрації (FilterRadius) визначає просторовий масштаб вибірок під час фільтрації. Чим більшим це значення є, тим ширшу область атласу охоплює фільтр і межа тіні стає м'якшою. Це особливо помітно для режимів Joint Bilateral та Poisson-based фільтрації, де зміщення вибірок відносно центральної точки безпосередньо залежить від радіуса. Але надмірне збільшення даного параметру може призвести до втрати локальної деталізації, розмивання дрібних тіней і появи небажаного проникнення тіні або світла через межі об'єктів.

Параметр PCFKernelSize визначає розмір ядра під час PCF фільтрації. Збільшення ядра дозволяє усереднювати більшу кількість сусідніх значень атласу, що робить перехід між освітленою та затіненою областю плавнішим. Проте вартість такої операції зростає разом із кількістю вибірок у ядрі.

					ІАЛЦ.467800.004 ПЗ	Арк.
						57
Змін.	Арк.	№ докум.	Підпис	Дата		

Параметри `DepthRejectionSharpness` і `NormalRejectionSharpness` відповідають за збереження геометричних меж під час Joint Bilateral фільтрації. Перший параметр визначає, наскільки сильно фільтр реагує на різницю глибини між центральною точкою та сусідньою вибіркою. Другий параметр керує чутливістю до відмінності нормалей поверхні. Якщо сусідня вибірка належить до іншого об'єкта або до поверхні з різко іншою орієнтацією, її вага зменшується. Водночас надмірно великі значення `DepthRejectionSharpness` або `NormalRejectionSharpness` можуть зробити фільтрацію занадто жорсткою. У такому випадку фільтр починає відкидати значну кількість сусідніх вибірок, через що тінь може залишатися шумною.

Параметр `NoiseScale` використовується для керування шумовою складовою в режимах, де застосовуються стохастичні або квазівипадкові шаблони вибірки. Зокрема, це стосується Poisson-based фільтрації та режимів з обертанням шаблону вибірок. Помірне використання шуму дозволяє зменшити регулярні просторові артефакти, які виникають при фіксованому або надто впорядкованому розташуванні вибірок. Разом із тим шумова компонента може погіршувати часову стабільність зображення. Якщо `NoiseScale` має надто велике значення, при русі камери або об'єктів можуть виникати мерехтіння та дрібні флуктуації тіньового фактора. Тому цей параметр не повинен використовуватися як основний засіб пом'якшення тіні. Його доцільно розглядати як інструмент тонкого налаштування, який допомагає зменшити регулярність артефактів, але потребує узгодження з просторовою фільтрацією та темпоральною стабілізацією кадру.

Окремо в системі передбачені параметри фінальної корекції тіньового фактора — `BlackLevel` і `ShadowContrast`. Параметр `BlackLevel` визначає нижню межу затемнення та дозволяє зробити тіні візуально глибшими або, навпаки, менш контрастними. `ShadowContrast` керує характером переходу між освітленими та затіненими областями, змінюючи розподіл значень тіньового фактора після фільтрації. Такі параметри не змінюють геометричну коректність

тіні, але впливають на її кінцеве сприйняття у сцені. Використання даних параметрів є корисним для налаштування візуального стилю, однак їх надмірне застосування може призвести до втрати деталей у напівтіні або до неприродно різкого затемнення.

3.7 Інтеграція атласу тіней у кластерне відкладене освітлення

Після генерації первинної карти видимості атлас тіней використовується безпосередньо на етапі кластерного відкладеного освітлення. Тіньовий фактор визначається в lighting pass для кожного релевантного джерела та одразу впливає на фінальний результат освітлення.

На початку lighting pass для поточного пікселя визначається кластер, до якого він належить, що наведено в фрагменті програмного коду:

```
uint tileIndex = ComputeClusterIndex(fragCoord, viewZAbs, ClustersData);  
uint lightCount = Clusters[tileIndex].Count;
```

Після цього з буфера Clusters зчитується кількість джерел світла, пов'язаних із цим кластером, та їхні індекси. Завдяки цьому система працює лише з тими джерелами світла, які впливають на відповідну область простору.

Для кожного джерела викликається функція яка використовує ідентифікатор джерела світла, екранні координати фрагмента, метадані атласу та відповідну область у текстурі атласу. У її межах виконується перехід від координат екрана до локального тайла джерела, вибірка значення видимості, апскейл і фільтрація згідно з поточними параметрами системи. Таким чином, атлас тіней інтерпретується як набір локальних областей, пов'язаних із конкретними джерелами світла.

Отриманий тіньовий фактор використовується як коефіцієнт видимості джерела. Значення, близьке до 1.0, відповідає повній видимості світла, а значення, близьке до 0.0, означає, що шлях до джерела перекритий геометрією сцени. У функціях освітлення цей коефіцієнт множиться на внесок джерела

світла. Наприклад, для точкового джерела це виконується безпосередньо перед поверненням результату освітлення:

```
float shadowFactor = GetShadowFactor(id, UV);
if (shadowFactor <= 0.0f)
{
    return float3(0.0f, 0.0f, 0.0f);
}

float3 resultLight = CalculatePBRLighting(
    N, V, L, radiance,
    albedo, metallic, roughness, F0
);

return resultLight * shadowFactor;
```

Загальна обробка освітлення виконується в циклі по джерелах поточного кластера. Для кожного джерела зчитується його структура з буфера Lights, після чого залежно від типу викликається відповідна функція освітлення. У середині цих функцій виконується звернення до атласу тіней, тому кількість таких звернень залежить від кількості джерел у поточному кластері, а не від загальної кількості джерел у сцені.

3.8 Особливості програмної реалізації

Програмна реалізація розробленої системи побудована як частина власного графічного рушія на базі Direct3D 12 та DirectX Raytracing. Основна особливість реалізації полягає в тому, що підсистема тіней не існує як ізольований модуль, який лише формує окрему текстуру результату. Вона інтегрована в загальний конвеєр рендерингу кадру.

З погляду програмної організації система складається з декількох взаємопов'язаних частин: модуля керування атласом рей-трейсингових тіней, модуля джерел світла, підсистеми кластерного освітлення, проходів формування GBuffer, DXR-проходу генерації атласу тіней, шейдера освітлення та допоміжних підсистем постобробки. Такий підхід дозволяє відокремити всі етапи генерації кадру.

					ІАЛЦ.467800.004 ПЗ	Арк.
						60
Змін.	Арк.	№ докум.	Підпис	Дата		

3.8.1 Основні програмні модулі реалізації

Центральним модулем підсистеми тіней є AtlasRTShadowSystem. Він відповідає за керування атласом рей-трейсингових тіней, службовими метаданими, картою відповідності texel-to-light, GPU-ресурсами атласу та запуском DXR-проходу. Саме цей клас виконує роль зв'язувального рівня між джерелами світла, даними камери, ресурсами GBuffer, структурою TLAS і фінальним етапом освітлення.

Підсистема джерел світла надає дані про активні освітлювачі сцени: тип джерела, положення, напрям, колір, інтенсивність, радіус впливу та інші параметри, необхідні для освітлення. Для зв'язку між джерелом світла та його областю в атласі використовується структура LightAtlasMeta. Вона містить індекс джерела, координати тайла, його розмір і параметри екранної області, у межах якої застосовується відповідне тіньове представлення.

На рівні GPU реалізація поділяється на два основні блоки. Перший блок становлять DXR-шейдери, які формують первинні значення видимості в атласі тіней. Другий блок становить шейдер освітлення. Саме в цьому шейдері виконується отримання тіньового фактора для конкретного джерела світла.

3.8.2 Організація виконання GPU-проходів у межах кадру

Виконання GPU-проходів у розробленому рушії організовано як граф задач із явними залежностями між ними. Окремі етапи кадру не запускаються як незалежні послідовні виклики, а повертають об'єкти ExecutionHandle, які надалі використовуються для синхронізації наступних проходів. Такий підхід дозволяє явно вказувати, які ресурси повинні бути підготовлені перед запуском конкретного GPU-проходу.

У межах кадру створюються задачі для оновлення камери, джерел світла, анімації, підготовки геометрії, побудови структур прискорення, формування GBuffer, генерації атласу тіней і виконання lighting pass. Частина проходів

					ІАЛЦ.467800.004 ПЗ	Арк.
						61
Змін.	Арк.	№ докум.	Підпис	Дата		

виконується умовно. Наприклад, підготовка DXR-ресурсів і генерація атласу тіней пропускаються, якщо сцена працює в неосвітленому режимі або якщо трасування променів недоступне. Це дозволяє уникати запуску непотрібних GPU-задач.

Після генерації атласу створюється задача освітлення, яка очікує готовності всіх необхідних вхідних ресурсів. До них належать дані джерел світла, кластерні структури, результат рендерингу геометрії, атлас тіней та IBL-ресурси. Після завершення lighting pass результат передається до наступних етапів кадру: відображення скайбоксу, ТАА, tonemapping і презентації.

3.8.3 Багатопотокова підготовка даних і виконання графічних команд

Оскільки система працює з великою кількістю динамічних джерел світла та складним графічним конвеєром, значна частина роботи виконується не безпосередньо в основному потоці, а через багатопотокову систему підготовки та виконання команд. До таких операцій належать підготовка метаданих атласу, оновлення службових буферів, запис командних списків, очікування залежностей і подання готових команд до GPU-черг. Послідовне виконання всіх цих задач в одному потоці призвело б до додаткових затримок у межах кадру.

Особливістю реалізації є використання підходу з “рецептами” команд. Графічна операція описується не як негайне виконання команд, а як об’єкт CommandRecipe, який містить логіку запису команд у командний список. Після створення такого рецепта він передається до RenderThreadManager, який організовує його подальшу обробку: побудову списку у worker-потоці та подальшого передавання до рендер потоку, очікування залежностей і надсилання у відповідну GPU-чергу.

Ця система дозволяє розділити відповідальність між кількома рівнями. Основний потік описує, яку роботу необхідно виконати. Робочі потоки будують командні списки на основі рецептів. Рендер потік відповідає за впорядковане

					ІАЛЦ.467800.004 ПЗ	Арк.
						62
Змін.	Арк.	№ докум.	Підпис	Дата		

подання команд у GPU-черги та враховує залежності між задачами. `AsyncCommandQueue` виконує безпосереднє подання команд на GPU й повертає об'єкт синхронізації, який надалі використовується іншими задачами як залежність. Узагальнена схема подання задач показана у фрагменті програмного коду:

```
m_xCommandsBuilder.PostTask(
    [recipe, batch, deps = std::move(dependencies), this]()
    {
        auto* pool = m_mPools[recipe->GetType()].get();
        auto* q = PickCommandQueue(recipe->GetType());

        auto list = pool->BuildFromRecipe(recipe);
        batch->Lists.push_back(std::move(list));

        m_xRenderThread.PostTask({
            [q, batch, deps]()
            {
                for (auto& d : deps) {
                    if (d && d->GetFence()) {
                        q->WaitFence(d);
                    }
                }

                q->Submit(batch);
            },
            deps
        });
    }
);
```

Описана архітектура важлива не лише для розвантаження основного потоку, а й для завчасного формування черги робіт для GPU. Основний потік у такій моделі не очікує негайного завершення кожної графічної операції, а лише описує наступні етапи кадру та задає залежності між ними. У цей час робочі потоки можуть паралельно будувати командні списки, а рендер потік — впорядковувати їхнє подання у відповідні GPU-черги. Завдяки цьому GPU отримує підготовлені пакети команд без очікувань між проходами і CPU не витрачає основний час кадру на синхронне очікування кожної проміжної операції. Повна схема взаємодії об'єктів синхронізації наведена у додатку 4.

3.8.4 Реалізація темпорального згладжування кадру

Після формування освітленого зображення в рушії може виконуватися темпоральне згладжування кадру. Цей етап не бере участі безпосередньо у генерації атласу тіней, однак є важливим для фінального сприйняття результату.

					ІАЛЦ.467800.004 ПЗ	Арк.
						63
Змін.	Арк.	№ докум.	Підпис	Дата		

Оскільки тіньовий фактор формується в атласі зі зниженою або адаптивною роздільною здатністю і під час його застосування можуть використовуватися фільтрація, jitter або стохастичні шаблони вибірки, у фінальному зображенні можуть виникати дрібні просторові та часові коливання значення. Темпоральне згладжування допомагає зменшити їхню помітність.

У програмній реалізації темпоральне згладжування оформлено як окрема підсистема TAASystem. Вона використовує два буфери історії кадру, які по чергово виконують дві ролі: один зберігає результат попереднього кадру для читання, а інший використовується для запису оновленого результату.

Під час виконання ТАА-pass для кожного пікселя зчитується поточний колір, вектор руху, глибина та колір з буферу історії. На основі вектора руху й різниці між поточним і попереднім jitter обчислюються координати попереднього кадру. Далі історичне значення змішується з поточним кольором. При цьому враховується швидкість руху та різниця глибин: якщо піксель швидко рухається або відбулася дизоклюзія, вплив історії зменшується, що дозволяє знизити помітність шлейфових артефактів.

У системі камери формується невелике субпіксельне зміщення проєкції на основі послідовності Халтона. Це зміщення додається до проєкційної матриці, внаслідок чого різні кадри мають різні позиції вибірки. Темпоральне згладжування використовує ці зміщення для накопичення додаткової інформації з кількох кадрів і згладжування високочастотних артефактів [18]. Для коректної компенсації в шейдер передаються як поточне, так і попереднє значення jitter.

3.9 Обмеження реалізованої системи та напрями подальшого розвитку

Попри реалізовану підтримку атласу рей-трейсингових тіней для динамічних джерел світла, реалізована система має низку обмежень, які впливають на якість результату, стабільність зображення та подальшу масштабованість. Ці обмеження не заперечують працездатність

					ІАЛЦ.467800.004 ПЗ	Арк.
						64
Змін.	Арк.	№ докум.	Підпис	Дата		

запропонованого підходу, однак визначають напрями його вдосконалення та формують основу для подальшого експериментального аналізу.

Одним з основних обмежень є спрощена оцінка значущості джерела світла під час вибору розміру тайла. У реалізованій системі цей параметр переважно пов'язаний із відстанню від джерела до камери. Такий підхід є простим і придатним для практичного використання, однак він не враховує всі чинники, що можуть впливати на помітність тіні: видиму площу джерела або його тіні, інтенсивність освітлення, розмір освітленої області, кут огляду та фактичну важливість джерела у фінальному кадрі. У подальшому функцію пріоритету доцільно розширити, враховуючи не лише відстань, а й екранну площу впливу та внесок джерела у зображення.

Іншим обмеженням є залежність якості тіні від розміру тайла в атласі. Якщо джерело отримує малу область, первинна карта видимості містить недостатньо деталей, і навіть подальша фільтрація не завжди може повністю усунути дискретність або втрату дрібних елементів тіні. В той самий час збільшення розмірів тайлів для великої кількості джерел швидко вичерпує площу атласу та підвищує вартість трасування. Тому система потребує компромісу між кількістю активних джерел, роздільною здатністю їхніх тайлів і вибраним режимом фільтрації.

Також важливим напрямом подальшого вдосконалення є підвищення часової стабільності тіней. Стохастичні шаблони вибірки, jitter-зміщення та обертання Poisson шаблонів допомагають зменшувати регулярні просторові артефакти, однак за невдалого вибору параметрів можуть спричиняти мерехтіння під час руху камери. Темпоральне згладжування частково зменшує цю проблему, але не замінює необхідності стабільнішої просторово-часової фільтрації самого тіньового фактора.

ВИСНОВКИ ДО РОЗДІЛУ 3

У третьому розділі було розглянуто практичну реалізацію системи атласу рей-трейсингових тіней для динамічних джерел світла. Описано загальну організацію роботи підсистеми у складі гібридного графічного конвеєра, де GBuffer використовується як джерело даних про поверхні сцени, DirectX Raytracing забезпечує побудову первинних значень видимості, а кластерне відкладене освітлення виконує застосування тіньового фактора під час обчислення внеску джерел світла.

У межах розділу описано формування спільного атласу тіней. Показано, що такий підхід дозволяє зберігати тіньові дані багатьох джерел у межах одного текстурного ресурсу та використовувати їх під час подальшого освітлення сцени.

Розглянуто генерацію первинної карти видимості методом трасування променів. Для цього система відновлює параметри точки поверхні з GBuffer, формує тіньовий промінь відповідно до типу джерела світла, застосовує normal-dependent bias для зменшення самозатінення та виконує трасування в TLAS із раннім завершенням після першого знайденого перекриття. Результатом цього етапу є значення видимості, записане у відповідний тексель атласу.

Окрему увагу приділено застосуванню тіньового фактора під час lighting pass. Було показано, що апскейл і фільтрація виконуються під час обчислення освітлення для конкретного джерела. Це дозволяє змінювати баланс між швидкодією, м'якістю меж тіні та збереженням геометричних контурів.

Описано особливості програмної реалізації: основні програмні модулі, організацію GPU-проходів у вигляді графа задач із залежностями, багатопотокову підготовку команд і використання темпорального згладжування кадру. Визначено обмеження поточної реалізації, пов'язані з пріоритезацією джерел світла, розміром тайлів, підтримкою прямокутних джерел, часовою стабільністю та пакуванням областей у спільному атласі.

4. ТЕСТУВАННЯ РОЗРОБЛЕНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ

4.1 Мета та завдання тестування системи

Метою тестування розробленої комп'ютерної системи є перевірка коректності роботи підсистеми атласу рей-трейсингових тіней, оцінювання візуальної якості сформованих тіней та визначення впливу основних параметрів системи на продуктивність рендерингу. У межах тестування перевіряється формування спільного атласу тіней і карти відповідності texel-to-light, генерація первинних значень видимості методом трасування променів, застосування тіньового фактора під час lighting pass, а також робота системи для різних типів джерел світла. Досліджуються режими апскейлу та фільтрації, вплив розміру тайлів, кількості динамічних джерел світла і параметрів якості на час кадру та стабільність зображення. Отримані результати повинні підтвердити працездатність запропонованого підходу, визначити його обмеження та сформулювати рекомендації щодо вибору оптимального режиму роботи системи.

4.2 Методика тестування

Тестування розробленої системи виконувалося на одному апаратному стенді в однакових умовах запуску для всіх експериментів. Це необхідно для коректного порівняння результатів, оскільки час генерації атласу тіней, виконання lighting pass, фільтрації, постобробки залежить від характеристик тестового стенду. Основні характеристики наведені у таблиці 4.1

Таблиця 4.1 – характеристики тестового апаратного середовища

Параметр	Значення
Процесор	Ryzen 5 5600H
Відеокарта	RTX 3060 Laptop
Об'єм оперативної пам'яті	16 ГБ

					ІАЛЦ.467800.004 ПЗ	Арк.
						67
Змін.	Арк.	№ докум.	Підпис	Дата		

Для тестування використовувалися сцени на основі моделі Sponza. Вона містить складну геометрію, велику кількість поверхонь, колон, арок і ділянок із різною глибиною. Така сцена є придатною для перевірки коректності обчислення тіней для джерел світла. Для різних груп тестів використовувалися окремі конфігурації освітлення, що відрізнялися кількістю джерел світла, їхнім типом, розташуванням, розміром тайлів атласу та режимами апскейлу і фільтрації. На рисунку 4.1 наведений приклад тестової сцени.



Рисунок 4.1 – Приклад тестової сцени.

Оцінювання результатів виконувалося за двома групами критеріїв: функціонально-візуальними та продуктивністю. До функціонально-візуальних критеріїв належать коректність формування тайлів у атласі, правильність карти texel-to-light, відповідність тіней положенню джерел світла й оклюдерів, чіткість меж тіні, наявність шуму, мерехтіння, надмірного розмивання або витоку освітлення через геометричні межі. До продуктивнісних критеріїв

належать середній час кадру, час виконання DXR-проходу, час lighting pass, вплив кількості джерел світла, розміру тайлів і режиму фільтрації на загальну швидкодію системи.

4.3 Функціональне тестування системи.

4.3.1 Перевірка формування атласу тіней і карти texel-to-light.

Перевіримо коректність формування службових ресурсів, необхідних для генерації атласу рей-трейсингових тіней. Основна увага надається правильності виділення областей у спільному атласі, відповідності цих областей активним джерелам світла, заповненню карти texel-to-light та обробці неактивних ділянок атласу.

На рисунку 4.2 наведено візуалізацію карти texel-to-light. Ця карта є службовим ресурсом, у якому кожний тексель активного тайла пов'язується з відповідним ідентифікатором джерела світла. Неактивні області позначаються службовим значенням і не повинні брати участь у трасуванні променів.

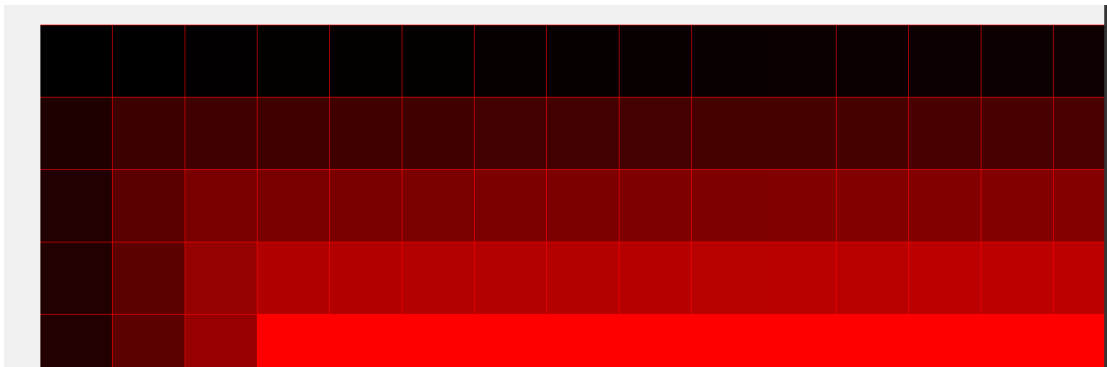


Рисунок 4.2 – Візуалізація карти відповідності texel-to-light

На рисунку 4.3 показано результат генерації атласу тіней. Видно, що активні області атласу заповнюються значеннями видимості, отриманими внаслідок трасування променів, тоді як неактивні ділянки не містять корисних тіньових даних. Наявність окремих прямокутних областей підтверджує, що тіньові дані

зберігаються не в окремих текстурах для кожного джерела, а в межах одного спільного атласу.

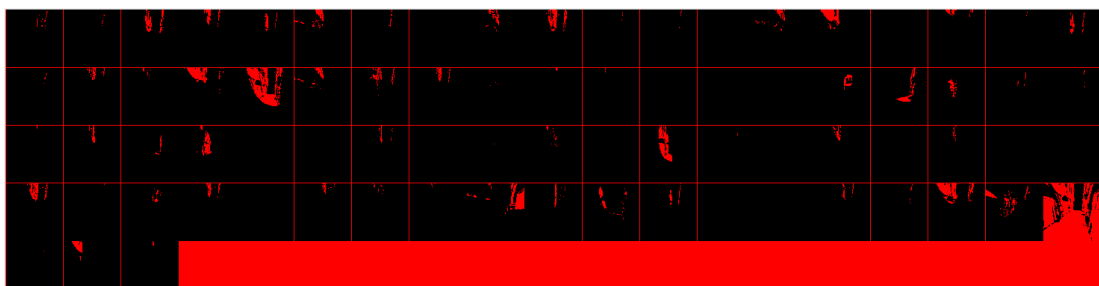


Рисунок 4.3 – Візуалізація карти відповідності texel-to-light

Додаткова перевірка відповідності між джерелами світла та областями атласу виконувалася за метаданими LightAtlasMeta, у яких для кожного активного джерела зберігаються індекс джерела, координати тайла, його ширина та висота. Отже, результати тестування показали, що атлас тіней і карта texel-to-light формуються коректно та можуть використовуватися як вхідні ресурси для подальшої обробки.

4.3.2 Перевірка генерації тіней для різних типів джерел світла

На даному етапі тестування перевіряється коректність формування тіней для різних типів джерел світла, які підтримуються розробленою системою, а саме: точкове, прожекторне та напрямлене джерела світла. Метою перевірки було встановити, що тіньовий фактор формується відповідно до типу джерела, а напрям і форма тіні узгоджуються з положенням оклюдера та параметрами освітлення.

Для тестування використовувалася спрощена сцена, що містить площину та один оклюдер. Такий підхід дозволяє наочно оцінити напрям тіні, її форму та відповідність положенню джерела світла без впливу складної геометрії сцени.

На рисунку 4.4 наведено результат для точкового джерела світла. У цьому

випадку тінь формується від локального джерела світла, тому її напрям визначається положенням джерела відносно оклюдера. Отриманий результат підтверджує, що тінь коректно проєктується на площину та змінює свою форму відповідно до просторового розташування освітлювача.

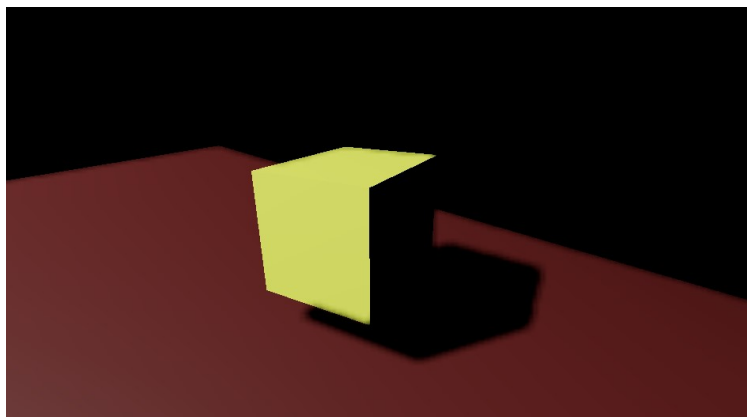


Рисунок 4.4 – Результат формування тіні для точкового джерела світла

На рисунку 4.5 показано результат для прожекторного джерела світла. Для цього типу джерела додатково враховується напрям випромінювання та обмеження конусом освітлення. Отримана тінь відповідає положенню оклюдера і формується лише в межах області, яка освітлюється прожектором, що свідчить про правильне врахування параметрів світлового конуса.

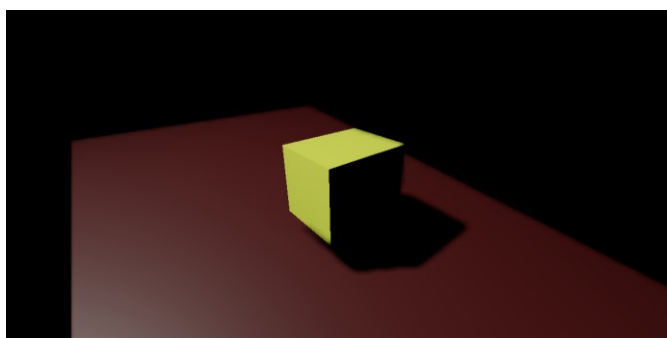


Рисунок 4.5 – Результат формування тіні для прожекторного джерела світла

На рисунку 4.6 подано результат для напрямленого джерела світла. У цьому випадку тінь формується для напрямленого джерела світла, промені якого

вважаються паралельними. Тому напрям тіні визначається лише напрямом освітлення, а не відстанню до джерела. Отриманий результат підтверджує коректне формування тіні для даного типу джерела.

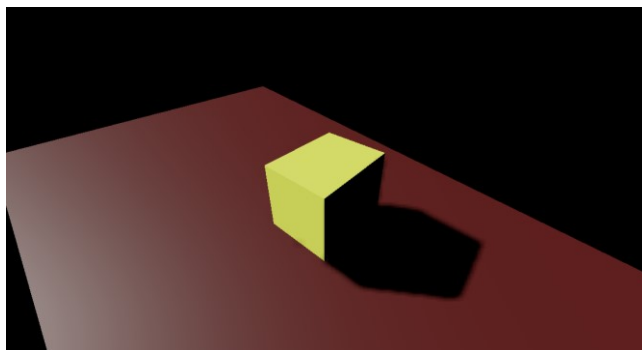


Рисунок 4.6 – Результат формування тіні для напрямленого джерела світла

4.4 Дослідження візуальної якості тіней

4.4.1 Порівняння режимів апскейлу та фільтрації

Для оцінювання візуальної якості реалізованих режимів апскейлу та фільтрації тіньового фактора було виконано порівняння результатів у межах одного й того самого фрагмента сцени за незмінних параметрів камери, положення джерела світла та геометрії. Усі режими застосовувалися до одного й того самого тіньового атласу, що дозволяє коректно порівнювати саме вплив способу вибірки та фільтрації на фінальний вигляд тіні.

На рисунку 4.7 наведено результати для режимів Point, Bilinear, PCF 3×3, Joint Bilateral, Joint Bilateral + PCF 3×3 та Poisson Based Filtering.

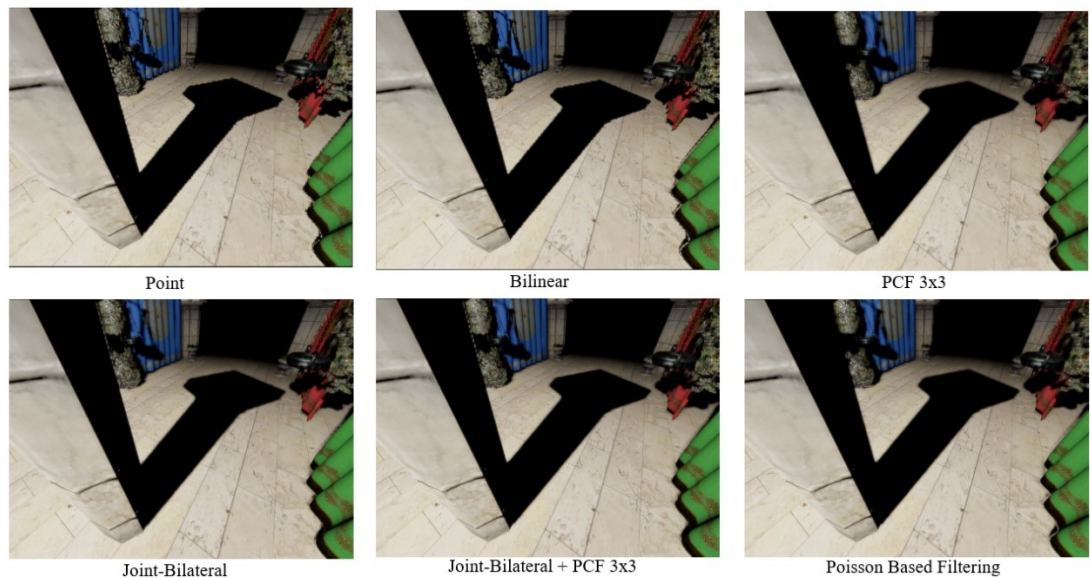


Рисунок 4.7 – Порівняння режимів фільтрації тіньового фактора

Найпростішим є режим Point, який забезпечує мінімальну обчислювальну вартість, проте формує найбільш помітну дискретність меж тіні. Bilinear filtering дещо згладжує межу, однак не враховує геометричні особливості сцени, через що може призводити до надмірного розмивання переходу. Режим PCF 3×3 забезпечує більш м'який контур тіні та зменшує вираженість сходинкоподібних артефактів, хоча при цьому частково втрачається локальна чіткість межі.

Joint Bilateral filtering демонструє кращий результат у збереженні форми тіні, оскільки під час фільтрації враховує додаткові характеристики поверхні. Поєднання Joint Bilateral + PCF 3×3 дозволяє отримати більш збалансований результат, у якому досягається пом'якшення межі без суттєвого розмивання контурів. Poisson Based Filtering також формує візуально м'якшу тінь і зменшує регулярність дискретних артефактів, однак характер його результату більше залежить від параметрів вибірки.

Проведене порівняння показує, що найнижчу якість формує Point-вибірка, тоді як найкращий візуальний баланс між плавністю межі тіні та збереженням її форми у даному фрагменті сцени забезпечують Joint Bilateral filtering і комбінований режим Joint Bilateral + PCF 3×3.

4.4.2 Аналіз впливу параметрів фільтрації та корекції тіні

Окрім вибору самого режиму апскейлу та фільтрації, на вигляд сформованої тіні впливають додаткові параметри, які керують радіусом вибірки, чутливістю до геометричних меж та фінальною корекцією тіньового фактора. У межах тестування основну увагу було приділено параметрам `FilterRadius` , `DepthRejectionSharpness` , `NormalRejectionSharpness` , `BlackLevel` та `ShadowContrast`, оскільки саме вони найбільш помітно впливають на візуальний результат.

Параметр `FilterRadius` визначає просторовий радіус вибірок під час фільтрації. Зі збільшенням цього параметра межа тіні стає м'якшою, однак надмірне значення може призводити до втрати дрібних деталей і надмірного розмивання. На рисунку 4.8 показано порівняння результатів для різних значень радіуса фільтрації в однакових умовах сцени.

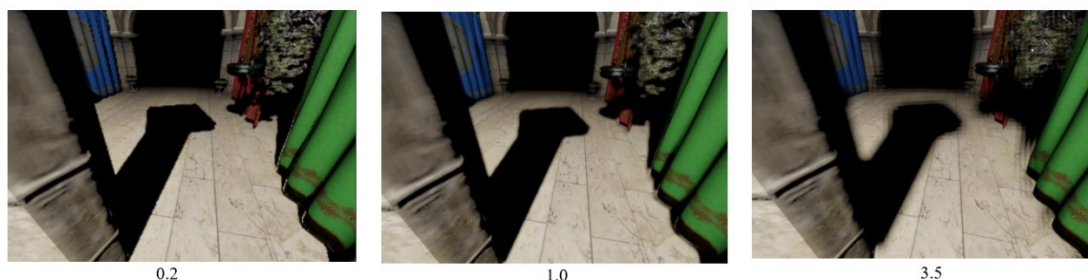


Рисунок 4.8 – Вплив радіуса фільтрації на вигляд тіні

Параметри `DepthRejectionSharpness` і `NormalRejectionSharpness` використовуються в режимах, які враховують геометричні характеристики поверхні. Вони визначають, наскільки сильно фільтрація повинна відкидати сусідні вибірки з відмінною глибиною або нормаллю. Помірні значення цих параметрів дозволяють краще зберігати контури об'єктів, тоді як надто великі значення можуть зробити фільтрацію жорсткішою та підвищити помітність локальних нерівномірностей.

Окремо було перевірено вплив параметрів фінальної корекції тіні —

BlackLevel і ShadowContrast. Перший параметр керує мінімальним рівнем затемнення, а другий впливає на виразність переходу між освітленою та затіненою областю. На рисунку 4.9 наведено приклад зміни вигляду тіні без корекції, з помірною корекцією та з надмірним підсиленням контрасту.

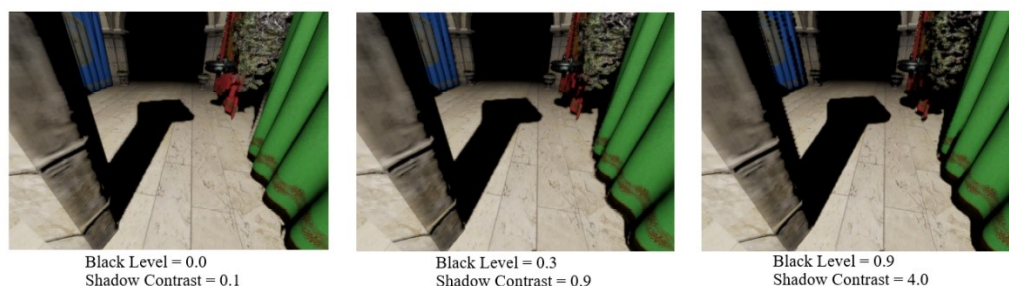


Рисунок 4.9 – Вплив параметрів BlackLevel та ShadowContrast на фінальний вигляд тіні.

Узагальнений результат впливу основних параметрів на якість тіні наведено в таблиці 4.2.

Таблиця 4.2 – Вплив параметрів на якість тіні.

Параметр	Візуальний ефект при збільшенні параметру	Можливий недолік
FilterRadius	М'якіша межа тіні	Розмивання деталей
PCFKernelSize	Сильніше згладжування	Збільшення вартості обчислень
DepthRejectionSharpness	Краще збереження меж за глибиною	Зернистість, недостатнє згладжування
NormalRejectionSharpness	Зменшення змішування різних поверхонь	Жорсткі переходи
NoiseScale	Зменшення регулярних патернів	Мерехтіння при русі камери
BlackLevel	Глибше затемнення	Втрата деталей у темних ділянках

Параметри фільтрації та корекції тіні дозволяють гнучко налаштовувати фінальний вигляд результату. Найбільш збалансований ефект досягається за помірних значень радіуса фільтрації, обмеженої чутливості до перепадів глибини й нормалей та помірної фінальної корекції контрасту. Надмірне збільшення цих параметрів може покращити окремі характеристики тіні, але водночас спричинити розмивання, зернистість, мерехтіння або втрату деталей.

4.5 Дослідження продуктивності системи

4.5.1 Вплив кількості динамічних джерел світла

Для оцінювання масштабованості розробленої системи було проведено тестування на одній і тій самій сцені з різною кількістю активних динамічних джерел світла. У всіх запусках використовувалися однакова позиція камери, однакова геометрія сцени та однаковий режим фільтрації тіньового фактора. Як основний режим для цього тесту було обрано Joint Bilateral, оскільки він забезпечує якісніше збереження геометричних меж тіні порівняно з простішими режимами вибірки.

Під час тестування змінювалася лише кількість джерел світла. Для кожної конфігурації вимірювався час кадру, який відображає повний час кадру з урахуванням CPU-підготовки та виконання GPU-проходів. Окремо фіксувалися час виконання lighting pass та час DXR-проходу генерації атласу тіней. Результати тестування наведені у таблиці 4.3.

Таблиця 4.3 – Час кадру при різній кількості джерел світла.

Кількість джерел	Час кадру, мс	Lighting pass, мс	DXR RayGen, мс
16	6.02	0.32	2.55
64	7.10	1.20	3.60
128	10.2	3.01	4.60
256	16.21	6.67	5.93

Продовження таблиці 4.3

Кількість джерел	Час кадру, мс	Lighting pass, мс	DXR RayGen, мс
512	25.10	15.60	6.12
1024	48.84	38.78	6.23

За результатами тестування видно, що зі збільшенням кількості джерел світла загальний час кадру поступово зростає. При 16 джерелах час кадру становив 6.02 мс, тоді як при 1024 джерелах він збільшився до 48.84 мс. Основне зростання спостерігається саме у lighting pass: з 0.32 мс при 16 джерелах до 38.78 мс при 1024 джерелах. Це пояснюється тим, що під час освітлення система повинна обробляти більше релевантних джерел, виконувати більше звернень до атласу тіней та застосовувати фільтрацію тіньового фактора для більшої кількості освітлювачів.

Час DXR-проходу також збільшується зі зростанням кількості джерел, однак після 256 джерел його приріст стає значно менш вираженим. Так, при 256 джерелах час DXR RayGen становив 5.93 мс, а при 1024 джерелах — 6.23 мс. Це свідчить про те, що в даній конфігурації основним обмежувальним фактором при великій кількості джерел стає не генерація атласу тіней, а подальше застосування освітлення та фільтрації в lighting pass.

4.5.2 Вплив режиму фільтрації

Для оцінювання впливу режиму фільтрації на продуктивність було виконано тестування на одній і тій самій сцені з фіксованою кількістю динамічних джерел світла. У всіх запусках використовувалося 256 джерел, однакова позиція камери, однакова геометрія сцени та однакова конфігурація атласу тіней. Змінювався лише режим апскейлу та фільтрації тіньового фактора. Даний підхід дозволяє оцінити саме вартість фільтрації під час lighting pass без впливу інших параметрів сцени.

У межах тестування вимірювався повний час кадру, а також окремо час виконання lighting pass, оскільки саме в цьому проході виконується вибірка з атласу, апскейл і фільтрація тіньового фактора для релевантних джерел світла. Результати вимірювання наведено в таблиці 4.4.

Таблиця 4.4 – Вартість режимів фільтрації.

Режим фільтрації	Час кадру, мс	Lighting pass, мс
Point	10.31	1.7
Bilinear	10.3	1.69
PCF 3x3	12.98	3.86
PCF 5x5	14.15	5.40
PCF 16x16	62.43	44.80
Joint Bilateral	12.70	3.76
Joint Bilateral + PCF 3x3	14.82	5.71
Joint Bilateral + PCF 5x5	26.66	17.60
Joint Bilateral + PCF 16x16	212.43	202.79
Poisson-based	13.78	4.60
Joint Bilateral + Poisson-based	25.30	16.80

Отримані результати показують, що найменшу вартість мають режими Point і Bilinear, оскільки вони потребують мінімальної кількості вибірок з атласу тіней. Режими PCF 3x3, PCF 5x5, Joint Bilateral і Poisson-based збільшують час lighting pass, але залишаються придатними для використання в межах інтерактивного рендерингу. Найбільше зростання часу спостерігається для великих ядер фільтрації, зокрема PCF 16x16 і Joint Bilateral + PCF 16x16, оскільки кількість вибірок і додаткових обчислень у цих режимах суттєво зростає.

4.6 Узагальнення результатів тестування

Узагальнене порівняння режимів роботи фільтрації наведено в таблиці 4.5.

Оцінювання виконувалося з урахуванням візуального порівняння тіней, стабільності результату та вимірної вартості lighting pass у тесті з 256 динамічними джерелами світла.

Таблиця 4.5 – Узагальнене порівняння режимів роботи системи.

Режим	Якість тіні	Стабільність	Продуктивність	Рекомендація використання
Point	Низька	Висока	Висока	Діагностика атласу тіней
Bilinear	Низька	Висока	Висока	Швидкий режим для дуже великої кількості джерел
PCF 3x3	Середня	Висока	Середня	Збалансований режим
PCF 5x5	Висока	Висока	Середня	Для м'якших тіней
Joint Bilateral	Висока	Висока	Середня	Режим із збереженням меж
Joint Bilateral + PCF	Висока	Середня	Низька	Режим із помірним збереженням меж та зменшенням зубчастості тіней
Poisson-Based	Висока	Середня	Середня	Зменшення артефактів, але з необхідністю у використанні антиаліасингових засобів
Joint Bilateral + Poisson-Based	Висока	Середня	Низька	Режим із помірним збереженням меж та зменшенням зубчастості тіней але з необхідністю у використанні антиаліасингових засобів

З таблиці видно, що найшвидшими режимами є Point і Bilinear, оскільки вони мають мінімальну кількість вибірок з атласу. Однак їхня візуальна якість є обмеженою, особливо на межах тіні. Режими PCF 3x3, Joint Bilateral і Poisson-based забезпечують кращу якість за помірного зростання вартості. Найдорожчими є режими з великим ядром фільтрації та комбіновані режими.

На основі отриманих результатів можна визначити декілька практичних

режимів використання системи. Для діагностики формування атласу та перевірки первинної карти видимості доцільно використовувати Point, оскільки він найменше змінює початкові дані. Для сценаріїв із великою кількістю джерел світла доцільнішими є Bilinear або PCF 3x3 фільтрації, оскільки вони забезпечують прийнятний баланс між швидкістю та якістю.

Для сцен, де важливо зберігати геометричні межі тіні, рекомендованим режимом є Joint Bilateral. Він має вищу вартість за прості режими, однак краще враховує глибину та нормалі поверхонь. Для режимів підвищеної якості можна використовувати Joint Bilateral + PCF або Joint Bilateral + Poisson-based, однак їх доцільно застосовувати лише за обмеженої кількості джерел або за наявності достатнього запасу продуктивності.

					ІАЛЦ.467800.004 ПЗ	Арк.
						80
Змін.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ ДО РОЗДІЛУ 4

У четвертому розділі було проведено тестування розробленої системи атласу рей-трейсингових тіней. У межах функціонального тестування перевірено коректність формування спільного атласу тіней, побудови карти відповідності texel-to-light, генерації тіней для різних типів джерел світла та застосування тіньового фактора під час lighting pass. Отримані результати підтвердили, що система коректно формує службові ресурси, прив'язує області атласу до відповідних джерел світла та використовує сформовані значення видимості під час фінального освітлення сцени.

Під час дослідження візуальної якості було порівняно різні режими апскейлу та фільтрації тіньового фактора. Найпростіші режими, зокрема Point і Bilinear, мають найменшу обчислювальну вартість, однак гірше приховують дискретність меж тіні. Режими PCF, Joint Bilateral і Poisson-based забезпечують кращу якість зображення, але збільшують час виконання lighting pass. Комбіновані режими дають вищу якість, проте потребують значно більшої кількості вибірок і тому є доцільними лише для режимів підвищеної якості або сцен із меншим навантаженням.

Дослідження продуктивності показало, що зі збільшенням кількості динамічних джерел світла зростає повний час кадру. При цьому найбільший вплив на продуктивність у тестах мав саме lighting pass, оскільки в ньому для релевантних джерел виконується вибірка з атласу, апскейл і фільтрація тіньового фактора. Час DXR-проходу генерації атласу також збільшується зі зростанням кількості джерел, однак у проведених тестах його приріст був менш різким порівняно з вартістю фінального освітлення.

Таким чином, тестування підтвердило працездатність розробленої системи та показало, що її ефективність значною мірою залежить від кількості джерел світла, вибраного режиму фільтрації та допустимого бюджету кадру.

					ІАЛЦ.467800.004 ПЗ	Арк.
						81
Змін.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У межах дипломного проєкту було розроблено програмні засоби атласу рейтрейсингових тіней для динамічних джерел світла в системі комп'ютерної графіки реального часу. Розроблене рішення інтегровано у графічний рушій і поєднує растеризаційний рендеринг, кластерне відкладене освітлення, DirectX Raytracing та спільне атласне представлення тіньових даних.

У роботі проаналізовано існуючі підходи до побудови тіней у системах комп'ютерної графіки реального часу. Розглянуто класичні методи на основі карт тіней, зокрема Shadow Mapping, Variance Shadow Maps, Moment Shadow Mapping, Cascaded Shadow Maps, Screen-space підходи, методи фільтрації та рейтрейсингові тіні. У результаті аналізу було встановлено, що традиційні методи мають обмеження при роботі з великою кількістю динамічних джерел світла, оскільки потребують значної кількості окремих тіньових ресурсів або складного керування роздільною здатністю.

Реалізовано формування метаданих активних джерел світла, адаптивний вибір розміру тайлів, розміщення тайлів у спільному атласі та побудову карти texel-to-light. Для генерації первинної карти видимості використовується DXR RayGen, який для активних текселів атласу відновлює параметри точки поверхні з GBuffer, формує тіньовий промінь, виконує трасування в TLAS і записує результат видимості в атлас. Для зменшення помилкового самозатінення використовується normal-dependent bias, а для підвищення продуктивності застосовується раннє завершення трасування після першого знайденого перетину.

Також було реалізовано інтеграцію тіньового атласу в кластерне відкладене освітлення. Тіньовий фактор обчислюється безпосередньо під час lighting pass для конкретного джерела світла, а не як окремий постпроцес. Це дозволяє виконувати вибірку, апскейл і фільтрацію лише для тих джерел, які реально впливають на поточний фрагмент. У системі передбачено кілька режимів фільтрації.

					ІАЛЦ.467800.004 ПЗ	Арк.
						82
Змін.	Арк.	№ докум.	Підпис	Дата		

Функціональні тести підтвердили коректність формування атласу тіней, карти texel-to-light, генерації тіней для різних типів джерел світла та застосування тіньового фактора під час lighting pass. Дослідження візуальної якості показало, що прості режими вибірки мають найменшу вартість, але гірше приховують дискретність меж тіні, тоді як PCF, Joint Bilateral і комбіновані режими забезпечують кращий візуальний результат за рахунок більшої кількості вибірок і додаткових обчислень.

Таким чином, поставленої мети роботи було досягнуто: розроблено та протестовано програмні засоби реалізації атласу рей-трейсингових тіней для багатьох динамічних джерел світла. Запропоноване рішення забезпечує компактне зберігання тіньових даних у межах одного атласу, використовує трасування променів для визначення видимості джерел світла та інтегрує тіньовий фактор у кластерне відкладене освітлення. Подальший розвиток системи може бути спрямований на вдосконалення пріоритезації джерел світла, покращення темпоральної стабільності, оптимізацію пакування тайлів і розширення підтримки тіней від площинних джерел світла.

					ІАЛЦ.467800.004 ПЗ	Арк.
						83
Змін.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Klyatchenko Ya., Samoilenko D. Ray-traced shadow atlas for multiple dynamic light sources integrated with rasterization rendering. Збірник тез доповідей Вісімнадцятої наукової конференції «Прикладна математика та комп'ютинг ПМК 2025», 19–21 листопада 2025 р. Київ: КПІ, 2025. С. 161–164. Режим доступу: <https://ela.kpi.ua/handle/123456789/78417>
2. Williams L. Casting Curved Shadows on Curved Surfaces. Proceedings of SIGGRAPH. 1978. P. 270–274.
3. Reeves W. T., Salesin D. H., Cook R. L. Rendering Antialiased Shadows with Depth Maps. Proceedings of SIGGRAPH. 1987. P. 283–291.
4. Fernando R. Percentage-Closer Soft Shadows. GPU Gems: Programming Techniques, Tips, and Tricks for Real-Time Graphics. Addison-Wesley, 2007. P. 303–329.
5. Donnelly W., Lauritzen A. Variance Shadow Maps. Proceedings of the 2006 Symposium on Interactive 3D Graphics and Games. 2006. P. 161–165.
6. Lauritzen A. Variance Shadow Mapping. GPU Gems 3 / ed. H. Nguyen. Addison-Wesley, 2008. P. 279–292.
7. Peters C., Klein R. Moment Shadow Mapping. Proceedings of the 19th Symposium on Interactive 3D Graphics and Games. 2015. P. 7–14.
8. Engel W. Cascaded Shadow Maps. ShaderX5: Advanced Rendering Techniques. Charles River Media, 2006.
9. McGuire M. The Graphics Codex. 2024. URL: <https://graphicscodex.com/>.
10. Pharr M., Jakob W., Humphreys G. Physically Based Rendering: From Theory to Implementation. 4th ed. MIT Press, 2023.
11. Akenine-Möller T., Haines E., Hoffman N., Pesce A., Iwanicki M., Hillaire S. Real-Time Rendering. 4th ed. CRC Press, 2018.
12. Olsson O., Billeter M., Assarsson U. Clustered Deferred and Forward Shading. Proceedings of High Performance Graphics. 2012. P. 87–96.
13. Microsoft. Direct3D 12 Graphics. URL: <https://learn.microsoft.com/en-us/windows/win32/direct3d12/direct3d-12-graphics>.

					ІАЛЦ.467800.004 ПЗ	Арк.
						84
Змін.	Арк.	№ докум.	Підпис	Дата		

14. Microsoft. DirectX Raytracing (DXR) Functional Spec. URL: <https://microsoft.github.io/DirectX-Specs/d3d/Raytracing>.
15. Schied C., Dachsbacher C., Wimmer M., Salvi M. Spatiotemporal Variance-Guided Filtering: Real-Time Reconstruction for Path-Traced Global Illumination. Proceedings of High Performance Graphics. 2017. P. 1–12.
16. Karis B. High Quality Temporal Supersampling. Advances in Real-Time Rendering in Games, SIGGRAPH Courses. 2014.
17. Heitz E., Dupuy J., Hill S., Neubelt D. Real-Time Polygonal-Light Shading with Linearly Transformed Cosines. ACM Transactions on Graphics. 2016. Vol. 35, No. 4. Article 41.
18. McGuire M. Computer Graphics Archive: Crytek Sponza. URL: <https://casual-effects.com/data>.

ДОДАТОК 1

до дипломного проекту

на тему: Комп'ютерні засоби реалізації атласу рей-трейсингових тіней
для динамічних джерел світла

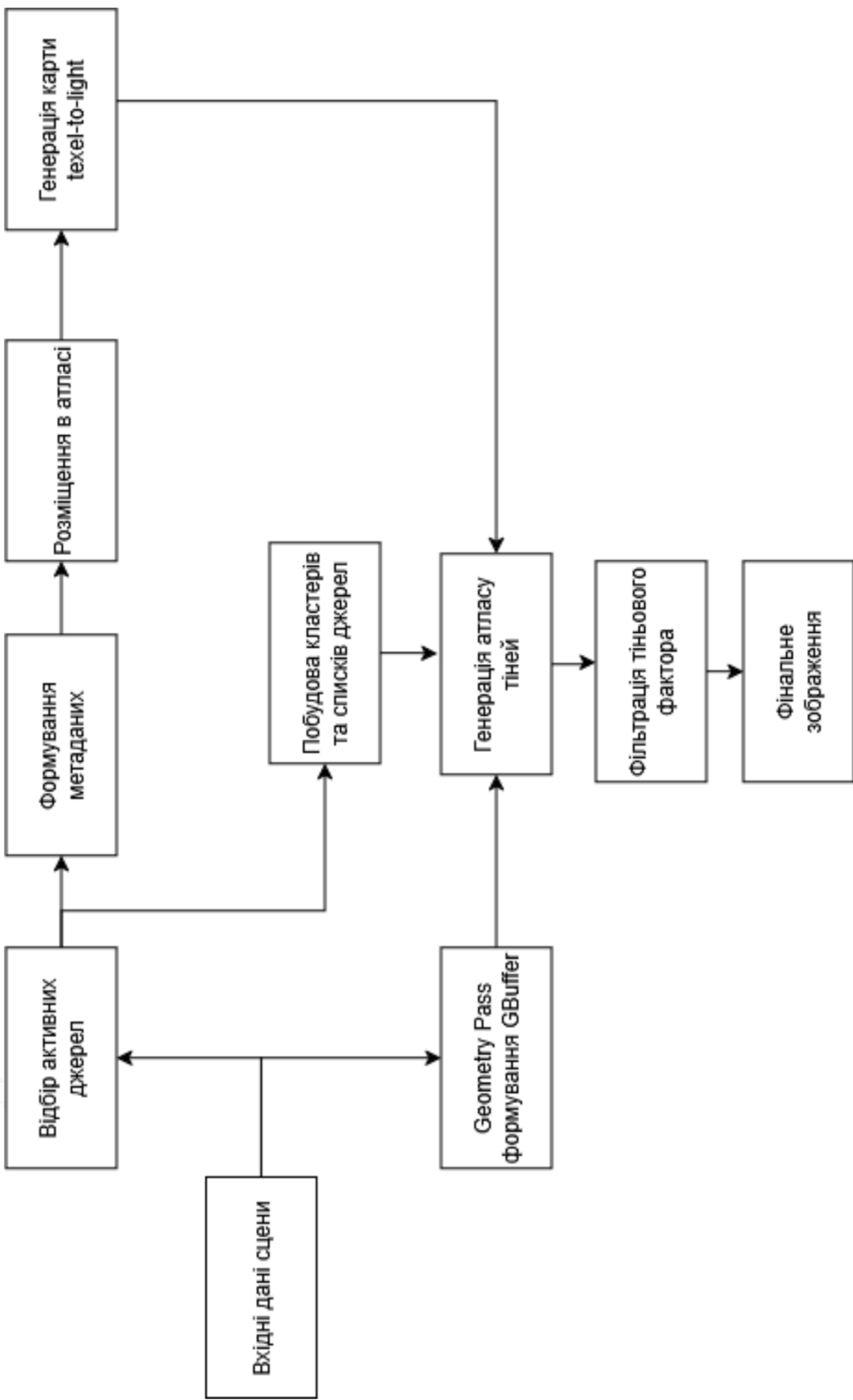
Загальна архітектура системи атласу рей-трейсингових тіней.

Схема структурна.

ІАЛЦ.467800.005 Д1

Аркушів 1

Київ – 2026 року



ДОДАТОК 2

до дипломного проекту

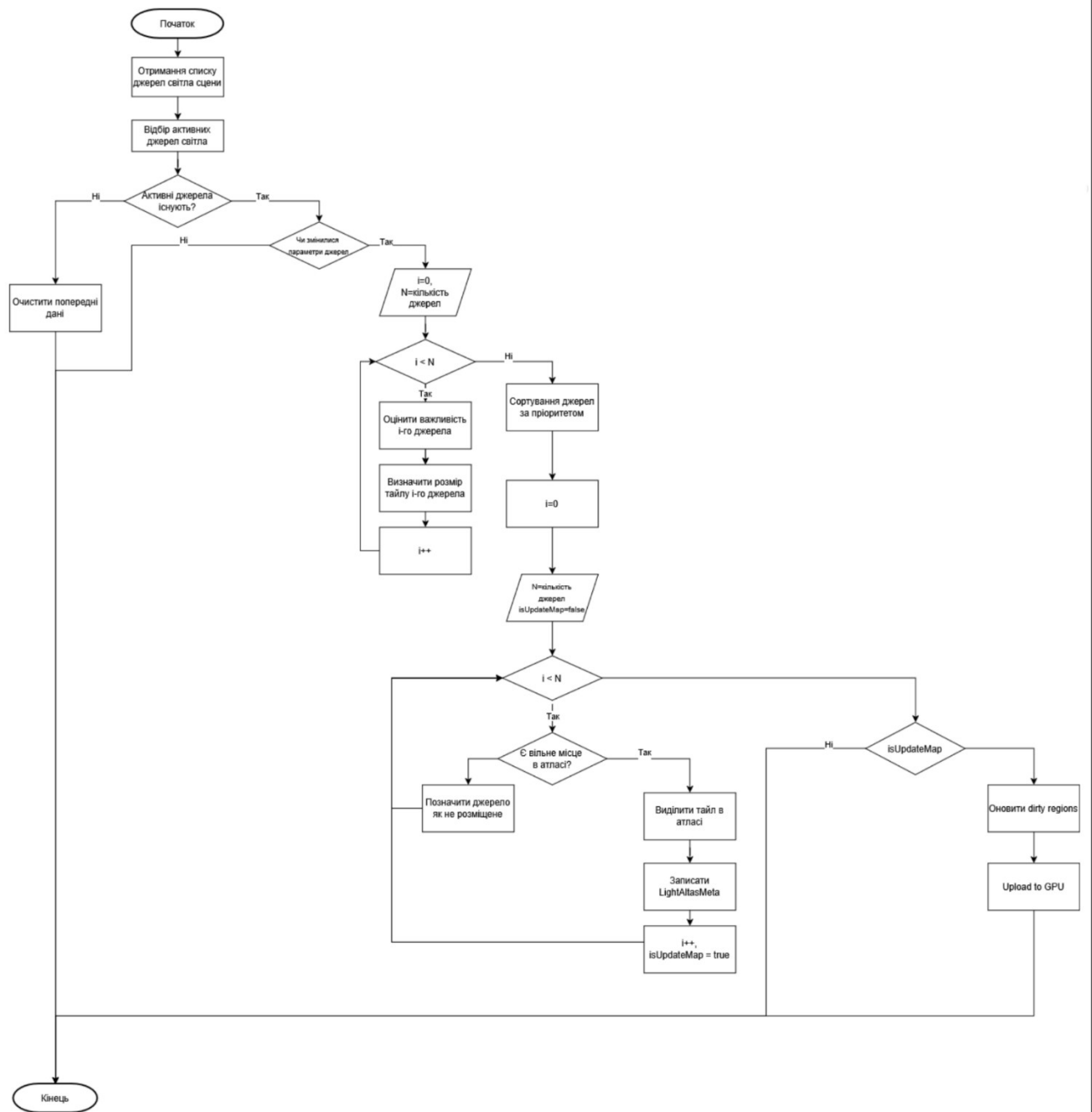
на тему: Комп'ютерні засоби реалізації атласу рей-трейсингових тіней
для динамічних джерел світла

Алгоритм формування атласу рей-трейсингових тіней та карти texel-to-light. Схема алгоритму.

ІАЛЦ.467800.006 Д2

Аркушів 1

Київ – 2026 року



ДОДАТОК 3

до дипломного проекту

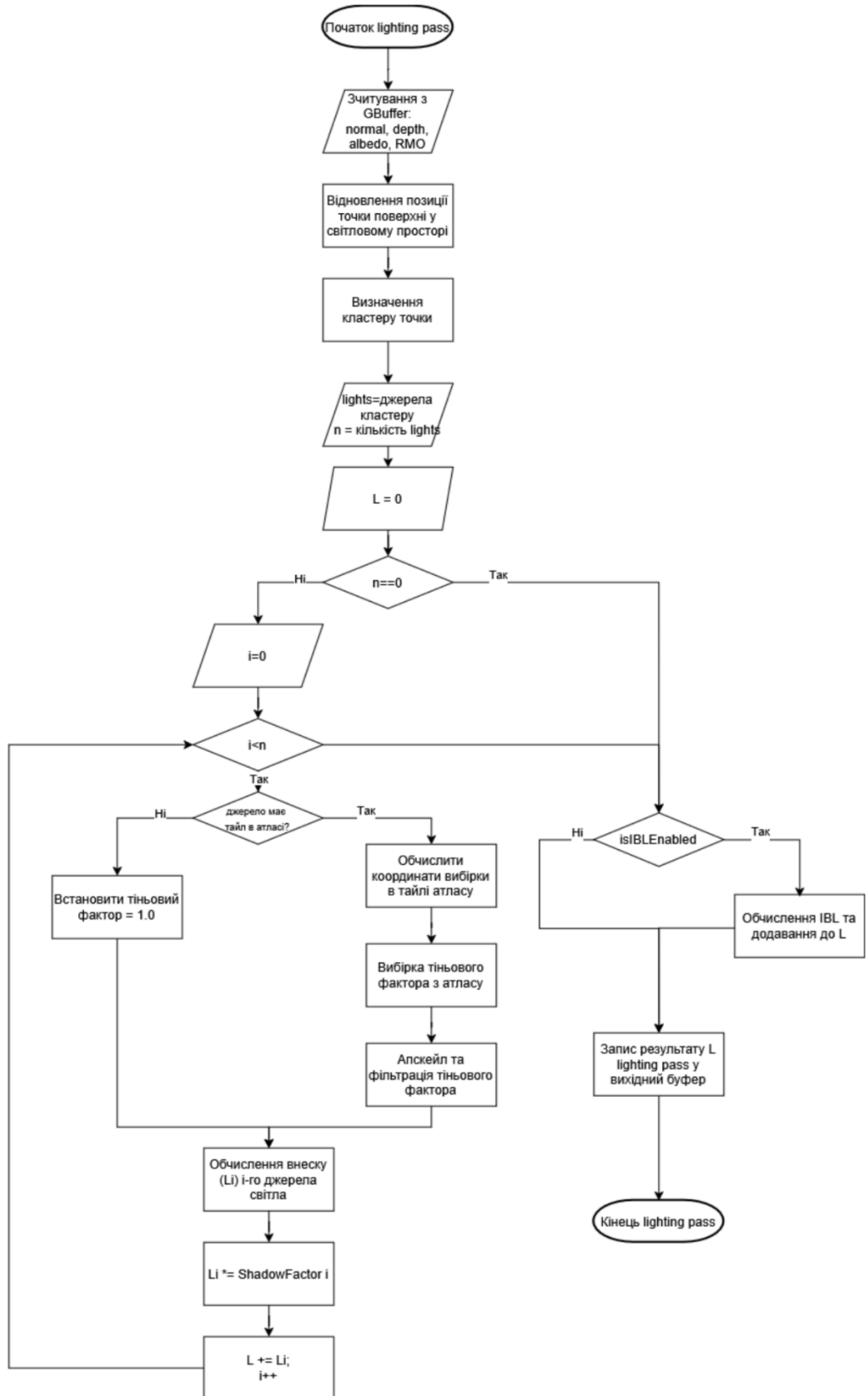
на тему: Комп'ютерні засоби реалізації атласу рей-трейсингових тіней
для динамічних джерел світла

**Застосування атласу рей-трейсингових тіней у кластерному відкладеному
освітленні. Схема алгоритму.**

ІАЛЩ.467800.007 ДЗ

Аркушів 1

Київ – 2026 року



ІАЛЦ.467800.007 ДЗ					
Вибір	Алг.	Модуль	Повтор	Дат.	
Вибір	Специфікація				
Вибір	Використання				
Вибір	Використання				
Вибір	Використання				

Застосування атласу реєстрації тіней у кластерному відображенні освітлення
Схема алгоритму

КПІ ім. І.Скоропадського
ФІСТІМ, КВ-21

ДОДАТОК 4

до дипломного проекту

на тему: Комп'ютерні засоби реалізації атласу рей-трейсингових тіней
для динамічних джерел світла

**Багатопотокова підготовка та виконання GPU-команд у графічному рушії.
Схема структурна.**

ІАЛЦ.467800.008 Д4

Аркушів 1

Київ – 2026 року

