

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра комп'ютерної інженерії

Пояснювальна записка

до кваліфікаційної роботи на ступінь вищої освіти магістр

на тему: **«УДОСКОНАЛЕННЯ СПОСОБУ РЕГУЛЮВАННЯ
ПЛАВНОСТІ ЗОБРАЖЕННЯ ІГРОВИХ ДОДАТКІВ»**

Виконала: студентка 6 курсу, групи КСДМ–61
спеціальності

123 Комп'ютерна інженерія
(шифр і назва спеціальності)

Свищук М.М.

(прізвище та ініціали)

Керівник Торошанко Я.І.

(прізвище та ініціали)

Рецензент _____

(прізвище та ініціали)

Київ – 2021

ВСТУП

Ігрова індустрія стає все більшою з кожним роком, оскільки все більше людей відкривають для себе можливості, які існують у цій діяльності.

Велика проблема назріває, коли доводиться мати справу із зависаннями, збоями, гальмуванням, повільним завантаженням та іншими проблемами, коли ви всі роздуті посеред гри. Однак ці проблеми не є рідкістю. Багато геймерів використовують звичайні ноутбуки та настільні комп'ютери, які не були розроблені, щоб відповідати потужності найновіших, складних комп'ютерних ігор.

Консольним геймерам в цьому плані набагато легше — усе їхнє обладнання ідеально налаштовано для чудового ігрового процесу, а ігри завжди розробляються з урахуванням обмежень продуктивності консолі. З іншого боку, ігри для ПК можуть бути набагато менш послідовними, з невідповідним обладнанням, старими графічними картами, застарілими програмними драйверами, що завжди призводить до низької продуктивності в іграх.

Налаштування комп'ютера для покращення його ігрових можливостей вимагатиме від вас деяких технічних змін, наприклад дефрагментації жорсткого диска, знищення непотрібних фонових служб, оновлення драйверів, звільнення місця тощо.

Ми розглянемо, як можна отримати плавні картинки в іграх на слабкому пристрої, оптимізувати його та гру таким чином, щоб користувач в підсумку отримав стабільні 60 кадрів у секунду.

Ми не будемо розглядати покращення компонентів ПК і покупку нового як один із варіантів досягнення наших цілей, а розглянемо, як на іменному та усталеному обладнанні отримувати впевнені результати простими та доступними інструментами.

У ролі допоміжних інструментів для досягнення наших цілей виступатимуть звичайні налаштування Віндовс, софт для оптимізації окремих компонентів ПК, а також внутрішньоігрові налаштування.

Додатково буде розглянуто вплив високо- та низько-полігональних моделей на ресурсоспоживання ПК і чому для розробників важливо оптимізувати ігри для кінцевих пристроїв.

1 ГРАФІЧНІ ПРОЦЕСОРИ ТА 3D-ГРАФІКА

1.1 Поява та розвиток графічних процесорів

Еволюція сучасного графічного процесора починається з появи перших 3D-карт у 1995 році, після чого послідує широке поширення 32-розрядних операційних систем і доступного персонального комп'ютера.

Графічна індустрія, яка існувала до цього, здебільшого складалася з більш прозаїчної 2D архітектури, не пов'язаної з ПК, з графічними платами, більш відомими за буквено-цифровими умовами найменування їхніх мікросхем і їх величезними цінниками. 3D-ігри та комп'ютерна графіка віртуалізації зрештою об'єдналися з таких різноманітних джерел, як аркадні та консольні ігри, військові, робототехнічні та космічні симулятори, а також медичні зображення.

Перші дні споживчої 3D-графіки були Диким Заходом конкуруючих ідей. Від того, як реалізувати апаратне забезпечення, до використання різних методів візуалізації та їх застосування та інтерфейсів даних, а також постійної гіперболи імен. Ранні графічні системи мали конвеєр із фіксованими функціями (FFP) та архітектуру, яка відповідала дуже жорсткому шляху обробки, що використовувала майже стільки графічних API, скільки було виробників 3D-чипів.

Хоча 3D-графіка перетворила досить нудну індустрію ПК на світлове та чарівне шоу, вона завдячує своїм існуванням поколінням інноваційних зусиль.

Перша справжня 3D-графіка почалася з ранніх контролерів дисплея, відомих як перемикачі відео та генератори відеоадреси. Вони виконували роль прохідної між основним процесором і дисплеєм. Вхідний потік даних був перетворений у послідовний бітовий відеовихід, такий як яскравість, колір, а також вертикальна та горизонтальна композитна синхронізація, яка зберігала рядок пікселів у генерації дисплея та синхронізувала кожен наступний рядок

разом із інтервалом гасіння (час між тим як один рядок сканування закінчується і починається наступний).

У другій половині 1970-х років з'явився шквал дизайнів, що заклало основу для 3D-графіки, якою ми її знаємо. Наприклад, відеочіп «Pixie» RCA (CDP1861) у 1976 році був здатний виводити сумісний з NTSC відеосигнал з роздільною здатністю 62x128 або 64x32 для злочасної консолі RCA Studio II.

Через рік після відеочіпа з'явився адаптер телевізійного інтерфейсу (ТІА) 1А, який був інтегрований в Atari 2600 для створення екранного дисплея, звукових ефектів і контролерів входу зчитування. Розробкою ТІА керував Джей Майнер, який пізніше також очолив дизайн власних мікросхем для комп'ютера Commodore Amiga.



Рисунок 1.1 – Відеочіп Atari 2600

У 1978 році Motorola представила генератор відеоадреси MC6845. Це стало основою для адаптерів монохромного та кольорового дисплея IBM PC (MDA/CDA) 1981 року і забезпечило ту ж функціональність для Apple II. У тому ж році Motorola додала генератор відеодисплеїв MC6847, який увійшов до ряду персональних комп'ютерів першого покоління, включаючи Tandy TRS-80.

Подібне рішення від дочірньої компанії Commodore MOS Tech, VIC, забезпечило вихід графіки для старовинних домашніх комп'ютерів Commodore 1980-83 років.

У листопаді наступного року в Atari 400 представили ANTIC (контролер алфавітно-цифрового телевізійного інтерфейсу) і співпроцесор CTIA/GTIA (колір або графічний телевізійний інтерфейсний адаптер). ANTIC обробляв інструкції 2D відображення за допомогою прямого доступу до пам'яті (DMA). Як і більшість співпроцесорів відео, він міг генерувати графіку ігрового поля (фон, титульні екрани, відображення оцінок), тоді як CTIA генерував кольори та рухомі об'єкти. Yamaha і Texas Instruments постачали подібні мікросхеми різним раннім постачальникам домашніх комп'ютерів.

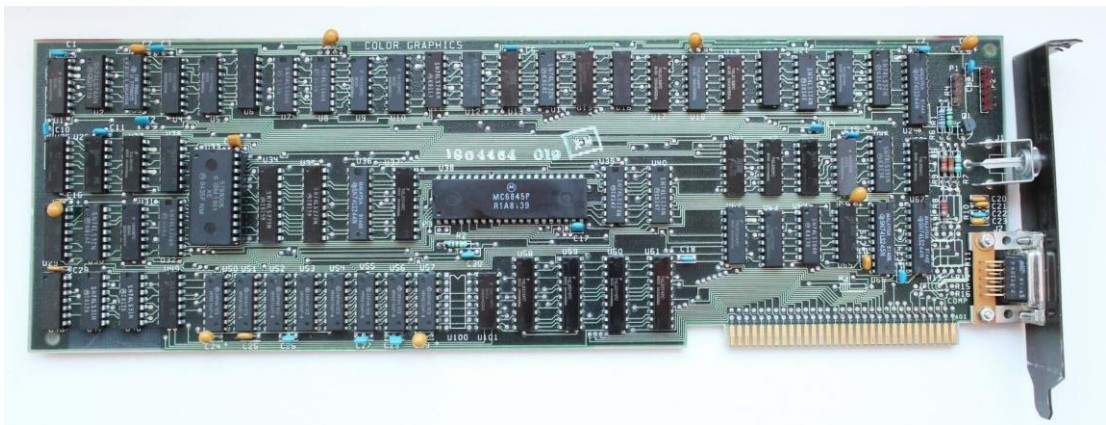


Рисунок 1.2 – Монохромний дисплей адаптер від IBM PC

Наступні кроки в еволюції графіки були насамперед у професійних сферах.

Intel використала свій графічний чіп 82720 як основу для багатомодової плати відеографічного контролера iSBX 275 за 1000 доларів. Він був здатний відображати дані восьми кольорів з роздільною здатністю 256x256 (або монохромне зображення з роздільною здатністю 512x512). Його 32 КБ оперативної пам'яті було достатньо для малювання ліній, дуг, кіл, прямокутників і растрових зображень символів. Чіп також мав можливість масштабування, розділення екрана та прокрутки.

SGI швидко випустила свою IRIS Graphics для робочих станцій - графічну плату GR1.x з можливістю окремих дочірніх плат для параметрів кольору, геометрії, Z-буфера та Overlay/Underlay.

Промислова та військова 3D віртуалізація була відносно добре розвинена на той час. IBM, General Electric і Мартін Марієтта (які мали купити аерокосмічний підрозділ GE у 1992 році) разом із безліччю військових підрядників, технологічних інститутів і NASA реалізували різні проекти, які потребували технології для військового та космічного моделювання. ВМС також розробили симулятор польоту з використанням 3D віртуалізації з комп'ютера Whirlwind Массачусетського технологічного інституту в 1951 році. Крім оборонних підрядників, були компанії, які охопили військові ринки професійною графікою.

Evans & Sutherland, які повинні були надати професійні відеокарти, такі як Freedom і REALimage, також надали графіку для симулятора польоту СТ5, пакета вартістю 20 мільйонів доларів, керованого мейнфреймом DEC PDP-11. Іван Сазерленд, співзасновник компанії, у 1961 році розробив комп'ютерну програму під назвою Sketchpad, яка дозволяла малювати геометричні фігури та відображати на CRT в режимі реального часу за допомогою світлового пера.



Рисунок 1.3 – Симулятор польоту від Evans & Sutherland

Це був родоначальник сучасного графічного інтерфейсу користувача (GUI).

У менш езотеричній сфері персональних обчислень серія EGA (Extended Graphics Adapter) від Chips and Technologies 82C43x створила необхідну конкуренцію адаптерам IBM і була встановлена в багатьох клонах PC/AT приблизно в 1985 році. Рік був визначним для Commodore Amiga, яка постачається з чіпсетом OCS. Чіпсет складався з трьох основних компонентів — Agnus, Denise та Paula — що дозволило певній кількості обчислень графіки та звуку не залежати від ЦП.

У серпні 1985 року троє іммігрантів з Гонконгу, Квок Юань Хо, Лі Лау і Бенні Лау, заснували Array Technology Inc в Канаді. До кінця року назва змінилася на ATI Technologies Inc.

Наступного року ATI випустила свій перший продукт, OEM Color Emulation Card. Він використовувався для виведення монохромного зеленого, жовтого або білого люмінофорного тексту на чорному тлі на монітор TTL через 9-контактний роз'єм DE-9. Карта була оснащена як мінімум 16 КБ пам'яті і відповідала за великий відсоток продажів ATI від 10 мільйонів канадських доларів за перший рік роботи компанії. Здебільшого це було зроблено завдяки контракту, який поставляв близько 7000 мікросхем на тиждень до Commodore Computers.

Поява кольорових моніторів і відсутність стандарту серед конкурентів привели до утворення Асоціації стандартів відеоелектроніки (VESA), членом-засновником якої була ATI разом з NEC і шістьма іншими виробниками графічних адаптерів.

У 1987 році ATI додала серію Graphics Solution Plus до своєї лінійки продуктів для OEM, яка використовувала 8-розрядну шину IBM PC/XT ISA для IBM PC на базі Intel 8086/8088. Чіп підтримував графічні режими MDA, CGA та EGA за допомогою DIP-перемикачів (монохром, кольоровий

графічний адаптер та посилений графічний адаптер). По суті, це був клон плати Plantronics Colorplus, але з місцем для 64 Кб пам'яті. PEGA1, 1a і 2a (256 КБ) Paradise Systems, випущені в 1987 році, також були клонами Plantronics.

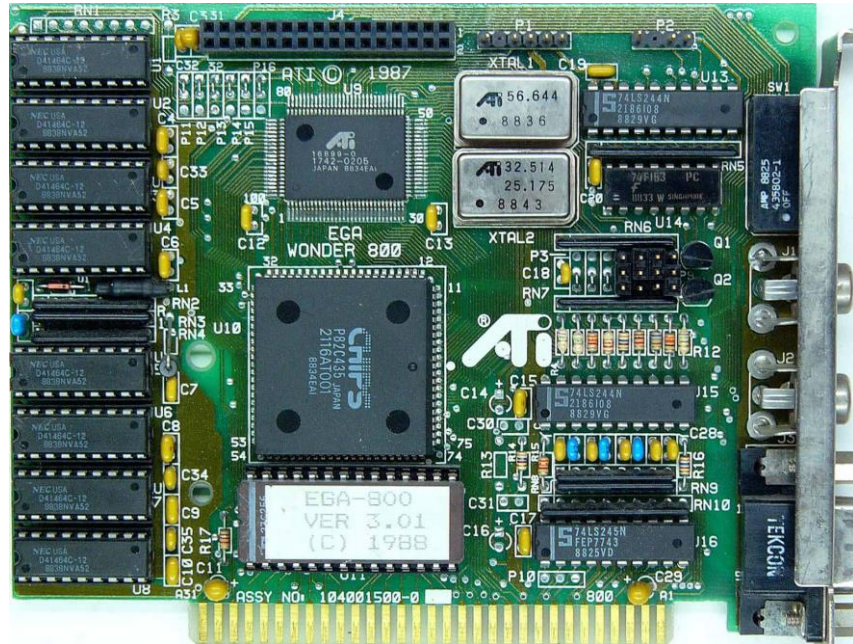


Рисунок 1.4 – Чіп моделі ATi EGA 800 з емуляцією VGA і підтримкою розширення 800x600

Серії EGA Wonder з 1 по 4 надійшли в березні за 399 доларів США і мають 256 КБ DRAM, а також сумісність з емуляцією CGA, EGA і MDA з роздільністю до 640x350 і 16 кольорами. Розширений EGA був доступний для серій 2,3 і 4.

Додатковим доповненням була EGA Wonder 800 з 16-кольоровою емуляцією VGA і підтримкою роздільної здатності 800x600, а також карта VGA Improved Performance (VIP), яка була в основному EGA Wonder з цифрово-аналоговим перетворювачем (DAC) доданим для забезпечення обмеженої сумісності з VGA. Останній коштував 449 доларів США плюс 99 доларів за модуль розширення Compaq.

ATi була далеко не самотньою, керуючись хвилиною споживчого апетиту до персональних комп'ютерів.

Того року з'явилося багато нових компаній і продуктів. Серед них були Trident, SiS, Tamerack, Realtek, Oak Technology, LSI's G-2 Inc., Hualon, Cornerstone Imaging і Winbond – усі створені в 1986-87 роках. Тим часом такі компанії, як AMD, Western Digital/Paradise Systems, Intergraph, Cirrus Logic, Texas Instruments, Gemini і Genoa, виробляють свої перші графічні продукти протягом цього періоду.

Серія Wonder від ATI продовжувала отримувати приголомшливі оновлення протягом наступних кількох років.

1993 рік започаткував шквал нових графічних конкурентів, особливо Nvidia. Нестабільність ринку вже змусила ряд графічних компаній вийти з бізнесу або бути поглиненими конкурентами. Однією з компаній, яка розвивалася від міцності до сили, була ATI, які анонсували чіп ATI 68890 PC Чіп міг знімати відео зі швидкістю 320x240 при 15 кадрах в секунду або 160x120 при 30 кадрах в секунду, а також стискати/декомпресувати в режимі реального часу завдяки вбудованому Intel i750PD VCP (процесор стиснення відео). Він також міг зв'язуватися з графічною платою через шину даних, що позбавляє потреби в ключах або портах і стрічкових кабелях.

Silicon Graphics Inc (SGI) випустила OpenGL 1.0, інтерфейс програмування агностичних прикладних програм (API) для 2D і 3D. Спочатку OpenGL був націлений на професійний бізнес на базі UNIX. Але з підтримкою, зручною для розробників, він швидко перейшов до 3D-ігор. Microsoft розробляла власний API-конкурент під назвою Direct3D, хоч і OpenGL працював набагато краще на системі Windows.

Через кілька років ситуація досягла краху, коли Джон Кармак з id Software, чий раніше випущений Doom зробив революцію в іграх на ПК, переніс Quake до OpenGL у Windows і відкрито критикував Direct3D.



Рисунок 1.5 – оригінальний Quake проти Quake на OpenGL

Пізні зміни від Microsoft і запуск DirectX SDK лишили виробників плат неможливими безпосереднього доступу до обладнання для відтворення цифрового відео. Це означало, що практично всі дискретні відеокарти мали проблеми з функціональністю в Windows 95. На відміну від цього, драйвери під Win 3.1 від різних компаній, як правило, були бездоганними.

Перша публічна демонстрація його відбулася на конференції з відеоігор E3, що відбулася в Лос-Анджелесі в травні наступного року. Сама карта стала доступною через місяць. 3D Rage об'єднав двовимірне ядро Mach64 з можливостями 3D. ATI оголосила про свій перший чіп 3D-прискорювача, 3D Rage (також відомий як Mach 64 GT), у листопаді 1995 року.

Графічна карта Voodoo від 3Dfx, запущена в листопаді 1996 року, складалася з карти лише для 3D, яка вимагала підключення кабелю VGA від окремої 2D-карти до Voodoo, який потім підключався до дисплея. Карти продавала велика кількість компаній. Orchid Technologies була першою, що випустила на ринок плату Orchid Righteous 3D за 299 доларів США, відому завдяки наявності механічних реле, які «клацали», коли чіпсет був у

використанні. Пізніші редакції використовували твердотільні реле відповідно до інших постачальників. За карткою йшли Monster 3D Diamond Multimedia, Voodoo Mania від Colormaster, Canopus Pure3D, Quantum3D, Miro Hiscore, Skywell (Magic3D) і 2theMAX Fantasy FX Power 3D.

Voodoo Graphics зробив революцію в графіці персональних комп'ютерів майже миттєво і зробив багато інших проектів застарілими, включаючи велику кількість виробників лише 2D-графіки. У 1996 році 3D-ландшафт віддав перевагу S3, який займав близько 50% ринку. Однак незабаром це мало змінитися. Підраховано, що 3Dfx становив 80-85% ринку 3D-прискорювачів у період розквіту правління Voodoo.

Менш ніж через два місяці після оголошення Rage Fury MAXX, Nvidia анонсувала GeForce 256 SDR 1 жовтня, а потім у лютому 2000 року з'явилася версія DDR. Це буде перша карта, яка використовує таку форму оперативної пам'яті. 23-мільйонний транзисторний чіп, побудований за технологією TSMC 220 нм, це був перший графічний чіп, який насправді отримав назву GPU (Graphics Processing Unit), на основі додавання механізму трансформації та освітлення (TnL або T&L).

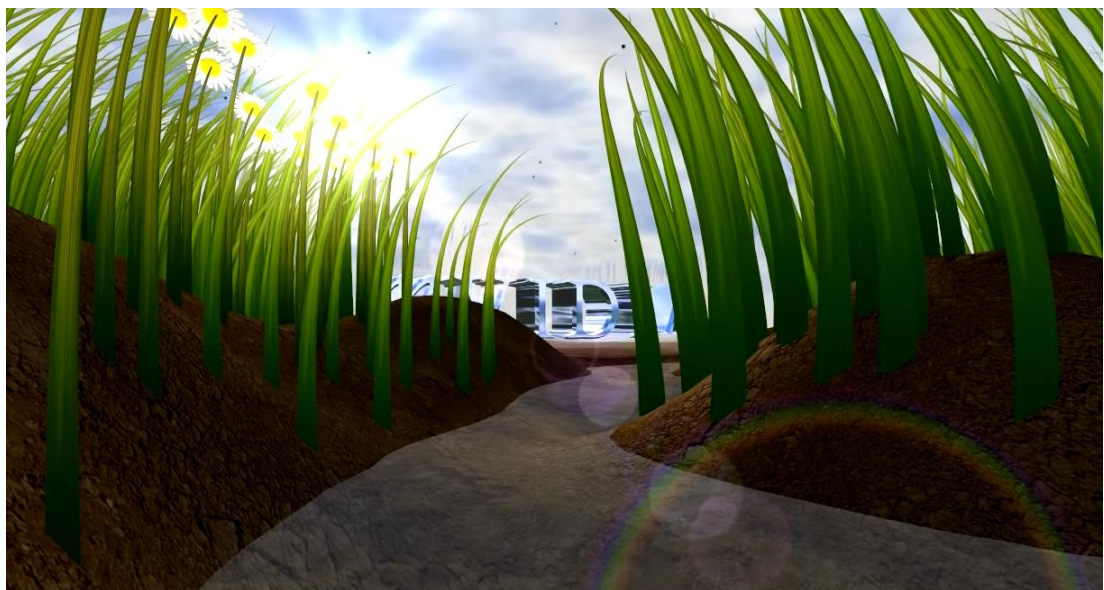


Рисунок 1.6 – Графічне демо Nvidia GeForce 256

Цей механізм дозволив графічному чіпу виконувати інтенсивні обчислення з плаваючою комою, трансформуючи тривимірні об'єкти та сцени – та пов'язане з ними освітлення – у 2D-подання відтвореного зображення. Раніше це обчислення проводилося центральним процесором, який міг легко обмежити робоче навантаження і обмежував доступні деталі.

Статус GeForce 256 як першої, яка включає програмовані піксельні шейдери з використанням T&L, довгий час був предметом дискусій. Це тому, що низка проектів також включала T&L або на стадії прототипу (Rendition Vérité V4400, BitBoys Pyramid3D, 3dfx Rampage), на рівні, що наближається до нерелевантності (3DLabs GLINT, Matrox G400 WARP), або через окремий вбудований чіп Thriller.

Однак жодна з них не мала комерційної функціональності. Більше того, будучи першою, яка прийняла архітектуру з чотирма конвейєрами, Nvidia мала перевагу в продуктивності перед конкурентами. Це в поєднанні з механізмом T&L дозволило компанії випустити GeForce 256 як плату для професійних робочих станцій.

Через місяць після того, як настільний варіант став доступним, Nvidia оголосила про свій перший діапазон професійних карт Quadro для робочих станцій, SGI VPro V3 і VR3, заснованих на GeForce 256. Карти використовували графічну технологію SGI, до якої Nvidia отримала доступ через міжліцензійну угоду, підписано в липні 1999 р.

До появи Voodoo 5 ATI оголосила Radeon DDR як «найпотужніший графічний процесор, коли-небудь розроблений для настільних ПК». Попередні перегляди карти вже були оприлюднені 25 квітня, і лише через двадцять чотири години Nvidia виступила проти анонсом GeForce 2 GTS (GigaTexel Shader). Останній включав версію Nvidia Pixel Tapestry Architecture, названу Nvidia Shading Rasterizer, що дозволяє застосовувати такі ефекти, як дзеркальне затінення, об'ємний вибух, заломлення, хвилі, змішування вершин,

об'єми тіней, відображення рельєфу та відображення висот, які можна застосовувати до кожного пікселя. бази через апаратне забезпечення.

Вважалося, що ця функція була доступна до попереднього чіпа NV10 (GeForce 256), але вона залишилася відключеною через апаратну несправність. GTS також слідував за технологією ATI Charisma Engine, дозволяючи всі обчислення трансформації, відсікання та освітлення підтримувати GPU. Тим не менш, ATI пішла ще далі, створивши скіни вершин для більш плавного переміщення полігонів та інтерполяцію ключових кадрів, де розробники розробили початкову та кінцеву сітку для анімації, а ядро Charisma розраховувало проміжні сітки. Протягом довгого часу лінійка ATI залишалась головним конкурентом Nvidia.

До появи DirectX 10 не було сенсу додавати зайву складність за рахунок збільшення площі матриці, що збільшувало функціональність вершинних шейдерів на додаток до підвищення точності піксельних шейдерів з плаваючою комою з 24 до 32 біт, щоб відповідати вимогам до вершинні операції. З появою DX10 вершинні та піксельні шейдери підтримували великий рівень спільних функцій, тому перехід до уніфікованої шейдерної арки усунув багато непотрібного дублювання блоків обробки. Першим графічним процесором, який використовував цю архітектуру, був культовий G80 від Nvidia.

За чотири роки розробки та 475 мільйонів доларів бюджету було створено 8800 GTX і 8800 GTS 640 МБ, що складався із 681 мільйонів транзисторів.

За допомогою нового алгоритму згладжування Coverage Sample (CSAA) Nvidia отримала задоволення, побачивши, що її GTX знищила кожного конкурента з подвійною графікою в повній продуктивності. Незважаючи на цей успіх, частка ринку дискретної графіки компанії в четвертому кварталі

впала на три відсоткові пункти – пункти, які AMD підняла завдяки контрактам з OEM.

У червні Nvidia випустила свої GTX 260 і GTX 280 з графічним процесором GT200, деталлю розміром 576 мм², що представляє собою найбільшу серійну матрицю графічного процесора на сьогоднішній день (Larrabee від Intel оцінювалася в 600-700 мм²) і найбільший виробничий чіп будь-якого типу, виготовлений TSMC.

GT200 підтвердив бажання Nvidia підштовхнути GPGPU в центр уваги, включивши в дизайн спеціалізоване обладнання подвійної точності (FP64) і обчислювальне обладнання. Архітектурні зміни, орієнтовані на ігри, були скромнішими, але це не завадило Nvidia встановити ціну на 280 за 649 доларів або запустити драйвери 3D Vision (3D-ігри та відео) у поєднанні з 3D-окулярами та ІЧ-випромінювачем - дуже дорогий пакет.



Рисунок 1.7 – Графічне демо Nvidia GeForce GTX 200

У березні 2011, Nvidia випустила GTX 590, яка об'єднала два повністю функціональні GF110 з однією платою.

Плати працювали з драйвером, який не вмикав обмеження потужності в належному ступені і який був у парі з BIOS, який дозволяв високу напругу. Цей недогляд дозволив транзисторам згорати при високій напрузі. Nvidia виправила ситуацію за допомогою більш стриманого BIOS і драйвера, але

діяльність у день запуску викликала кілька різких відгуків і принаймні одне популярне відео на YouTube. GTX 590 досягла не більше ніж паритету продуктивності з двотижневою HD 6990, власною подвійною картою AMD.

Наступники AMD Northern Islands, Southern Islands, розпочали поетапний графік випуску серії, починаючи з 9 січня, з флагманської карти HD 7970. Це була перша карта PCI-E 3.0 і перший одержувач архітектури AMD GCN, побудованої на 28-нм технології TSMC з мікросхемою Tahiti. Лише через три тижні до 7970 приєдналася друга карта на HD 7950.

Таблиця 1.1 – Порівняння продуктивності (%) 7970 та інших карт в різних іграх

2500x1660	HD6970	HD6990	GTX580	GTX590	GTX580SLI
Alien vs Predator	+23%	-26%	+16%	-18%	-34%
Metro 2033	+35%	-24%	+24%	-9%	-22%
Battlefield 3	+56%	-11%	+24%	-22%	-33%
Modern Warfare 3	+35%	-25%	+18%	-17%	-34%
Crysis 2	+78%	-3%	+39%	-11%	-27%
Dragon Age II	+20%	-31%	+9%	-12%	-30%
Deus EX	+35%	-16%	+43%	-10%	-30%
Hard Reset	+42%	+5%	+16%	-13%	-31%
The Elder Scrolls	+34%	-13%	+8%	-23%	-38%

Одним з бонусів компонентів мікросхеми Tahiti було підтвердження того, що AMD залишила багато невикористаної продуктивності, доступної

через розгін. Це було компромісом між споживанням енергії та тепловіддачею в порівнянні з тактовою частотою, але призвело до консервативної частоти ядра та пам'яті. Потреба максимізувати вихід і недооцінку GTX 680/670 від Nvidia на базі Kepler, можливо, також увійшли до рівняння. Карти були хорошими, але не забезпечили карколомних ігрових покращень у порівнянні з попередніми платами на базі 40nm. Це в поєднанні з менш конкурентоспроможними цінами, які були основним продуктом AMD з часів серії HD 2000, відсутність драйверів WHQL протягом двох місяців і нефункціональний механізм відеокодеків (VCE), стримував ентузіазм багатьох потенційних користувачів і рецензентів.

Nvidia випускає наступного гіганта, під назвою GTX Titan. Відсутність цифрового номера моделі посилює бажання Nvidia розглядати картку як модель окремо від існуючої (і, ймовірно, наступної) серії Kepler. Ціна \$999 Titan орієнтована на ультра-ентузіастів і бенчмаркерів. Nvidia також розширила привабливість для дослідників і професіоналів з обмеженим бюджетом, оскільки це перший раз, коли компанія дозволила карті GeForce зберегти ту саму обчислювальну функціональність, що й її професійні побратими Tesla і Quadro.



Рисунок 1.8 – Графічне демо GTX Titan

Карта чудово показувала себе в ігрових тестах, що особливо помітно в роздільній здатності кількох моніторів із застосуванням суперсемплового

згладжування. Проте байдужа підтримка драйверів OpenCL від Nvidia та сплеск останніх ігор, пов'язаних із програмою AMD Gaming Evolved, пом'якшили вплив Titan так само, як і його непомірна ціна.

У 2013 році Nvidia і AMD, здавалося, боролися за частку ринку дискретної графіки, яка скорочувалася, оскільки розробка ігор і роздільна здатність екрана не відповідали успіхам інтегрованої графіки. На початку 2002 року частка ринку ПК-графіки Intel становила 14%. З появою Extreme Graphics (від 830 до 865 чіпсетів) частка компанії зросла до 33%, потім до 38% з чіпсетами DX 9 третього і четвертого поколінь, а до 2013 року ця цифра становила понад 50% з DX10 GMA. 4500 серія.

Інтеграція графічного процесора в процесор означала, що Intel відповідала за доставку близько 60% графіки ПК.

У 2014 році ми побачили нові архітектури, випущені більшістю основних постачальників, а також низку продуктів, що використовують старі технології. У випадку з AMD їх лінійка майже повністю складалася з попередніх систем. На ринку настільних комп'ютерів з'явилися моделі, які використовують Graphics Core Next (GCN) 1.0 і ще старішу архітектуру TeraScale 2. Хорошим прикладом першого стала Radeon R9 280, випущена в березні.

У 2015 році також були випущені версії програмного забезпечення, які в подальшому сформували напрямок, у якому графічні процесори та їхні постачальники будуть рухатися для своїх майбутніх архітектур. У березні на щорічній конференції розробників ігор Khronos Group публічно назвала проект, над яким працювали – Vulkan.

Цей новий графічний API пропонував значні переваги в порівнянні з OpenGL і Direct3D, в основному у вигляді передачі великої частини управління пам'яттю, потоками і самим GPU розробникам і подалі від

драйверів графічного процесора. Це допомогло б значно зменшити накладні витрати ЦП, з якими боролися існуючі на той момент системи.



Рисунок 1.9 – Порівняння OpenGL та Vulkan

Через чотири місяці Microsoft запустила Windows 10 і разом з нею DirectX 12.

Графічна частина цього API під назвою DirectX3D пропонувала подібні функції, що й Vulkan, хоча вона була обмежена лише новою операційною системою — користувачі зі старими версіями Windows були змушені залишатися з DirectX 11.

Перша гра з DirectX 12, звичайно, була від Microsoft – Gears of War: Ultimate Edition. Тайтл був катастрофічним, з численними помилками та жахливою продуктивністю. Інші ігри, що використовують новий API, мали з'явитися в наступному році, але знадобиться більше часу, щоб програмне забезпечення повністю розкрило свій потенціал.

Одна з його функцій, асинхронні обчислення, викликала особливий інтерес для розробників. Обчислювальні шейдери деякий час були частиною DirectX, вперше з'явилися в DX11 у 2008 році (і через розширення в OpenGL і програмному забезпеченні CUDA від Nvidia). Оброблені спеціально API DirectCompute, ці шейдери працювали на окремому конвеєрі від графічного (наприклад, вершинні, геометричні, піксельні шейдери) і використовували більші обчислювальні можливості загального призначення GPU.

Nvidia продовжить випуск трьох нових графічних процесорів з використанням архітектури Pascal: GP102, 104 і 106. Останні два будуть

працювати на таких моделях, як GTX 1070, 1060 і 1050 — усі вони добре прийняті та популярні. Перший буде використовуватися в найдорожчій графічній карті для настільних комп'ютерів Nvidia з одним GPU свого часу.

Titan X був запущений із ціною 1199 доларів США – на 70+% дорожче, ніж GTX 1080. Цій величезній цифрі відповідали характеристики графічного процесора: 3584 шейдерних одиниць забезпечують до 11 TFLOPS пропускної здатності FP32, 12 ГБ оперативної пам'яті GDDR5X, і 384 ГБ/с пропускної здатності пам'яті.

Але незважаючи на всю цю потужність, він був не на 70% швидше, ніж GTX 1080; багато тестів показали, що карта, в середньому, на 25% краще. Не те, щоб це мало значення, оскільки Titan X продавався так само добре, як і його менші побратими.

Таблиця 1.2 – Порівняння Titan X та GTX 1080

	GTX Titan X	GTX 1080	Різниця
Ціна	999\$	599\$	+66%
Макс. температура	83	80	0%
Шум вентилятора	51.6db	52.8db	+1%
Рекомендоване живлення	530 W	600 W	+13%
Середня продуктивність 1080p	109.6 FPS	97.9 FPS	+11%
Середня продуктивність 1440p	80.8 FPS	71.6 FPS	+9.2%

Продовження таблиці 1.2

Середній продуктивність 4К	48.2 FPS	43.0 FPS	+11%
Пам'ять	12 ГБ	8 ГБ	+33%

В той же час графічні процесори, інтегровані в процесори або системи на чіпі (SoC), також досягли значного прогресу. Хоча Intel не порушувала жодних нових технологічних кордонів зі своїми процесорами Coffee Lake, їхня графічна архітектура досягла Gen 9.5 і, залежно від моделі процесора, до 48 EUs - кожен із них тепер має ширину 7 потоків із до 4 інструкціями. -видається за годинник.

Природно, ігровій продуктивності все ще перешкоджали загальні масштаби графічного процесора, і вони досить погано порівнювалися з новими процесорами AMD на базі Zen з інтегрованими графічними процесорами Vega. Але це показало, що Intel все ще має намір покращити архітектуру.

У світі смартфонів також спостерігався постійний прогрес із їхніми графічними процесорами. Apple успішно використовувала власний дизайн, і хоча елементи системи використовували ліцензовану технологію від PowerVR, вони оголосили, що розійдуться.

Для шанувальників Android Arm запропонувала такі моделі, як Mali-T860, а Qualcomm мав цілком респектабельний графічний процесор у своїй серії мобільних SoC - Snapdragon 600. Навіть дворічний чіп Tegra X1 від Nvidia знайшов своє місце у Nintendo Switch.

Це мав бути ще один «золотий рік» для графічних процесорів. Існувало безліч моделей для будь-якого бюджету та сектора, і AMD і Nvidia здавалися досить добре поєднані, за винятком найвищої сходинки графічної продуктивності.

Однак те, що назрівалося у фоновому режимі протягом кількох років, раптово вибухнуло. Використання графічних процесорів для майнінгу криптовалюти різко зросло через різке зростання ціни біткойн.

Постачання нових графічних процесорів вичерпувались, а ціни на старі карти помітно зросли. Майнінг GPU став справжньою проблемою, тому що графічні процесори були визнані надзвичайно хорошими у масовому виконанні багатьох простих математичних процесів. Графічні процесори AMD були особливо хороші в обчисленнях, хоча Nvidia були більш енергоефективними.

Незалежно від різниці, як у середньому, так і в топових моделях спостерігалось постійне зростання ціни (а також постійна нестача в наявності), яка тривала протягом наступного року.

На тлі глобальної пандемії AMD, Intel і Nvidia випустили нові відеокарти з новою архітектурою та дизайном продуктів. Microsoft і Sony також випустили на ринок свіжі консолі з безліччю нових технологій і функцій, причому перша консолідувала кілька років оновлень API з випуском DirectX Ultimate.

Професійний світ обчислень і штучного інтелекту отримав подібні AMD Radeon Instinct MI100 і Nvidia A100, обидва з гігантськими графічними процесорами (750 і 856 мм² відповідно) з величезною потужністю (120 CU, що дають 23 FP32 TFLOPs або 4316 TFLOP, що виробляють 4316 TFLOP²).).

Перший представив нову архітектуру CDNA від AMD, GCN переродився на ринок лише для обчислень, тоді як Nvidia використовувала новий дизайн Ampere. Він був проданий як пряма заміна Volta, пропонуючи значне підвищення продуктивності для робочих навантажень штучного інтелекту.

Говорячи про штучний інтелект, Nvidia представила покращену версію DLSS у березні, яка використовувала зовсім інший процес, ніж перша ітерація.

Тепер тензорні ядра в графічних картах користувачів оброблятимуть алгоритм висновку для підвищення масштабу зображення, і в цілому нова система була добре прийнята.

Любителям настільних ПК довелось почекати до кінця року, щоб отримати нову партію графічних процесорів, але їх терпіння було винагороджено картами серії GeForce RTX 3000 і Radeon RX 6000. Моделі Nvidia принесли Ampere масам, хоча були значні відмінності між чіпом GA100 в A100 і GA102, які керували лінійкою RTX. Останній був, по суті, оновленням Тьюринга, в якому були вдосконалені ядра CUDA, Tensor і RT.

1.2 Ігрові рушії

Ігровий рушій - це програмний компонент, що дозволяє нам створювати та запускати відеоігри. Він надає розробникам інструменти для створення більшості компонентів гри, а потім дозволяє їм зібрати їх до купи. Рушій гри зачіпає всі компоненти гри, починаючи від рендерингу, фізики, звукового оформлення, скриптингу, створення ІІ та закінчуючи мережевими аспектами. А якщо щось створити з його допомогою і не можливо, ви можете створити це у спеціалізованій програмі та потім імпортувати це у гру.

Трохи нижче буде представлений список популярних рушіїв цього покоління. Це дійсно найкращі представники, що гордо сяють у руках їхніх творців, вони приносять вам той самий ігровий процес, що запам'ятовується. Ось приклад того, як гарні творчі ідеї приносять вам захоплюючий геймплей.

Варто зазначити, що не всі розробники говорять про свої ігрові рушії, багато хто тримає їх найсуворішому секреті. Тому багато хороших рушіїв, про які нам не відомо не потрапили в цей список, хоча вони цього й варті. Приклади таких інкогніто використовувалися та використовуються для створення: Metal Gear Solid 4, FIFA та God of War III.

Найпопулярніший серед ігрових рушіїв на даний момент – це Unity. На ньому можна розроблювати та запускати ігри на Windows, OS X, Windows

Phone, Android, Apple iOS, Linux, Wii, PlayStation 3, PlayStation 4, Xbox 360, Xbox One. До інтерфейсу Unity звикаєш за тиждень. Повністю освоюєш двигун приблизно за 0,5-1 рік. Unity розбитий на кілька вікон: Hierarchy, де ви можете спостерігати назви об'єктів на сцені, групувати їх, Scene, Inspector (налаштування виділеного об'єкта), Project, де видно всі матеріали проекту та ToolBar. Розрахунки фізики здійснює NVIDIA PhysX. Unity використовує C# та JavaScript як інструменти програмування додатків. Unity також має міцний Unity Asset Store, де зберігається широка бібліотека текстур та моделей, які можна використовувати у свої проектах. Unity націлений на розробку невеликих інди-проектів і робити в ньому щось на зразок Mortal Kombat буде дуже скрутним процесом, оскільки двигун не володіє достатньою оптимізацією для PC і Mac.

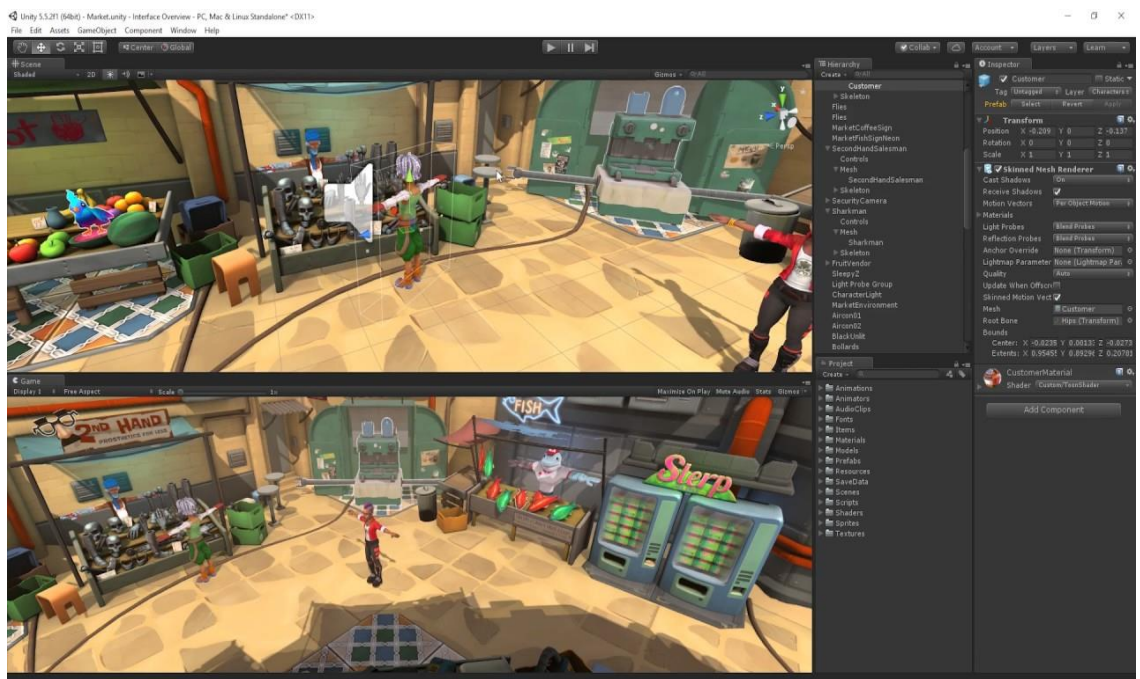


Рисунок 1.10 – Створення сцени в Unity

CryEngine 3 – ігровий рушій, розроблений компанією Crytek. CryEngine, на відміну від інших ігрових двигунів, орієнтований на створення ігор для PC та консолей. Crytek зробили дуже потужний двигун, який допомагає реалізувати найкращу графіку з усіх доступних рушіїв на даний момент. Тут є і динамічне світло, і затінювання і затуманювання в реальному часі, і

управління рівнем деталізації ландшафту. Якщо вам потрібна приємна зовнішня складова вашої гри, то CryEngine залишиться незамінним в цьому випадку. Ігри, що були розроблені на CryEngine: Crysis, Far Cry, Warface, Star Citizen, Ryse: Son of Rome, Homefront: The Revolution, Evolve, Проект Армата, Kingdom Come: Deliverance.



Рисунок 1.11 – Рушій CryEngine

Unreal Engine 4 - найпопулярніший ігровий рушій для створення фільмів та AAA-проектів. Цей двигун має високі графічні можливості. З Unreal Engine 4 можна розробляти ігри під ПК, Mac, консолі, IOS, Android, Windows Phone. На відміну від Unity, UE4 має потужний інструмент для дизайну ігрових рівнів прямо в двигуні та неймовірно зручну систему Blueprint, а також створювати проекти для VR-окуляра. Unreal Engine надає велику кількість документації багатьма мовами, проте їх магазин моделей та текстур відстає від конкурентів і для комфортного користування необхідно купувати моделі та текстури на сторонніх ресурсах. Рушій використовує мови програмування, такі як C++ та UnrealScript C#.



Рисунок 1.12 – Сцена створена за допомогою Unreal Engine

Cocos2d - фреймворк для створення програм та ігор під IOS, Android, Windows Phone. Двигун повністю безкоштовний і має безліч відгалужень: Cocos2d-ObjC, Cocos2d-x, Cocos2d-html5 та Cocos2d-XNA. Найчастіше гейм-девелопери використовують Cocos2d для створення ігор під IOS. Двигун має високу адаптивність, чудово пристосований для розробників-початківців. Якщо ви збираєтеся створювати 2D гру, то краще ніж Cocos2d не знайдете. За продуктивністю він обіграє Unity, але дизайн самого Cocos2d гірший за конкурента. Cocos2D може похвалитися такими проектами: Badland, Antutu Benchmark, Clash of Kings і т.д.

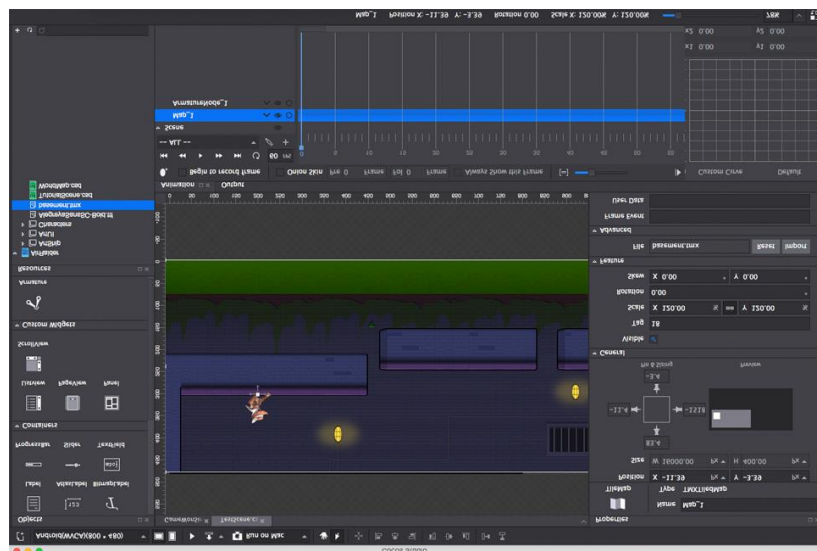


Рисунок 1.13 – Зовнішній вигляд Cocos2d-x

Rage – ігровий двигун, розроблений Rockstar (Rockstar Advanced Game Engine). Rage Engine надає широкий спектр можливостей, включаючи

графічний, фізичний, звуковий і анімаційний двигуни, П, роботу з мережею і т.д. Багато років Rockstar домагались, щоб машина у грі поведилася так само, як і в реальному житті. Зараз система враховує вагу машини, силу зчеплення шин із поверхнею та інші характеристики. Також вони приділили увагу безшовному відкритому світу. Рендер місцевості відбувається дуже швидко: деякі об'єкти віддаляються зі сцени, деякі додаються, а відбувається все за частки секунди. Фізика людини та П в Rage Engine вважаються дуже просунутими. Rockstar купила Euphoria – програмний компонент, який створює анімації “на льоту”. У тій же Grand Theft Auto 5 використовувався цей елемент і саме через нього персонажі вийшли такими «живими».



Рисунок 1.14 – Сцена створена у Rage

1.3 Частота кадрів в секунду

Одним із найпоширеніших критеріїв, що використовуються для вимірювання графічної продуктивності відеоігор, є частота кадрів в секунду. Частота кадрів у відеогрі відображає, як часто зображення, яке ви бачите на екрані, оновлюється для створення зображення та імітації руху/руху. Частота кадрів найчастіше вимірюється в кадрах в секунду або FPS (не плутати зі шутерами від першої особи – first person shooter).

Існує багато факторів, які впливають на визначення частоти кадрів у грі, але, як і в багатьох речах у технологіях, чим вище або швидше щось, тим краще. Низька частота кадрів у відеоіграх призведе до ряду проблем, які можуть виникнути в самий невідповідний момент. Приклади того, що може статися з низькою частотою кадрів, включають поривчасте або стрибкий руї під час послідовностей дій, які включають багато анімацій.

Існує ряд факторів, які впливають на частоту кадрів гри або продуктивність кадрів в секунду (FPS). Області, які можуть вплинути на частоту кадрів і FPS в грі, включають:

- системне обладнання, таке як відеокарта, материнська плата, процесор і пам'ять.
- улаштування графіки та роздільної здатності в грі.
- наскільки добре оптимізований і розроблений код гри для графічної продуктивності.

Найбільшим фактором, що впливає на частоту кадрів гри або продуктивність FPS, є відеокарта та процесор. Центральний процесор комп'ютера надсилає інформацію або інструкції від програм, додатків, у даному випадку гри, на відеокарту. Потім відеокарта, у свою чергу, оброблятиме отримані інструкції, відтворює зображення та надсилає його на монітор для відображення.

Між процесором і графічним процесором існує прямий зв'язок, при цьому продуктивність вашої відеокарти залежить від ЦП і навпаки. Якщо процесор недостатньо потужний, не має сенсу оновлюватися до останньої та найкращої відеокарти, якщо він не зможе використовувати всю свою потужність обробки.

Немає загального правила для визначення найкращої комбінації відеокарта/процесор, але якщо 18-24 місяці тому процесор був середнього та низького рівня, є велика ймовірність, що він уже знаходиться на нижньому

рівні мінімальних системних вимог. Насправді, через 0-3 місяці після придбання значна частина апаратного забезпечення вашого ПК, ймовірно, буде перевершена новим і кращим обладнанням. Головне — спробувати знайти правильний баланс із графікою та налаштуваннями роздільної здатності гри.

Більшість відеоігор сьогодні розроблено з метою досягнення частоти кадрів 60 кадрів в секунду, але прийнятною вважається частота від 30 до 60 кадрів в секунду. Це не означає, що ігри не можуть перевищувати 60 кадрів в секунду, насправді багато хто робить, але при частоті нижче 30 кадрів в секунду анімація може почати перериватися і показувати відсутність плавного руху.

Фактичні кадри в секунду, які ви відчуваєте, змінюються протягом гри в залежності від обладнання та того, що може відбуватися в грі в будь-який момент. З точки зору апаратного забезпечення, як згадувалося раніше, ваша відеокарта та процесор відіграють роль у частоті кадрів в секунду, але також ваш монітор може впливати на FPS, який ви зможете бачити. Багато ПК-моніторів мають частоту оновлення 60 Гц, що означає, що все, що перевищує 60 FPS, не буде видно.

У поєднанні з вашим обладнанням такі ігри, як Doom (2016), Overwatch, Battlefield 1 та інші, які мають інтенсивні графічні послідовності дій, можуть впливати на FPS у грі через велику кількість рухомих об'єктів, фізику та обчислення гри, 3D-середовища тощо. Новіші ігри також можуть вимагати більш високих версій моделі шейдерів DirectX, яку може підтримувати відеокарта, якщо GPU не відповідає вимогам моделі шейдерів, часто може виникнути низька продуктивність, низька частота кадрів або несумісність.

Для вимірювання частоти кадрів або кадрів за секунду відеоігри під час гри є ряд інструментів і програм. Найпопулярніший і який багато хто вважає найкращим називається Fraps. Fraps — це окрема програма, яка працює поза

лаштуваннями для будь-якої гри, яка використовує графічні API DirectX або OpenGL (інтерфейс програмного забезпечення) і служить утилітою порівняння, яка відображатиме ваші поточні кадри в секунду, а також вимірює FPS між початковою та кінцевою точкою. . На додаток до функціональних можливостей порівняльного аналізу, Fraps також має функції для зйомки скріншотів ігор і відеозапису в грі в реальному часі. Хоча повна функціональність Fraps не є безкоштовною, вони пропонують безкоштовну версію з обмеженнями, які включають порівняльний аналіз кадрів у секунду, 30 секунд відеозйомки та скріншоти .bmp.

Існують деякі альтернативні програми Fraps, наприклад Bandicam, але вам доведеться заплатити і за них, якщо ви хочете повну функціональність.

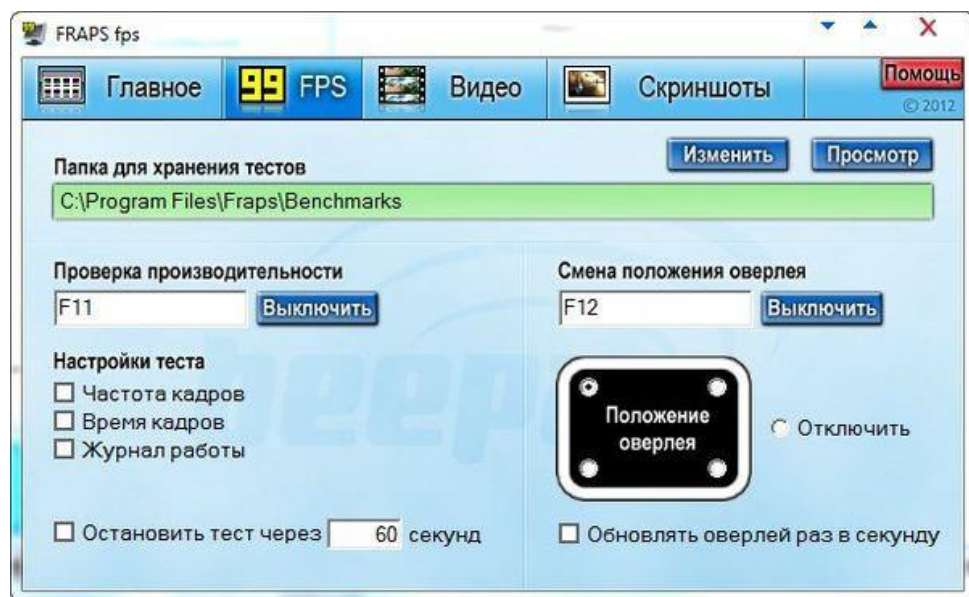


Рисунок 1.15 – Програма FRAPS

1.4 Оптимізація в ігрових додатках

Оптимізація в ігрових дискусіях не означає те саме, що означає в інформатиці. Замість того, щоб говорити про те, щоб зробити процес оптимальним, це, як правило, означає покращити плавність речей в грі. Основна відмінність полягає в тому, що в інформатиці оптимізація вважається такою лише в тому випадку, якщо вона дає точно такий же результат. Як дуже

простий приклад, якщо ви хочете обчислити: $a = b * c + b * c + b * c$. Натомість ви розраховуєте: $d = b * c$, і потім $a = d + d + d$.

Це, свого роду, споріднено із «оптимізацією», яку ми знаємо зараз — вона вимагає лише одного множення, а не трьох для того самого результату. Проте за цим суворим визначенням ідея порівняння рівня оптимізації в різних іграх, які намагаються досягти різних речей, стає абсолютно непрактичною.

Отже, як ми говоримо про оптимізацію гри, зберігаючи певний рівень об'єктивності? Ось визначення цьому, яке дає QLOC: «хороша оптимізація означає, що гра працює з однаковою частотою кадрів у широкому діапазоні апаратних характеристик, у тому числі в бюджетних конфігураціях». Підтримка надійної частоти кадрів – це те, у чому вони бачать важливість графічних налаштувань, оскільки вони можуть подолати великий технологічний розрив між ПК низького та високого класу.

З одного боку, ні, ви не можете порівнювати, оптимізовані різні ігри чи ні, але ви можете скласти власне судження, дивлячись на якість результату, отриманого двома різними іграми, і відносну продуктивність, яку вони досягають. Хоча під поверхнею також є багато речей, які впливають на те, як гра може працювати.

Які речі? Зрозуміти оптимізацію та те, що на неї впливає, означає відповісти на це запитання. Ми почнемо з оцінки впливу на продуктивність графічних ефектів, які особливо вимогливі до поточного обладнання.

Якщо ви не отримуєте бажаної продуктивності від гри, є велика ймовірність, що винні її налаштування. Це може бути найбільшою перешкодою для досягнення плавного зображення, і ви маєте налаштувати її власноруч, щоб досягти цілі 60 кадрів в секунду.

Можливо, для оптимізації порівняння також дуже легко врахувати єдиний найбільш інтенсивний параметр: роздільну здатність візуалізації. Оскільки він зазвичай зводиться лише до двох чисел (наприклад, 1920x1080) і

доступний майже в усіх іграх. Важливим фактом, який іноді не помічають, є те, що зусилля візуалізації масштабуються відповідно до кількості затінених пікселів — 4k приблизно в чотири рази більше 2k, а не в два рази.

Згладжування є набагато складнішою темою, коли справа доходить до порівняння. У сучасній грі ви можете знайти будь-яке з наступного:

- методи чистого простору екрана, як-от FXAA або SMAA (1x), що дивляться на остаточне відтворене зображення та намагаються знайти й пом'якшити артефакти згладжування.
- методи згладжування тимчасового накопичення, що відтворюють зображення з невеликим зміщенням кожного кадру і накопичують результати, намагаючись компенсувати рух.
- мультисемплінг (MSAA), що на різних рівнях передбачає взяття кількох вибірок на піксель на краях багатокутника.
- суперсемплінг (SSAA), що на різних рівнях використовує кілька вибірок для всіх пікселів.
- комбінації перерахованого вище (наприклад, TXAA).

Вони перераховані вище в порядку зростання для очікуваного впливу на продуктивність, за винятком категорії комбінації, яка залежить від вартості її компонентів. Методи екранного простору, як правило, дешеві, чисте тимчасове накопичення є ресурсозатратним, за яким слідує мультисемплінг і, нарешті, суперсемплінг. Мультисемплінг представляє особливу складність у порівнянні продуктивності, оскільки його вартість у продуктивності може значно відрізнятись в залежності від реалізації, від дуже економічно ефективного до майже супервибірки. Суперсемплінг відносно легко зв'язати з роздільною здатністю: суперсемплінг Nx за вартістю подібний до візуалізації з вищою роздільною здатністю в N разів.

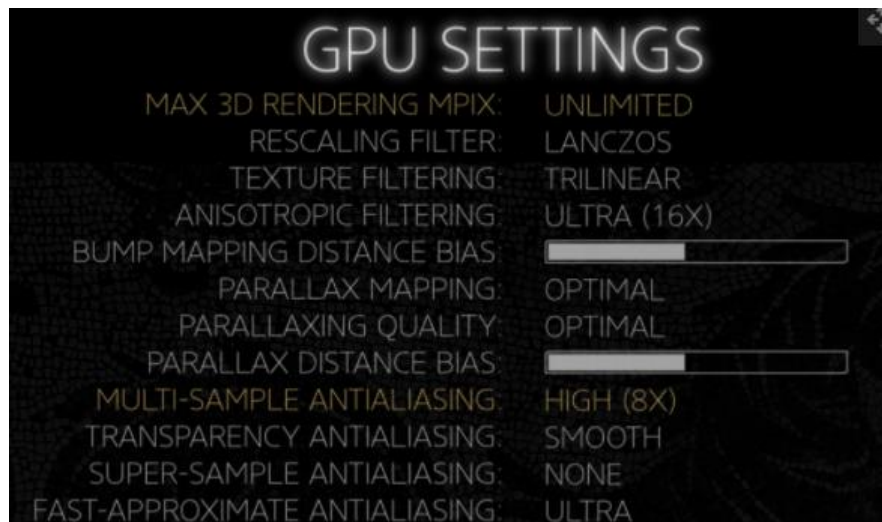


Рисунок 1.16 – Меню опцій згладжування у The Talos Principle

1.4.1 Освітлення та затемнення

Поверхні та атмосфери, що реалістично реагують на світло, є основою 3D-графіки і водночас її святим Граалем. Більш складні ефекти постійно розробляються, і з огляду на точність, доступну сьогодні, подальші помітні покращення часто супроводжуються значною вартістю продуктивності.

З точки зору освітлення, майже все, що вважається об'ємним або наближеним до глобального освітлення, часто виправдано є ресуртновитратним. Це включає в себе високоякісні реалізації так званих «променів бога» (god-rays), які намагаються відтворити ефект яскравого світла, що висвітлює частинки повітря або пил перед темнішим фоном.

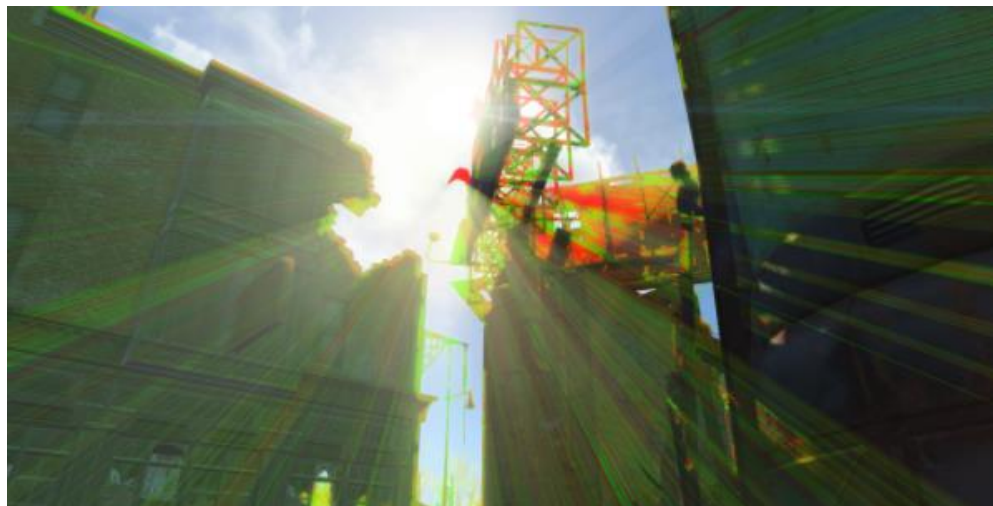


Рисунок 1.17 – «Промені бога» у Fallout 4

Там, де є світло, повинні бути і тіні, але, на жаль, це відбувається не так природно в іграх, як у реальному світі. У сучасних іграх зазвичай є принаймні два джерела тіней у реальному часі: оклюзія навколишнього середовища, яка намагається врахувати загальне зниження освітленості, як правило, ґрунтуючись на оточенні близької відстані, і пряме затемнення.

Амбієнтна оклюзія, коли вона була введена в Crysіs, сама по собі була значно ресурсоемним ефектом. У наші дні високоякісна оклюзія простору екрана більше не є великою витратою продуктивності. Однак нові методи, такі як воксельна оклюзія або оклюзія на основі полів відстані, які спрямовані на усунення проблем, властивих методам екранного простору, знову мають значний вплив на продуктивність на даний момент часу.

При прямих тінях контактне зміцнення важко повторити в іграх. Це описує той факт, що контури тіні здаються чіткішими і чіткішими поблизу від заклинача, і поступово м'якше, чим далі від нього віддаляється точка, де відкидається тінь. Існує ряд різних реалізацій цієї поведінки (наприклад, PCSS), проте тиражування такого ефекту на вашому комп'ютері потребуватиме значних ресурсів.

Одна важлива річ, яку слід пам'ятати під час обговорення оптимізації: все, що реагує на непередбачувані зміни в реальному часі — переміщення світла чи зміна інтенсивності, рух гравців або, у гіршому випадку, навіть зміна місцевості — набагато інтенсивніше, ніж поведінка світла та тіні, які вже були попередньо обчислені. Якщо в грі є динамічні цикли денного світла або деформація рельєфу, це потрібно ретельно враховувати при обговоренні ефективності її рішень для освітлення та тіні.

1.4.2 Різні ефекти

Багато інших ефектів, які не вписуються в жодну з категорій вище, стосуються імітації оптичної поведінки системи камери. Вони часто не надто

інтенсивні, за винятком високоякісних ефектів глибини «боке». Чим більш розмитою має бути розфокусована частина зображення, тим вище потенційна вартість продуктивності.

Відносно недавня розробка, яка часто зустрічається разом із фізичним рендерингом (PBR), — це відображення простору екрана. Це потребує менше ресурсів, ніж традиційні відображення, які потребують повторного відтворення великих частин сцени, але все одно можуть мати значний вплив на продуктивність. Це подвійно вірно, якщо вони моделюють передові характеристики, такі як контактне зміцнення.

1.5 Технічні аспекти оптимізації

Тепер, коли ми маємо деяке уявлення про деякі з найдорожчих ефектів у сучасних іграх, ми можемо розібратися в тому, що потрібно для створення гри, яка, сподіваємося, буде вважатися оптимізованою. Одним із аспектів оптимізації буде, звісно, наявність графічних пресетів у налаштуваннях гри.

Графічні пресети — звичайні «низький», «середній», «високий», «дуже високий», «ультра» — майже ніколи не можна порівняти безпосередньо в іграх, але вони дуже важливі: вони направляють гравців, які не хочуть занурюватися глибоко в індивідуальні налаштування кожного елемента.

Деякі розробники встановлюють «середні» налаштування на початку розробки гри, виходячи з обмежень та очікувань обладнання, і весь дизайн та ілюстрація відповідають цьому стандарту. Ближче до випуску технічна команда потім визначає налаштування для кожного пресета, намагаючись збалансувати графічну пишність і продуктивність на кожному рівні.



Рисунок 1.18 – Меню налаштувань продуктивності за категоріями у The Talos Principle

Для QLOC, які в основному займаються портуванням існуючих консольних ігор на ПК, стандартні налаштування консолі зазвичай перекладаються на «середні» попередні налаштування версії для ПК, хоча деякі аспекти можуть бути змінені, щоб врахувати істотні відмінності платформи. Потім надаються параметри масштабованості, наскільки це можливо. Як попередні налаштування, так і вимоги постійно оцінюються протягом усього процесу оптимізації, починаючи з моменту тестування та роботи засобу візуалізації та інших основних функцій.

Геймерам часто невідомо, як пов'язані пресети та вимоги до обладнання. Хоча кожен розробник має свої власні стандарти, для Croteam (розробники The Talos Principle) «мінімальні» специфікації означають, що гра буде добре працювати на низьких налаштуваннях, і якщо ці вимоги не будуть дотримані, повна технічна підтримка не надаватиметься. «Рекомендовано», з іншого боку, означає, що гру можна грати за призначенням з роздільною здатністю 1080p.

1.6 Оцінювання оптимізованих ігор

Тепер розглянемо, як ми (гравці) можемо оцінити якість оптимізації тієї чи іншої гри на кількох прикладах.

Таким предметом для дискусії є *Dying Light*, і помилковий прийом оптимізації, який він отримав. *Dying Light* — гра з відкритим світом із великою кількістю рухомих персонажів і частин, а також динамічним циклом день/ніч, усі інгредієнти, які складають технічно складну гру. Він постачається з величезним діапазоном налаштувань, зокрема для відстані малювання, що в цьому типі досвіду сильно впливає як на CPU, так і на GPU.



Рисунок 1.19 – Дальність промалювання у *Dying Light*

Dying Light не відповідав (довільним) стандартам на максимальних налаштуваннях. Як виявилось, повзунок дистанції малювання в початковій версії гри вже був вище налаштувань консолі в найнижчому положенні, а піднімався незрівнянно вище. Через це розробникам довелось зменшити діапазон повзунка до 55% від його колишнього максимуму в ранньому виправленні.

Чи вважали б гру набагато більш «оптимізованою», якби цей тривіальний крок був зроблений до випуску? Звісно так. Чи насправді він був би «краще оптимізований»? Ні, абсолютно ні. *Dying Light* є чудовим

прикладом того, наскільки важко судити про оптимізацію (через велику кількість аспектів), а також про те, що розробники можуть обмежити під час впровадження параметрів гри.

Подібна проблема виникла із Deus Ex: Mankind Divided, в якій помітно показувалося налаштування MSAA (технологія мультипробового згладження). Оскільки в грі використовується відкладене затінення — поширена техніка візуалізації, яка робить просту реалізацію MSAA з апаратним прискоренням набагато менш життєздатною — це налаштування принесло надзвичайний вплив на продуктивність, який можна було б очікувати від такої візуалізації. Знову ж таки, гра могла б вважатися набагато більш «оптимізованою», якби вона не включала цю опцію.

Ігри рідко є єдиним монолітним об'єктом, який є оптимізованим або неоптимізованим. Можуть бути окремі ефекти, які не дуже добре оптимізовані, в той час як велика її частина працює чудово на будь-яких пристроях. Це особливо часто зустрічається з новими функціями: коли оригінальний Crysis вперше включив оклюзію навколишнього середовища, ефект не був особливо оптимізований порівняно з сучасними реалізаціями. Аналогічно, перші реалізації АО на основі вокселів у Rise of the Tomb Raider, що можуть бути оптимізовані краще в майбутньому із розвитком технології.

Metro 2033 була однією з перших ігор, у якій широко реалізовано об'ємне освітлення, і вважалася духовним наступником Crysis за його впливом на продуктивність на більш високих налаштуваннях. Те ж саме стосується останніх і майбутніх реалізацій «зміцнення контактних тіней». Експерименти із новими функціями є важливими, навіть якщо їх початкові реалізації можуть бути непридатними для широкого використання через вплив на продуктивність. Таким чином ігри прогресують до більш просунутих, більш оптимізованих ефектів у майбутньому. Принадність платформи ПК полягає в тому, що те, що було «неоптимізованим» ефектом у 2010 році, можна

відтворити з чудовою частотою кадрів і зробити гру краще навіть на обладнанні середнього рівня у 2021 році.

Звичайно, існують ігри, які в цілому є якісними і справді неоптимізованими. Зазвичай для такого можуть бути свої причини: обмежені ресурси або брак технічних навичок розробників. Тим не менш, коли гра гальмується навіть за умовчанням на потужному обладнанні або порівняно проста 2D-гра падає нижче 20 FPS на сучасній консолі, тоді навіть найдобріший аналіз не залишає місця висновків, що гра банально не оптимізована.

1.7 Задачі оптимізації на етапі розробки

Фактичний процес справжньої оптимізації коду (у значенні практичної інформатики) є складним. Для більшості розробників «найгірший кошмар оптимізації» — це «подивитися на вихід профайлера (інструмент, що повідомляє програмісту про час обробки функції ділянки коду) й побачити, що верхня функція займає 3% часу».

Коли верхня функція займає 3% часу, це означає, що навіть якщо вам вдасться покращити її, щоб вона була вдвічі швидшою — що може вимагати величезних зусиль — повна програма пришвидшиться лише на 1,5%. Оскільки такі ситуації є поширеними у великих, грамотно написаних кодових базах, їх подальша оптимізація стає поступовим, трудомістким завданням.

QLOC, який переносять ігри на основі широкого спектру технологій, бачить, що їхні проблеми сильно відрізняються в кожному проекті. «В одній грі все може бути дуже складно, а в іншій – набагато менше». Один загальний, складний аспект, який вони ідентифікують, — це переконатися, що гра добре працює на дуже широкому діапазоні обладнання.

2 МЕТОДИ ОПТИМІЗАЦІЇ ВИСОКОПОЛІГОНАЛЬНИХ МОДЕЛЕЙ

2.1 Алгоритми видалення невидимих граней та ліній

В цьому розділі на практиці буде показано, як зміна графічних моделей може посприяти оптимізації ігрової сцени.

На сучасному етапі розвитку 3D редакторів та засобів моделювання стало досить легко створювати деталізовані 3D моделі для використання в різних програмах, що працюють з тривимірною графікою. Особливо активно деталізовані моделі використовуються у виробництві спецефектів для кінофільмів або в іграх. Однак при спробі використовувати ці моделі в інтерактивних програмах для мобільних пристроїв, симуляторах та ГІС виникають проблеми. Основні перешкоди полягають у необхідності виділення великого обсягу обчислювальних ресурсів, складності обробки об'єктів. Також наявність великої кількості полігонів у сцені або моделі обмежує можливість запуску програми на пристроях з низькою обчислювальною потужністю. Таким чином існує проблема, що вимагає оптимізації вже готових тривимірних деталізованих моделей для використання в додатках на пристроях з низькою обчислювальною потужністю.

Виділяють три класи алгоритмів "видалення невидимих граней і ліній":

Алгоритми, що працюють у просторі екрану. Вони засновані на розрахунку влучення граней у зону видимості спостерігача з певної точки.

Алгоритми, які працюють у просторі об'єктів (сцени). Найчастіше в моделях існують грані, які розташовуються всередині об'єкта, або на звороті прихованої від спостерігача. Для обчислення видимих поверхонь використовується порівняння та розрахунок розташування всіх об'єктів що знаходяться в сцені та видалення всіх невидимих граней.

Алгоритми зі списком пріоритетів. Є гібридною модифікацією перших двох і працюють, використовуючи обидва алгоритми поперемінно.



Рисунок 2.1 - Приклад використання алгоритмів видалення невидимих граней для користувача в звичайному режимі

До переваг алгоритмів видалення невидимих граней і ліній можна віднести високий рівень оптимізації та відсутність втрат якості видимого для спостерігача. Недоліками є низька швидкість обробки, високе споживання обчислювальних ресурсів і відсутність можливості обробки в реальному часі на більшості пристроїв.

2.2 Алгоритми редукції полігонів

Редукція – процес спрощення 3D моделі шляхом зменшення кількості полігонів. Принцип роботи методу полягає у заміщенні групи полігонів одним, найбільш близьким до вихідної групи за розташуванням вершин. Більшість 3D редакторів мають можливість задавати інтенсивність редукції, що дозволяє вибрати потрібне співвідношення між рівнем опрацювання моделі та економією обчислювальної потужності.

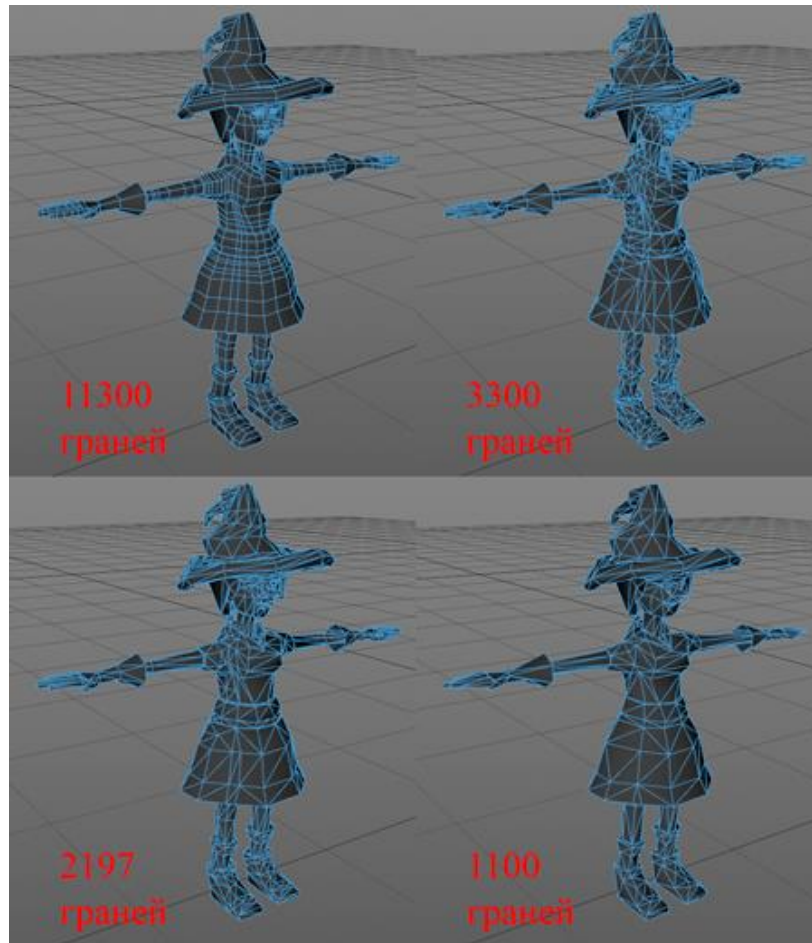


Рисунок 2.2 - Метод редукції полігонів на прикладі персонажа та вибір оптимальної конфігурації.

До переваг відносять легкість використання та високу швидкість обробки. Однак недоліками є спотворення моделі при сильному або неправильному застосуванні методу та необхідність перемальовування текстур, якщо на первинну модель вже була накладена текстура.

2.3 Алгоритми ручного створення низькополігональних моделей

Низькополігональні (low-poly) моделі використовуються коли не потрібна висока деталізація, якщо прийнятну якість зображення можна отримати, за допомогою опрацьованих текстур, карт нормалей та інших візуальних ефектів. Найчастіше такі моделі використовуються для економії обчислювальних ресурсів у 3D-додатках, де є необхідність відображення моделей та анімації в режимі реального часу.

При створенні низькополігональних моделей слід прагнути за мінімально можливу кількість вершин і ребер створити модель об'єкта досить схожу на реальний прототип. Існує кілька способів створення візуального представлення тривимірного об'єкта з мінімальними втратами якості:

Завдання оптимальної кількості полігонів (достатнє для збереження форми об'єкта, але не зайве). Слід враховувати сферу застосування моделі та доступні ресурси обчислювальної техніки.

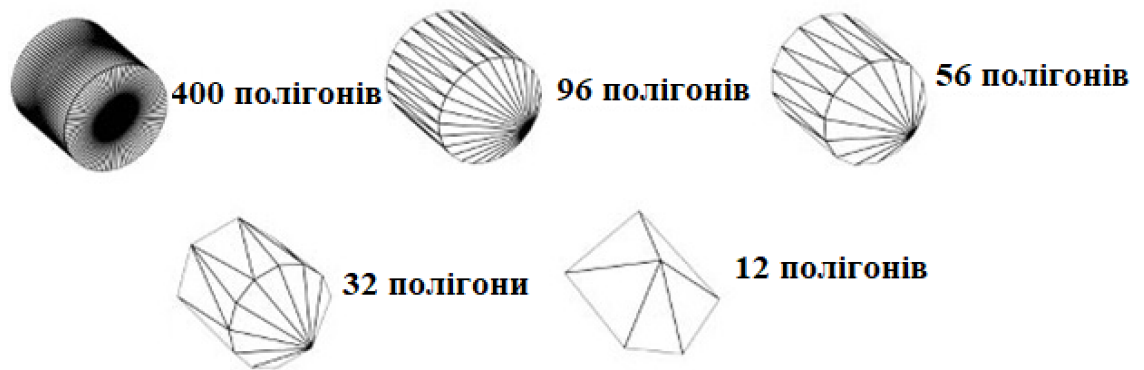


Рисунок 2.3 - Приклади завдання примітиву різними кількостями полігонів

Злиття сусідніх близьких вершин. У ряді випадків модель може мати кілька вершин зосереджених на малій відстані один від одного або що знаходяться на одній прямій. Найоптимальнішим рішенням у разі буде об'єднання всіх вершин в одну, що зменшить кількість дисфункційних полігонів.

"Заокруглення" шляхом зниження кількості граней на сторонах циліндричних об'єктів. Деякі сцени з використанням таких об'єктів не мають на увазі необхідності високої деталізації самої поверхні циліндра. Саме в таких випадках слід застосовувати скруглення, оскільки візуальне сприйняття циліндра слабо залежить від його граней.

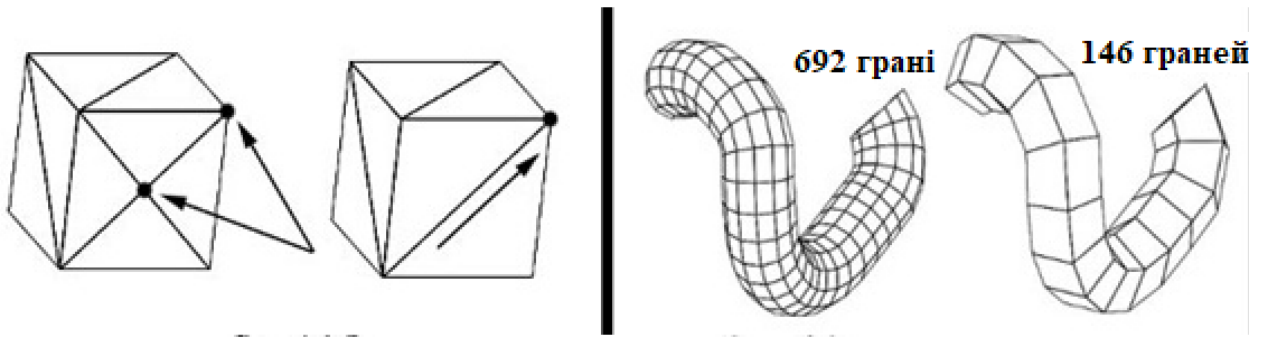


Рисунок 2.4 - Приклади використання методів "Злиття вершин" та "Закруглення".

Використання трикутних полігонів для моделювання сфер. Незважаючи на більш складне розташування полігонів у порівнянні зі сферою з чотирикутних полігонів, сфера із трикутників дозволяє обійтися меншим числом полігонів при меншій мірі апроксимації.

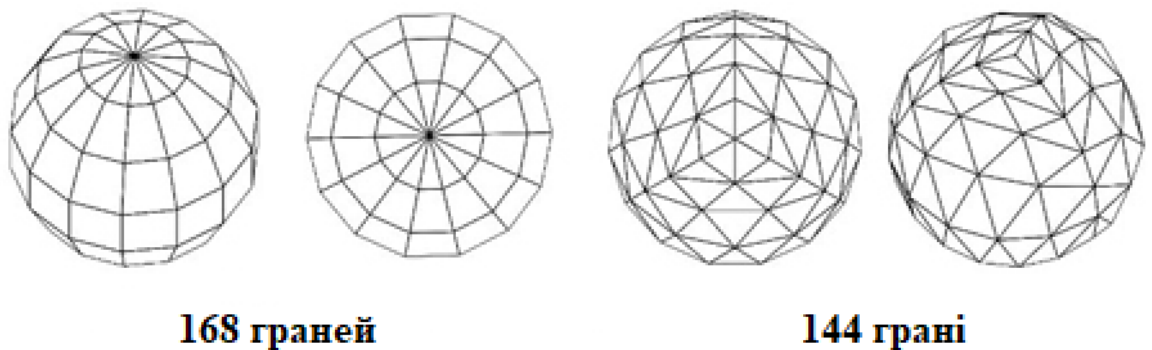


Рисунок 2.5 - Демонстрація відмінності сфер із трикутних та чотирикутних полігонів

До переваг ручного створення низькополігональних моделей відносять дуже високу фінальну якість та мінімальну кількість полігонів. Ключовими недоліками є висока трудомісткість та дуже низька швидкість моделювання.

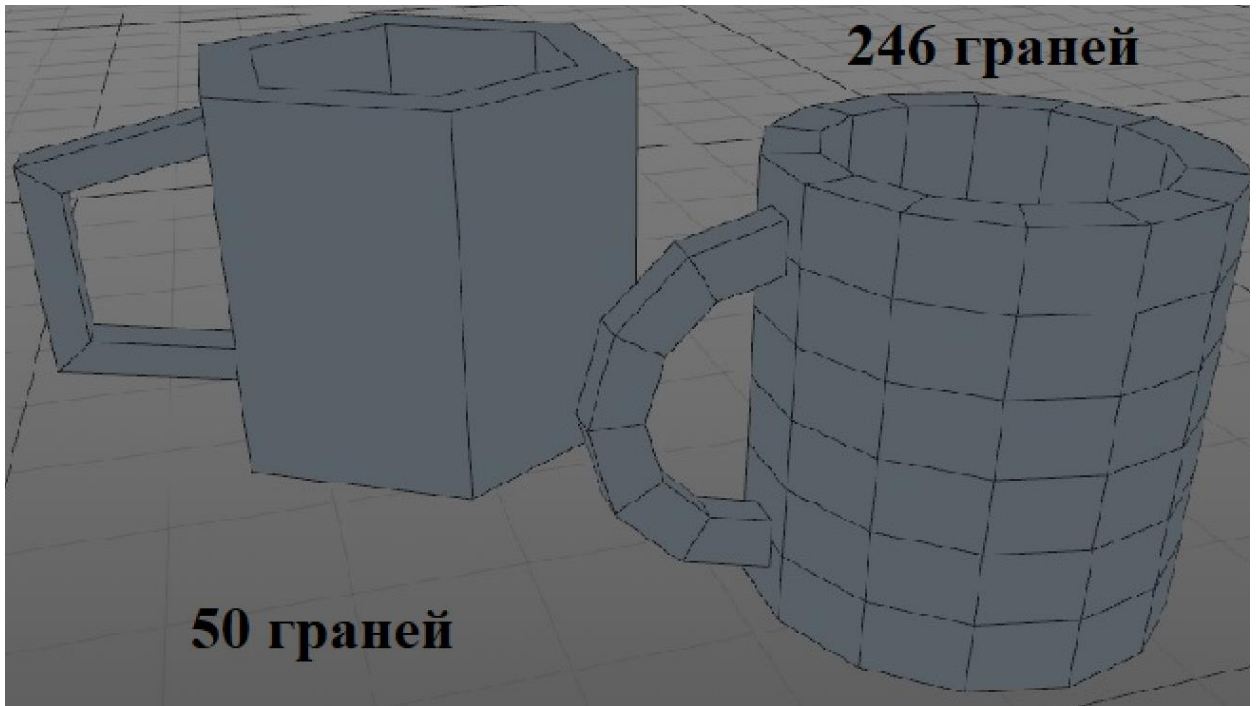


Рисунок 2.6 - Порівняння високополігональної моделі гуртка (246 граней) та низькополігональної (50 граней).

2.4 Тестова сцена

Для того щоб оцінити ефективність оптимізації високо деталізованих 3D моделей, була створена сцена з різних типів об'єктів, до яких були застосовані описані вище методи та алгоритми.



Рисунок 2.7 - Оптимізована сцена для запуску на пристроях із низькою кількістю обчислювальних ресурсів.

Наприклад, до будівель був застосований алгоритм видалення невидимих граней і ліній, для оптимізації персонажів використовувався алгоритм редукції полігонів, а предмети інтер'єру були заново змодельовані з використанням техніки створення низькополігональних об'єктів. Таким чином вдалося досягти значного зменшення кількості полігонів у сцені, що наочно показано у таблиці 2.1.

Таблиця 2.1 – Зниження кількості полігонів за допомогою застосування різних технологій оптимізації графічних моделей

Кількість полігонів	З повною мальовкою	Оптимізований режим
Будівля	7460	2062
Персонаж	11300	3300
Предмети інтер'єру	2536	688
Усього	21296	6050

Тести та оцінки продуктивності проводилися на базі Unity 3D. Зрештою скорочення кількості полігонів з 21292 до 6050 дало значний приріст продуктивності та зменшення навантаження на центральний та графічний процесор пристрою.

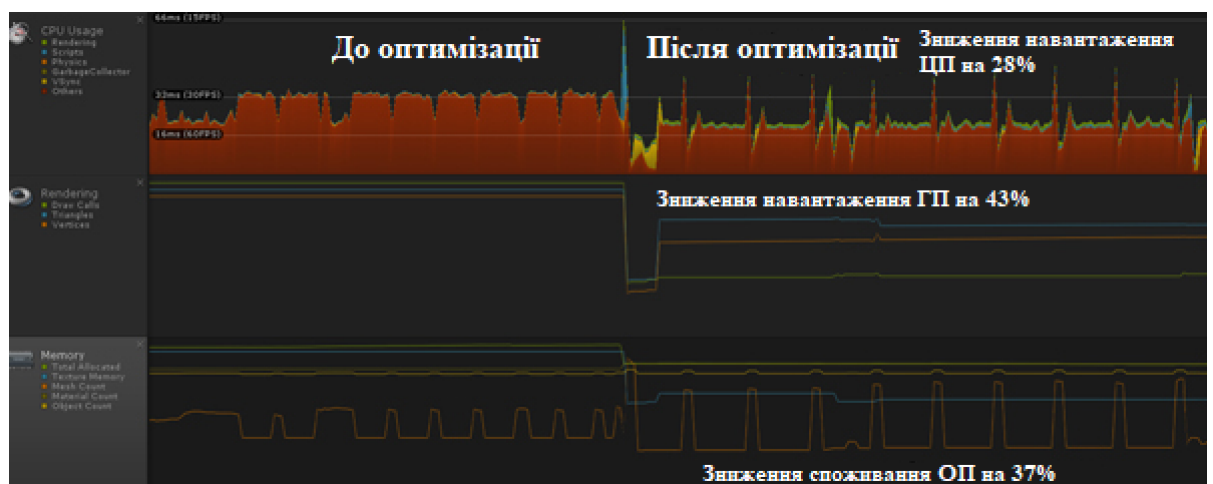


Рисунок 2.8 - Графік навантаження на центральний та графічний процесор до та після оптимізації

В результаті використання методів оптимізації описаних у розділі високополігональна сцена була перетворена на низько деталізовану. Кінцевим результатом маніпуляцій стало зменшення навантаження центрального та графічного процесора на 28% та 42% відповідно. Це дало можливість запуску сцени на мобільному пристрої з обмеженими обчислювальними ресурсами, що показує ефективність цих методів та перспективи розвитку напряму загалом.

3 ШЛЯХИ ОПТИМІЗАЦІЇ ПК ТА ІГРОВИХ ДОДАТКІВ

3.1 Оптимізація ПК

Перш ніж зазирнути в глибини поліпшення плавності картинки за допомогою різного софту та редагування вихідних файлів ігор, ми постараємося також налаштувати те, що лежить на поверхні і трохи покращить плавність зображення на екрані в будь-яких іграх. Це шляхи забезпечення оптимізації ігрових додатків за допомогою стандартних налаштувань Windows та допоміжних додатків.

Ми не можемо вступати в бій, вирушати на гоночний трек або вилітати в космос, не встановивши на вашому графічному процесорі останні графічні драйвери Nvidia або графічні драйвери AMD. Це одна з найважливіших перевірок перед грою, яку ми можемо зробити, і вона також проста.

На нашій системі вже є інструмент автоматичного оновлення Nvidia і ми можемо перевірити його, знайшовши його на панелі завдань. Якщо ж його немає, нам необхідно завантажити додаток Nvidia (або AMD, залежно компонентів ПК), який аналізуватиме обладнання нашої системи та отримає потрібні драйвери з Інтернету. Ці драйвери також доступні для ручного завантаження та встановлення на сайтах Nvidia або AMD.

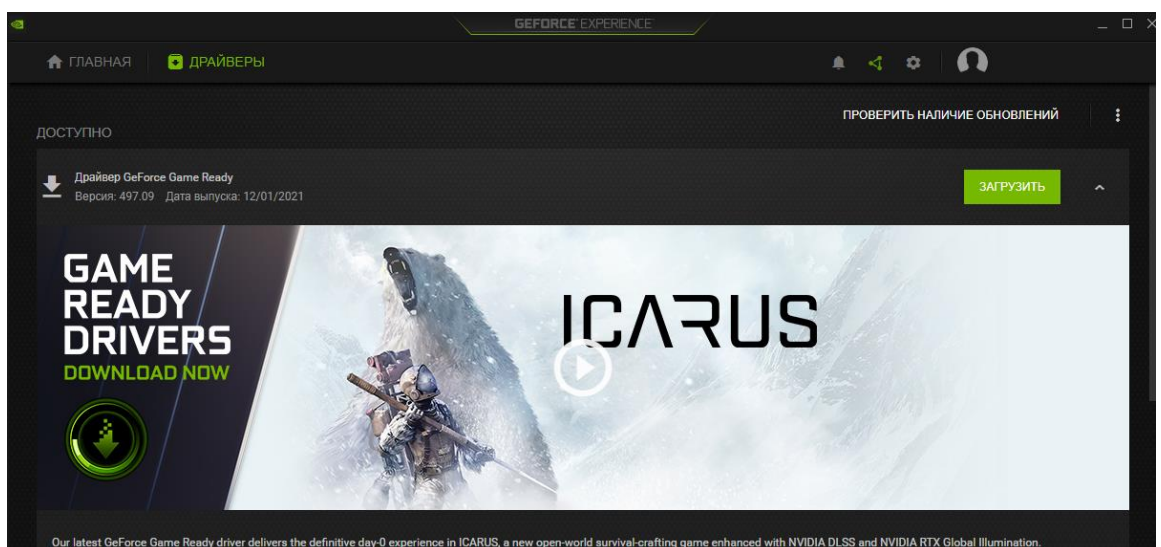


Рисунок 3.1 – Nvidia GeForce

Як ми можемо побачити малюнку 2.1, наші драйвери не оновлені і ми можемо завантажити нову версію драйверів і встановити їх за допомогою кнопки "Завантажити".

Якщо ви не впевнені, яку відеокарту ви встановили, знайдіть «диспетчер пристроїв» на панелі завдань і запустіть програму, яка відображається як найкраща. Відкрийте список адаптерів дисплея, щоб побачити свій графічний процесор — ви навіть можете клацнути правою кнопкою миші запис і вибрати «Оновити драйвер».

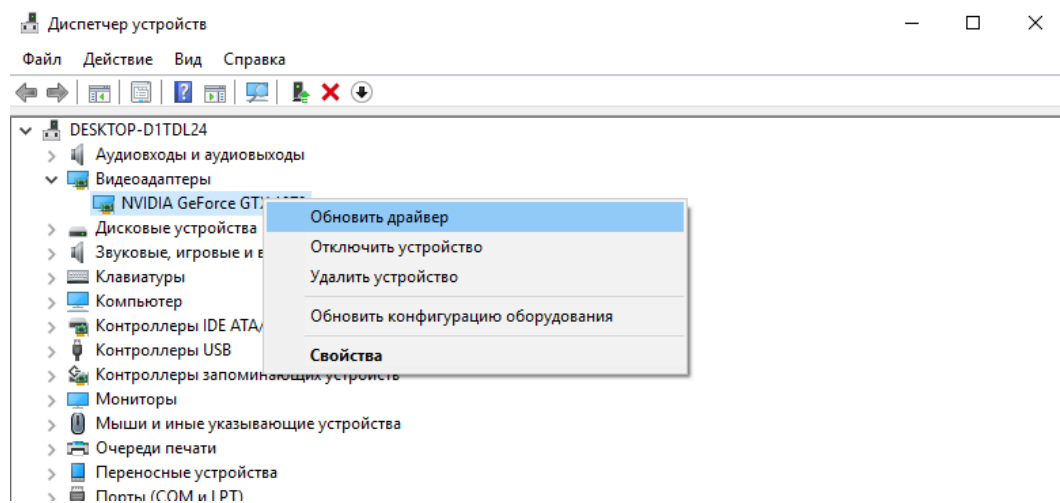


Рисунок 3.2 – Оновлення драйверів через диспетчер пристроїв

Драйвери забезпечують зв'язок між вашим графічним обладнанням і програмним забезпеченням у вашій системі (включаючи встановлені ігри). Чим новіші та сучасніші ваші драйвери, тим швидшим і плавнішим буде зв'язок. Це також стосується оновлень програмного забезпечення на рівні ОС — переконайтеся, що перевірте, чи є у вас якісь, які очікують на розгляд за допомогою Оновлення та безпеки в налаштуваннях Windows.

Хороша продуктивність в іграх також залежить від швидкодії системи. Якщо система засмічена непотрібними файлами та забитим кешем, це може певною мірою впливати на швидкість запуску та роботи додатків. Таке очищення можна виконати як вручну, так і за допомогою допоміжних програм на кшталт CCleaner.

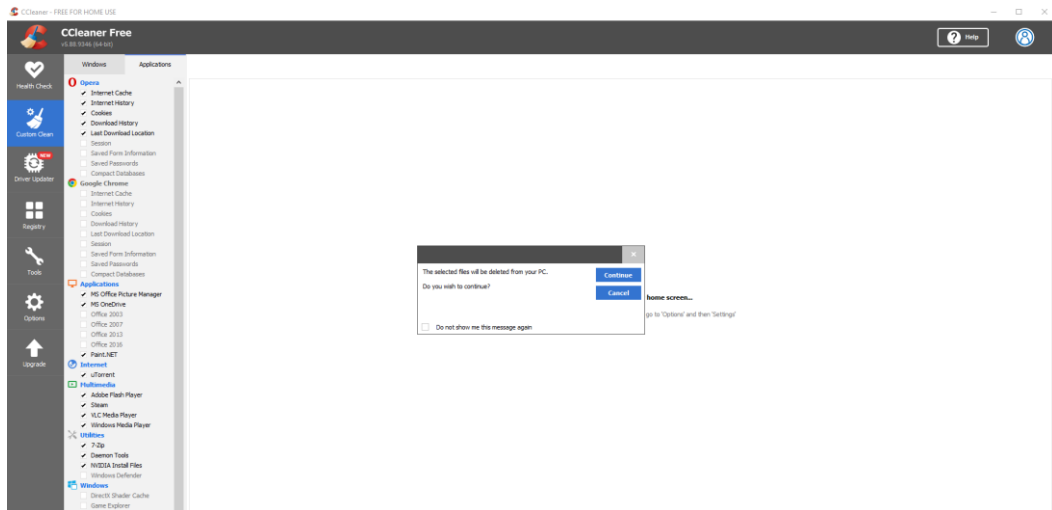


Рисунок 3.3 – Очистка непотрібних файлів системи в CCleaner

Зачищаємо непотрібні файли кешу і можемо перейти до чищення дисків. Головне тримати в чистоті диск С, на якому розташовані файли нашої системи, щоб не перевантажувати вільний доступ до необхідних компонентів системи, під час запуску додатків.

Ігри обробляють величезні обсяги даних, коли ви блукаєте по створеним ними віртуальним світам, тому чим більше вільного місця на диску ми зможемо їм запропонувати, тим краще. Якщо у вашій системі мало місця для зберігання, можливо, ви вже помічали, що ігри починають працювати з труднощами та уповільнюються в результаті.

Переконайтеся, що на нашому жорсткому диску достатньо вільного місця, — це нудний, але ефективний спосіб максимізувати наші шанси на безперебійну ігрову сесію. Позбавтеся від ігор, програм і файлів, які ви більше не використовуєте, і скористайтеся перевагами — ви можете переносити файли в хмару, якщо вам це потрібно, і видаляти програми з розділу «Програми» налаштувань Windows.



Рисунок 3.4 – Стан дисків

Під диск С виділено мало пам'яті, тому основне завантаження лише з боку системних файлів. Диск D можемо зачистити за допомогою "Встановлення та видалення програм", видаливши всі непотрібні та невикористовувані програми та файли.

Багато програм Windows люблять працювати у фоновому режимі. Ці несуттєві процеси можуть іноді забирати дорогоцінні ресурси, що використовує наш ПК. Щоб перевірити, що працює у фоновому режимі, ми можемо зайти у «Диспетчер завдань», потім «Процеси» та вимкнути все зі списку, чим ми насправді не користуємося, вибравши елементи та натиснувши «Завершити процес». Якщо ми не впевнені, що це за конкретний процес, ми можемо пошукати його значення в Інтернеті — деякі з них можуть знадобитися для ігор і самої системи Windows.

Имя	Состояние	4% ЦП	34% Память	0% Диск	0% Сеть	2% GPU	Яд
> Google Chrome (12)		0,3%	741,4 МБ	0 МБ/с	0 Мбит/с	0%	
EpicWebHelper		0%	554,6 МБ	0 МБ/с	0 Мбит/с	0%	
> EpicGamesLauncher (3)		0,2%	327,9 МБ	0 МБ/с	0 Мбит/с	0%	
> Telegram Desktop (32 бита)		0,1%	200,0 МБ	0 МБ/с	0,1 Мбит/с	0%	
> Microsoft Word (32 бита)		0%	99,9 МБ	0 МБ/с	0 Мбит/с	0%	
> Kaspersky Lab launcher (32 бита)		0,8%	80,9 МБ	0 МБ/с	0 Мбит/с	0%	
> Проводник (2)		0,4%	79,1 МБ	0 МБ/с	0 Мбит/с	0%	
Google Chrome		0%	78,8 МБ	0 МБ/с	0 Мбит/с	0%	
Google Chrome		0%	65,1 МБ	0 МБ/с	0 Мбит/с	0%	
EpicWebHelper		0%	56,1 МБ	0 МБ/с	0 Мбит/с	0%	
Google Chrome		0%	55,3 МБ	0 МБ/с	0 Мбит/с	0%	
Google Chrome		0%	50,6 МБ	0 МБ/с	0 Мбит/с	0%	
NVIDIA Share		0%	39,4 МБ	0 МБ/с	0 Мбит/с	0%	
NVIDIA Container		0%	36,9 МБ	0 МБ/с	0 Мбит/с	0%	
Google Chrome		0%	33,4 МБ	0 МБ/с	0 Мбит/с	0%	
> CCleaner		0%	32,6 МБ	0 МБ/с	0 Мбит/с	0%	
Диспетчер окон рабочего стола		0,5%	32,2 МБ	0 МБ/с	0 Мбит/с	1,5%	
> Диспетчер задач		0,4%	27,4 МБ	0 МБ/с	0 Мбит/с	0%	
Google Chrome		0%	24,4 МБ	0 МБ/с	0 Мбит/с	0%	

Рисунок 3.5 – Стан «Диспетчеру завдань»

Як показує «Диспетчер завдань», наразі наш ПК перебуває у «спокійному стані», ресурси неперевантажені і загалом система працюватиме без заповільнень у разі, якщо ми зараз запустимо гру.

Для того, щоб показати наскільки може бути ефективна оптимізація ПК, а саме побачити результат плавної картинки, ми використовуватимемо як приклад одну з найпопулярніших ігор на даний момент - Grand Theft Auto 5. Однак перш ніж порівняти результати наших маніпуляцій, нам також необхідно зробити внутрішнє ігрове налаштування.

Таблиця 2.1 – Мінімальні вимоги до ПК

ОС	Windows 8.1, 8, 7, Vista 64 біт
Процесор	Intel Core 2 Quad CPU Q6600 @ 2.40GHz (4 ядра) / AMD Phenom 9850 Quad-Core Processor (4 ядра)
ОП	4ГБ
Відеокарта	NVIDIA 9800 GT 1 Гб/AMD HD 4870 1 Гб (версія не нижче DX 10, 10.1, 11)
Звукова карта	100% сумісна з DirectX 10
Місце на диску	65 ГБ

Проводити тест ми будемо на ноутбуці з характеристиками у Таблиці 2.2

Таблиця 2.2 – Характеристики ноутбука

Процесор	Чотирьохядерний Intel Pentium Silver N6000 (1.1 - 3.3 ГГц)
ОП	6 ГБ

Такі компоненти ноутбука будуть відповідати мінімальним вимогам та гра з труднощами, але запускатиметься і працюватиме.

Продовжимо оптимізацію системи за рахунок активації додаткової швидкодії. Переходимо до "Мого комп'ютера" і заходимо у вкладку "Властивості", звідки переходимо до "Параметри швидкодії".

Забираємо всі галочки та залишаємо тільки "Виведення ескізів замість значків", "Згладжування нерівностей екранних шрифтів" та "Збереження виду ескізів панелі завдань".

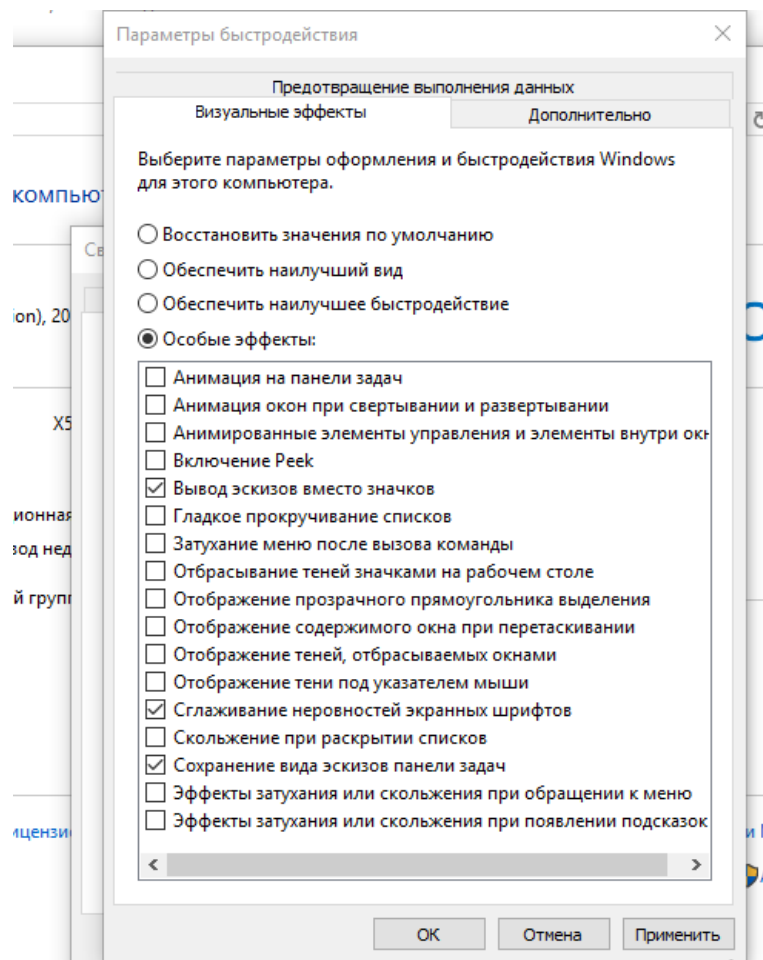


Рисунок 3.6 – Налаштування параметрів швидкодії

Також, щоб знизити навантаження з диска, на якому встановлено програму, ми можемо зайти у властивості диска та вимкнути індексування.

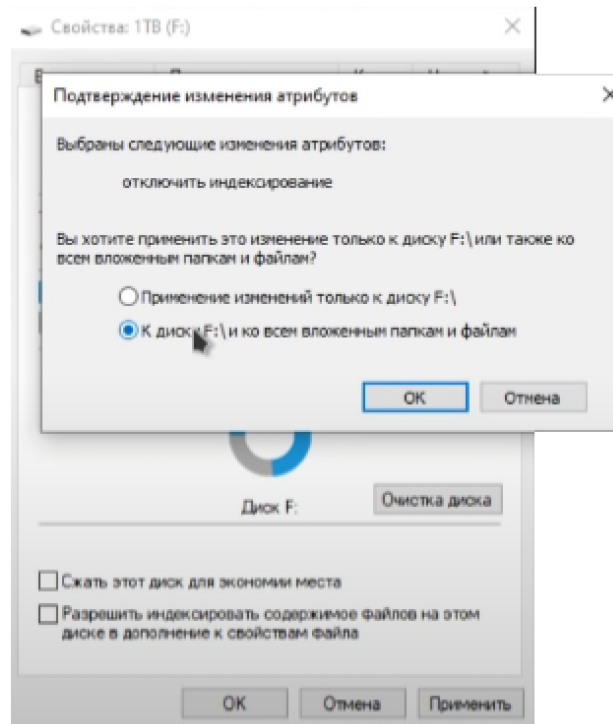


Рисунок 3.7 – Відключення індексації на диску

Також нам необхідно настроїти енергоспоживання. Для цього нам необхідно зайти в "Панель управління", потім у розділ "Електроживлення", позначити пункт "Висока продуктивність" замість стандартної "Збалансованої" та переходимо в меню налаштувань і ставимо значення в пункті «Стан (мін.)» на 999.

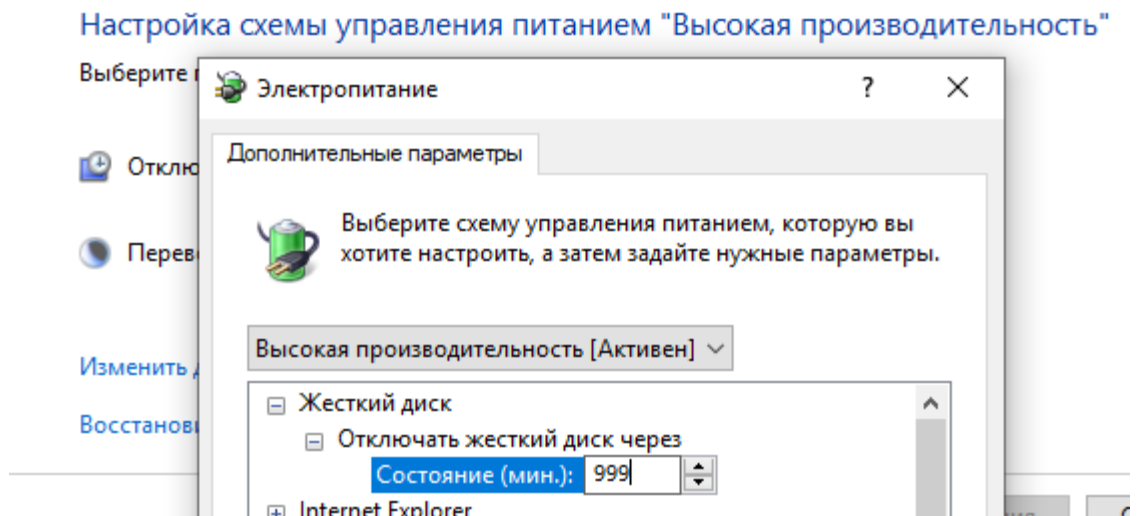


Рисунок 3.8 – Налаштування енергоспоживання

Тепер нам необхідно попрацювати з файлами вихідної папки з грою. Заходимо в каталог, де містяться файли гри та клацаємо по виконавчому файлу (.exe). Переходимо до вкладки «Сумісність» та помічаємо чекбокс «Відключити оптимізацію у весь екран», а також змінюємо параметри DPI і помічаємо чекбокс про «перевизначення режиму масштабування високої роздільної здатності».

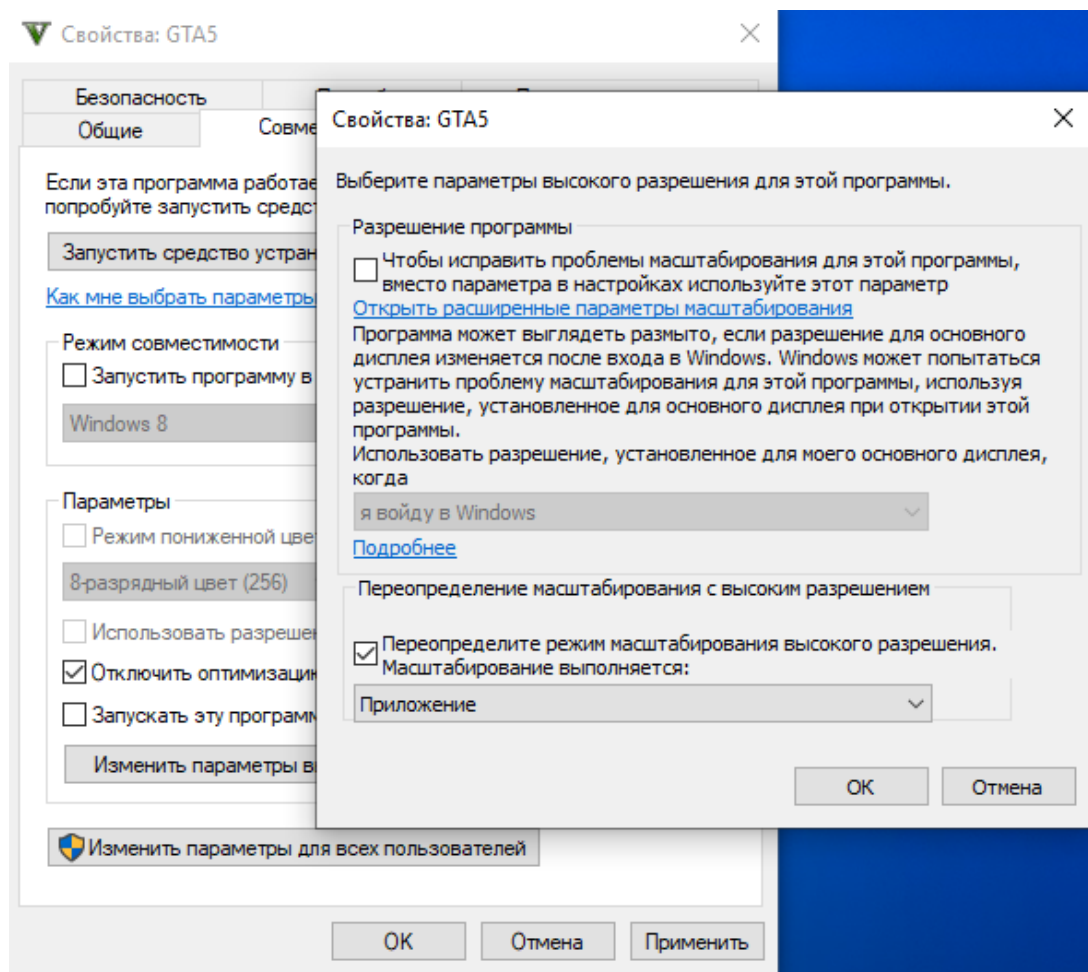


Рисунок 3.9 – Налаштування виконавчого файлу

На цьому базові установки ноутбука закінчені, тепер ми можемо запустити гру і подивитися наскільки плавне у нас буде зображення. Для перевірки таких параметрів як FPS, навантаження процесора і т.д. нам

необхідно використати спеціальний софт. Таких програм безліч, але в нашому випадку, для зручності, нам знадобиться FPSMonitor.

FPSMonitor - просунута програма, яка показує під час гри не тільки FPS, а й дані компонентів комп'ютера. Наприклад, на екран можна вивести завантаженість, температуру ядер процесора та кількість вільної оперативної пам'яті.

Інша важлива особливість FPSMonitor — можливість візуального налаштування інформації, що відображається на екрані. Якщо в описаному вище Fgaps можна було вибрати лише кут, де розміщується лічильник FPS, то можна підібрати колір, шрифт, включати або відключати показ різних датчиків і так далі.

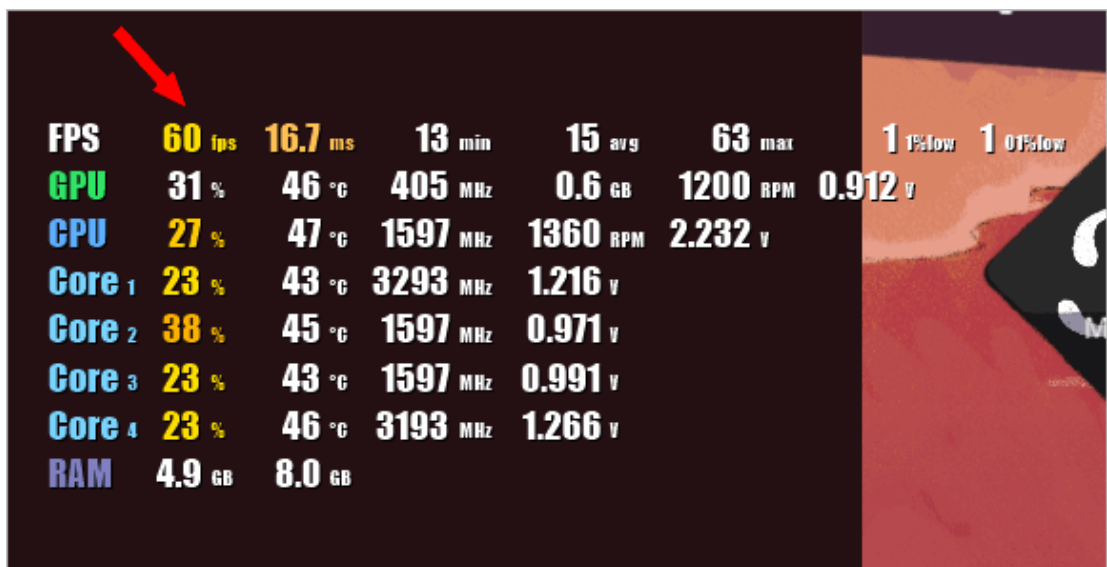


Рисунок 3.10 – Зовнішній вигляд оверлею FPSMonitor

Функції FPSMonitor:

- підтримка датчиків всіх популярних процесорів та відеокарт;
- гнучке налаштування оверлея (виводиться на екран зображення);
- виведення на екран кількох оверлеїв;
- робота із гарячими клавішами;
- обмеження FPS за рахунок емуляції функції вертикальної синхронізації (частоти кадрів гри та частоти зміни оновлення екрана);

- ведення логів;
- стандартні передумовки для популярних завдань;
- підтримка «сцен» (різних варіантів візуального відображення даних, між якими можна швидко перемикається).

Після проведеної оптимізації гра запустилася з розширенням 800x600 та за середнього FPS 15-25.



Рисунок 3.11 – GTA 5 після оптимізації Windows

При цьому кількість FPS в 15-25 досягається тільки під час «спокійної» гри. Коли гра досягає «пікових» моментів при великій кількості об'єктів на екрані, відбуваються природні «фризи екрану» і FPS опускається до 5. Цей результат непоганий, але все одно його не можна вважати грабельним. Тому ми продовжимо оптимізувати гру у відповідності з нашими планами і подивимося наскільки її фрейм-рейт зможе змінитись на краще після виконаних маніпуляцій.

Ми повертаємося в меню Віндоус і тепер працюватимемо з додатковими програмами оптимізації, розподілом ресурсів компонентів ПК та настройками самої гри.

Якщо ви купили готовий ігровий комп'ютер або ноутбук від відомого ігрового бренду, наприклад Razer, Alienware, Gigabyte та багатьох інших, він може мати додаткову або дві утиліти для оптимізації ваших ігор. Ми не можемо розповісти про кожен варіант програмного забезпечення для кожного ігрового бренду, але якщо ви подивитесь на встановлені програми в меню «Пуск», ви зможете знайти будь-яких очевидних кандидатів. Наприклад, ігрові ноутбуки Gigabyte оснащені центром керування Aorus, який надає зворотній зв'язок щодо поточного навантаження на графічний процесор, процесор та пам'ять. Це також дозволяє підвищити швидкість процесора та відеокарти в обмін на більш високу споживану потужність і гучніший шум вентилятора, якщо ви хочете отримати додаткову продуктивність зі своєї системи.

Якщо ви не можете знайти жодної такої утиліти, але вважаєте, що вона має бути там, завжди варто звертатися безпосередньо до виробника (що зазвичай означає читання відповідних форумів підтримки в Інтернеті). Можливо, вам знадобиться вручну завантажити утиліту, якщо вона не була встановлена під час збирання комп'ютера (або якщо ви ненавмисно видалили програму, не усвідомлюючи, що це таке).

Нам не потрібно задовольнятися пакетом програмного забезпечення для оптимізації, що постачається разом із ПК, оскільки існує багато альтернатив. Razer Cortex є безкоштовним і одним з найкращих, і для його використання не потрібні пристрій Razer. Він розумно розподілятиме системні ресурси для ігор, коли вони потребують, обмежуватиме неігрові додатки та загалом витискатиме якомога високу частоту кадрів із обладнання.

Game Fire — це ще один варіант, який пропонує подібний набір налаштувань і оптимізацій — закриття непотрібних фонових процесів, переконання, що наша установка ефективно використовує оперативну пам'ять, і спостереження за потенційними сповільненнями інтернет-

з'єднання. Це обійдеться в 18 доларів, але є безкоштовна версія з базовими функціями, щоб ми могли спочатку вирішити, чи дійсно це ефективно для нас.

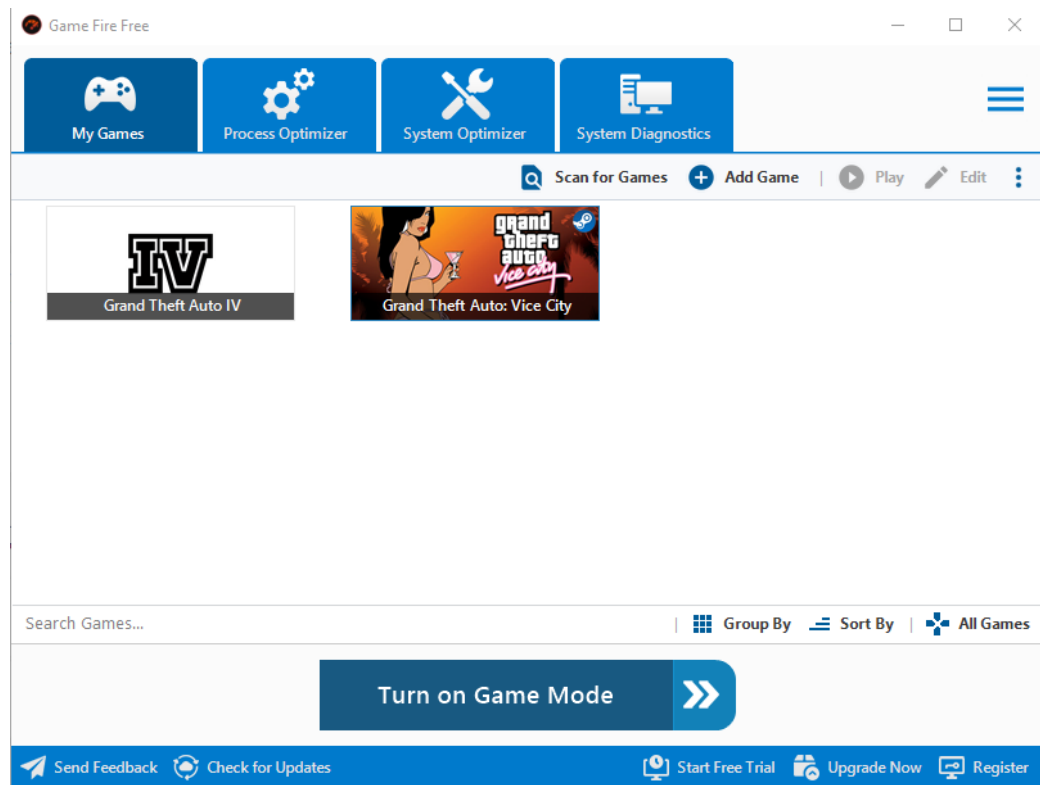


Рисунок 3.12 – Оптимізація ресурсів ПК за допомогою Game Fire

Ми будемо використовувати безкоштовний додаток Game Fire Free. Після того, як ми натиснемо на «Turn on Game Mode», додаток запропонує нам вручну закрити фонові програми, потім видалить самостійно непотрібні процеси і почне стежити, щоб ресурси ПК розподілялися таким чином, щоб гра не сповільнювалася, коли запущена.

Також, як одну із альтернатив «пришвидшення» ПК, ми можемо розглядати його «розгін». Раніше розгін — виведення компонентів вашої системи за межі дозволених виробником — був винятковою сферою ентузіастів-геймерів, що хотіли досягти максимальної продуктивності від ПК, але тепер практично будь-хто може спробувати. Користуватися програмними додатками простіше, ніж будь-коли, хоча ми рекомендуємо спочатку трохи ознайомитися з цією темою, щоб зрозуміти наскільки система зможе потягнути

такий «розгін». Також важливо зазначити, що розгін виконується на ваш власний ризик (ви майже анулюєте свої гарантії на обладнання).

Розігнані компоненти означають швидший ігровий процес, але вам знадобиться процесор та/або графічний процесор, створений з урахуванням розгону. Якщо ви придбали готову ігрову установку, призначену для розгону, ви можете виявити, що вона постачається з утилітою для цього завдання.

Це не точна наука, але ви в основному підвищуєте швидкість свого ЦП та/або графічного процесора, поки не помітите помилки та збої, а потім повертаєте його до безпечного рівня. Такі програми, як Intel Extreme Tuning Utility, AMD Ryzen Master, MSI Afterburner і Asus GPU Tweak, є хорошими для початку, щоб побачити, що можливо у вашій системі.

Ми не будемо поки що ризикувати із «розгоном» системи і подивимось на що здатен наш ПК після того, як ми виконаємо усі прості способи оптимізувати його.

Тепер коли ми оптимізували наш ПК, вичистили і видалили все непотрібне, ми можемо перейти до налаштувань гри і розглянути, що ми можемо змінити на користь підвищення гладкості та плавності зображення.

3.2 Налаштування гри під ПК

Перш ніж ми перейдемо до налаштувань Grand Theft Auto 5, ми пропонуємо короткий опис загальних налаштувань, що доступні майже в кожній грі

Згладжування. Яке зазвичай називають AA (Antialiasing), — це техніка розробки комп'ютерної графіки для згладжування грубих піксельних або нерівних країв у графіку. Більшість із нас стикалися з цією комп'ютерною графікою з піксельною чи нерівною зовнішністю. AA робить для кожного пікселя на вашому екрані вибірку навколишніх пікселів і намагається змішати їх, щоб вони виглядали гладкими. Багато ігор дозволяють увімкнути або

вимкнути AA, а також встановити частоту дискретизації AA, виражену як 2x AA, 4x AA, 8x AA тощо. Найкраще встановити AA у поєднанні з роздільною здатністю графіки/монітора. Більш висока роздільна здатність має більше пікселів і може знадобитися лише 2x AA, щоб графіка виглядала гладкою та добре працювала, тоді як нижча роздільна здатність може потребувати встановлення 8x, щоб згладити ситуацію. Якщо ви шукаєте пряме підвищення продуктивності, то зниження або повне вимкнення AA має дати вам приріст.

Анізотропна фільтрація. У комп'ютерній 3D-графіці зазвичай віддалені об'єкти в 3D-середовищі використовують нижчу якість текстурних карт, які можуть здаватися розмитими, тоді як ближчі об'єкти використовують високоякісні текстурні карти для більшої деталізації. Надання високоякісних текстурних карт для всіх об'єктів у 3D-середовищі може мати великий вплив на загальну продуктивність графіки, і саме тут на допомогу приходить налаштування анізотропної фільтрації або AF (Anisotropic Filtering).

AF досить схожий на AA з точки зору налаштувань і того, що він може зробити для підвищення продуктивності. Зменшення параметра має свої недоліки, оскільки більша частина перегляду використовуватиме текстуру нижчої якості, завдяки чому об'єкти, що знаходяться поблизу, виглядатимуть розмитими. Частота дискретизації AF може коливатися від 1x до 16x, і налаштування цього параметра може забезпечити помітне покращення продуктивності старої відеокарти; Цей параметр стає все меншою причиною зниження продуктивності на новіших відеокартах.

Дальність промальовки/поле зору. Налаштування відстані відображення об'єктів або параметри відстані перегляду та поля зору використовуються для визначення того, що ви побачите на екрані, і є найбільш релевантними як для шутерів від першої, так і від третьої особи. Налаштування відстані промальовки або перегляду використовується, щоб визначити, наскільки далеко ви бачите відстань, тоді як поле зору визначає більшу частину периферійного огляду персонажа в FPS. У разі відстані промальовки та поля

зору, чим вище налаштування, відеокарті потрібно буде працювати інтенсивніше, щоб відтворити та відобразити вигляд, однак вплив, здебільшого, має бути досить мінімальним, тому зниження цих налаштувань значно не вплине на підвищення відображених кадрів в секунду.

Освітлення/Тіні. Тіні у відеогрі сприяють загальному вигляду гри, додаючи відчуття несподіванки в історію, що розповідається на екрані. Налаштування якості тіней визначає, наскільки детально чи реалістично тіні виглядатимуть у грі. Вплив цього може відрізнятись від сцени до сцени залежно від кількості об'єктів та освітлення, але це може мати досить великий вплив на загальну продуктивність. Хоча тіні можуть зробити сцену чудовою, це, ймовірно, перше налаштування, яке потрібно зменшити або вимкнути для підвищення продуктивності під час роботи зі старішою відеокартою.

Роздільна здатність. Налаштування роздільної здатності залежить як від того, що доступно в грі, так і від монітора. Чим вища роздільна здатність, тим краще буде виглядати графіка, усі ці додаткові пікселі додають деталі середовища та об'єктів, покращуючи їх зовнішній вигляд. Однак вища роздільна здатність має компроміс, оскільки на екрані відображається більше пікселів, відеокарта повинна працювати більше, щоб відображати все, і, отже, може знизити продуктивність. Зменшення налаштування роздільної здатності в грі є надійним способом покращити продуктивність і частоту кадрів, але якщо ви звикли грати з вищими роздільними можливостями та бачити більше деталей, можливо, вам захочеться переглянути деякі інші параметри, такі як вимкнення AA/AF або регулювання освітлення/тінь на користь збільшеної роздільної здатності.

Деталі/якість текстури. Простіше кажучи, текстури можна розглядати як шпалери для комп'ютерної графіки. Це зображення, які накладаються на об'єкти/моделі в графіку. Цей параметр зазвичай не впливає на частоту кадрів у грі, якщо взагалі, тому досить безпечно мати цей набір у вищій якості, ніж інші налаштування, такі як освітлення/тінь або AA/AF.

Заходимо у внутрішньоігрові налаштування графіки. Нам пропонують вибрати версію DirectX. В нашому випадку це не буде ні на що впливати, тому ми можемо залишити, як і DirectX10, так і вибрати DirectX11 для якої вона і розроблювалась.

Після цього виставимо мінімальне значення «тіней», якості текстур та анізотропної фільтрації, про які обговорювалось раніше. Вони займатимуть основну кількість відеопам'яті і їх зниження сприяє плавності зображення під час гри.

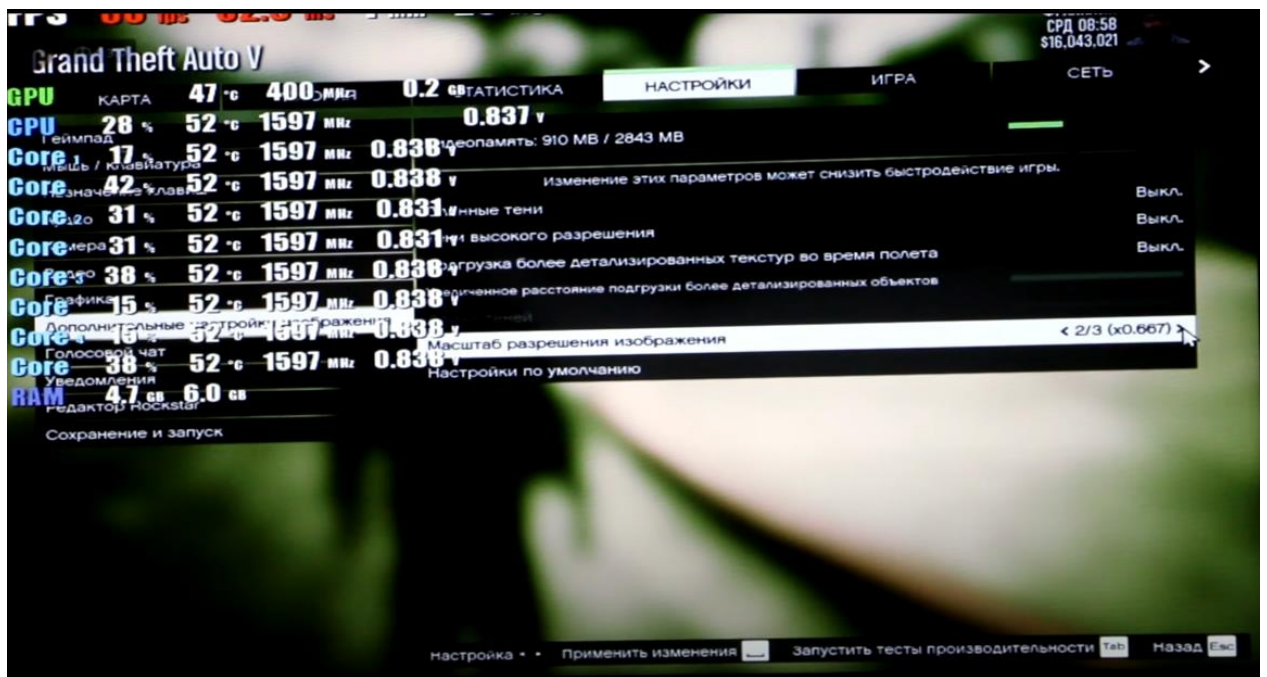


Рисунок 3.13 – Налаштування відображення GTA 5.

У розділі «додаткові параметри зображення», ми можемо також випробувати різні параметри «масштабу роздільної здатності зображення», спробуємо взяти 2.3x (0.667) і переглянути результат.

Завдяки нашим маніпуляціям картинка в цілому покращилася і фрейм-рейт став стабільнішим, проте фризи екрану досі присутні і нам необхідно дивитися налаштування далі, щоб їх подолати



Рисунок 3.14 – Покращення плавності зображення в GTA 5

Замість того, щоб продовжити редагувати налаштування у внутрішньоігровому меню, для досягнення наших цілей, нам необхідно відредагувати .xml файл налаштувань гри, оскільки тут ми можемо накрутити найточніші налаштування будь-якого елемента або прибрати його зовсім, а також використовувати параметри, які просто недоступні в ігровому меню.

У файлі settings.xml представлено багато параметрів, які можуть бути змінені на користь позитивного впливу на фрейм-рейт. Проте ми поговоримо лише про параметри, які нададуть найбільший ефект від редагування. Кожен з параметрів може приймати мінімальне значення до -1, та максимальне значення 1.

3.3 Налаштування параметрів гри через .xml файл

LODs. Це налаштування дистанції промальовки об'єктів. В цій категорії у нас буде декілька параметрів:

"LodScale". Контролює відстань промальовування світу ("Масштабування відстані" у налаштуваннях гри). Ми ставимо негативне

значення -0.200000 цьому параметру, щоб промальовка карти не займала багато відеопам'яті.

"PedLodBias". Контролює відстань промальовки для пішоходів. В цьому параметрі також ставимо значення -0.400000.

"VehicleLodBias". Контролює відстань промальовки для машин. В цьому параметрі ставимо значення -0.700000.

"MaxLodScale". Цей параметр контролює загальну відстань малювання і є сумішшю трьох значень вище. Він також контролює відстань малювання для високодеталізованих моделей світу. Ми поставимо тут значення "-0,600000", і матимемо більш плавний FPS майже без графічних втрат.

Матимемо наступні зміни в settings.xml:

```
<LodScale value="-0.200000" />
```

```
<PedLodBias value="-0.400000" />
```

```
<VehicleLodBias value="-0.700000" />
```

```
<MaxLodScale value="-0.600000" />
```



Рисунок 3.15 – Зміни промальовки об'єктів

Загалом можемо побачити, що зміни даних параметрів не особливо псуєть навколишній світ, і якщо ми зробили налаштування недостатньо позитивними для впливу на наш фреймрейт, ми завжди можемо поставити нижчі значення для досягнення кращої мети. За нашими налаштуваннями змінена дальність промальовки міста, тепер об'єкти, що знаходяться в даліні, відображаються помітно жакливіше, проте така маніпуляція додає нам запас фрейм-рейту.

Переходимо до налаштувань тіней. Зміна налаштувань тіней також дуже впливає на продуктивність і можна досягти хороших результатів, змінивши значення кількох параметрів. Такі в нас будуть значення параметрів після змін:

```
<ShaderQuality value="0" />
```

```
<Shadow_SoftShadows value="0" />
```

```
<UltraShadows_Enabled value="true" />
```

```
<Shadow_ParticleShadows value="false" />
```

```
<Shadow_Distance value="0.000000" />
```

```
<Shadow_LongShadows value="true" />
```

```
<Shadow_SplitZStart value="0.0001" />
```

```
<Shadow_SplitZEnd value="0.0001" />
```

```
<Shadow_aircraftExpWeight value="0.990000" />
```

```
<Shadow_DisableScreenSizeCheck value="false" />
```

Можемо зайти в гру і порівняти з тим, що було до цього.



Рисунок 3.16 – Зміни тіней в грі

Нашими змінами ми буквально прибрати тіні з далеких об'єктів. При наближенні гравця до об'єкта починає відображатися тінь, як це можна побачити на малюнку 3.9, якщо розглянути огорожу. Тінь там обривається, бо гравець перебуває надто далеко.

Спробуймо відредагувати ще кілька елементів, перш ніж зробимо фінальний тест.

Параметри `CityDensity` та `PedVarietyMultiplier` відповідають за кількість об'єктів машин та людей, що одночасно відображаються на екрані. Ми не збираємось помітно змінювати цей параметр, тому що він вплине на густість об'єктів світу і сам світ буде здаватись неживим. Поставимо значення параметрів – 0.8.

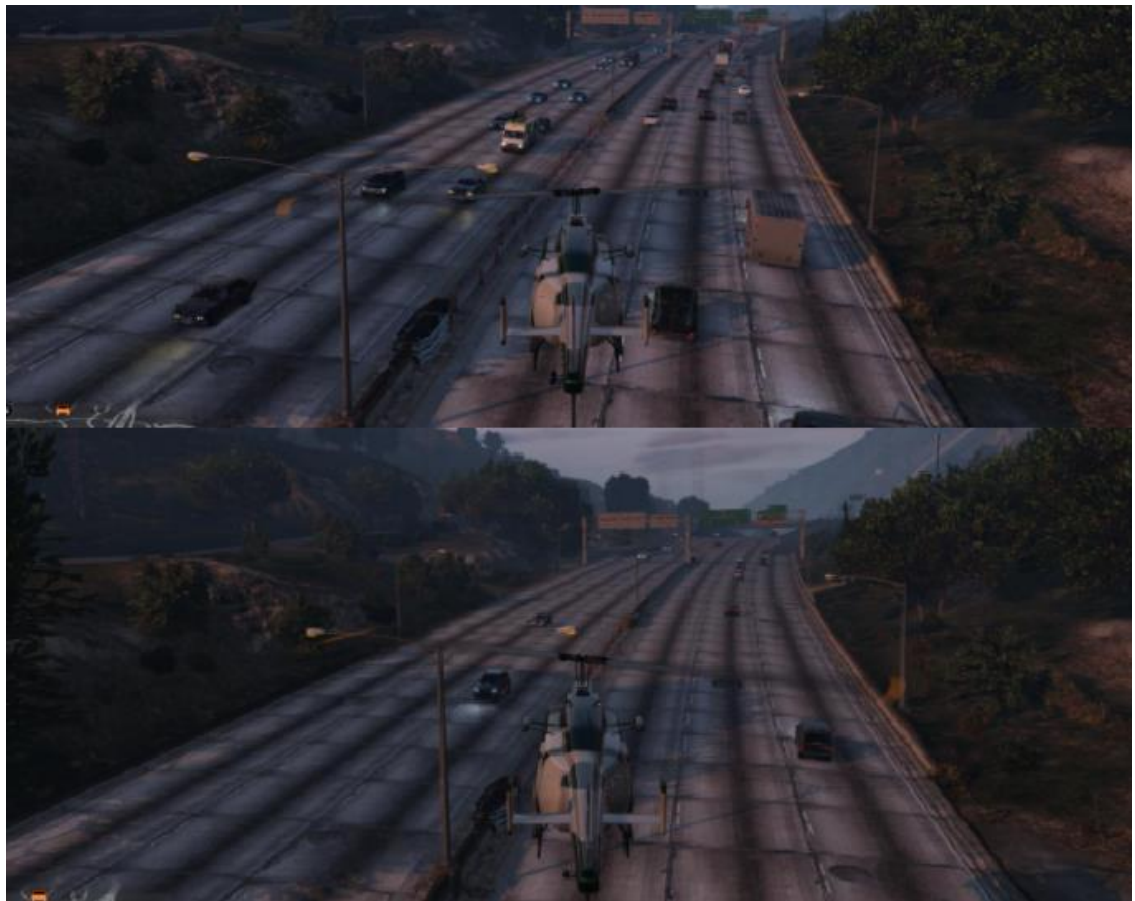


Рисунок 3.17 – Змінена кількість об'єктів, що генерується водночас

Також змінюємо значення `PostFX`, яке відповідає за постефекти на 0.3. Змінюємо якість відображень (у дзеркалах машин тощо) за допомогою

параметрів ReflectionQuality та ReflectionMSAA. У ReflectionQuality буде значення 0.5, а у ReflectionMSAA - 1. Також знижуємо значення якості відображення трави та води за допомогою параметрів GrassQuality та WaterQuality, можемо встановити їм значення 0.4. Можемо також змінити значення MotionBlurStrength (розмитості в русі), встановивши його також у діапазоні від 0.3 до 0.5, хоча це особливо впливати на продуктивність не буде.

Загалом матимемо наступні параметри:

```
<SamplingMode value="0" />
```

```
<TextureQuality value="0.3" />
```

```
<ParticleQuality value="0.3" />
```

```
<WaterQuality value="0.4" />
```

```
<GrassQuality value="0.4" />
```

```
<ShaderQuality value="0.5" />
```

```
<CityDensity value="0.800000" />
```

```
<PedVarietyMultiplier value="0.800000" />
```

```
<VehicleVarietyMultiplier value="0.800000" />
```

Тепер можемо зробити фінальний тест та порівняти результати з тим, що було спочатку.



Рисунок 3.18 – Фінальний тест

Завдяки налаштуванням, які ми зробили, ми адаптували гру під нелегкі умови нашого ноутбука і запустили з плавною грабельною картинкою. Проведемо також порівняння зниження споживаних ресурсів ПК.

Имя	Состояние	ЦП	Память	Диск	Сеть	GPU	Ядро GPU	Энергопотреб
> Grand Theft Auto V (5)		37,1%	2 492,8 МБ	0,5 МБ/с	0 Мбит/с	0,1%	Графический процессор 0 - Сору	Очень высоко
> Grand Theft Auto V (3)		23,7%	2 345,2 МБ	0,3 МБ/с	0 Мбит/с	0,1%	Графический процессор 0 - Сору	Очень высоко

Рисунок 3.19 – Ресурси ПК до та після

Поліпшення плавності зображення на користь погіршення графіки дало свої плоди. Ми можемо бачити покращення продуктивності, як у грі, так і на диспетчері завдань. Гра набагато менше використовує доступних ресурсів, що може сприяти спокою системи та недопущення перегрівів. Фрейм-рейт у грі коливається від 60 до 80 фпс і це відмінний результат, враховуючи що компоненти нашого ноутбука ледве наближені до мінімальних.

Практично більшість ігор зберігають налаштування користувача в окремих файлах у форматі .xml і розібратися зі значеннями параметрів можна, шукаючи їх в інтернеті. Така процедура часто використовується

користувачами для покращення ігрового досвіду на користь втрати якості зображення. Зрозуміло, що першочерговою річчю при нестабільній плавності ігрового зображення має бути апгрейд вашої ігрової станції, проте такі маніпуляції також цілком допустимі і їх можна застосовувати навіть у невеликих кількостях, щоб відчувати помітний результат (у нашому випадку даунгрейд якості картинки був зроблений кардинально, тому що система не дозволяла досягти плавної картини навіть на найнижчих налаштуваннях).

ВИСНОВКИ

У роботі була поетапно розглянута та описана еволюція графічних карт і як змінювалася їх архітектура, описані актуальні движки на яких пишуться та запускаються ігри.

Була створена сцена Unity 3D, щоб на практиці показати, як високо- та низько-полігональні моделі впливають на продуктивність ПК. Також була проведена оптимізація ПК та даунгрейд графіки гри Grand Theft Auto 5, щоб провести аналіз та порівняння, як можна досягти плавної та помірної картинки у грі без заморозків екрану, за рахунок погіршення якості ефектів, відображень, тіней, дальності промальовування тощо.