Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Медіасистем та технологій
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Дослідження технологій створення 3D-моделей
для ААА відеоігор
(тема)

Виконав:
студент 2 курсу, групи ТЕМВм-22-1

Мельніченко Д.О.

Спеціальності 186 Видавництво та поліграфія

Тип програми Освітньо-професійна

Освітня програма
Технології електронних
мультимедійних видань

Керівник проф. Дейнеко Ж.В.

Допускається до захисту
Зав. кафедри МСТ _____ Дейнеко Ж.В.
(підпис) (прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
Кафедра Медіасистеми та технології
Рівень вищої освіти другий (магістерський)
Спеціальність 186 Видавництво та поліграфія
Тип програми Освітньо-професійна
Освітня програма Технології електронних мультимедійних видань
(шифр і назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри МСТ_____
(підпис)
« 30 » жовтня 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові *Мельніченку Дмитру Олександровичу*
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження технологій створення 3D-моделей для ААА відеоігор
затверджена наказом по університету від "27" жовтня 2023 р. № 1248 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 25.01.2024

3. Вихідні дані до роботи: Документація розробки для Autodesk 3D Studio MAX, Вимоги до тривимірної моделі за кількістю полігонів, картами текстур. Формати файлів текстур – png, jpg,

4. Перелік питань, що потрібно опрацювати в роботі
Вступ; 1 Аналіз завдання на кваліфікаційну роботу; 2 Аналітичний огляд літератури за темою роботи; 3 Огляд і аналіз існуючих підходів до створення моделей; 4 Розробка рекомендацій щодо створення інтерфейсу; 5 Розробка методики і складання плану проведення експерименту; 6 Проведення експерименту, оцінка ефективності результату; 7 Економічне обґрунтування дослідження. Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Вихідні дані та мета роботи; Актуальність дослідження, Мета роботи, Задачі дослідження, Аналіз існуючих підходів до розробки тривимірних моделей анімованих персонажів; Розробка рекомендацій щодо створення інтерфейсу; Розробка методики і складання плану проведення експерименту; Проведення експерименту, оцінка ефективності результату; Економічне обґрунтування дослідження. Висновки

6. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	проф. каф. МСТ, Дейнеко Ж.В.		10.01.2024
Економічна частина	ас. каф. ЕК, Помогалова Н.В.		20.01.2024

КАЛЕНДАРНИЙ ПЛАН

Номер	Назва етапів роботи (проекту)	Термін виконання етапів роботи	Примітка
1	Аналіз завдання на кваліфікаційну роботу	30.10.23 – 05.11.23	виконано
2	Аналітичний огляд літератури за темою роботи	06.11.23 – 19.11.23	виконано
3	Огляд предметної області дослідження	20.11.23 – 03.12.23	виконано
4	Дослідження дитячого навчання у мобільних додатках	04.12.23 – 24.12.23	виконано
5	Розробка методики проектування дизайну для мобільного додатку з навчання дітей	25.11.23 – 28.12.23	виконано
6	Практична реалізація проекту	29.12.23 – 01.01.24	виконано
8	Економічне обґрунтування дослідження	02.01.24 – 04.01.24	виконано
10	Оформлення пояснювальної записки	25.12.23 – 10.01.24	виконано

Дата видачі завдання «*30* » *жовтня* 20*23* р.

Студент _____ Мельніченко Д.О.
(підпис) (прізвище, ініціали)

Керівник роботи _____ Дейнеко Ж.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка містить 73 стор., 7 табл., 62 рис., 17 джерел.

3D-МОДЕЛЬ, UV-РОЗГОРТКА, МОДЕЛЮВАННЯ, ПОЛІГОН, РЕНДЕРІНГ, КОНЦЕПТ-АРТ, ТЕКСТУРА, КАРТИ ТЕКСТУР.

В роботі розглянуто проблеми створення тривимірних моделей в сучасній індустрії ААА-відеоігор, проведено аналіз сучасних аналогів пайплайнів та корисних засобів для розробки моделей. На основі виявлених проблем та існуючих методів сформульовано рекомендації щодо оптимізації 3D-моделей для пристроїв. Відповідно до аналізу процесу рендерингу виявлено основні критерії оцінки для моделей ААА-відеоігор. Розроблено декілька 3D-моделей для проведення дослідження та обрано метод проведення дослідження. На основі обраних критеріїв оцінено моделі у 3D-редакторі та рушії гри. Результати експерименту проаналізовано та на основі них складено висновки. Таким чином, було розроблено сучасні рекомендації щодо оптимізації 3D-моделей для відеоігор.

Метою даної роботи є розробка сучасних рекомендацій щодо оптимізації 3D-моделей.

Об’єктом дослідження є технологія створення 3D-моделей для ААА відеоігор. Область застосування – використання тривимірних моделей у ААА-відеоіграх.

ABSTRACT

The explanatory note contains 73 p., 7 tabl., 62 fig., 17 sources.

3D MODEL, UV SCANNING, MODELING, POLYGON, RENDERING, CONCEPT ART, TEXTURE, TEXTURE MAPS.

The paper examines the problems of creating three-dimensional models in the modern industry of AAA video games, analyzes modern analogues of pipelines and useful tools for developing models. Recommendations for optimizing 3D models for devices are formulated based on the identified problems and existing methods. According to the analysis of the rendering process, the main evaluation criteria for AAA video game models were revealed. Several 3D models have been developed for conducting the research and the method of conducting the research has been chosen. Models in the 3D editor and game engine are evaluated based on the selected criteria. The results of the experiment were analyzed and conclusions were drawn based on them. Thus, modern guidelines for optimizing 3D models for video games have been developed.

The object of research is the optimization of 3D models of video games.

The purpose of this work is to develop modern recommendations for the optimization of 3D models. Areas of application – the use of the model in AAA video games.

ЗМІСТ

С.

СКОРОЧЕНЯ ТА УМОВНІ ПОЗНАКИ 8
ВСТУП…… 9
1 ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ..... 12
1.1 Теоретичні аспекти та задачі тривимірної графіки 12
1.2 Ключові компоненти в 3D віртуальній сцені..... 15
1.3 Проблематика створення 3D контенту для відеоігор 17
2 АНАЛІЗ СПЕЦИФІКИ PHYSICAL BASED RENDERING 25
2.1 Вимоги PBR в ігровій індустрії 25
2.2 Робочі процеси PBR 29
3 АНАЛІЗ МЕТОДІВ 3D МОДЕЛЮВАННЯ ДЛЯ ААА ВІДЕОІГОР..... 40
3.1 Subdiv-моделінг 40
3.2 CAD моделювання..... 41
3.3 Boolean моделювання..... 43
3.4 Скульптинг моделювання 44
4 РОЗРОБКА МЕТОДИКИ СТВОРЕННЯ ТРИВИМІРНИХ МОДЕЛЕЙ 46
4.1 Характерні особливості пайплайнів 46
4.2 Особливості оптимізації моделей Weighted Normals пайплайна..... 47
4.3 Особливості створення моделей з Weighted Normals пайплайн 49
4.4 Особливості триангуляції сітки моделей з
Weighted Normals пайплайн..... 51
5 АПРОБАЦІЯ МЕТОДИКИ. АНАЛІЗ РЕЗУЛЬТАТІВ 52
5.1 Процес створення mid poly моделі для ААА-відеігор..... 52
5.2 Використання масок..... 55
6 ЕКОНОМІЧНА ЧАСТИНА 63
6.1 Характеристика науково-дослідної роботи..... 63
6.2 Етапи виконання НДР, їх трудомісткість та заробітна плата 63
6.3 Розрахунок одноразових витрат на розробку НДР..... 66

6.4 Оцінка результатів науково-дослідної роботи 69
ВИСНОВКИ 71
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 72

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ААА-відеоігри – (англійською, «triple-A») — умовна підмножина відеоігор, створюваних й розповсюджуваних середніми й великими видавництвам, що зазвичай мають більше коштів на розробку й рекламу.

Low-poly –модель за малою кількістю полігонів.

High-poly – тривимірна модель з великою кількістю полігонів.

Пайплайн – сукупність процесів розробки, підготовки та виробництва ігрового продукту.

Полікаунт – кількість полігонів у моделі.

Півот (Pivot) – центр повороту та масштабування моделі. За замовчуванням знаходиться у геометричному центрі об’єкту.

Меш (Mesh) – сукупність вершин, ребер та граней, що створюють форму об’єкта.

GPU – центральний процесор.

CPU – графічний процесор.

Оверлап – накладання однакових шелів розгортки.

Шейдери – програма, що обробляється на відеокарті, ця програма властива кожному об'єкту на сцені і вона говорить комп'ютеру, як цей об'єкт потрібно обробляти.

Артефакт – некоректне відображення карт та текстур.

ВСТУП

Розвиток та удосконалення ігор тісно взаємопов'язане з розвитком комп'ютерних технологій. Зараз безліч комплектуючих комп'ютерів розробляється спеціально для сучасних ігор. Наприклад, високопродуктивні відеокарти, вартість яких досягає вартості двох комп'ютерів для роботи в офісі. Всі ігри розробляються на основі останніх інновацій комп'ютерної індустрії, реагуючи на всі розробки та посилюючи реалістичність зображення та звукового супроводу. На сьогоднішній день існують вражаючі своєю правдоподібністю ігри, з гарним графічним та звуковим оформленням, практично повністю повторюючи реальне життя. Існує і постійно виникає величезна кількість фірм, які розробляють нові і нові ігри.

На сьогодні індустрія ААА-відеоігор – один з найпопулярніших та наймасштабніших напрямів розробки ігор. З кожним роком індустрія розвивається дуже стрімко. Причина в тому, що сьогодні існують багато якісних інструментів для створення привабливої реалістичної графіки, але це потребує більших ресурсів від консолей або персональних комп’ютерів (ПК). Також кожна компанія конкурує з іншими аби створити найкращу та цікавішу гру. Актуальність розробки полягає в тому, що сучасні ігри наближаються до реалістичної графіки, анімації тощо.

Іноді складно зрозуміти в якому реальному чи віртуальному середовищі знаходиться гравець.

Мета кваліфікаційної роботи магістра – дослідження технологій створення 3D-моделей. Для досягнення поставленої мети необхідно вирішити наступні завдання:

- аналіз предметної галузі та літератури за темою дослідження;
- методи отримання 3D зображення;
- провести аналіз ключових компонентів в 3D віртуальній сцені;
- виявити проблематику створення 3D контенту для відеоігор;

– виявити сучасні вимоги в ігровій індустрії для створення 3D об’єктів;

– вивчити стандартний пайплайн для створення 3D моделей;

– аналіз нових пайплайнів, які більш досконалі для деяких видів 3D моделей.

Об'єктом дослідження в кваліфікаційній роботі є сучасні технології та засоби розробки, аналізу та підготовки 3D об’єктів в ігровому середовищі. Предметом дослідження є тривимірні моделі та інструменти для різних пайплайнів під час розробки моделей.

Теоретична значущість цього дослідження полягає в розробці та систематизації методичної інформації, необхідної для якісного опрацювання специфічних тривимірних моделей.

Практична значущість полягає в застосуванні отриманих знань для самостійної розробки та вдосконалення процесу створення тривимірних моделей, робота може бути використана в інформаційних цілях для зацікавлених осіб.

Для досягнення мети були використані такі методи дослідження, як аналіз літературних джерел, систематизація зібраного за темою матеріалу, класифікація і узагальнення, виявлення структури, проектування, практичне опрацювання такого матеріалу.

Структура кваліфікаційної роботи магістра обумовлена предметом, метою і завданнями дослідження.

У першому розділі розглянуто теоретичні аспекти та задачі тривимірної графіки, також вивчені нові технології, що виводять освітлення у комп’ютерній графіці на більш новий рівень тобто Ray Tracing та Rasterization. В другому розділі визначені ключові компоненти в 3D віртуальній сцені, а саме контент такі як: камера, типи джерел освітлення, 3D модель (геометрія), матеріал.

В третьому розділі виявлено проблематику створення 3D контенту для відеоігор – це обмеження поліакаунту та розмірів текстур.

В четвертому розділі описані вимоги індустрії в створенні 3D арта, а саме: ліміт часу на створення 3D моделі за мінімальний час та також обмеження Physical Based Rendering (RBR).

В п’ятому розділі наведено аналіз пайплайнів для різних типів 3D моделей, а також застосування запропонованого методу для створення 3D об’єктів оточення.

Надалі визначено витрати замовника на науково-дослідну роботу щодо розробки такої методики.

Наприкінці узагальнено висновки з проведеного дослідження.

1 ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ

1.1 Теоретичні аспекти та задачі тривимірної графіки

На сьогоднішній день тривимірна комп'ютерна графіка міцно увійшла до нашого життя, часом ми не звертаємо уваги на її прояви. Слід нагадати, що тривимірна графіка використовується не тільки в відеоіграх, а також у кіно, мультфільмах та дизайні. Але в дослідницькій роботі буде розглядатись тільки ігрова індустрія. З часом розвиток сучасної обчислювальної техніки, високоякісних технічно складних консолей і всесвітньої павутини змінило все це, перетворивши відеоігри на складні, реалістичні продукти, які розраховані на багато користувачів та високоякісну графіку, сучасне відео і музику. Сьогодні такі відеоігри відносяться до типу ААА – відеоігри, які мають великий бюджет та високі стандарти розробки, а також ігровий простір, тривимірні моделі і об'єкти оточення мають відповідати необхідним параметрам і високим вимогам щодо реалістичності та відповідності до ігрового середовищі гри [3].

Таким чином Gamedev-індустрія переросла від простих до складних відеоігор, в котрих використовуються сучасні технології для створення ультра-реалістичної картинки. Але для того щоб отримати реалістичне зображення в відеогрі створити красиві та якісні 3D моделі недостатньо, дуже важливу роль відіграє освітлення та тіні. Тому з розвитком індустрії з'явилися технології які можуть створювати освітлення якомога реалістичним – це Ray Tracing та Rasterization (табл. 1.1).

Ray Tracing – це реалістичне відображення світла і тіней. Цю технологію використовують майже в кожній сучасній ААА-відеогрі, щоб реалістично поєднати всі компоненти в ігровій сцені. Також у відеоіграх використання трасування променів може допомогти покращити візуальні ефекти. Проте ця технологія використовує дуже великі ресурси на відеокарті

тому, що кожну секунду випускається мільйони променів на кожний піксель об’єкту. Чим більше променів тим реалістичне та якісніше буде затінення об’єкту. Так як тіні — це єдиний спосіб показати об’єм тривимірної моделі на екрані. Тому, що фінальну картинку на дисплеї користувач отримує в вигляді 2D зображення (рис. 1.1, рис. 1.2).

Таблиця 1.1 – Порівняльна характеристика методів затінення

Порівняння	Метод затінення	Швидкість	Якість	Витрата ресурсів пристрою
Ray Tracing	Трасування променів	Дуже повільно	Ультра-реалізм	Потребує великих ресурсів
Rasterization	Проекція на площину	Дуже швидко	Недостатня якість та коректність затінення	Потребує малу кількість ресурсів

Рисунок 1.1 – Зображення 3D-об’єкта без трасування променів

Рисунок 1.2 – Зображення 3D-об’єкта з трасуванням променів

Для обробки мільйонів променів в real time тобто під час гри, навіть на самих потужний пристроях потрібно дуже багато часу. Тому аби заощадити ресурси пристрою та збільшити швидкість обробки даних в real time трасування променів використовують не повністю і не всюди але навіть при таких обмеженнях технологія потребує великого об'єму відеопам'яті на пристрої. Найчастіше Ray Tracing використовують на максимальних налаштуваннях в відеоіграх під час катсцен – це заздалегідь відрендерена сцена, в якій користувач не може виконувати будь-які рухи, тобто по суті це маленький відеоролик вставлений в гру.

Існує інша технологія котра дозволяє дуже швидко в real time обробляти тіні з меншими витратами відеопам'яті – це Rasterization. Проте Rasterization не дає такого реалістичного затінення об'єктів в ігровій сцені та не вміє виконувати непрямі обчислення зі світлом. Rasterization побудована на алгоритмі Брезенхема. За допомогою цього алгоритму виконується растеризація тривимірних об'єктів на площині (рис. 1.3, рис. 1.4).

Суть растеризації полягає в тому, що виконується проекція затінення на кожний піксель моделі для того, щоб отримати коректне затінення.

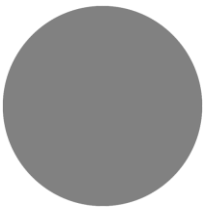


Рисунок 1.3 – Відображення 3D-об'єкта без пастеризації

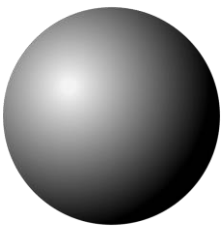


Рисунок 1.4 – Відображення 3D-об'єкта з растеризацією

З розвитком технологій растеризації та трасуванням променів з’явився гібридний спосіб затінення 3D об’єктів в ігровій сцені. Такий підхід дозволяє швидко та майже без втрати якості виконувати реалістичні та коректні затінення в real time. Так як трасування променів дуже повільний та затратний в ресурсах пристрою, а растеризація швидкий але не якісний спосіб затінення. В gamedev-індустрії це останнє ноу-хау.

Суть гібридного метода полягає в тому, що для затінення використовується растеризація так як вона швидше, а для реалістичних відображень поверхонь використовують трасування променів. Таким чином для отримання реалістичної та водночас оптимізованої картинки дві технології комбінують між собою та використовують сильні сторони кожного методу затінення. Такий гібридний спосіб використовує менше ресурсів відеопам’яті та набагато швидший в обчисленнях.

Отже, були вивчені сучасні методи затінення тривимірних моделей в ігрових рушіях.

1.2 Ключові компоненти в 3D-віртуальній сцені

В любій комп’ютерній відеогрі використовується багато різних компонентів для створення 3D віртуальної сцени. Існують компоненти котрі користувач не може бачити під час гри, а також є елементи які наповнюють світ ігрових локацій та взаємодіють з гравцем. Сама головна складова будь-якої ігрової сцени — це камера, без неї неможливо побачити інші компоненти локації такі як: світло, тривимірні моделі, анімації, текстури та VFX ефекти. Тому саме камеру створюють самою першою навіть коли локація гри ще ненаповнена артом (art) та іншими складовими. Далі дуже важливу роль в відеоіграх відіграє світло. Саме освітленням можна задати атмосферу гри або окремої локації. Температура, інтенсивність світла – все це впливає на атмосферу та реалізм гри.

Джерела світла бувають різних типів:

– глобальний – відіграє роль сонця в локації;
– точковий – сфокусоване джерело світла на певний об’єкт;
– прожектор – створення контуру та освітлення фону.

Візуальні компоненти 3D віртуальної сцени:

– вертекс (vertex, вершини), ребра (edges), полігони;
– примітиви;
– VFX ефекти;
– анімації;
– матеріали.

Важливу роль має матеріал об’єкту. Саме він передає фізичні характеристики поверхні 3D-моделі (рис. 1.5). Матеріал включає в себе не тільки створені текстури, а ще має шейдер.

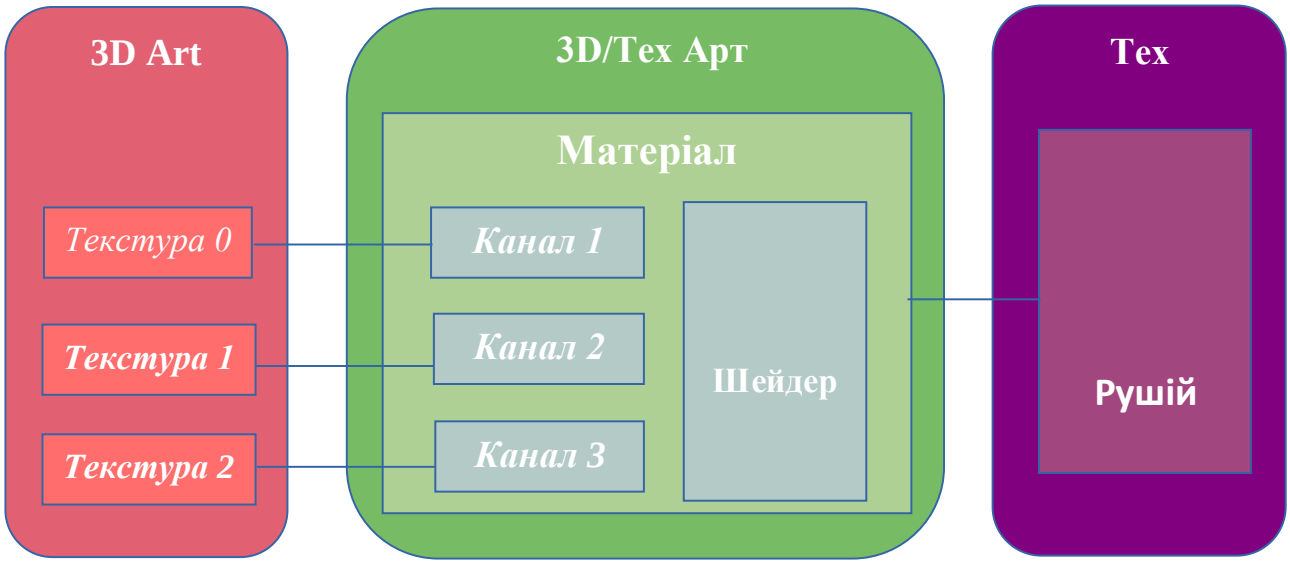


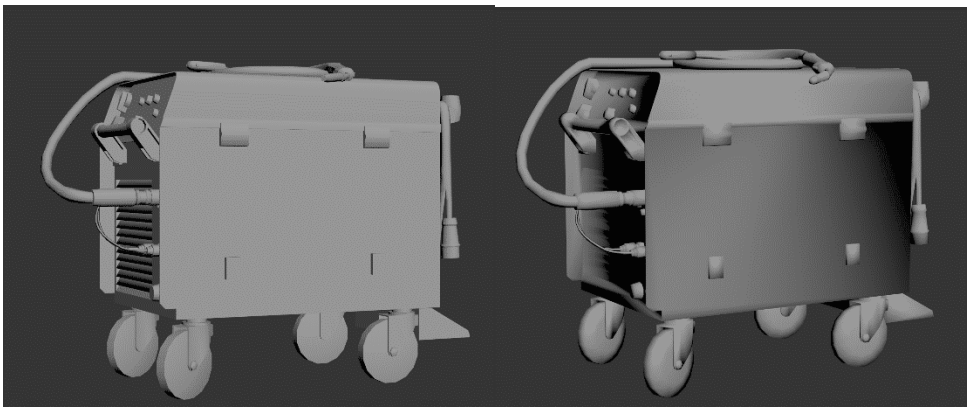
Рисунок 1.5 – Складова шейдера

На (рис. 1.5) зображена складова шейдера, з чого він складається і що необхідно для коректної роботи шейдера в ігровому рушії. Суть шейдерів, полягає в тому, що вони виконуються графічним процесором відеоадаптера, щоб уникнути маси повторюваних операцій в коді програми, знизити навантаження з центрального процесора і мати можливість звернення до процесора відеокарти безпосередньо.

1.3 Проблематика створення 3D контенту для відеоігор

Сьогодні в Gamedev-індустрії існує безліч програм для створення 3D об’єктів. Такі як, Maya, 3D Studio MAX, ZBrush, Blender, Fusion 360, Marvelous Designer – це стандартні програми game-індустрії, котрі використовуються для розробки моделей, кожна з цих програм призначена для створення окремих задач. Під час створення 3D-моделі виникає багато помилок та проблем, котрі навіть досвідчені розробники не можуть наперед передбачити. Основною проблемою під час моделювання – це топологія та шейдинг моделі, а саме здатність сітки моделі коректно реагувати на деформації, вона відповідає за те як саме полігони формують тривимірну модель (рис. 1.6). Необхідно завжди слідкувати за сіткою моделі, особливо це стосується при створенні Low Poly моделі, так як саме вона буде в ігровому рушії.

Зазвичай проблема вибору трикутників або чотирикутників зосереджена навколо шейдингу поверхні моделі. Це актуально при моделюванні органічних тіл, наприклад людини або тварини, особливо якщо ці моделі піддаватимуться майбутній анімації. Такі об'єкти краще моделювати з використанням чотирикутників. Вважається, що краще всього будь-які моделі завжди створювати за допомогою чотирикутної сітки, тому що її деформація передбачувана, на відміну від зміни поверхні, зробленої з трикутників.



Русинок 1.6 – Приклад шейдингу моделі
(зліва – правильно, справа – неправильно (заламаний меш))

При створенні будь-якої моделі, розробники завжди прагнуть зробити її як можна гладшою і природнішою. Існують алгоритми дроблення поверхонь, метою яких є додавання нових елементів для збільшення деталювання. Зазвичай їх застосовують до низькополігональних об'єктів, створюючи згладжену поверхню. Цей алгоритм краще застосовувати до поверхонь з чотирикутною полігональною сіткою, оскільки результати будуть більше передбачувані. А з трикутниками дроблення може стати занадто складним, і в результаті шейдингу на поверхні можуть виникнути видимі помилки. Це виникає через те, що трикутник має непарну кількість вершин. При розміщенні серед чотирикутників трикутника, можуть виникнути "артефакти" на поверхні моделі. Трикутники можна використовувати в моделюванні, і в більшості випадків дуже складно уникнути їх появи, але краще по можливості їх приховувати, щоб вони не створювали проблем. Проте, виключенням є низько-полігональні моделі, або статичні об'єкти, наприклад архітектурні, які не будуть анімовані по усій сітці, і тому в них допустиме використання трикутної полігональної сітки (рис. 1.7).

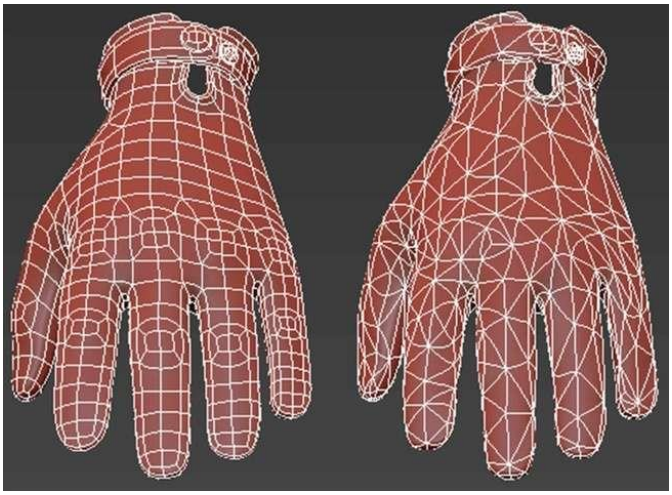


Рисунок 1.7 – Правильна і неправильна топологія

Наступна задача з котрою зіштовхується розробник – це обмежена кількість полігонів для низько-полігональнаї моделі (Low Poly). Низько-полігональна модель повинна мати максимальну деталізацію з мінімальною

кількістю полігонів, так як менше полігонів тим швидше пристрій буде обробляти кожну вершину моделі. Саме тому розробник повинен завжди слідкувати за кожним полігоном і необхідно розуміти як правильно розподіляти полікаунт (рис. 1.8).

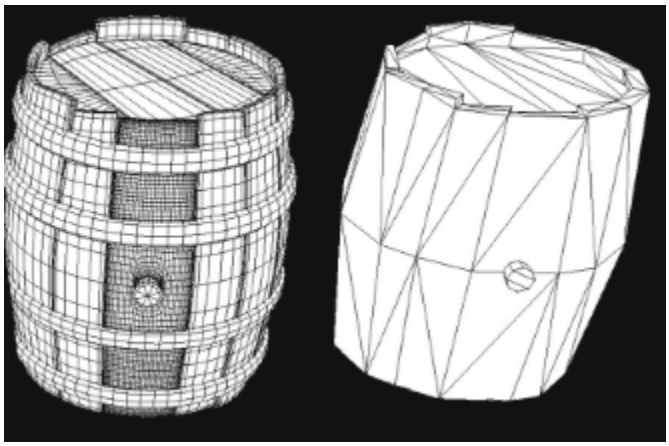


Рисунок 1.8 – Різниця кількості полігонів одного об’єкта
(зліва – неправильно, справа – правильно)

Далі по звершенню розробки Low-Poly, художник створює розгортку (UV). Чітких правил при створенні розгортки не існує але треба завжди пам`ятати, що від якості розгортки залежить майбутні текстури та карти.

Спочатку художник просто розрізає кожну деталь, тобто створює шви між шелами. Шви повинні надрізатися в тих місцях, де гравець не може побачити розрив розгортки, а значить внизу або в інших складних місцях, до яких гравець не добереться, – це робиться ще для того, якщо текстура буде в маленькому розмірі, наприклад, 512×512 пікселів то при таких обмеженнях текстура стає не такою чіткою і більш мутною, через це шов може відображатись ще сильніше. Після розробник виконує вирівнювання шелів, тобто кожне ребро повинно бути чітко вертикальними або горизонтальним-це робиться аби текстура на краях шелів не мала піксельних зубчиків. Якщо не вирівнювати кожне ребро на шелах то при запіканні карт, текстури на краях будуть мати нерівності – це псує якість відображення текстур у грі.

Після нарізання та вирівнювання робиться оверлап – це створюється для того, щоб заощадити місце на UV але оверлапи не завжди потрібні, якщо художник буде текстурити кожний об`єкт як унікальний тоді оверлап не використовуються. За допомогою оверлапів можно збільшити щільність пікселів на текстурі але накладені шели будуть мати однакову текстуру. Перед упаковкою розгротки, розробник масштабує шели до меншого розміру, а деякі шели до більшого розміру – це виконується для того, щоб збільшити щільність пікселів на дрібних деталях, при масштабуванні шели маленьких об’єктів будуть мати більше площу на розгортці, а отже матимуть більше пікселів на шелах, тому якість зображення на дрібних місцях буде якісніша (рис. 1.9). Така сама процедура виконується для місць, котрі гравець дуже рідко або ніколи не побачить, наприклад, дно автомобіля але такі шели навпаки максимально зменшують, такий підхід до різниці щільності пікселів створює новий простір для пакування в координатах від 0 до 1 – це називається Texel Density.

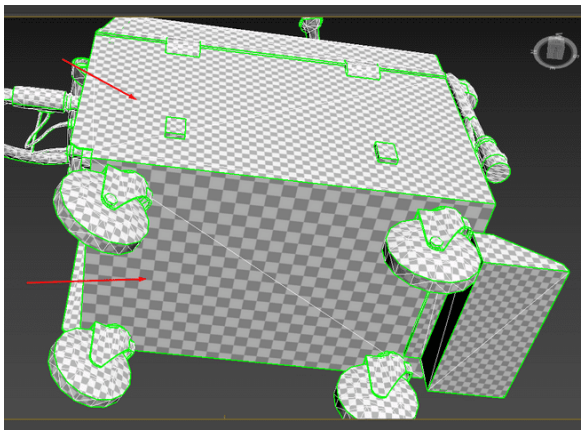


Рисунок 1.9 – Приклад різниці Texel Density

Після всіх етапів – робиться упаковка розгортки, її можна робити як вручну так і автоматично. Ручний спосіб більш нудний та займає більше часу але художник після упаковки легше орієнтується на розміщенні кожного шела. Більшість художників пакують розгортку автоматично – це заощаджує час та сучасні інструменти для упаковки роблять це якісно. При пакуванні

розгортки необхідно слідкувати за відстанню між шелами – це називається падінг. Не дозволяється аби шели накладались один на одного хоча б одним пікселем, якщо це не оверлап.

Створення оверлапів заощаджує місце на розгортці в координатах від 0 до 1 та також оверлап створюють для того, якщо модель має однакові об’єкти, наприклад, колеса зварювального апарату то шели одного колеса будуть знаходиться в координатах розгортки, а решта три чітко накладаються піксель в піксель один на одного та всі шели інших трьох коліс відсуваються на одну координату від основної розгортки. Відсування виконується аби запобігти артефактів та помилок при запікання Ambient Occlusion та Normal Map щоби програма розуміла на який шел запікати тінь або фаски (рис.1.10).

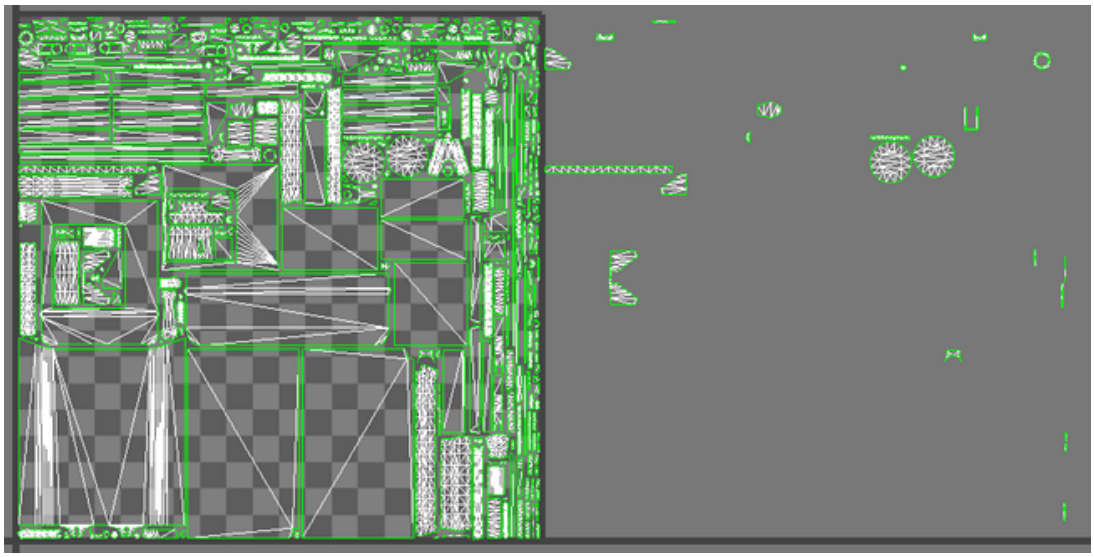


Рисунок 1.10 – Розгортка з оверлапами

Відстань між кусками розгортки залежить від розміру текстури:

– текстура 4096×4096 має відстань від 16 до 32 пікселів;

– текстура 2048×2048 має відстань від 8 до 16 пікселів.

Далі під час розробки карт для текстурування (рис. 1.11-1.14), а саме Normal Map, карту тіней (ambient occlusion), curvature – це основні карти, які допомагають при текстуруванні, можуть виникнути помилки при запіканні карт текстур. Для того щоб запобігти проблем необхідно аби низько-

полігональна модель п ідеально співпадала з формами з високо-полігональною моделлю. Якщо форми обох моделей не будуть співпадати, то надалі при запіканні карт будуть великі проблеми та артефакти, котрі довго видаляти. Також через це розробник втрачає час на розробку моделі тому , що він повертається на один крок назад щоб форми низько-полігональнаї моделі співпадали з високо-полігональною моделлю. Після запікання у кінці розробник може побачити як Low-Poly модель отримала деталізацію від High-Poly, та без додаткових полігонів. Все це робиться для оптимізації, бо в грі таких моделей може бути тисячі і всі вони різного розміру та полігонажу. Далі розробник створює карту тіней та курватуру, вони створюються таким же чином як і Normal Map [4].

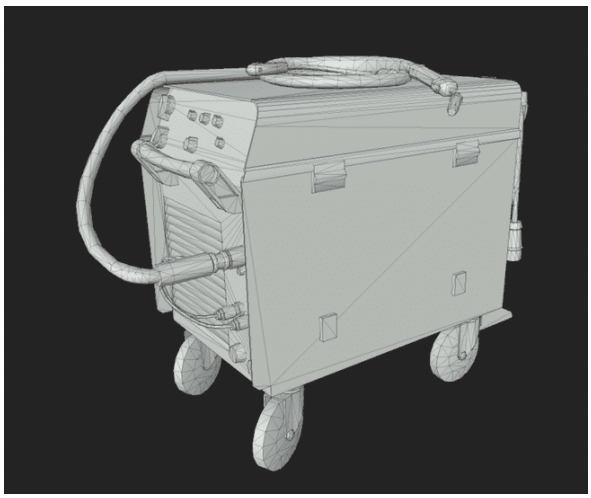


Рисунок 1.11 – Low-Poly модель без Normal Map

Рисунок 1.12 – Low-Poly модель з Normal Map

Рисунок 1.13 – Ambient Occlusion карта на моделі

Рисунок 1.14 – Карта Curvature на моделі

Останній та самий відповідальний крок розробки моделі – це створення текстур. Головна задача під час текстурингу – це передати історію моделі та розробити модель під вимоги ігрового рушія бо далі команда програмістів створює шейдера для покращення освітлення, відображення фізичних властивостей текстур. Також суворо забороняється використовувати на текстурах повністю чорний та білий колір, бо в реальному житті таких кольорів не існує і в рушію ці кольори будуть некоректно відображатись. До

таких суворих вимог відноситься канал металік, рафнес та базовий колір, не дозволяється робити рафнес та металік в значеннях 1 або 0.

Для розуміння, які максимальні значення дозволяються під окремий шейдер – використовується інструмент валідатор, він відображає правильні зони значень на текстурі, якщо якась зона підсвічуються червоним, то значення неправильні, якщо зелений – все добре. Це необхідно для розуміння, де трапилась помилка, на якому каналі (рафнес, металік або колір). Валідатор має свої налаштування, по яким можна обирати відповідні канали [5].

2 АНАЛІЗ СПЕЦИФІКИ PHYSICAL BASED RENDERING

2.1 Вимоги PBR в ігровій індустрії

Physically-Based Rendering або фізично коректний шейдинг – це метод рендерингу, який дає більш точне уявлення про те, як світло взаємодіє з властивостями матеріалу.

Досягення фотореалізму у графіці полягає у створенні особливо виразних зображень, які виглядають як фотографії. Подібним чином, PBR забезпечує точне відображення взаємодії світла з об’єктами. Це може покращити відображення. Тому коли користувач помічає щось, що виглядає несправжнім, їх увага більше не прикута до сюжету гри.

Світло є складним явищем, оскільки воно може проявляти властивості як хвилі, так і частинки. У результаті були створені різні моделі для опису його поведінки. Для текстур, необхідна променева модель світла, оскільки вона описує взаємодію світла та матеріалу. Розуміння того, як світлові промені взаємодіють із поверхневою речовиною, є важливим, оскільки робота розробника полягає у створенні текстур, які описують поверхню (рис. 2.1). Текстури та матеріали взаємодіють зі світлом тривимірних віртуальних сценах. Променева модель світла стверджує, що світловий промінь має траєкторію прямої лінії в однорідному прозорому середовищі, такому як повітря. Променева модель також говорить, що промінь поводитиметься передбачуваним чином при зустрічі з поверхнями, такими як непрозорі об’єкти, або при проходженні через інше середовище, наприклад з повітря у воду. Це дає змогу візуалізувати шлях світлового променя, коли він рухається від початкової точки до іншої точки, де він перетворюється на іншу форму енергії, наприклад тепло. Світловий промінь, який потрапляє на поверхню, називається падаючим променем, а кут, під яким він падає, називається кутом падіння [6].

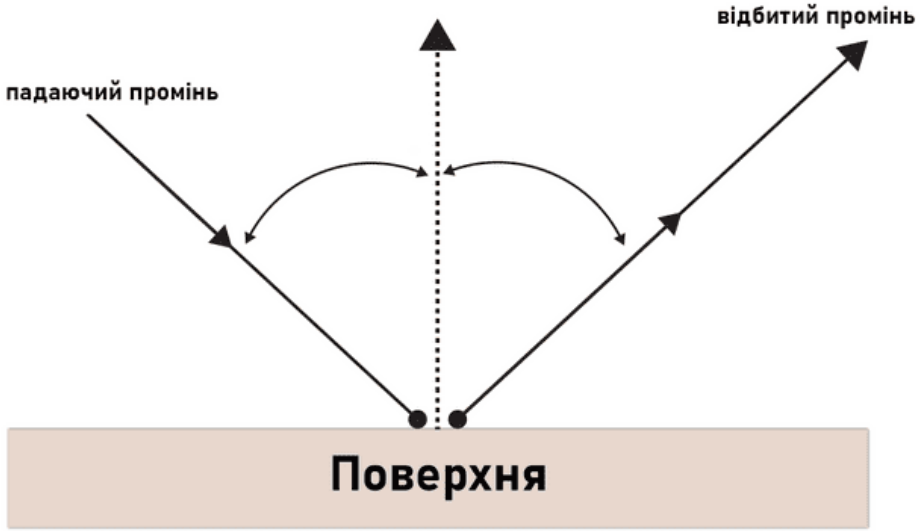


Рисунок 2.1 – Схема падаючого променя на поверхню

На поверхні промені світла розщеплюються на два напрямки: відбивання і заломлення. Також промінь або відбивається, або заломлюється, і в кінцевому підсумку може бути поглинений будь-яким середовищем. Однак поглинання не відбувається на поверхні матеріалу.

Specular Reflection – дзеркальне відображення світла, яке було відбито від поверхні (рис. 2.2). Світловий промінь відбивається від поверхні і рухається в іншому напрямку. Він відповідає закону відбиття, який стверджує, що на абсолютно плоскій поверхні кут відбиття дорівнює куту падіння. Однак більшість поверхонь нерівні, і напрямок відбитого світла змінюватиметься випадковим чином залежно від шорсткості поверхні. Це змінює напрямок світла, але інтенсивність світла залишається постійною. На шорсткіших поверхнях відблиски будуть більшими та тьмянішими. Гладкіші поверхні зосередять дзеркальні відображення, і вони виглядатимуть яскравішими або інтенсивнішими, якщо дивитися під відповідним кутом. Однак однакова загальна кількість світла відбивається в обох випадках.

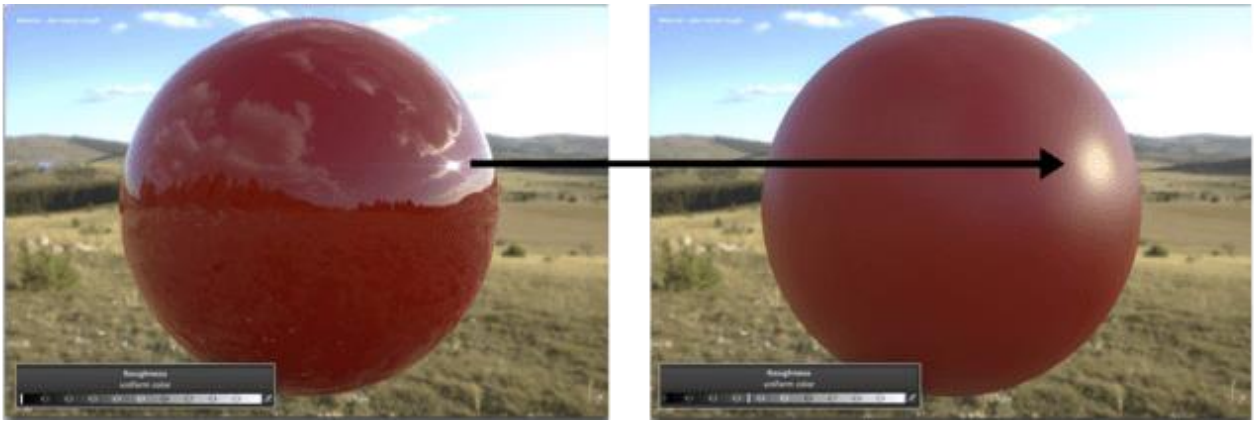


Рисунок 2.2 – Залежність відблиску об’єкту від жорсткості матеріалу

Reflection – зміна напрямку світлового променю (рис. 2.3). Коли світло переходить з одного середовища в інше, воно змінює швидкість і напрямок. Показник заломлення, або IOR, є оптичним вимірюванням, яке описує зміну напрямку, у якому рухається світловий промінь. Значення IOR використовується для визначення того, наскільки промінь буде зігнутий, коли він проходить через одне середовище в інше. Наприклад, вода має IOR 1,33, тоді як листове скло має IOR 1,52.

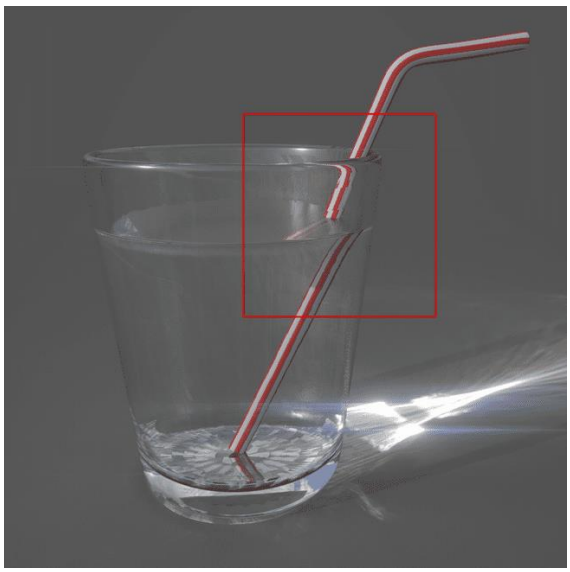


Рисунок 2.3 – Приклад заломлення

Diffuse reflection – це світло, яке заломилося. Світловий промінь переходить з одного середовища в інше. Потім світло багаторазово

розсіюється всередині об’єкта. Виходячи з об’єкта, промінь повертається до початкового середовища приблизно в тій самій точці, де він спочатку увійшов [7].

Дифузні матеріали поглинають світло (рис. 2.4). Якщо заломлене світло надто довго поширюється в такому матеріалі, воно може повністю поглинутися. Якщо світло все-таки виходить із цього матеріалу, воно, ймовірно, подолало лише дуже невелику відстань від точки входу. Тому відстань між точками входу та виходу можна вважати незначною. Модель Ламберта, яка використовується для дифузного відбиття в традиційному розумінні затінення, не враховує шорсткість поверхні. Однак інші моделі дифузного відбиття враховують цю шорсткість.

Матеріали, які мають як високе розсіювання, так і низьке поглинання, іноді називають середовищами-учасниками або напівпрозорими матеріалами. Прикладами таких є дим, молоко, шкіра, нефрит і мармур. Візуалізація останніх трьох може бути можливою за допомогою додаткового моделювання підповерхневого розсіювання, де різниця між вхідною та вихідною точкою світлового променя більше не вважається незначною. Для точного відтворення матеріалів із дуже різноманітними та дуже низькими розсіюванням і поглинанням, наприклад диму чи туману, можуть знадобитися ще дорожчі методи, котрі займають ще більше обчислювальних ресурсів пристрою та часу.

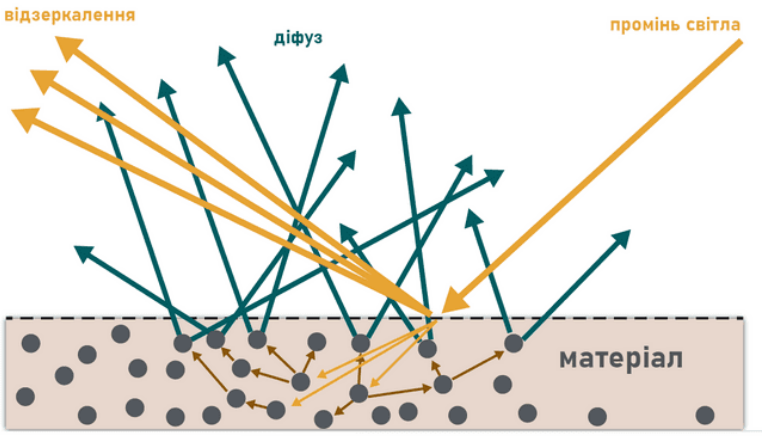


Рисунок 2.4 – Схема дифузного розсіювання світла

Дифузне і дзеркальне відображення залежать від нерівностей поверхні, де світлові промені перетинаються з середовищем. Однак, вплив шорсткості на дифузне відображення менш помітний через розсіювання, яке відбувається всередині матеріалу. У результаті вихідний напрямок променя практично не залежить від шорсткості поверхні та напрямку падіння.

Нерівності поверхні можуть мати кілька інших назв, зокрема шорсткість, гладкість, глянець або мікроповерхня, залежно від робочого процесу PBR, який використовується (рис. 2.5).

Усі ці терміни описують один і той самий аспект поверхні. Ці нерівності поверхні записуються на карті шорсткості або блиску залежно від робочого процесу, який використовується.

Нерівності поверхні на мікроскопічному рівні викликають дифузію світла. Наприклад, розмиті відблиски викликані розсіяними світловими променями. Промені не відбиваються паралельно, тому ми сприймаємо дзеркальне відображення як розмите.

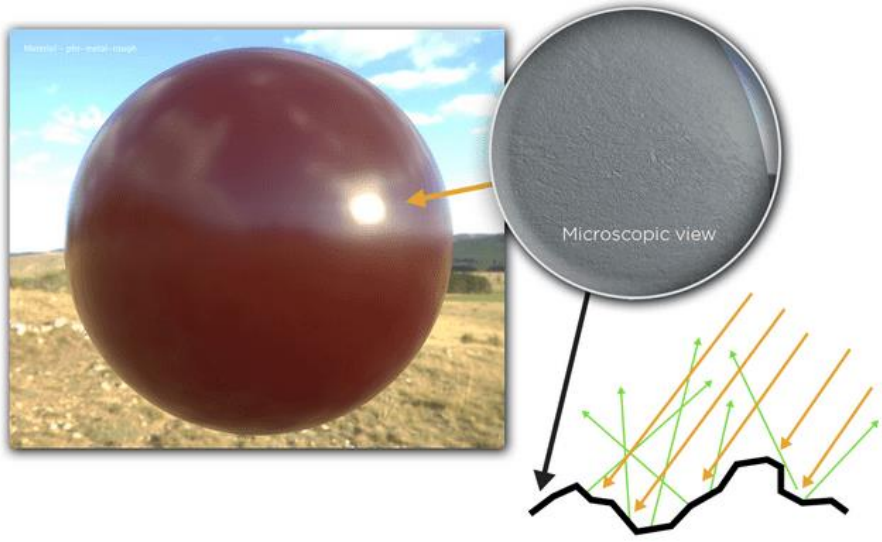


Рисунок 2.5 – Схема залежності відображення від нерівностей поверхні

2.2 Робочі процеси PBR

Фізична візуалізація (PBR) – це методологія, а не жорсткий стандарт. Є конкретні принципи та вказівки, але немає єдиного вірного правила, що

означає, що можуть бути різні реалізації. Типи карт і робочий процес можуть відрізнятися. В сучасній індустрії існують два найпоширеніші робочі процеси: metal/roughness і specular/glossiness (рис. 2.6, рис. 2.7). Набір інструментів для створення карт PBR, який складається з Substance Designer, Substance Painter підтримує обидва робочі процеси.

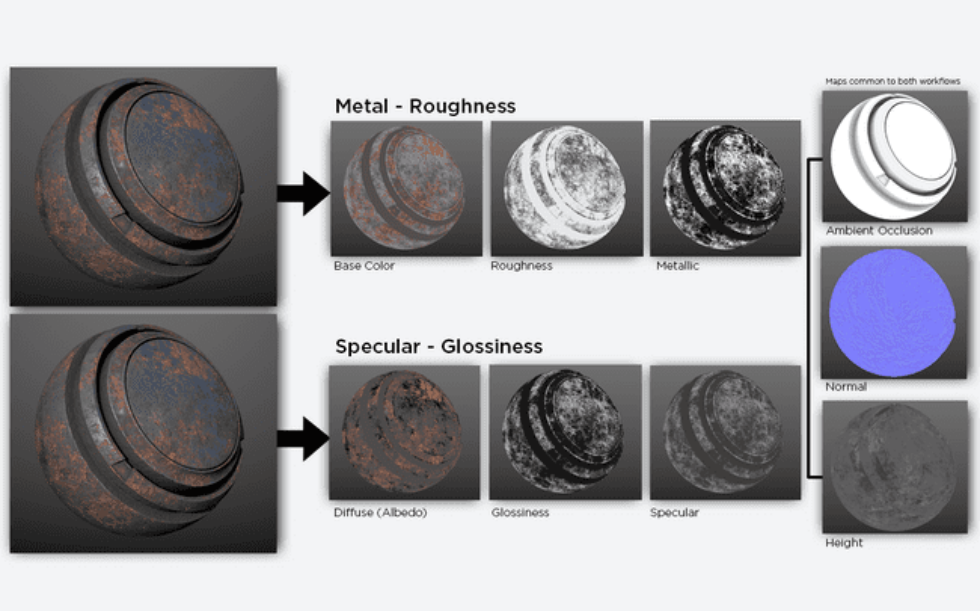


Рисунок 2.6 – Приклад робочих процесів PBR

Крім того, у наборі інструментів Substance підтримуються спеціальні шейдери, що означає, що ви можете адаптувати Substance до будь-якого налаштованого конвеєра. Хоча обидва робочі процеси мають переваги та недоліки у своїй реалізації, один не перевершує інший.

Саме базові концепції та рекомендації зроблять створені вами карти PBR точними, а не сам робочий процес.

Робочі процеси представляють ті самі дані, але вони реалізують їх по-різному. Переваги робочих процесів PBR:

– усуває здогад дзеркальності, оскільки його методологія та алгоритми базуються на фізично точних формулах. Тому легше створювати реалістичні текстури;

– тривимірні об’єкти виглядатимуть акуратно за будь-яких умов освітлення

– забезпечує робочий процес для створення узгоджених ілюстрацій, навіть між різними художниками.

Для розробників це дозволяє по-іншому думати про карти, які описують фізичні характеристики поверхні, оскільки прогрес у комп'ютерному обладнанні та рендерингу дозволяють тепер точніше імітувати фізику світла. Дозволяє викинути поняття дифузних і дзеркальних карт із традиційних робочих процесів візуалізації, оскільки ці карти служать лише обхідними шляхами для наближеної взаємодії світла з матеріалами.

Робочий процес metal/roughness визначається набором каналів, які подаються до семплера в шейдері PBR як текстури. Карти для робочого процесу металу/шорсткості – це основний колір, карта металу і шорсткість.

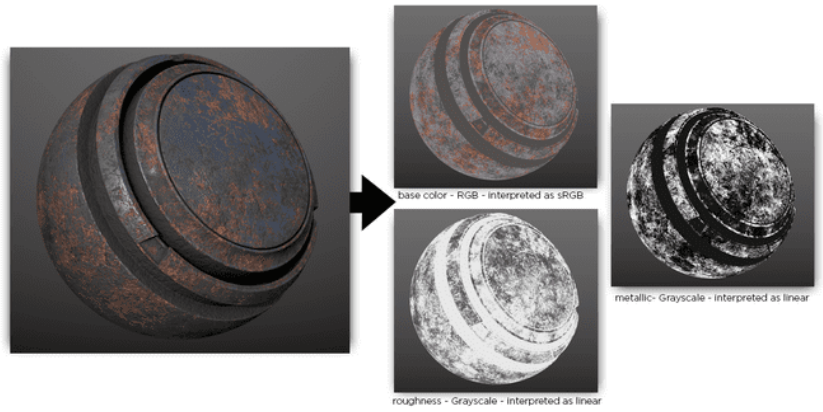


Рисунок 2.7 – Карти для metal/roughness

У робочому процесі метал/шорсткість значення відбиття для металів розміщується в базовій карті кольорів разом із відбитим кольором для діелектриків (рис.2.8).

Використовується металева карта, яка працює як маска для розрізнення даних про метал і діелектрик, знайдених у карті базових кольорів. Діелектричні значення не створюються розробником, оскільки їх обробляє шейдер. Коли шейдер бачить чорний колір на карті металу, він розглядає відповідну область на карті основного кольору як діелектрик і використовує значення коефіцієнта відбиття 4%.

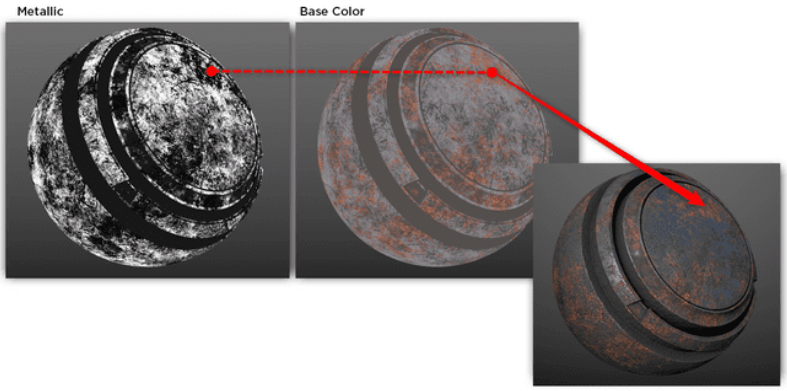


Рисунок 2.8 – Приклад вигляду карти металу та кольору

Значення 0 для звичайних діелектричних матеріалів зазвичай встановлюється на 0,04 (лінійний) 4% відбиття, так в PBR та реальному житті не існує ідеально матових або глянцевих покриттів. У робочому процесі метал/шорсткість це значення жорстко закодовано в шейдері.

Деякі реалізації металу/шорсткості, які є в наборі інструментів Substance та Unreal Engine 4 або Unreal Engine 5, мають дзеркальний контроль, який дозволяє художнику змінювати постійне значення F0 для діелектриків. У Substance цей результат позначається як «specularLevel» і надається семплером текстур у шейдері PBR metal/roughness (рис. 2.9).

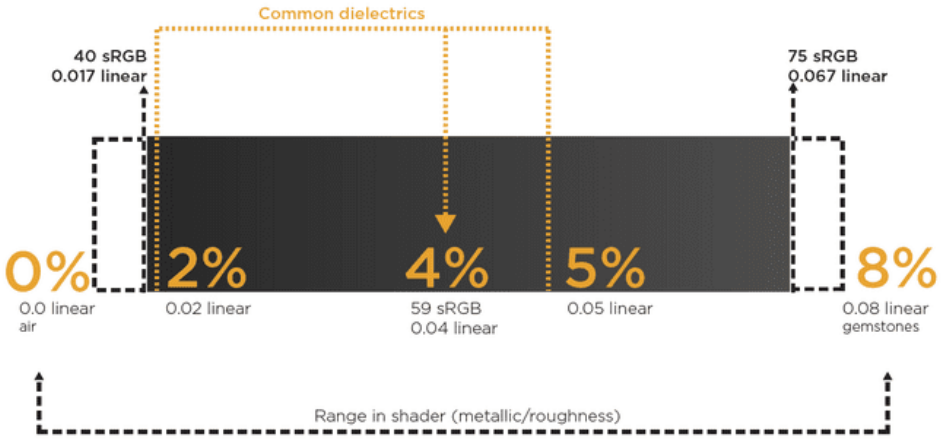


Рисунок 2.9 – Приклад допустимих значень шорсткості

Карта кольору – це RGB-карта, яка може містити 2 типи даних: дифузно відбитий колір для діелектриків і значення відбиття для металів (рис. 2.10). Колір, що представляє діелектрики, представляє відбиті довжини хвиль. Коефіцієнт відбиття значень металу позначений на металевій карті білим значенням.

Карту базового кольору вважається плоскою за тональністю. Тобто контрастність кольору нижча, ніж у традиційної дифузної карти. Небажано мати надто яскраві або надто темні значення. Об’єкти, як правило, набагато світліші за тонами. Наприклад, цей діапазон з точки зору найтемнішої речовини – вугілля, а найяскравішої – свіжого білого снігу. Вугілля темне, але не чорне 0,0. Значення кольорів, яке обирається, повинно залишатися в межах діапазону яскравості.

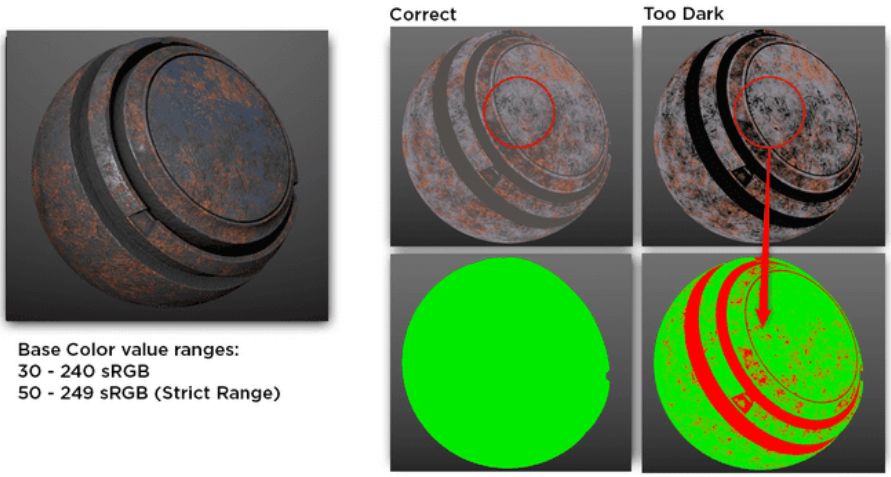


Рисунок 2.10 – Приклад допустимих значень кольору

Карта кольору містить дані для відбитого світла з точки зору діелектричних матеріалів і, отже, він повинен бути позбавлений інформації про освітлення, такої як оклюзія навколишнього середовища [8]. Можуть бути винятки для додавання мікрооккклюзії у випадках, коли шейдер не зможе представити цей рівень деталізації лише за допомогою каналу навколишньої оклюзії (рис. 2.11). Однак якщо мікрооккклюзія додається до карти, вона все ще потрібно регулювати діапазони яскравості. Значення на

карті, які вказують значення коефіцієнта відбиття для металів, повинні бути отримані з реальних виміряних значень. Ці значення складатимуть близько 70-100% віддзеркалення.

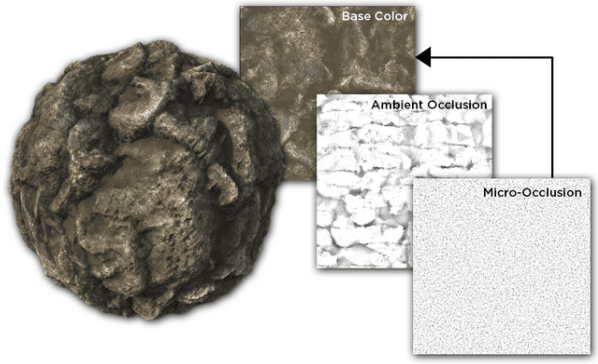


Рисунок 2.11 – Приклад додавання мікрооклюзії в карту кольору

Значення, які вказують на значення коефіцієнта відбиття для металів:

– колір представляє альбедо для неметалевих матеріалів і коефіцієнт відбиття для металів;

– темні значення не повинні бути нижчими за 30 sRGB (толерантний діапазон) до 50 sRGB (суворий діапазон);

– значення яскравості не повинні перевищувати 240 sRGB;

– коефіцієнт відображення необробленого металу буде високим у діапазоні 70-100% дзеркального зображення, яке ми можемо відобразити на 180-255 sRGB.

Недоліком використання робочого процесу метал/шорсткість є те, що він може створити артефакт білих країв (рис. 2.12).

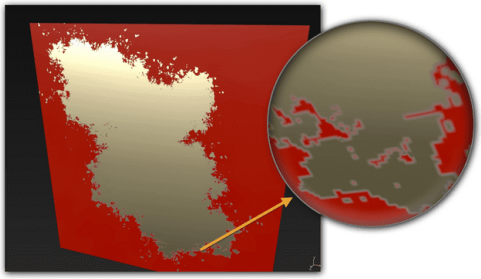


Рисунок 2.12 – Приклад артефакту білих країв

Робочий процес дзеркальності/глянцевості визначається набором карт, які подаються як текстури до семплера в шейдері PBR (рис. 2.13). Карти, специфічні для робочого процесу дзеркальності/глянцевості, — це дифузна, дзеркальна та глянцевість.

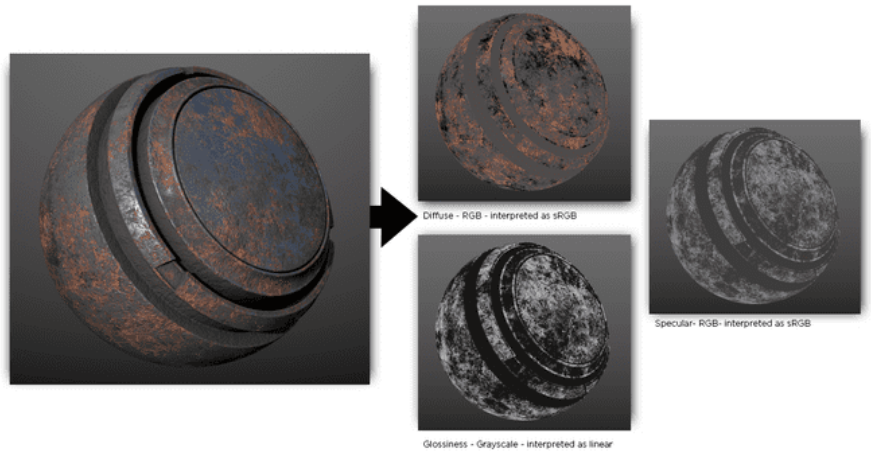


Рисунок 2.13 – Карти для specular/glossiness

Хоча робочий процес specular/glossiness використовує більш звичні назви, такі як diffuse та specular, важливо відзначити, що ці карти не такі самі, як їхні традиційні аналоги. Програма для текстурування, така як Substance Painter використовує термін дифузний, але деякі реалізації можуть посилатися на дифузний як на альбедо. Шейдер PBR також використовуватиме оклюзію навколишнього середовища, нормаль і, можливо, висоту для відображення впуклостей та опуклостей.

Дані, які представляють карти в робочому процесі specular/glossiness , такі самі, як і в робочому процесі метал/шорсткість. Тому розробники дотримуються тих самих вказівок під час створення текстур; однак різниця буде в тому, як створено карти. Дані будуть розміщені на різних картах, необхідно дотримуватися тих самих принципів. Як зазначалося раніше, усі значення, такі як діелектрик, коефіцієнт відбиття металу та діапазони яскравості для кольору альбедо, отримані з фактичних виміряних даних.

Карта кольору в робочого процесі specular/glossiness має лише колір альбедо (рис. 2.14). Області, які вказують на метал, будуть чорними (0,0), оскільки метал не має дифузного кольору. У випадках, коли відбулося окислення, металева область буде містити колір, оскільки вона більше не розглядається як метал. Те саме стосується бруду чи інших ефектів, які створюють діелектричний шар на необробленому металі.

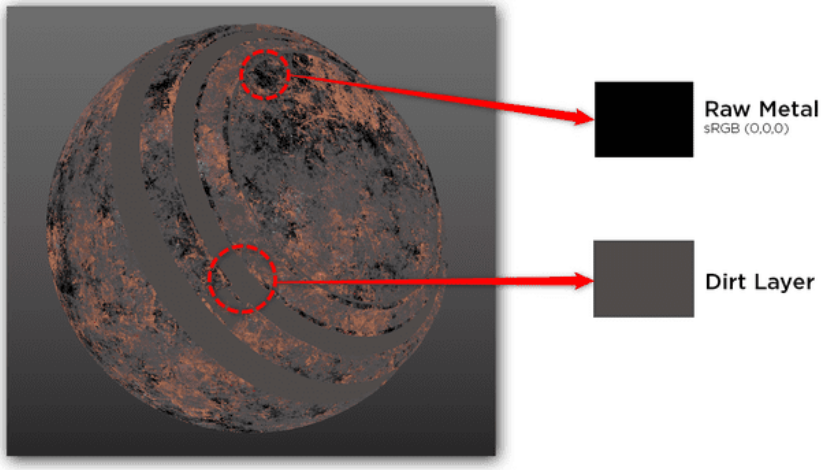


Рисунок 2.14 – Значення кольору для металу та діалектреків

Вказівки для карти кольору з точки зору тональності такі ж, як і для базової кольорової карти. Винятком є те, що якщо присутній необроблений метал, то допускається значення 0,0 (чорний);

– колір представляє альбедо для неметалевих матеріалів і чорний (0,0) для матеріалів з металу;

– основний колір не повинен містити інформації про освітлення, за винятком мікрооккклюзії;

– темні значення не повинні бути нижчими за 30 sRGB (допустимий діапазон) або 50 sRGB (суворий діапазон), за винятком випадків, коли необроблений метал є чорним;

– значення яскравості не повинні перевищувати 240 sRGB.

Карта дзеркального відображення визначає значення коефіцієнта відбиття для металу та для неметалу (рис. 2.15). Ця карта допускає різні значення для діелектричних матеріалів, які можна створити на карті. Це відрізняється від робочого процесу метал/шорсткість, де діелектрики жорстко закодовані з коефіцієнтом відбиття 4% і можуть бути змінені лише через канал «specularLevel». Так само, як в робочому процесі метал/шорсткість, дані F0 повинні бути отримані з реальних виміряних значень. F0 для діелектриків буде більш темним значенням. Коефіцієнт відбиття металу може бути забарвлений, оскільки деякі метали поглинають світло на різних довжинах хвиль. Значення F0 як для діелектриків, так і для металів визначено в дзеркальній карті RGB.

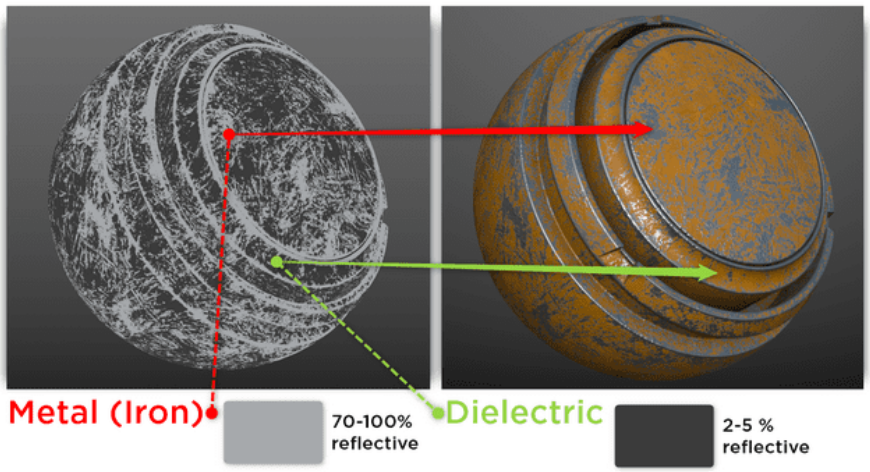


Рисунок 2.15 – Карта дзеркальності

Оскільки дзеркальна карта містить значення F0 як для металів, так і для неметалів, ми розділимо карту на окрему категорію для кожного типу матеріалу. Значення F0 має базуватися на реальних даних. Коефіцієнт відбиття необробленого металу потрібно знизити, якщо є окислення або якийсь шар, який вказує на неметал. У випадку робочого процесу віддзеркалення/глянцевості бруд або окислення підвищить дифузний колір необробленого металу на карті дифузії та знизить значення відбиття на карті

віддзеркалення. Бруд на дзеркальній карті містить відповідне значення F0 для діелектрика. У цьому випадку використовується 0,04 або 4% (рис. 2.16).

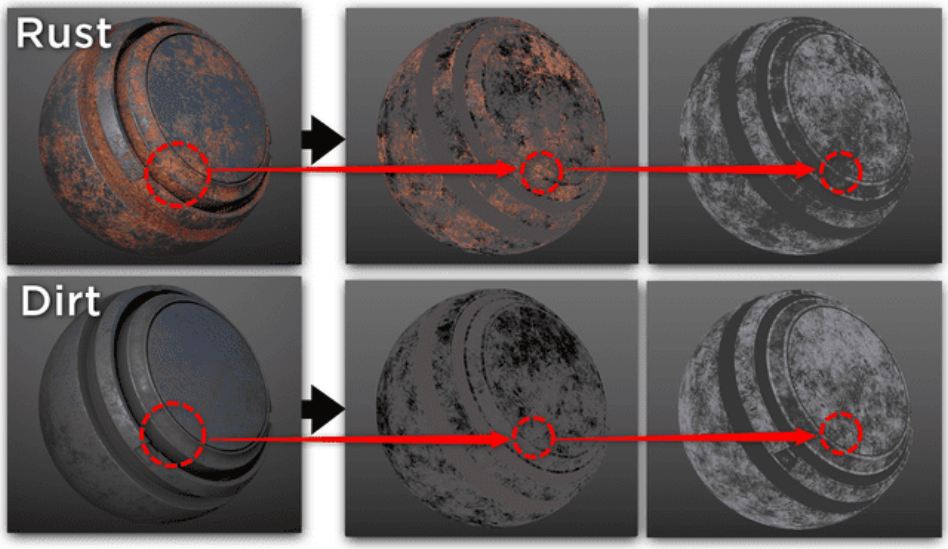


Рисунок 2.16 – Колір карти дзеркальності в діалектриках та металах

Перевагою використання робочого процесу дзеркальності/глянцевості є те, що він не має артефакти білих країв (рис. 2.17). Порівняння робочих процесів PBR наведено у табл. 2.1.

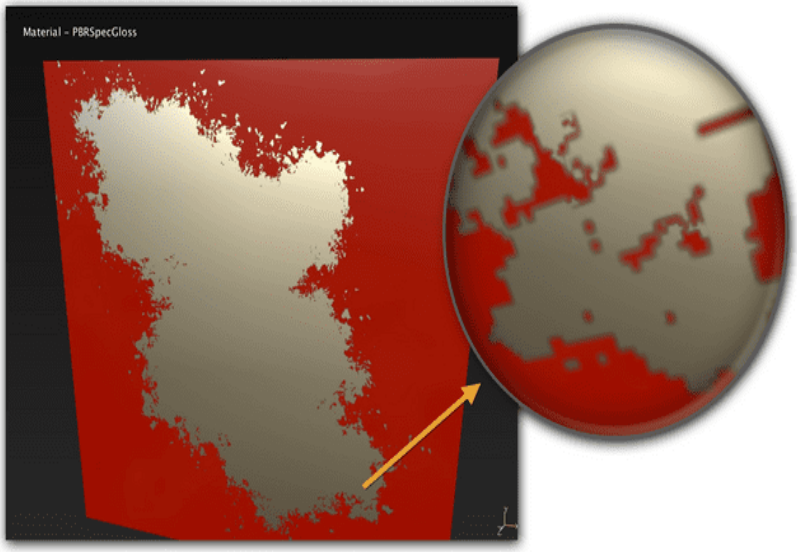


Рисунок 2.17 – Відсутність артефакту білих країв

Таблиця 2.1 – Порівняльна таблиця робочих процесів PBR

Порівняння	Наявність артефактів	Простота в використанні	Оптимізація пакування карт	Художня зручність для розробників
spec/gloss	Відсутні	Більш складний для використання	Неможливо пакувати канали в один атлас	Більш гнучкий для змішування кольорів
metal/rough	Присутні	Легкий та логічний в використанні	Можлива упаковка карт в один атлас	Незручний для змішування кольорів

В розділі було визначено, що робочі процеси PBR широко використовуються в сучасній геймдев індустрії. Проте кожен з процесів має свої недоліки та переваги в використанні під час розробки текстур та матеріалів, тому обидва методи необхідно використовувати для заздалегідь заготовлених шейдерів в ігрових рушіях. Наприклад, metal/roughness використовується більше для рушія Unreal Engine, а specular/glossiness для рушія IW Engine – ігровий рушій від компанії Infinity Ward.

3 АНАЛІЗ МЕТОДІВ 3D-МОДЕЛЮВАННЯ ДЛЯ ААА ВІДЕОІГОР

3.1 Subdiv-моделінг

Subdiv-моделінг – старий та класичний метод моделювання при якому розробик створює модель з додатковими сапорт лупами для отримання правильної форми моделі. При нульовому дивайді модель має гострі ребра, які закріплені сапорт-лупами по периметру [9]. Для згладжування застосовується інструмент subdiv. Існує популярне правило: модель має досягти потрібної гладкості після застосування двох сабдів (рис. 3.1).

Переваги методу:

– високий контроль за формою. Практично жодних проблем із внесенням правок. Просто видаляти зайві полігони, підчищати та перебудовувати топологію;

– можливість швидко виправити геометрію. Буквально за хвилину можна повністю прибирати одну деталь та перебудувати іншу. Сам метод побудови топології дозволяє це легко зробити;

– high poly є основою для low poly. Сабдив дозволяє взяти нульовий дивайд з гострими гранями, дублювати меш, почистити всі сапорти, видалити зайві трикутники та вершини та таким чином отримати готову low poly модель. Нічого не потрібно перебудовувати заново. High poly та low poly створені з одного меша;

– популярність використання сабдів-моделінгу у пайплайнах студій. Сабдів використовується не тільки в геймдеві, а й у кінопродакшні, рекламі, 3D-друкі та ін. Класичний сабдів-моделлер – універсальний боєць на ринку.

Недоліки методу:

– швидкість створення об'єктів. Робота методом сабдіва – досить тривалий процес. У комерційному продакшні це не завжди можна;

– важко створювати складні форми. Не кожен художник-моделлер зможе створювати складні форми цим способом. А в тих, хто може це займає багато часу;

– чищення high poly під low poly займає багато часу. Чистка сабдива може зайняти стільки ж часу, скільки і створення high poly. В ідеалі на створення low poly має йти втричі менше часу;

– сабдів вимагає багато часу для вивчення. Щоб стати досвідченим користувачем такого методу моделювання, потрібно щонайменше два-три роки практики.

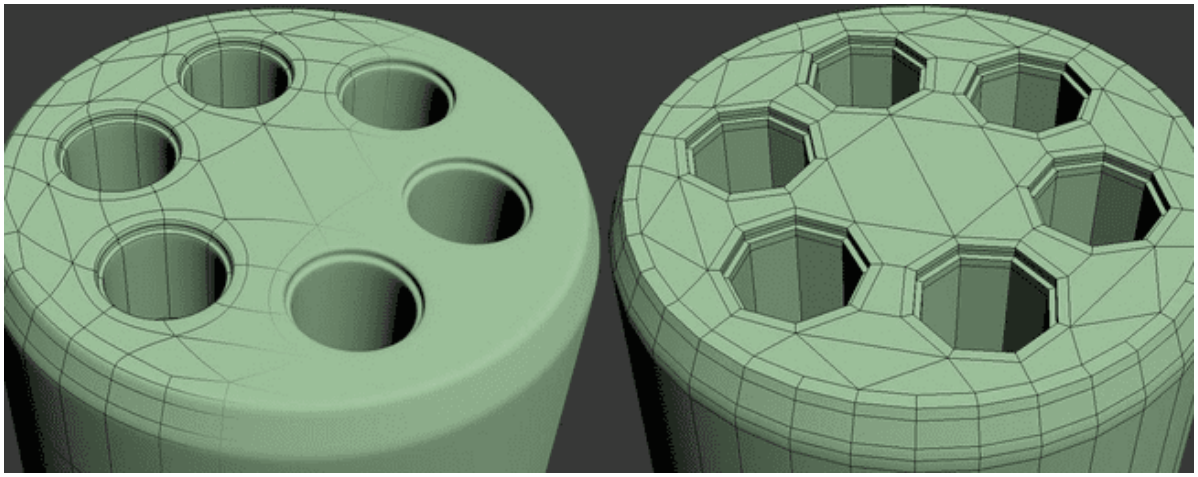


Рисунок 3.1 – Приклад Subdiv-моделінгу

3.2 CAD моделювання

Модель створюється із поверхонь NURBS. Метод дозволяє легко досягти потрібної плавності кіл за рахунок використання кривих третього, п'ятого порядку. У кіновиробництві розробникам не дозволяють використовувати CAD, але в іграх - це дуже поширений спосіб (рис. 3.2).

Перевага методу:

– простота та швидкість у створенні складних форм. Виконати булеві операції та з'єднати два об’єкти можна, зробивши буквально два рухи та натиснувши на одну кнопку. В результаті між об’єктами вийде плавний

перехід (фаска). Для того щоб зробити те саме в Subdiv-моделінгу, потрібно докласти в кілька разів більше зусиль та часу.

Недоліки методу:

– складність редагування після переведення моделі до полігонів. Моделі, що складаються з поверхонь NURBS, не сприймаються ігровими рушіями та софтом для полігонального моделювання. Щоб експортувати таку модель, її потрібно конвертувати до полігонів – усередині будь-якого CAD софту є необхідний функціонал. У процесі модель буде розділена на велику кількість абсолютно рандомних трикутників. Це ускладнює роботу з сіткою та залишає гнучкість редагування форми. Щоб виправити таку хаотичну сітку моделі потрібно повернутися до моделі CAD та внести зміни при конвертації та знову експортувати;

– низький контроль над низьким low poly. При експорті моделі з CAD не можна задати необхідну кількість полігонів та граней. І тим більше не можна задавати параметри для окремих частин моделі – все буде хаотично. Після CAD модель завжди доведеться редагувати вручну.

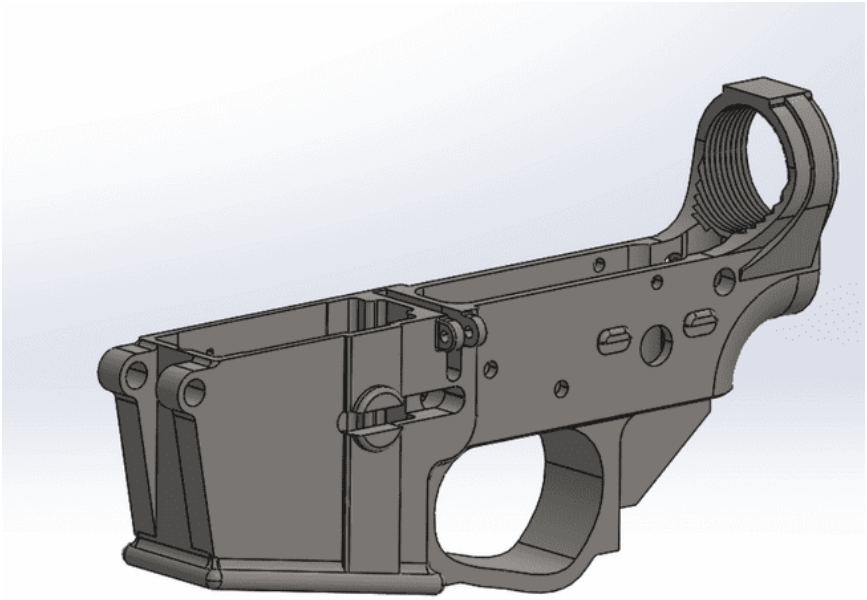


Рисунок 3.2 – Приклад CAD-моделінгу

3.3 Boolean моделювання

Болванка моделі та об'єкти для булевих операцій створюються в Maya, 3D MAX або Blender. Булеві операції проводяться в ZBrush за допомогою інструмента Live Boolean. Для згладжування фасок між з’єднаними об’єктами використовується Dynamesh та Polish, після чого до моделі застосовується Decimate (рис. 3.3).

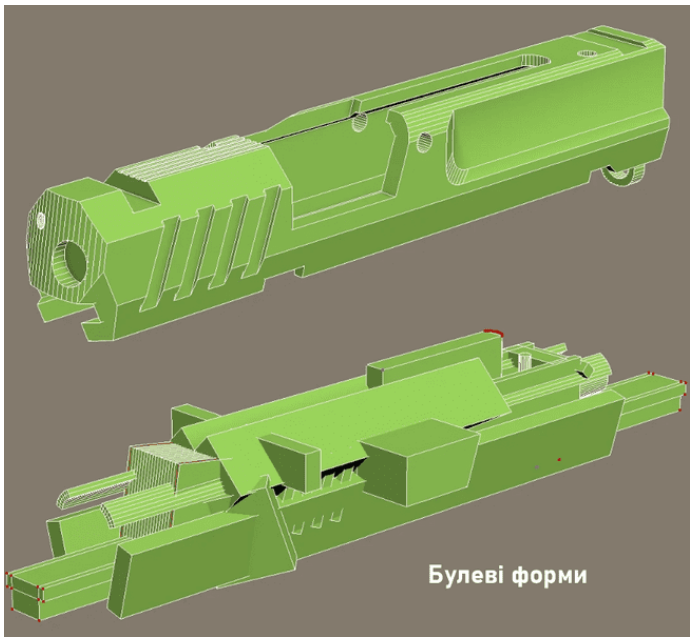


Рисунок 3.3 – Приклад булевих форм та болванки моделі

Переваги методу:

– відносна простота. Цей метод простий в виконанні складних форм;

– висока швидкість створення складних форм. Не потрібно нічого промоделювати вручну – просто поєднали заготовки, провели булеві операції, застосували Dynamesh;

– фактично не потрібен час створення low poly. Зайві полігони вичищаються дуже швидко, порівняно з моделлю, зробленою під сабдів. У цьому методі high poly і low poly створюються з однієї і тієї ж болванки, що дозволяє уникнути етапу ретопології.

Недоліки методу:

– складність редагування сітки high poly. Після бульової операції та експорту з ZBrush не можна редагувати сітку. Так, є можливість відкотитися для редакції назад у ZBrush, але це незручно і займає додатковий час;

– рівномірні фаски по всій довжині. Модель із рівномірними фасками втрачає реалістичність та зчитується як 3D. Вирішити це просто.

3.4 Скульптинг моделювання

Скульптинг – метод цифрової ліпнини в тривимірному просторі. За допомогою скульптингу можна швидко створювати форми органіки не звертаючи уваги на топологію (рис. 3.4).

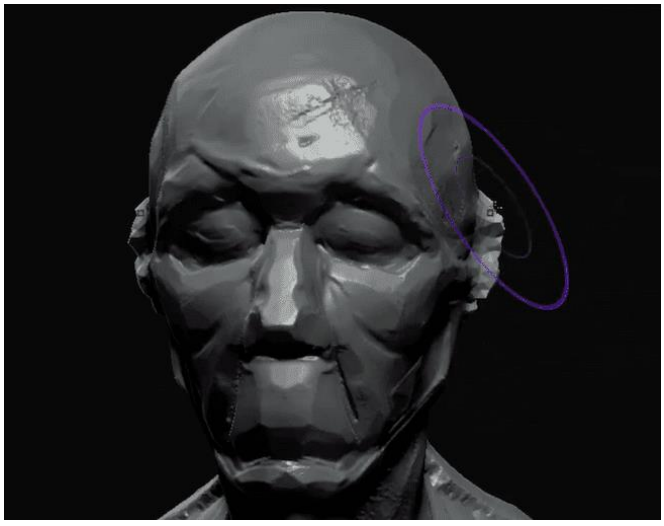


Рисунок 3.4 – Приклад скульптингу моделі

Для такого методу моделювання в індустрії використовують Zbrush, саме ця програма надає легкості та зручності створення персонажів, тварин – це ідеальне програмне забезпечення для таких задач.

Zbrush має доволі не складний інтерфейс для використання, так як найголовніший інструмент в програмі – це пензлики, котрі мають свої характеристики для моделювання [12].

Переваги методу:

– займає менше часу розробки, через вільне керування при створенні моделі;

– під час розробки моделі методом скульптингу не обов'язково слідкувати за топологією.

Недоліки методу:

– неможливість ретельно слідкувати за кількістю полігонів;

– складно створювати ідеально рівні площини на моделі.

Порівняння методів моделювання наведено табл. 3.1.

Таблиця 3.1 – Порівняльна таблиця методів моделювання

Порівняння	Метод	Швидкість та ресурс-потрібність	Для яких об'єктів призначено	Топологія
Subdiv	Полігональне моделювання	Створення моделей займає багато часу	Hard Surface	Необхідно слідкувати за сіткою та її рівномірністю
CAD	NURBS, криві	Не займає багато часу але витрачає багато ресурсів	Hard Surface	Топологія не підлягає редагуванню
Boolean	Відрізання форм	Швидкий метод створення форм моделі	Hard Surface	Створюється багато зайвих вершин
Скульптинг	Цифрова ліпнина	Займає менше часу, через вільне керування при створенні моделі	Органіка та об'єкти навколишньої середи	Підлягає редагуванню будь-яка топологія

В розділі були визначені методи моделювання тривимірних моделей для ААА-відеоігор за сучасними вимогами в GameDev індустрії.

4 РОЗРОБКА МЕТОДИКИ СТВОРЕННЯ ТРИВИМІРНИХ МОДЕЛЕЙ

4.1 Характерні особливості пайплайнів

Для дослідження характерних особливостей алгоритмів тривимірного моделювання був розглянутий класичний пайплайн створення моделей в ААА-відеоіграх і виявлені наступні його особливості.

Даний алгоритм створення моделей займає багато часу на розробку. Також для використання класичного пайплайну необхідний великий досвід розробника, так як даний цикл розробки вимагає використовувати велику кількість інструментів [13].

Вибір за яким пайплайном буде створюватися тривимірна модель багато залежить від ролі асета в грі, його розмірів та об’ємом ресурсів котрі виділені на об’єкт. Велике значення грають габарити моделі, так як за класичним пайплайном великі об’єкти витрачали би багато ресурсів пристрою та займали багато часу на розробку. Саме для таких моделей необхідний використання Weighted Normals пайплайн [17].

Виявлено, що для великих за розміром асетів не витрачається час на розробку high poly та low poly моделей, а створюється одразу mid poly модель. Mid poly модель не має карт нормалей та інших мап, котрі необхідні для стандартного пайплану, все через те, що mid poly має геометричні фаски (рис. 4.1).

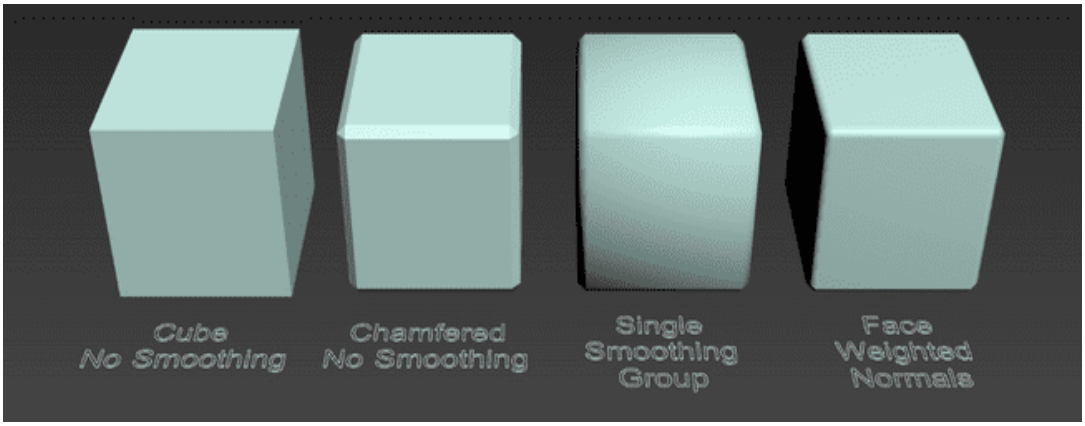


Рисунок 4.1 – Приклад вигляду моделей з геометричними фасками та без

4.2 Особливості оптимізації моделей Weighted Normals пайплайна

Було розглянуто сучасні методи оптимізації моделей, особливо оптимізація текстур, так як на всі моделі, котрі є в грі, неможливо застосувати 4к або 8к текстури. Дані розміри атласів великі за розмірами і вони потребують великих ресурсів відеопам’яті пристрою.

Для використання невеликих розмірів атласів і при тому, щоб текстури були не мильні, а чіткі, було необхідне використання mid poly моделей. На таких моделях дозволяється створювати розгортку без особливих правил упаковки в певних координатах атласу. Тому навіть при розмірі атласа 512×512 пікселей, великі шели розгортки можна класти за межі координат текстуроного атласу. Таким чином отримується тайлова текстура (рис. 4.2). На тайлову текстуру, котра виходить за межі атласу, неможливо розробити карту нормалей, як це робиться для класичного пайплану.

Так як в класичному алгоритмі кожен шел розгортки має окреме унікальне місце на текстурному атласі (рис. 4.3). Тому аби модель мала блік всіх граней (фасок) без карти нормалей, необхідно створювати mid poly модель, яка має геометричні фаски [15].

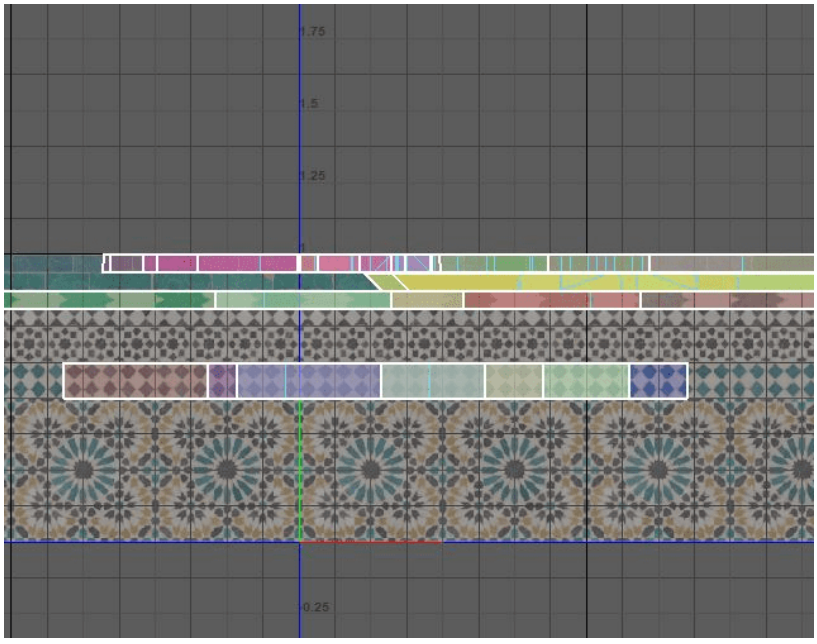


Рисунок 4.2 – Приклад розгортки mid poly моделі для тайлових текстур

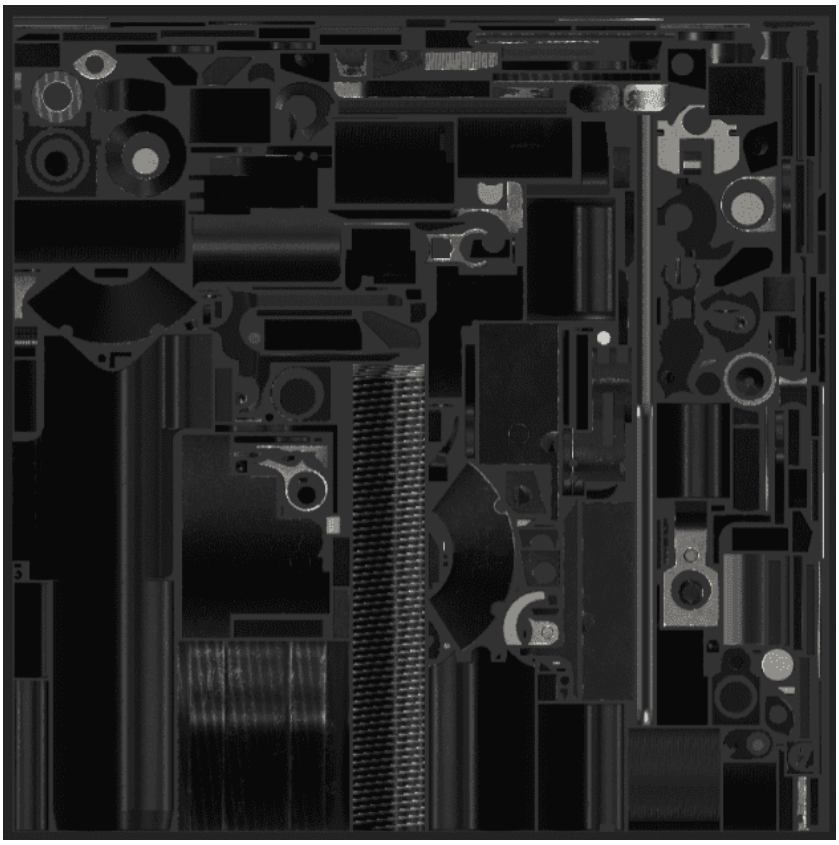


Рисунок 4.3 – Приклад розгортки low poly моделі для унікального текстурування

Основною перевагою створення mid poly моделі – це швидкість розробки асету та оптимізація розміру текстурних атласів. Наприклад, на створення звичайної mid poly моделі автівки витратиться біля 60 годин. Для тієї ж самої моделі за класичним пайплайном на розробку об’єкту витратиться біля 100 годин. Проте на розробку mid poly моделі витрачається більше полігонів, що негативно впливає швидкість обчислень пристрою.

Резюмуючи вищесказане, мною були виведені вимоги моделі для використання Weighted Normals пайплайн:

- роль моделі в грі;
- габарити моделі;
- виділений об’єм полігонів на об’єкт;
- розмір текстурного атласу для моделі.

4.3 Особливості створення моделей з Weighted Normals пайплайн

Було виявлено, що Weighted Normals пайплайн передбачає те, що об'єкт не має унікальної карти нормалей, щоб шейдинг на моделі виглядав коректно потрібно правильно робити фаски на гранях і вирівнювати вертекс нормалі, це збільшує полігонаж, тому такі моделі називають mid poly. Для розробки правильних геометричних фасок, на моделі необхідно створювати трохи більшу товщину стінок об’єкту, бо якщо початкова товщина стінки буде занадто тонка, то при створенні фасок виникають артефакти (рис. 4.4).

Також з’ясувалось, що фаски не повинні бути занадто тонкі [11]. Так як при віддаленні камери від моделі в ігрових рушіях спрацьовує компресія текстур та лодування об’єкту, тому необхідно створювати фаски трохи жирніші, щоб на відстані було помітно блік граней та зчитувався об’єм моделі (рис. 4.5).

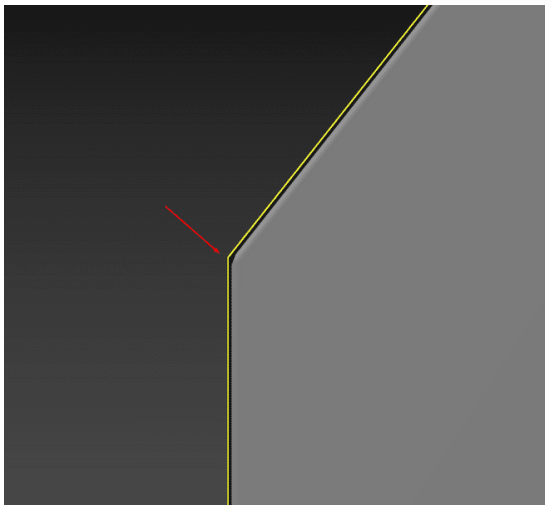


Рисунок 4.4 – Артефакт фасок при тонкій товщині стінки

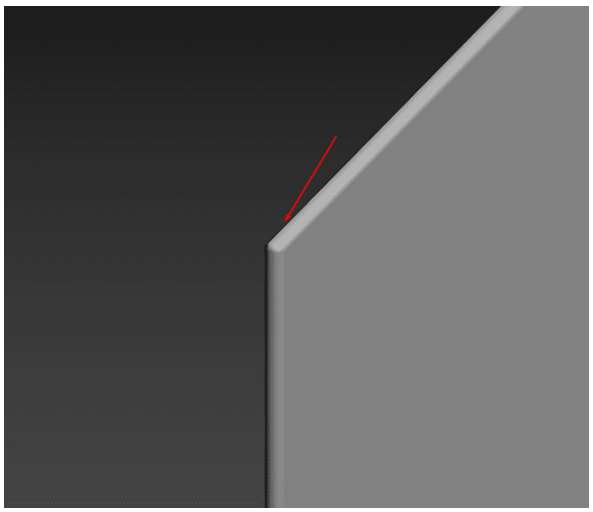


Рисунок 4.5 – Правильна товщина стінки з коректними фасками

На будь-якій тривимірній моделі необхідно створювати коректний шейдинг [16]. Проте, було виявлено, що при розробці mid poly моделі шейдинг повинен бути максимально рівний та коректний, на відміну від допустимих артефактів шейдингу на low poly моделі (рис. 4.6). Так як low

poly меш має створену карту нормалей, котра додатково вирівнює шейдинг low poly моделі (рис. 4.7, рис. 4.8), то на mid poly (рис. 4.9), шейдинг котрий створив розробник, буде мати такий же вигляд в грі (рис. 4.10).

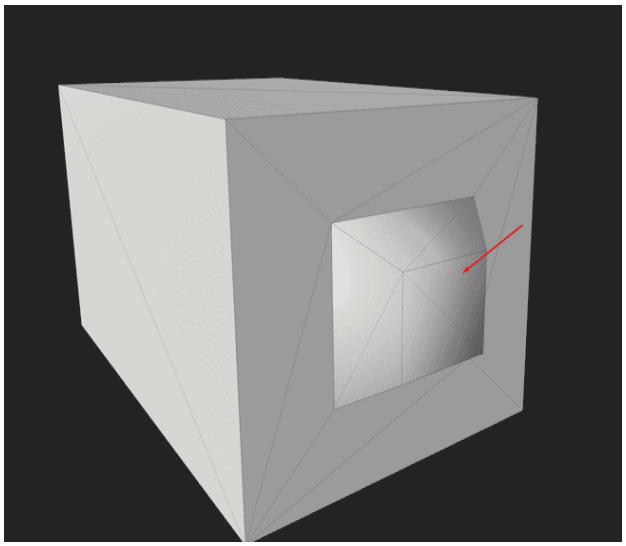


Рисунок 4.6 – Некоректний шейдинг на low poly моделі

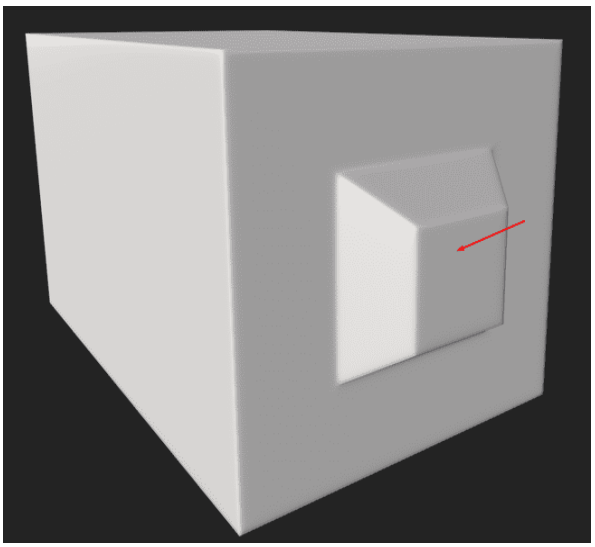


Рисунок 4.7 – Вирівнювання шейдинг на low poly моделі картою нормалей

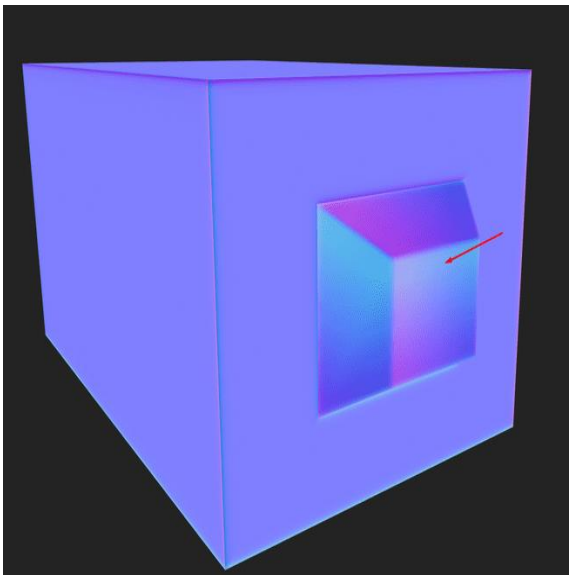


Рисунок 4.8 – Компенсація некоректного шейдингу за допомогою карти нормалей

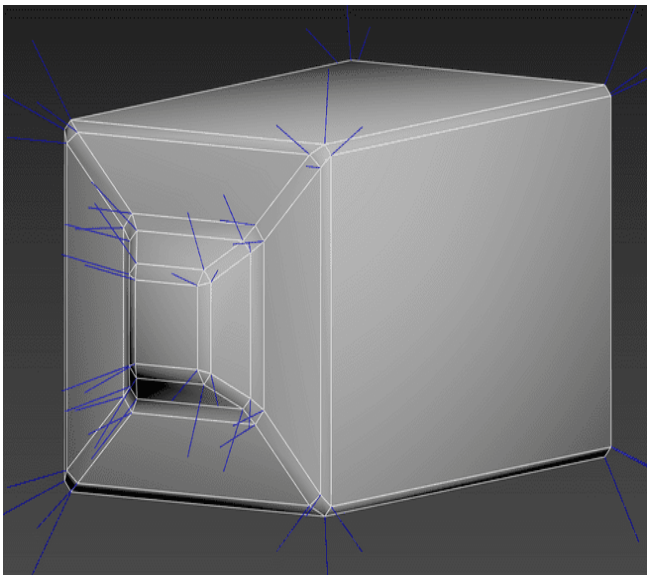


Рисунок 4.9 – Некоректний шейдинг на mid poly без вирівнювання нормалей

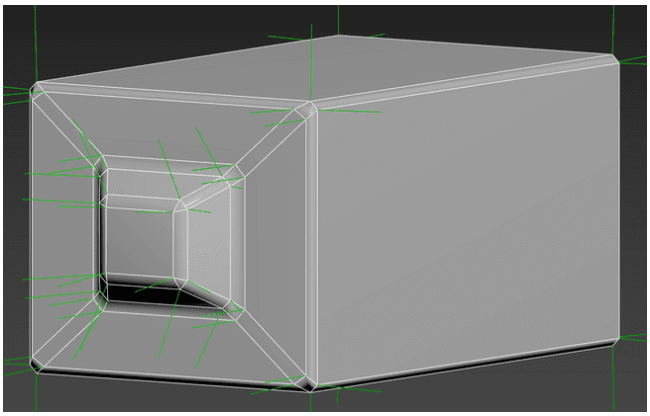


Рисунок 4.10 – Коректний шейдинг на mid poly з вирівнюванням нормалей

4.4 Особливості триангуляції сітки моделей з Weighted Normals пайплайн

Триангульована сітка – це тип редагованої створеної сітки моделі. Триангуляція необхідна для будь-якої моделі, котра буде використовуватися в грі [16].

Було з’ясовано, що триангуляція потребується для коректного шейдингу моделей в ігровому рушії. При експорті в рушій модель автоматично триангулюється, через те, що обчислювальна техніка, тобто пристрій, зчитує полігони тільки трикутного вигляду. Саме на цьому етапі розробки моделі виникають проблеми з шейдингом асету. Так як, при створенні об’єкту необхідно вручну триангулювати модель ще перед експортом в ігровий рушій аби запобігти автоматичної зворотної триангуляції (рис. 4.11).

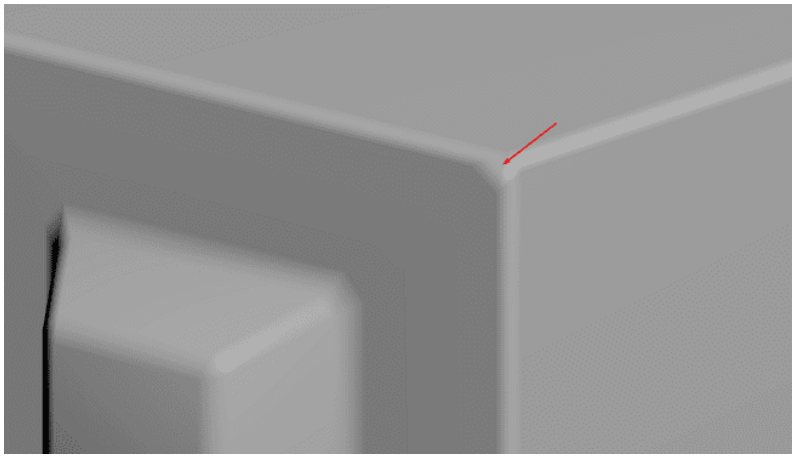


Рисунок 4.11 – Приклад зворотної триангуляції

5 АПРОБАЦІЯ МЕТОДИКИ. АНАЛІЗ РЕЗУЛЬТАТІВ

5.1 Процес створення mid poly моделі для ААА-відеоігор

У попередньому розділі були описані особливості роботи над створенням mid poly та low poly моделей з урахуванням всіх сучасних вимог індустрії до 3D арта.

Далі наведений детальніший опис дій, необхідних для отримання бажаних результатів.

Як основне програмне забезпечення для створення моделі, були вибрані програми 3D Studio MAX компанії Autodesk та Substance Painter компанії Adobe. Для створення шейдеру та тестування готової моделі використовувався ігровий рушій Unreal Engine. Це професійні пакети для тривимірної графіки, які нині широко застосовується в ігровій індустрії.

Для створення моделі, було вирішено використовувати Weighted Normals пайплайн, щоб уникнути розробки low poly, high poly моделей та усіх необхідних карт для текстурування.

Тому треба було відразу враховувати необхідність створення правильної топології, ще на етапі моделювання.

Тривимірна модель не дуже складна, так що створення правильної топології не склало великих труднощів. Важливо було звернути особливу увагу на місця стиків окремих об’єктів і створити коректно правильну сітку моделі для правильного фінального шейдингу. Для круглих деталей також потрібна ущільнена сітка з правильним розташуванням ліній уздовж основних подібних форм, це значно поліпшить загальний вигляд круглих об’єктів, наприклад, колес.

Перед початком створення моделі були зібрані референси, на котрі, при подальшій розробці спираюсь як на реальний прототип. Для референсів

підійшли будь-які фотоматеріали та креслення реального об’єкту аби якомога точно створити форми та пропорції майбутньої моделі.

Далі починаємо роботу над створенням моделі, приділяючи особливу увагу фаскам та їх товщині. Проблемні місця вирішуватимуться за допомогою додаткових полігонів, що в фінальному вигляді коригують плавність та округлість деталей (рис. 5.1).

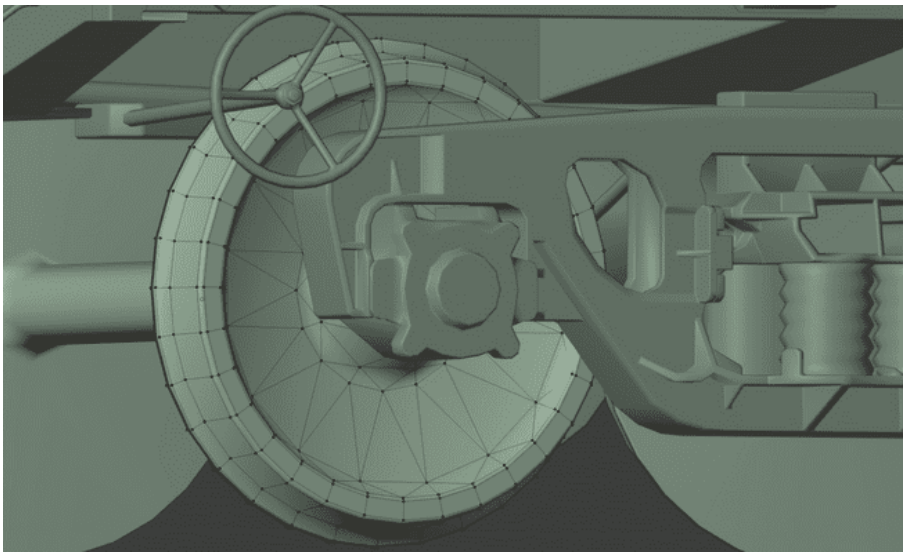


Рисунок 5.1 – Приклад додавання полігонів в округлих місцях

На даному етапі моделювання не звертається особлива увага на шейдинг моделі. Так як зараз необхідно створити точну копію реального об’єкту з мінімальною кількістю полігонів.

В цілому, дана модель це відносно простий ассет для моделювання, багато простих форм і елементів, що повторюються. Для більшості об'єктів використався звичайний моделлінг з геометричними фасками. Зв'язування інструменту Bevel – для створення фасок та модифікатора Weighted Normals – для створення гладкого шейдингу, дозволяє не перевантажуючи вихідну сітку зайвими полігонами отримувати згладжені форми, а геометрію без застосованих модифікаторів легко редагувати. З усіх частин вагона найбільшу складність становить лише візок. Все, що можна дзеркалити, дзеркалять модифікатором Symmetry, щоб не моделювати зайве, не створювати дрібних об’єктів (рис. 5.2, рис. 5.3).

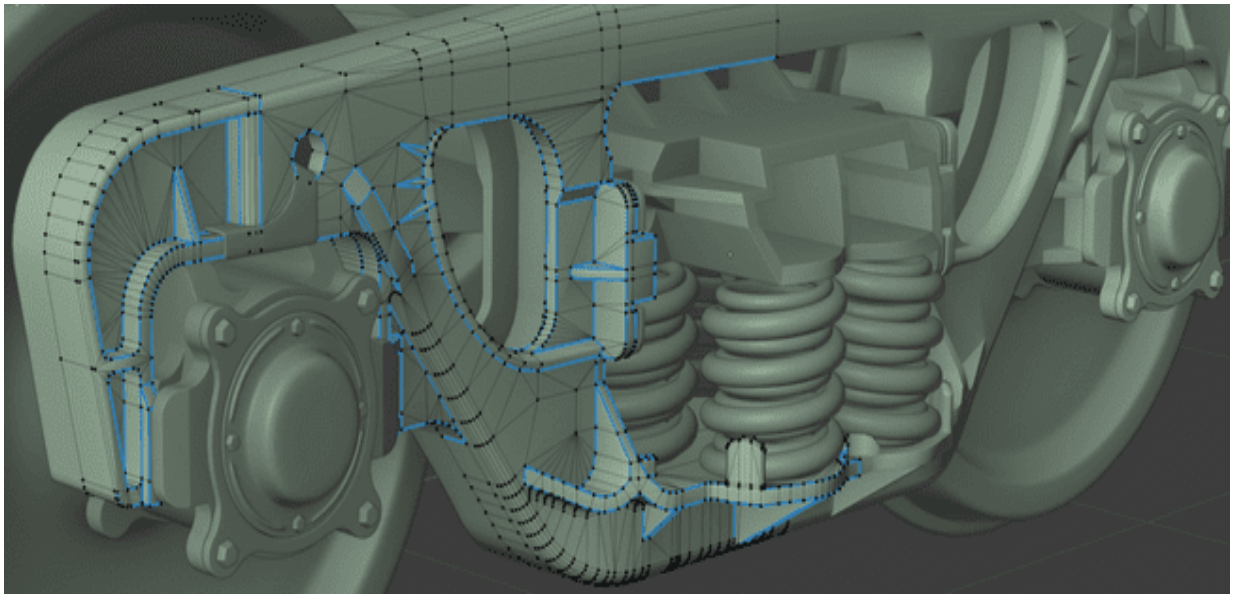


Рисунок 5.2 – Використання симетрії

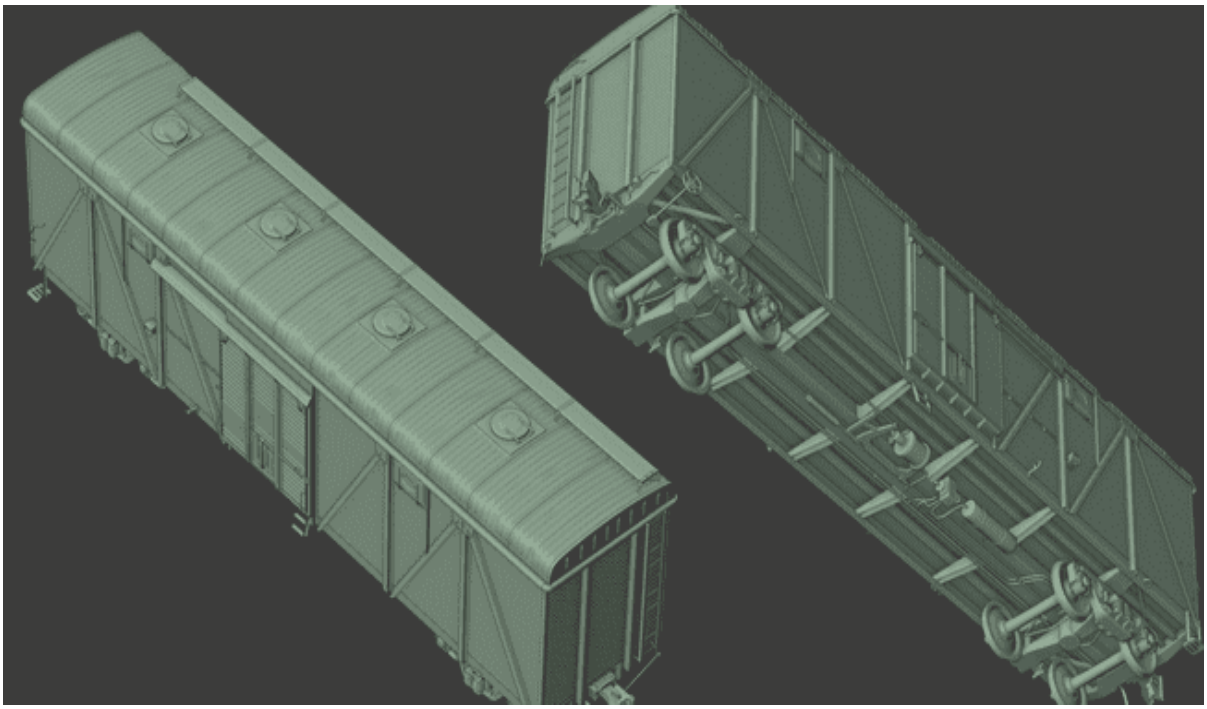


Рисунок 5.3 – Готова сітка моделі

Після завершення моделінгу можна переходити до вирівнювання нормалей, тобто створення коректного шейдингу моделі.

У 3DsMAX вирівнювати нормалі можна за допомогою модифікатора Weighted Normals, він автоматично вирівнює нормалі (рис. 5.4).

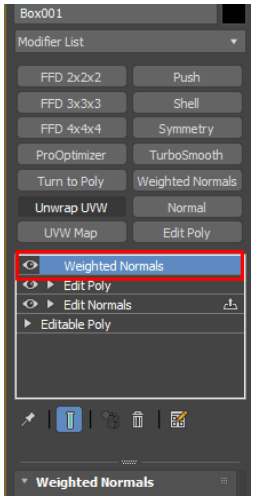


Рисунок 5.4 – Меню для використання модифікатора Weighted Normals

5.2 Використання масок

Починаємо процес створення шейдингу, для надійності можна скопіювати будь-який полігон на моделі, який дивиться в потрібну сторону і використовувати його як зразок для інших нормалей (рис. 5.5). Далі виділяємо нормалі які ми хочемо вирівняти і останнім виділяємо наш полігон зразок, використовуємо модифікатора Weighted Normals і нормалі зі зразка переносяться на виділені полігони.

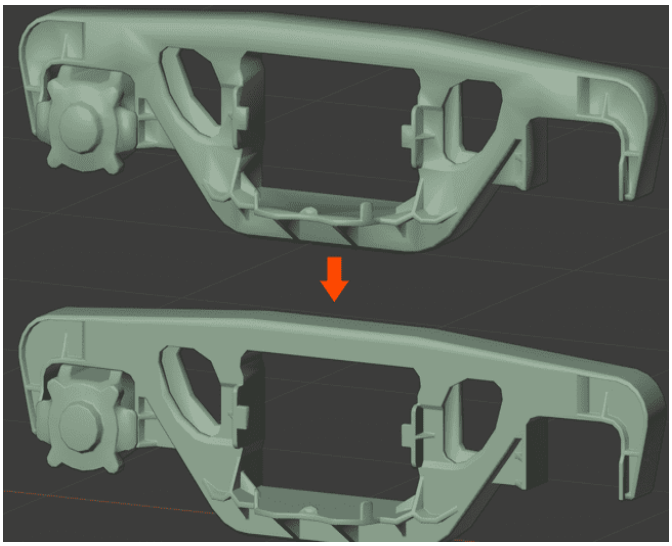


Рисунок 5.5 – Створення коректного шейдингу моделі

Після завершення моделювання та вирівнювання нормалей переходимо до UV розгортки та підготовки моделі до текстурування. Для цього створюється окрема сцена, в якій залишаються тільки унікальні елементи вагона. Деякі об'єкти розташовуються на відстані, щоб легше було контролювати процес створення текстур.

Після того як UV шелли вирівняні використовується модифікатор Texel Density щоб задати всім шеллам однакову щільність пікселей на сантиметр текстурного атласу (рис. 5.6, рис. 5.7).

Було виявлено, що тексель варіюється в залежності від видимості деталей, наприклад нижня частина вагона має нижчу щульність пікселей так як знаходиться постійно в тіні і поза увагою умовного гравця.

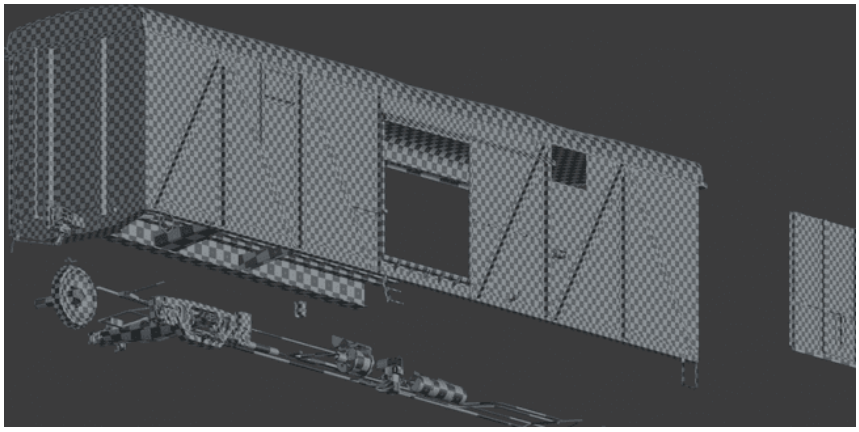


Рисунок 5.6 – Приклад різниці щільності пікселей різних шеллів

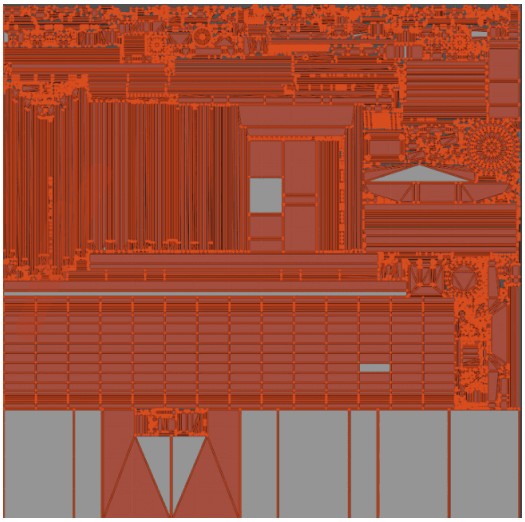


Рисунок 5.7 – Готова упакована розгортка

Після завершення розробки розгортки та вирівнювання шеллів переходимо до текстурування. Для цього насамперед потрібно створити три користувацькі канали (рис. 5.8). Для цього заходимо в програмі Substance Painter в Texture Set Settings тиснемо на + і додаємо три User канали, для зручності називаємо їх R, G, B.

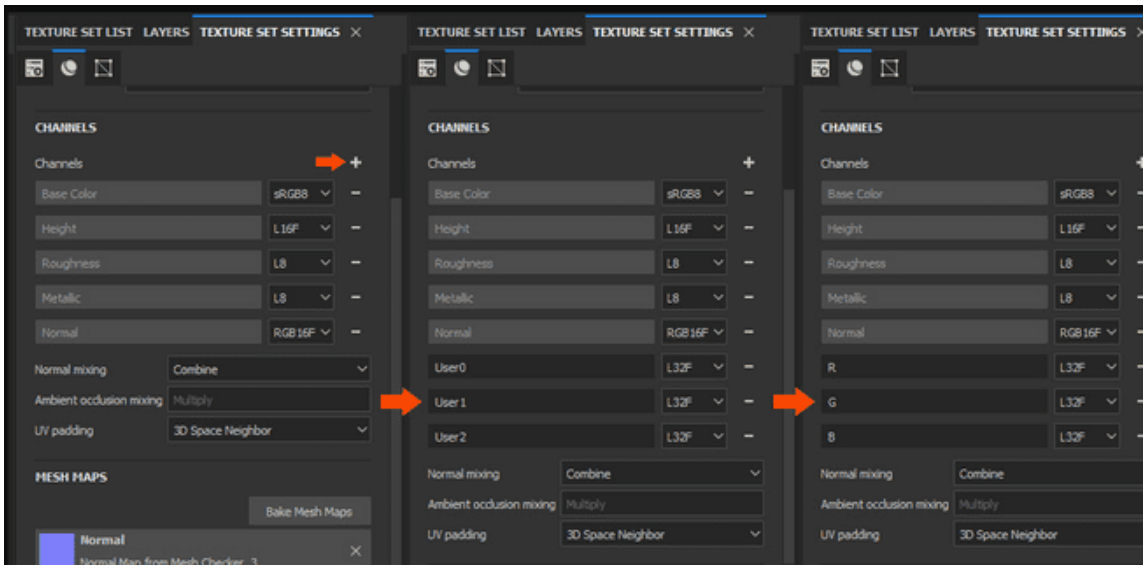


Рисунок 5.8 – Налаштування каналів

Далі після налаштування каналів R, G, B створюємо пустий шар (Fill Layer) в ноьму залишаємо тільки наші R, G, B канали і ставимо їм нульові значення, цей шар завжди повинен бути в самому низу, він даватиме чорнеий фон нашим маскам. Далі створюються шари в яких малюватимемо маски.

Створюється Fill Layer, залишаємо в цьому шарі тільки канал Base Color, він ні на що не впливатиме, потрібен тільки для візуальної ідентифікації каналів, для зручності кольору призначаються відповідно до каналів: R – червоний G – зелений B – синій.

Значення для каналу користувача ставимо на 1 або будь-яке число більше 0, цим повзунком можна додатково регулювати інтенсивність прояву маски. Після позначення кольорів каналів R, G, B починається процес малювання текстур моделі, але на відміну від класичного, унікального

текстурингу створюються тільки кольорові маски, в яких будуть проявлятися матеріали іржі, бруду та іншого (рис. 5.9).

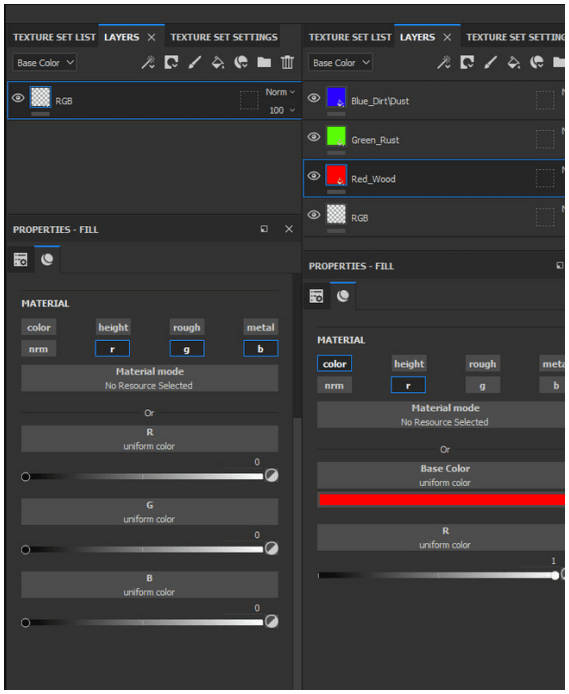


Рисунок 5.9 – Створення кольорів для каналів R, G, B

Білий колір проявлятиме матеріал іржі, там де чорний будуть усі нижні шари. Далі просто малюються маски. Наприклад, зелений канал відповідає за іржу. Створюємо цьому каналу Black Mask та починаємо малювати текстуру іржі. Можна створювати любу кількість Fill Layer, також дозволяється змішувати їх різними способами, загалом процес малювання схожий на звичайний текстуринг.

Єдине, що потрібно пам'ятати, в межах однієї текстури є 3 канали і відповідно можна змішати тільки 3 матеріали.

Після того, як робота над масками закінчена, потрібно їх правильно експортувати в ігровий рушій. Для цього в File\Export Textures\Output Templates натискаємо на + і створюємо свій пресет експорту текстур. Вибираємо R+G+B перетягуємо у слоти наші канали User 0/1/2 використовуючи режим Gray Channel (рис. 5.10).

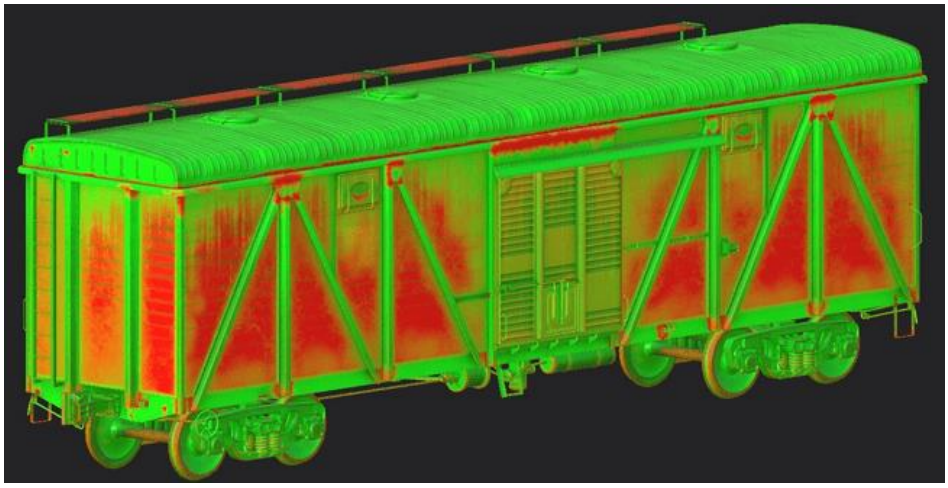


Рисунок 5.10 – Готові маски для проявлення текстур

Після завершення розробки масок в Substance Painter та малювання текстур моделі переходимо до налаштування шейдера (рис. 5.11) та тестування готової моделі в ігровому рушії Unreal Engine [19].

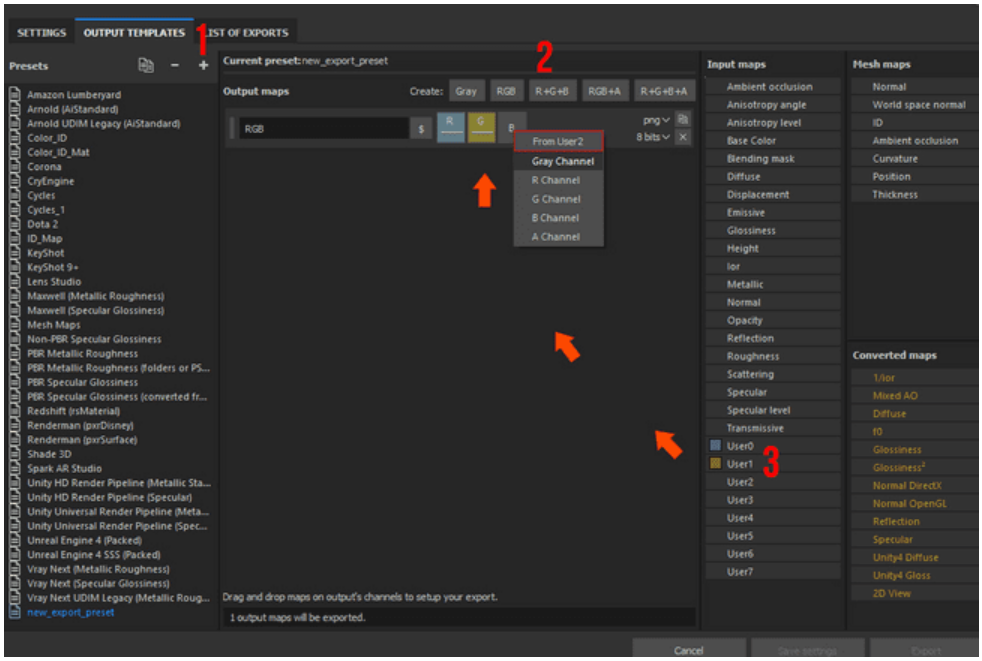


Рисунок 5.11 – Створення пресету експорту текстур

В першу чергу включаємо в налаштуваннях проекту функцію Layered Materials, щоб вся система працювала. Можна використовувати систему Material Function, але система Layers більш компактна. Потрібно буде

створити Master Material, шари Material Layers і Layer Blends функції для змішування шарів (рис. 5.12).

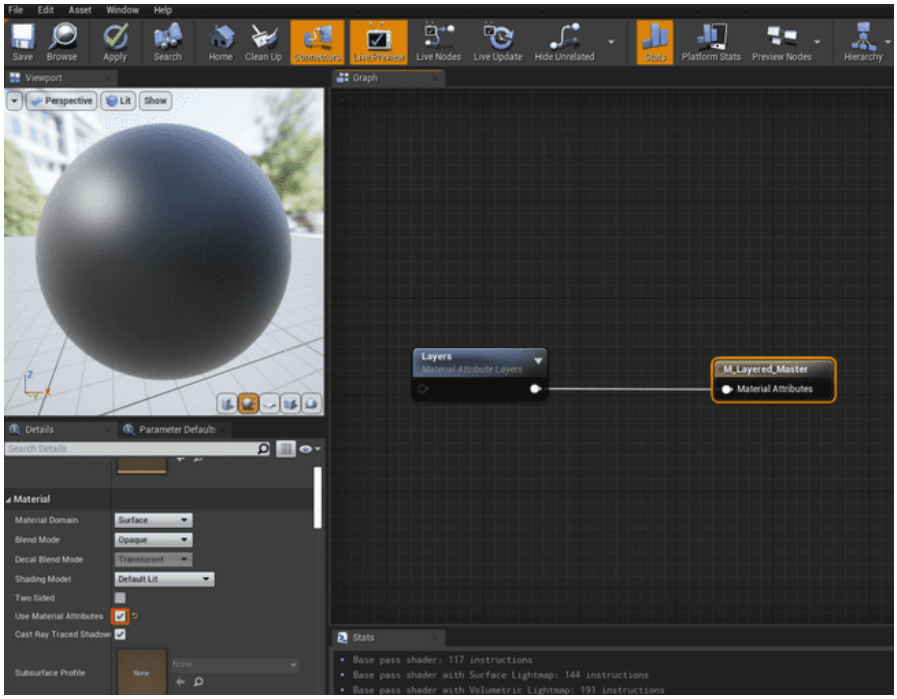


Рисунок 5.12 – Створення Master Material

Створюємо звичайний матеріал, переводимо його в режим Material Attributes, підключаємо в нього ноду Material Attribute Layers, тепер майстер матеріал готовий. Після того, як Master Material готовий, потрібно створити шари в Material Layers.

Для кожного типу матеріалів створюється свій Material Layer, формуючи таким чином бібліотеку для нашого проекту. Після того як шари зроблені необхідно зробити функцію Layer Blend для змішування масок або використовувати заготовлені [17]. Layer Blend це інструмент, який змішуватиме наші шари. Беремо створену RGB текстуру з Substance Painter і підключаєм її в Channel Mask Parameter (рис. 5.13 та рис. 5.14).

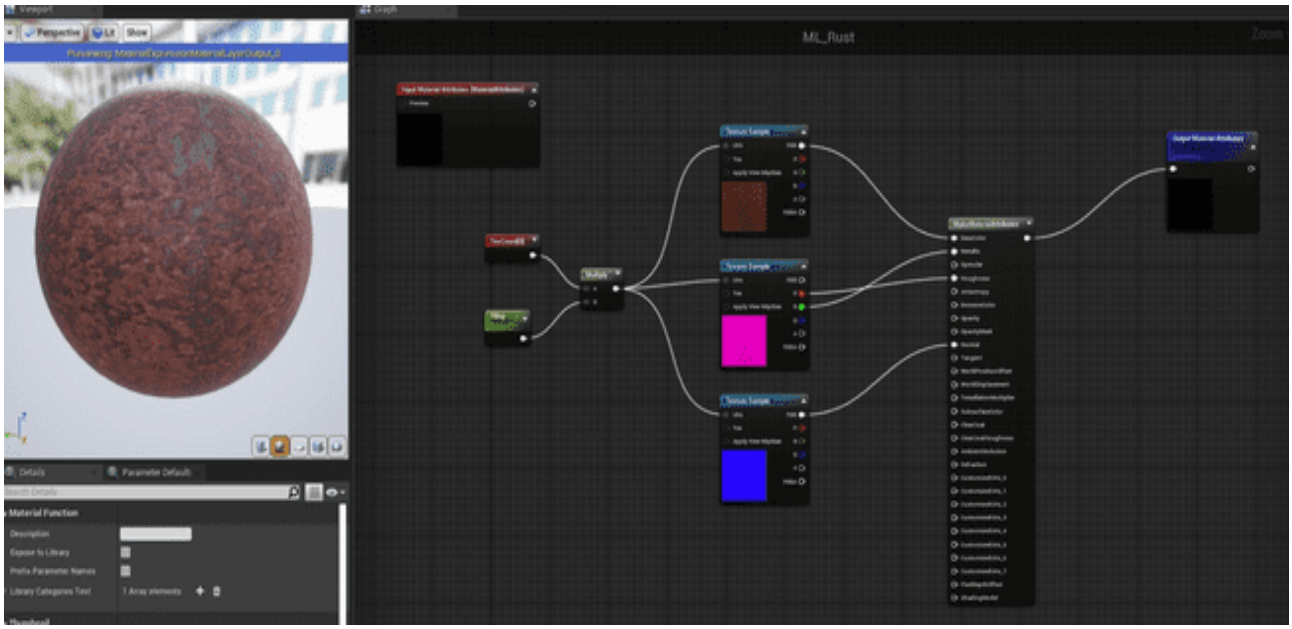


Рисунок 5.13 – Створення нодів в Master Material

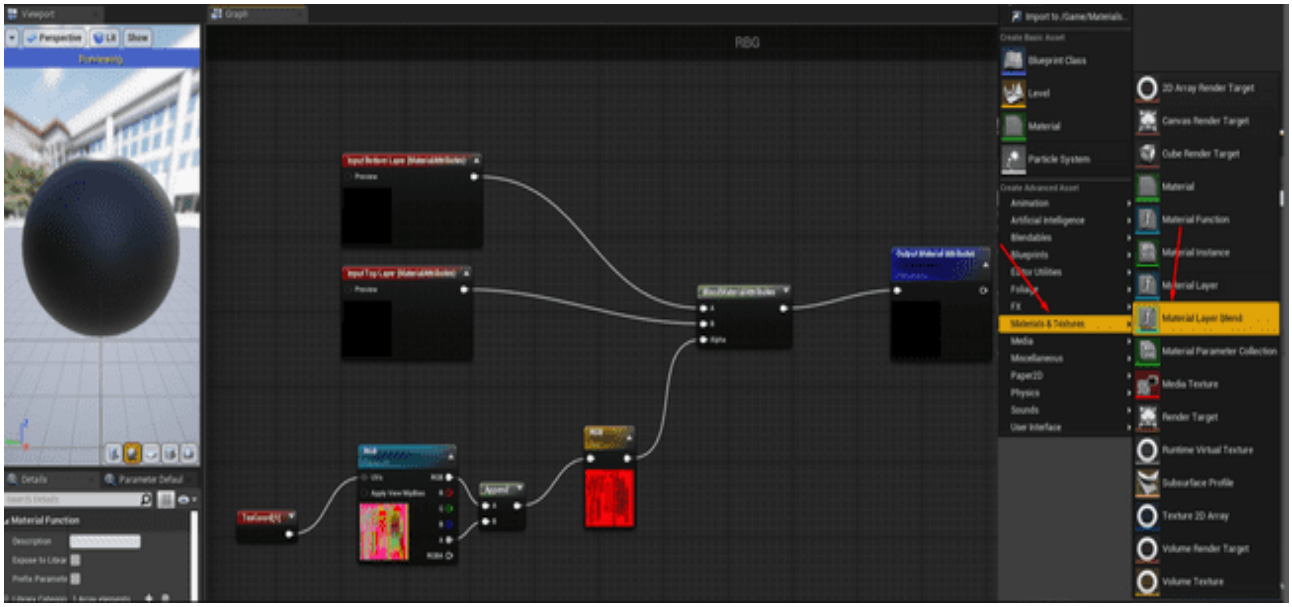


Рисунок 5.14 – Створення функції змішування масок

Після того, як Layer Blend готовий, далі створюємо копію Master Material і в ньому збираю створені шари в розділі Layer Parameters. У шарі Background у лежить матеріал, в якому змішані дерево і фарба, всі інші шари лягають поверх нього. У налаштуваннях шару відображаються параметри, які надаються шару при створенні шейдера. Налаштування для регулювання сили впливу маски, які створив у Blend Layer (рис. 5.15).

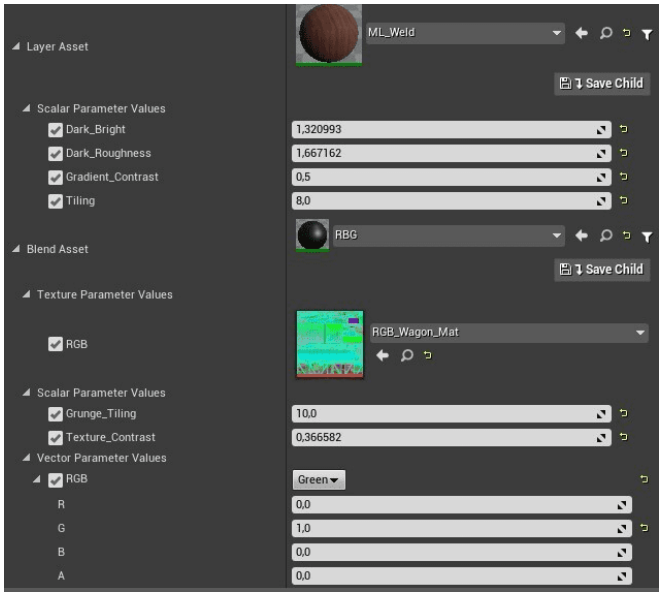


Рисунок 5.15 – Налаштування шарів в Master Material

Таким чином створюється решта шарів для різних матеріалів, котрі будуть проявлятися в масках на моделі (рис. 5.16).

Рисунок 5.16 – Готова модель в ігровому рушії Unreal Engine

Були визначені характерні риси, яких слід дотримуватись для створення правдоподібної моделі вагона потяга.

Був узагальнений алгоритм створення моделі, з урахуванням рекомендованих пропорцій між елементами моделі, та виявлені специфічні операції, яких необхідно дотримуватись.

6 ЕКОНОМІЧНА ЧАСТИНА

6.1 Характеристика науково-дослідної роботи

В економічній частині кваліфікаційної роботи обґрунтовано економічну доцільність формування методичного підходу щодо створення тривимірної моделі для ААА-відеоігор, а також умов застосування запропонованої методики. В роботі досліджується поетапний процес створення тривимірної моделі, а також наведено оцінювання отриманих даних.

Реалізація запропонованої методики дозволяє скоротити час на підготовку на створення карт для тривимірної моделі за рахунок нової методики моделювання.

На підставі проведеного аналізу результатів така методика дозволяє поліпшити робочий процес без суттєвих втрат якості моделі.

Метою даного розділу є економічне обґрунтування витрат на проведення науково-дослідної роботи (НДР) для дослідження та розробки методики створення тривимірної моделі за новим пайплайном. Буде проведено розрахунок трудовитрат та заробітної плати працівникам, одноразових витрат, прибутку, оцінка економічної ефективності НДР.

Реалізація НДР передбачає такі етапи:

– аналіз літературних та мережевих джерел, формулювання основних вимог до технологічного процесу створення тривимірної моделі;

– аналіз існуючих технологій тривимірного моделювання;

– розробка тривимірної моделі відповідно до етапів розробленої методики;

– обробка та аналіз результатів дослідження.

6.2 Етапи виконання НДР, їх трудомісткість та заробітна плата

У процесі виконання науково-дослідної роботи було проведено огляд існуючих методів для розробки та оцінки полігональних тривимірних моделей, на основі аналізу спеціальної літератури розглянуто принципи, на

яких будується оптимізація зовнішнього вигляду моделі, та принцип формування критеріїв для їх оцінки. Після чого проведено експеримент, методами анкетування експертної групи та попарного порівняння виконано чисельне оцінювання якості розробленої моделі. На основі отриманих даних методика розробки тривимірної моделі оцінена позитивно.

Умовно НДР можна розділити на такі етапи: підготовчий, основний і заключний. На стадії виконання підготовчого етапу здійснено порівняльне оцінювання методик створення тривимірної моделі та особливості моделювання для даної методики.

На етапі виконання основної частини НДР були виконано такі роботи:

– огляд існуючих аналогічних методів;

– проведення експерименту;

– обробка та аналіз результатів експерименту;

– розробка методики.

У заключній частині здійснюється оцінка ефективності виконання НДР, складання звіту по НДР, захист звіту.

Найбільш складною й відповідальною частиною під час планування НДР є розрахунок трудомісткості робіт, тому що трудові витрати часто становлять основну частину вартості науково-дослідних робіт і безпосередньо впливають на строки розробки.

Для проведення аналізу якості створеної моделі шляхом порівняння її з моделями конкурентів на підставі короткого опитування було залучено до роботи 15 експертів, на умовах погодинної оплати. Загальний обсяг часу, відведеного для опитування, складав 1 годину на експерта, з оплатою 120,00 грн/год. Також було запрошено спеціаліста-аналітика в сфері тривимірного моделювання, заробітна плата якого становить 14000,00 грн/міс. Заробітна плата керівника роботи – 12000,00 грн/міс, основного виконавця робіт – 11000,00 грн/міс.

Проведемо розрахунок трудовитрат і заробітної плати виконавців робіт.

Середньоденна заробітна плата виконавця робіт ($З_{ср.дн.}$) розраховується за формулою:

$$З_{ср.дн.} = \frac{З_{ср.міс.}}{n}, \quad (6.1)$$

де $З_{ср.міс.}$ – середньомісячна зарплата виконавця роботи;

n – число робочих днів у місяці, (n = 22).

В даному випадку середньоденна заробітна плата аналітика складає 636,36 грн/день, заробітна плата керівника проекту складає 545,45 грн/день, заробітна плата виконавця проекту складає 500,00 грн/день.

Етапи виконання НДР, перелік і зміст робіт, трудомісткість їх виконання, заробітна плата виконавця робіт представлені в табл. 6.1.

Таблиця 6.1 – Розрахунок трудовитрат і заробітної плати виконавця робіт

Перелік робіт	Кіль-кість вико-навців	Трудо-місткість робіт, люд.-днів	Середньо-денна заробітна плата, грн	Сума заробітної плати, грн
1	2	3	4	5
1. Підготовчий етап				
1.1. Розробка та затвердження ТЗ	1	1	545,45	545,45
1.2. Підготовка довідкових матеріалів та даних для виконання НДР	1	2	545,45	1090,90
2. Основний етап				
2.1 Постановка задачі	1	1	545,45	545,45
2.2 Аналіз існуючих технологічних процесів створення тривимірних моделей	1	3	636,36	1909,08
2.4 Створення методичних рекомендацій	1	1	636,36	636,36
2.4 Розробка методики створення персонажа	1	4	500,00	2000,00
2.5 Практичний експеримент, перевірка розробленої методики	1	2	500,00	1000,00

Продовження таблиці 6.1

1	2	3	4	5
2.6 Аналіз якості розробленої моделі	15	1,875	960,00	1800,00
3. Заключний етап				
3.1 Аналіз результатів проведення роботи	2	2	636,36 500,00	1272,72 1000,00
3.2 Формування висновків та пропозицій за темою проекту	2	1	636,36 500,00	636,36 500,00
Усього		20		12936,32

6.3 Розрахунок одноразових витрат на розробку НДР

Калькуляція собівартості розраховується відповідно до існуючих нормативних актів України. До складу калькуляції входять такі статті витрат:

– матеріальні витрати;

– витрати на оплату праці;

– єдиний соціальний внесок;

– амортизація основних засобів (вартість машинного часу);

– витрати на спожиту електроенергію;

– інші витрати.

До інших витрат відносяться адміністративні витрати (водопостачання, водовідведення, опалення, освітлення) та вартість послуг зв’язку.

Матеріальні витрати визначаються витратами на матеріали, визначені їх потребою для виконання робіт, і цін, що діють на момент складання відповідної калькуляції.

Матеріальні витрати розраховуються за такою формулою:

$$M = \sum_{j=1}^{n} Q_j \times Ц_j, \qquad (6.2)$$

де M – сумарні витрати на матеріали, в тому числі малоцінні предмети, що швидко зношуються (носії, папір, канцелярське приладдя тощо), або на літературу, яка необхідна для проведення роботи, тощо;

Q_j – кількість використаних одиниць *j*-го виду матеріалів, $j = (1..n)$;

$Ц_j$ – ціна одиниці *j*-го виду матеріалів.

Розрахунок матеріальних витрат представлено в табл. 6.2.

Таблиця 6.2 – Розрахунок матеріальних витрат

Найменування	Од. вим.	Кількість, (Q_j)	Ціна ($Ц_j$), грн	Сумарні витрати на матеріали (М), грн
Олівець механічний	шт.	2	23,00	46,00
Блокнот	шт.	2	40,00	80,00
Усього				126,00

Витрати на оплату праці розраховуються виходячи з необхідного для виконання робіт складу й кількості працівників, а також із середньомісячної заробітної плати. Відповідно до проведених розрахунків витрати на оплату праці виконавців роботи дорівнюють 12936,32 грн.

Єдиний внесок на загальнодержавне соціальне страхування (ЄСВ) – консолідований страховий внесок, збір якого здійснюється в систему загальнообов'язкового державного соціального страхування в обов'язковому порядку і на регулярній основі з метою забезпечення захисту у випадках, передбачених законодавством, прав застрахованих осіб і членів їх сімей на отримання страхових виплат (послуг) за діючими видами загальнообов'язкового державного соціального страхування.

Відповідно до чинного законодавства ставка єдиного соціального внеску дорівнює 22 % від витрат на оплату праці, тобто розмір ЄСВ дорівнює 2845,99 грн.

При виконанні НДР застосовувався комп'ютер вартістю 15000,00 грн.

Вищенаведене устаткування є власністю організації виконавця, тому доцільно розрахувати суму амортизаційних відрахувань на період виконання НДР. Амортизація основних засобів розраховується за формулою:

$$AB = \sum_{k=1}^{L} \frac{BO_k}{TE_k} \times T \quad , \tag{6.3}$$

де AB – сума амортизаційних відрахувань, нарахованих під час проведення науково-дослідницької роботи;

BO_k – вартість основних засобів k-го виду;

TE_k – термін експлуатації основних засобів k-го виду, днів;

T – термін науково-дослідницької роботи, днів;

L – кількість видів обладнання.

Підставивши відомі значення у формулу (6.3), визначимо величину амортизаційних відрахувань з урахуванням, що термін експлуатації комп'ютера в середньому 3 роки, сума амортизаційних відрахувань складе 273,97 грн.

Витрати на використану обладнанням електроенергію розраховуються за наступною формулою:

$$З_e = M \cdot t \cdot T_{кВт} \, , \tag{6.4}$$

де M – потужність устаткування, тобто кількість енергії, споживаної за одиницю часу (кВт/година);

t – кількість годин використання устаткування за період проведення науково-дослідної роботи;

$T_{кВт}$ – тариф, тобто вартість використання 1 кВт електроенергії.

Споживна потужність комп'ютера складає 0,5 кВт за годину, тариф споживачів складає 2,64 грн/кВт. Підставивши значення у формулу (6.4), отримаємо величину витрат на спожиту електроенергію у розмірі 211,20 грн.

До інших статей витрат відносяться адміністративні витрати (водопостачання, водовідведення, освітлення, опалення), які прийнято у розмірі 20 % від витрат на оплату праці та вартість оплати послуг зв'язку.

Вартість оплати послуг зв’язку, а саме Інтернет становитиме 250,00 грн за 20 днів виконання НДР.

За час виконання НДР витрати на відрядження, аутсорсинг, інформаційні послуги та маркетингові заходи не мали місця.

Результати розрахунку кошторису витрат, тобто одноразових витрат, на виконання НДР наведені в табл. 6.3.

Таблиця 6.3 − Кошторис витрат на розробку НДР

№ з/п	Стаття витрат	Сума, грн
1	Заробітна плата	12936,32
2	Єдиний соціальний внесок (22,0 % від п.1)	2845,99
3	Матеріальні витрати	126,00
4	Амортизація основних засобів	273,97
5	Витрати на спожиту електроенергію	211,20
6	Інші витрати, у тому числі:	
6.1	адміністративні витрати (20% від п.1)	2587,26
6.2	вартість послуг зв’язку	250,00
	Усього витрати на розробку (*Вр*)	19230,74

Кошторис витрат на виконання даної НДР складає 19230,74 грн.

6.4 Оцінка результатів науково-дослідної роботи

Результат – це наслідок послідовності дій виконаних при НДР, виражений якісно або кількісно. В загальному випадку оцінка результатів НДР – це визначення ефективності отриманих рішень порівняно з сучасним науково-технічним рівнем. Відповідно до теми даної кваліфікаційної роботи розглянемо скорочення часу на створення тривимірної моделі як позитивний результат впровадження НДР (табл. 6.4).

Результат від впровадження НДР визначається за такою формулою:

$$\Delta P_j = \left| Хб_j - Хн_j \right|, \tag{6.5}$$

де ΔP_j – покращення *j*-ої характеристики досліджуваного процесу за рахунок впровадження результатів НДР (*j =1,m*);

m – кількість досліджуваних характеристик;

$Хб_j$ – базове значення *j*-ої характеристики;

$Хн_j$ – нове значення *j*-ої характеристики після впровадження НДР.

Таблиця 6.4 – Відсоток досягнення мети користувачем та оцінка якості дизайну

Характеристика	До впровадження	Після впровадження
1	2	3
Час розробки моделі, год	18	15
Загальний час роботи над моделлю, год	46	43

Підставивши відповідні значення до формули (6.5), визначимо результат від впровадження НДР у чисельному вигляді:

$$\Delta P_{ч.1} = |18 - 15| / 18 = 16{,}6\ \%;$$
$$\Delta P_{як.1} = |46 - 43| / 46 = 6{,}5\ \%.$$

Результат дослідження можна вважати вдалим, оскільки загальний час роботи над моделлю зменшено на 6,5 %, а її якість оцінено позитивно (п. 6.3).

Під час економічного обґрунтування НДР було дана характеристика науково-дослідної роботи, наведено етапи виконання НДР, розраховано їх трудомісткість та заробітна плата виконавців. Розраховано одноразові витрати на розробку НДР, дано оцінку результатів науково-дослідної роботи та визначено економічну ефективність результатів НДР. Дана науково-дослідна робота має позитивний показник економічної ефективності. Роботу у цілому можна вважати ефективною або такою, що має науковий та технічний рівень.

ВИСНОВКИ

Дослідження присвячене розробці методики розробки тривимірної моделі за Weighted Normals пайплайн для ААА-відеоігор. Були досліджені теоретичні матеріали і науково-методичні статті за технологією і етапами розробки моделей. Проведено дослідження специфіки розробки моделей різними пайплайнами.

Також були вивчені характерні особливості прототипу моделі і особливості підготовки його до розробки в тривимірному вигляді. В результаті дослідження були виявлені основні критерії необхідні для створення правдоподібної моделі. На основі яких був створений метод розробки моделі вагона потяга. Цей алгоритм був надалі використаний для створення власної тривимірної моделі.

Розроблена методика розрахована на використання прикладних засобів тривимірного моделювання, за допомогою яких можна створювати складні та не стандартні за алгоритмом створення моделі. Як основні програми для роботи були використані програми 3DsMAX, Substance Painter та Unreal Engine це професійні пакети для роботи з тривимірною графікою, які використовують великі ААА студії для вирішення складних завдань.

Створення тривимірних моделей являється дуже популярним на сьогодні в багатьох сферах життя суспільства – реклама, кіно, мультиплікація, ігри.

Привабливі та фотореалістичні тривимірні моделі затребувані в іграх оскільки позитивно впливають на сприйняття графіки гравцями.

Створена дана модель, з некласичним підходом текстурування та моделювання, готова для впровадження в реальні ААА-відеоігри.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Grafting Physically-based materials. Unreal Energy Documentation. URL:https://docs.unrealengine.com/4.26/enUS/RenderingAndGraphics/Materials/PhysicallyBased/ (дата звернення 10.06.2023).

2. Десять кращих ігрових рушіїв. URL: https://senfil.net/index.php?newsid=321 (дата звернення 10.11.2023).

3. Unreal Engine 4: учебное пособие по созданию анимации URL: https://stdpub.com/unrealengine/unreal-engine-4-uchebnoe-posobie-po-sozdaniyu-animaczii (дата звернення 10.10.2023).

4. Unreal Engine 4 Tutorial: работа с Искусственным Интеллектом – Режим доступу до ресурсу. URL: https://stdpub.com/unrealengine/unreal-engine-4-tutorial-rabota-s-iskusstvennym-intellektom (дата звернення 10.12.2023).

5. Матеріал по Unreal Engine Знайомство з движком і його основними функціями. URL: https://habr.com/post/344394 (дата звернення 15.05.2023).

6. Матеріал по Unreal Engine Blueprints та його функції. URL: https://habr.com/post/344446/ (дата звернення 15.05.2023).

7. Сіденко Л.А. Комп'ютерна графіка та геометричне моделювання: Навчальний посібник. Київ: ArtHuss, 2009. 224 с.

8. Шелл Д. Геймдизайн. К.: ООО «Альпина Паблишер», 2019. 284 с.

9. Optimized View Frustum Culling Algorithms: Ulf Assarsson and Tomas M¨oller. URL: http://www.cse.chalmers.se/~uffe/vfc.pdf. (дата звернення 15.05.2023)

10. Гундольф С., Фрейермут С. Ігри. Геймдизайн. Дослідження ігор. Київ: Гуманитарный центр. 2021. 250 с.

11. Використання NURBS-моделювання для створення тривимірних об'єктів в 3D Studio Max. URL: http://esate/uroki/3d-max/kurs_modelirovaniya/3D_max_urok_7_1/ (дата звернення 10.09.2023).

12. Lee J. Unreal Engine: Game Development from A to Z / J. Lee, J. P. Doran, N. Misra., 2016. 837 с.

13. Edmonds M. Mastering Game Development with Unreal Engine 4 / Matt Edmonds., 2018. 356 с.

14. Дейнеко Ж.В., Бондар А.С. Використання штучного інтелекту для створення 3D-моделей // Поліграфічні, мультимедійні та web-технології: тези доп. 2023. Т. 1. С. 203-204.

15. Гаманець Є.О., Дейнеко Ж.В. Особливості створення анімованого тривимірного персонажу // Поліграфічні, мультимедійні та web-технології: матеріали молодіжної школи-семінару IV Міжнародної науково-технічної конференції, Харків: ТОВ «Друкарня Мадрид», 2021. Т. 2. С. 76-78.

16. Мельніченко Д.О., Дейнеко Ж.В. Порівняння методів моделювання hard surface моделей для ААА відеоігор // Поліграфічні, мультимедійні та web-технології: тези доп. VIII Міжнар. наук.-техн. конф. (16-20 травня 2023, м. Харків) / редкол.: І.Б. Чеботарьова, О.В. Вовк, Ж.В. Дейнеко. Харків: ТОВ «Друкарня Мадрид», 2023. Т1. 2. С. 215-216.

17. Воронцова Д.В., Маслаковець А.Б. Аналіз інструментарію моделювання у програмних комплексах Autodesk 3DS Max та Blender // Поліграфічні, мультимедійні та web-технології: тези доп. VIII Міжнар. наук.-техн. конф. (16-20 травня 2023, м. Харків). 2023. Т. 1. 2. С. 220-221.