

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МІЖНАРОДНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ
Факультет кібербезпеки, програмної інженерії та комп'ютерних наук
Кафедра комп'ютерних наук

Соловська І.М., Мазур Г.Д.

КОМП'ЮТЕРНА ГРАФІКА

Конспект лекцій

Одеса 2025р

Затверджено Вченою Радою Міжнародного гуманітарного університету
(протокол № 5 від 20.01.2025 р.)

Соловська І.М., Мазур Г.Д.

Комп'ютерна графіка: конспект лекцій для здобувачів [Електронне видання]. /
Соловська І.М., Мазур Г.Д. Кафедра комп'ютерних наук Міжнародного
гуманітарного університету. Одеса, 2025. – 53 с.

Конспект лекцій з курсу «**Комп'ютерна графіка**» розроблено відповідно до навчального плану. Матеріали призначено для студентів факультету кібербезпеки, програмної інженерії та комп'ютерних наук Міжнародного гуманітарного університету, які навчаються за першим (бакалаврським) рівнем вищої освіти в галузі знань – 12 Інформаційні технології, спеціальності 121 «Інженерія програмного забезпечення», 122 «Комп'ютерні науки», 123 «Комп'ютерна інженерія», 125 «Кібербезпека та захист інформації», галузі знань 17 – Електроніка, автоматизація та електронні комунікації, спеціальність 172 «Електронні комунікації та радіотехніка», галузі знань 01 – Освіта/Педагогіка, спеціальність 014 «Середня освіта. Інформатика».

ЗМІСТ

Лекція 1. Вступ до комп'ютерної графіки. Історія виникнення та основні поняття.....	3
Лекція 2. Растрова графіка: основи та характеристики.....	9
Лекція 3. Векторна графіка: принципи побудови та переваги.....	13
Лекція 4. Фрактальна та тривимірна графіка.....	20
Лекція 5. Колориметрія. Адитивна модель RGB.....	24
Лекція 6. Субтрактивна модель CMYK та модель LAB.....	27
Лекція 7. Моделі HSB і HLS. Практичне використання колірних моделей.....	30
Лекція 8. Робота з двовимірною графікою. Шари та допоміжні засоби.....	33
Лекція 9. Робота з текстом. Інструменти моделювання та редагування.....	36
Лекція 10. Криві Без'є та сплайни для створення довільних фігур.....	39
Лекція 11. Основи тривимірного моделювання. Булеві операції.....	42
Лекція 12. Матеріали і текстури в 3D графіці.....	45
Лекція 13. Налаштування рендера. Візуалізація тривимірних сцен.....	48
Список джерел літератури.....	51

ЛЕКЦІЯ 1

ВСТУП ДО КОМП'ЮТЕРНОЇ ГРАФІКИ. ІСТОРІЯ ВИНИКНЕННЯ ТА ОСНОВНІ ПОНЯТТЯ

Комп'ютерна графіка являє собою одну з найважливіших галузей сучасної інформатики, яка займається методами створення, обробки та відображення зображень за допомогою обчислювальної техніки. Ця дисципліна об'єднує знання з математики, фізики, програмування та мистецтва, створюючи потужний інструментарій для вирішення широкого спектра завдань – від наукової візуалізації до розважальної індустрії. Сучасне розуміння комп'ютерної графіки включає не лише статичні зображення, але й анімацію, інтерактивну графіку реального часу, віртуальну та доповнену реальність.

Історія виникнення комп'ютерної графіки бере свій початок у середині двадцятого століття. Перші експерименти з відображення графічної інформації на екранах електронно-променевих трубок були проведені ще в 1950-х роках. Знаковою подією стала розробка інтерактивної графічної системи Sketchpad Іваном Сазерлендом у 1963 році, яка дозволяла користувачеві створювати та редагувати графічні об'єкти за допомогою світлового пера. Ця система заклала основи сучасних систем автоматизованого проектування та продемонструвала потенціал інтерактивної графіки для інженерних застосувань.

У 1970-х роках розвиток комп'ютерної графіки прискорився завдяки появі перших алгоритмів приховання невидимих поверхонь, затінення та текстуровання. Дослідження в галузі трасування променів, розпочаті в цей період, згодом стали основою для створення фотореалістичних зображень. 1980-ті роки ознаменувалися появою перших комерційних систем тривимірного моделювання та анімації, які почали використовуватися у кіновиробництві та телебаченні.

Розвиток персональних комп'ютерів у 1990-х роках зробив інструменти комп'ютерної графіки доступними для широкого загалу. З'явилися потужні графічні редактори для роботи з растровою та векторною графікою, системи тривимірного моделювання почали інтегруватися у виробничі процеси різних галузей. Початок двадцятого століття характеризується інтенсивним розвитком графіки реального часу, обумовленим прогресом у галузі апаратного прискорення графічних обчислень та створенням спеціалізованих графічних процесорів.

Сучасна комп'ютерна графіка поділяється на кілька основних напрямків, кожен з яких має свої особливості та сфери застосування. Наукова візуалізація займається представленням складних наукових даних у зрозумілій графічній формі, дозволяючи дослідникам аналізувати результати експериментів та математичного моделювання. Інформаційна графіка фокусується на візуалізації даних та створенні інфографіки для ефективного передавання інформації.

Інженерна графіка використовується в системах автоматизованого проектування для створення креслень, схем та тривимірних моделей технічних об'єктів. Медична візуалізація дозволяє отримувати та аналізувати зображення внутрішніх органів людини за допомогою комп'ютерної томографії, магнітно-резонансної томографії та інших методів діагностики. Розважальна індустрія, включаючи кінематограф, телебачення та відеоігри, є одним з найбільших споживачів технологій комп'ютерної графіки.

Класифікація комп'ютерної графіки може здійснюватися за різними критеріями. За способом представлення зображень розрізняють растрову, векторну, фрактальну та тривимірну графіку. За характером взаємодії з користувачем графіка поділяється на пасивну, коли зображення створюється заздалегідь і лише відтворюється, та інтерактивну, яка реагує на дії користувача в реальному часі. За часовою характеристикою розрізняють статичну графіку, що

представляє нерухомі зображення, та анімацію, яка імітує рух шляхом послідовної зміни кадрів.

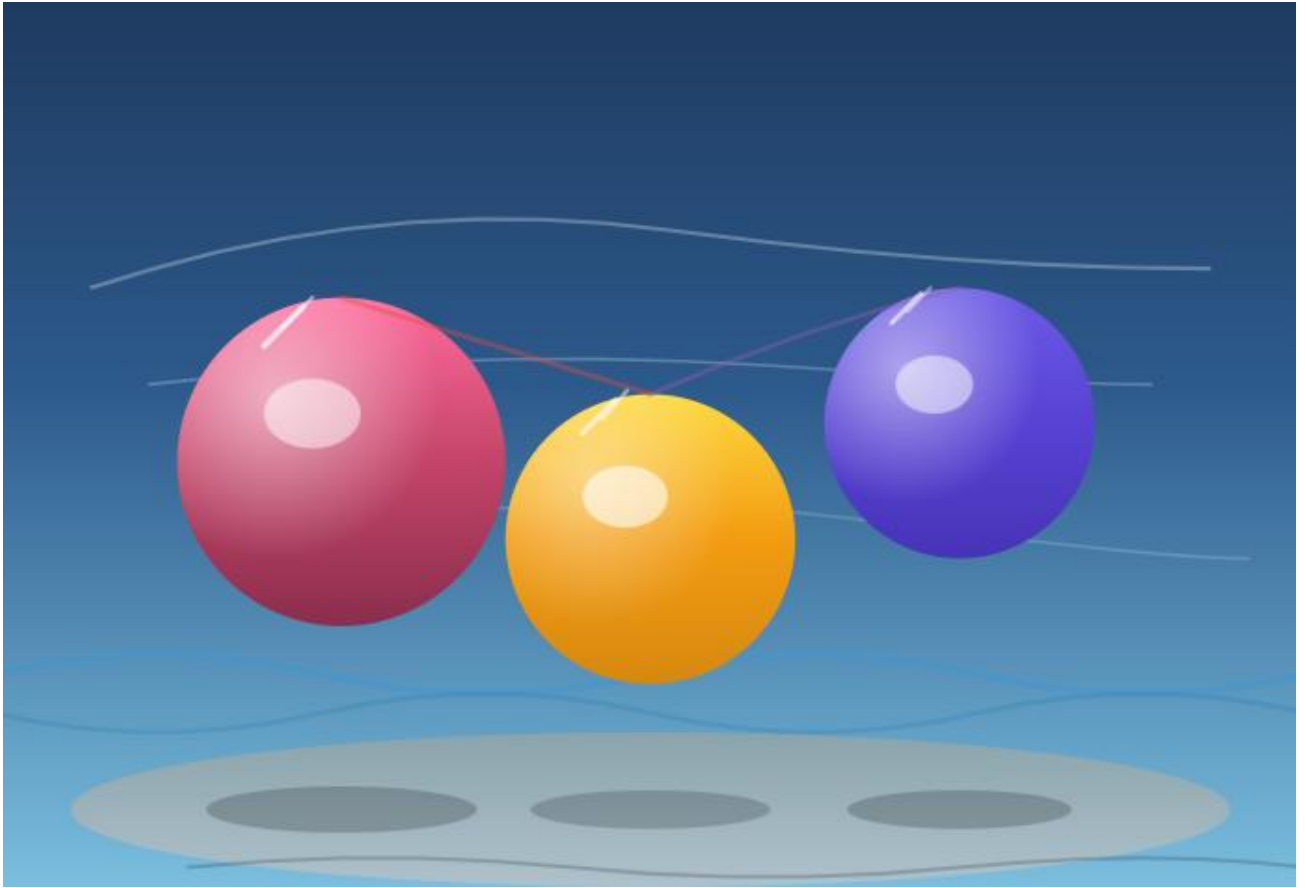


Рисунок 1.1 - Приклад сучасної комп'ютерної графіки: фотореалістичне 3D-зображення з реалістичним освітленням та текстурами

Растрова графіка базується на представленні зображення у вигляді прямокутної матриці пікселів, кожен з яких має визначений колір. Цей підхід дозволяє досягти високої реалістичності зображень, особливо при роботі з фотографіями та складними візуальними ефектами. Однак растрові зображення мають суттєвий недолік – вони залежать від роздільної здатності, що означає втрату якості при масштабуванні.

Векторна графіка використовує математичний опис геометричних примітивів, таких як лінії, кола, криві та багатокутники. Такий підхід забезпечує незалежність від роздільної здатності – векторні зображення можуть

масштабуватися без втрати якості. Векторна графіка ідеально підходить для створення логотипів, ілюстрацій, технічних креслень та шрифтів.

Фрактальна графіка ґрунтується на математичних об'єктах, які мають властивість самоподібності – кожна частина фрактала подібна до цілого. Цей вид графіки дозволяє створювати складні природні форми, такі як хмари, гори, берегові лінії, за допомогою відносно простих математичних формул. Фрактальна графіка знаходить застосування в моделюванні природних явищ, створенні текстур та художній візуалізації.

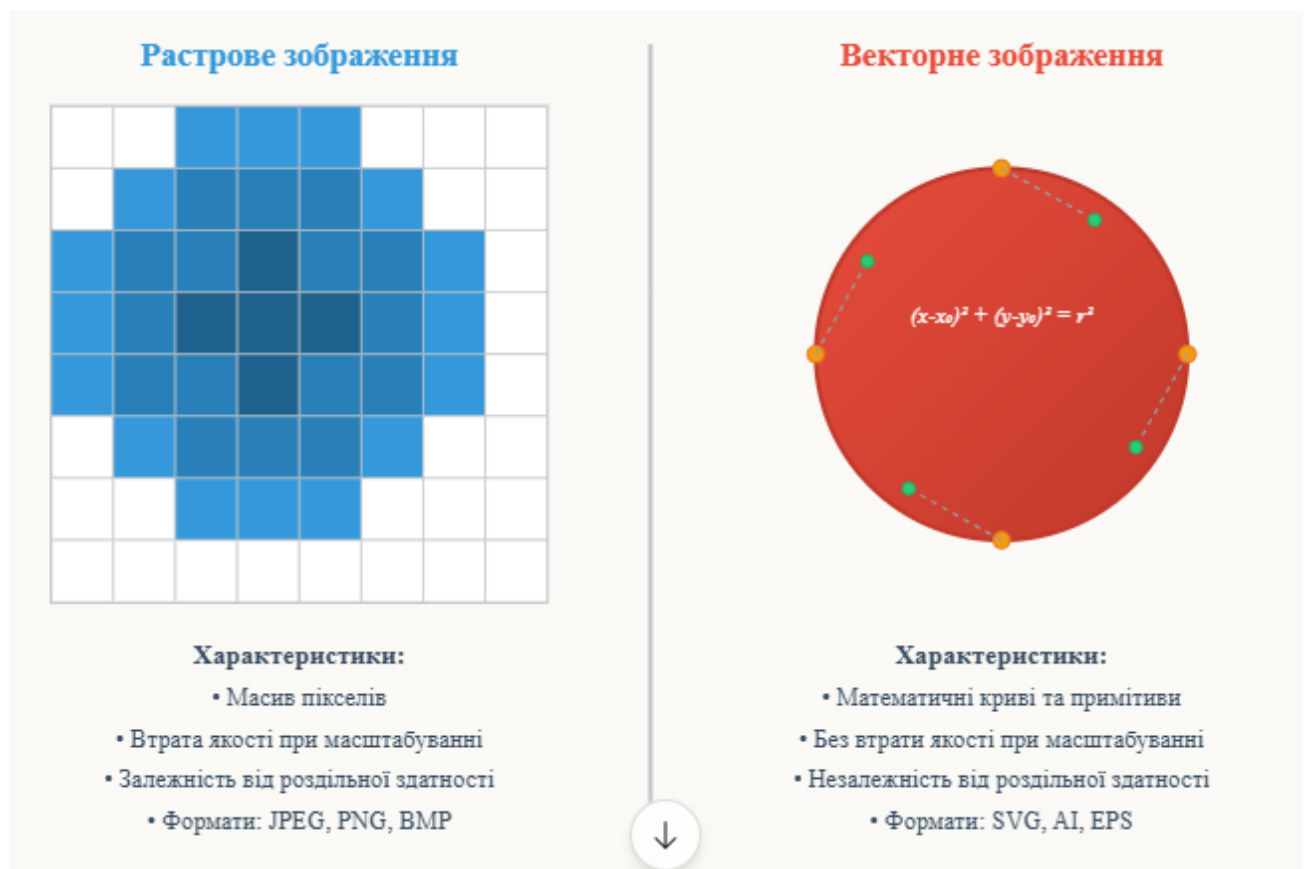


Рисунок 1.2 - Порівняння растрового та векторного підходів до представлення зображень

Тривимірна графіка займається створенням та маніпулюванням об'єктами у віртуальному тривимірному просторі. Процес створення тривимірного зображення включає моделювання геометричної форми об'єктів, визначення матеріалів та текстур, налаштування освітлення та камери, і нарешті, рендеринг – перетворення тривимірної сцени у двовимірне зображення. Тривимірна графіка є основою сучасної комп'ютерної анімації, візуальних ефектів у кіно та відеоігор.

Розуміння основних понять комп'ютерної графіки є фундаментальною основою для подальшого вивчення цієї дисципліни. Студенти повинні усвідомлювати відмінності між різними видами графіки, їхні переваги та обмеження, сфери застосування. Це дозволить у майбутньому обирати оптимальні інструменти та методи для вирішення конкретних завдань у професійній діяльності.

Комп'ютерна графіка тісно інтегрована з іншими галузями інформатики, такими як штучний інтелект, обробка зображень, комп'ютерний зір, віртуальна та доповнена реальність. Сучасні тенденції розвитку включають використання машинного навчання для генерації та обробки зображень, створення фотореалістичних зображень у реальному часі, розвиток технологій трасування променів на споживчому апаратному забезпеченні.

Програмне забезпечення для роботи з комп'ютерною графікою поділяється на кілька категорій. Растрові графічні редактори, такі як Adobe Photoshop, GIMP, Krita, призначені для обробки фотографій та створення растрових ілюстрацій. Векторні редактори, включаючи Adobe Illustrator, CorelDRAW, Inkscape, використовуються для створення векторної графіки. Системи тривимірного моделювання, такі як Blender, Autodesk Maya, 3ds Max, Cinema 4D, надають потужні інструменти для роботи з тривимірною графікою.

Апаратне забезпечення відіграє критичну роль у продуктивності графічних застосунків. Сучасні графічні процесори містять тисячі обчислювальних ядер, оптимізованих для паралельної обробки графічних даних. Технології апаратного

прискорення, такі як CUDA, OpenCL, Vulkan, дозволяють ефективно використовувати можливості графічних процесорів не лише для візуалізації, але й для загальних обчислень.

Вивчення комп'ютерної графіки вимагає знань з лінійної алгебри, аналітичної геометрії, математичного аналізу, оскільки більшість алгоритмів базується на математичних перетвореннях координат, векторів та матриць. Розуміння фізичних принципів поширення світла, відбиття та заломлення необхідне для створення реалістичних зображень. Програмування є невід'ємною частиною роботи з комп'ютерною графікою, оскільки навіть при використанні готових програмних продуктів часто потрібно автоматизувати процеси або створювати власні плагіни та скрипти.

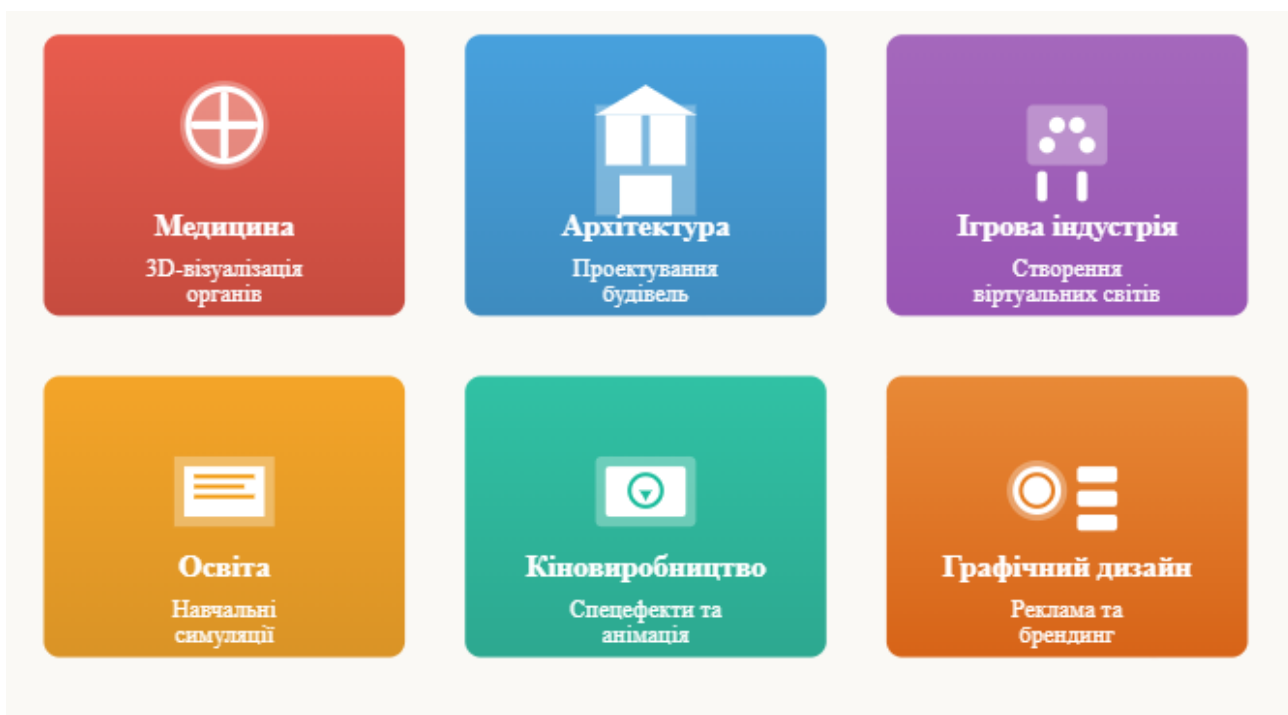


Рисунок 1.3 - Приклади застосування комп'ютерної графіки в різних галузях: наукова візуалізація, медична діагностика

ЛЕКЦІЯ 2

РАСТРОВА ГРАФІКА: ОСНОВИ ТА ХАРАКТЕРИСТИКИ

Растрова графіка є найпоширенішим способом представлення цифрових зображень і базується на концепції дискретизації візуальної інформації. Основним елементом растрового зображення є піксель – найменший неподільний елемент зображення, який має визначений колір та координати. Піксель утворюється в результаті перетину рядків і стовпців прямокутної сітки, яка називається растром. Сукупність пікселів утворює двовимірний масив, який зберігається у пам'яті комп'ютера та відображається на пристроях виведення.

Растрова графіка створюється за допомогою пікселів, які розташовані на прямокутній базовій матриці.

Кожен піксель характеризується своїм кольором, який кодується за допомогою певної кількості бітів. Глибина кольору визначає кількість можливих кольорових відтінків, які може мати окремий піксель. Найпростіший випадок – бінарне зображення, де кожен піксель може бути або чорним, або білим, що вимагає один біт пам'яті на піксель. Відтінки сірого використовують вісім біт на піксель, забезпечуючи 256 рівнів яскравості. Повноколірні зображення зазвичай використовують 24 біти на піксель, по вісім біт на кожний з трьох каналів червоного, зеленого та синього кольорів, що забезпечує понад 16 мільйонів можливих кольорів.

Роздільна здатність растрового зображення визначається кількістю пікселів по горизонталі та вертикалі. Наприклад, зображення з роздільною здатністю 1920 на 1080 пікселів містить понад два мільйони пікселів. Чим вища роздільна здатність, тим більше деталей може відтворити зображення, але тим більше пам'яті воно займає. Щільність пікселів вимірюється в пікселях на дюйм і визначає якість відтворення зображення на фізичних носіях. Для екранного

відображення зазвичай достатньо 72-96 пікселів на дюйм, тоді як для якісного друку потрібно 300 і більше пікселів на дюйм.

Растрова графіка залежить від роздільної здатності, тобто при збільшенні масштабу зображення втрачається якість.

Ця властивість називається залежністю від роздільної здатності і є основним обмеженням растрової графіки. При збільшенні масштабу зображення програмне забезпечення змушене інтерполювати нові пікселі на основі існуючих, що призводить до появи ступінчастих країв, розмиття та інших артефактів. Цей ефект особливо помітний при значному збільшенні, коли окремі пікселі стають видимими, утворюючи характерну "східчасту" структуру.

Кольорова модель визначає спосіб кодування кольору пікселя. Найпоширенішою моделлю для растрової графіки є RGB, яка базується на адитивному змішуванні червоного, зеленого та синього кольорів. Кожен канал може мати значення від 0 до 255, що забезпечує широкий діапазон кольорів. Інші кольорові моделі, такі як CMYK для друку, HSB для інтуїтивного вибору кольорів, LAB для професійної обробки зображень, також використовуються в растровій графіці залежно від конкретних завдань.

Формати файлів для збереження растрових зображень поділяються на дві основні категорії: формати зі стисненням без втрат та формати зі стисненням з втратами. Формати без втрат, такі як PNG, TIFF, BMP, зберігають усю інформацію про зображення без будь-яких спотворень, але займають більше місця на диску. Формати зі стисненням з втратами, найпопулярнішим з яких є JPEG, досягають значного зменшення розміру файлу шляхом відкидання частини візуальної інформації, яка менш помітна для людського ока.

Алгоритм стиснення JPEG базується на перетворенні зображення з просторової області в частотну за допомогою дискретного косинусного перетворення. Високочастотні компоненти, які відповідають за дрібні деталі та різкі переходи, стискаються сильніше, ніж низькочастотні, що відповідають за

плавні градієнти та загальну структуру зображення. Ступінь стиснення може регулюватися користувачем, дозволяючи знаходити баланс між якістю зображення та розміром файлу. Однак багаторазове збереження зображення у форматі JPEG призводить до накопичення артефактів стиснення та поступової деградації якості.

Формат PNG підтримує стиснення без втрат за алгоритмом DEFLATE, який поєднує алгоритми LZ77 та кодування Хаффмана. Цей формат підтримує прозорість через альфа-канал, що дозволяє створювати зображення з напівпрозорими ділянками. PNG ідеально підходить для збереження зображень з різкими переходами кольорів, таких як скріншоти, ілюстрації, логотипи. Формат TIFF є гнучким контейнером, який підтримує різні методи стиснення, багат шаровість, різні кольорові простори та глибоку кольоровість, що робить його стандартом у поліграфії та професійній фотографії.

Растрові зображення складаються з мільйонів крихітних пікселів, кожен з яких має свій колір for-k.com.ua.

Обробка растрових зображень включає широкий спектр операцій, від базових, таких як зміна розміру, обрізання, поворот, до складних, таких як фільтрація, корекція кольору, ретушування. Фільтри застосовують математичні операції до матриці пікселів, змінюючи значення кольорів на основі значень сусідніх пікселів. Розмиття, підвищення різкості, виявлення країв – всі ці ефекти реалізуються за допомогою згортки з різними ядрами.

Корекція кольору дозволяє змінювати яскравість, контрастність, насиченість, кольоровий баланс зображення. Гістограма зображення, яка показує розподіл пікселів за рівнями яскравості, є потужним інструментом для аналізу та корекції тонального діапазону. Криві та рівні дозволяють точно налаштувати тональну компресію, розтягнути або стиснути діапазон яскравостей, виправити проблеми з експозицією.

Маски та виділення є фундаментальними інструментами для селективної обробки зображень. Виділення дозволяє ізолювати певну область зображення для подальших операцій, таких як копіювання, видалення, застосування фільтрів. Маски зберігають виділення у вигляді grayscale-зображення, де білий колір відповідає повністю виділеній області, чорний – невиділеній, а відтінки сірого – частковому виділенню. Це дозволяє створювати плавні переходи між обробленими та необробленими ділянками.

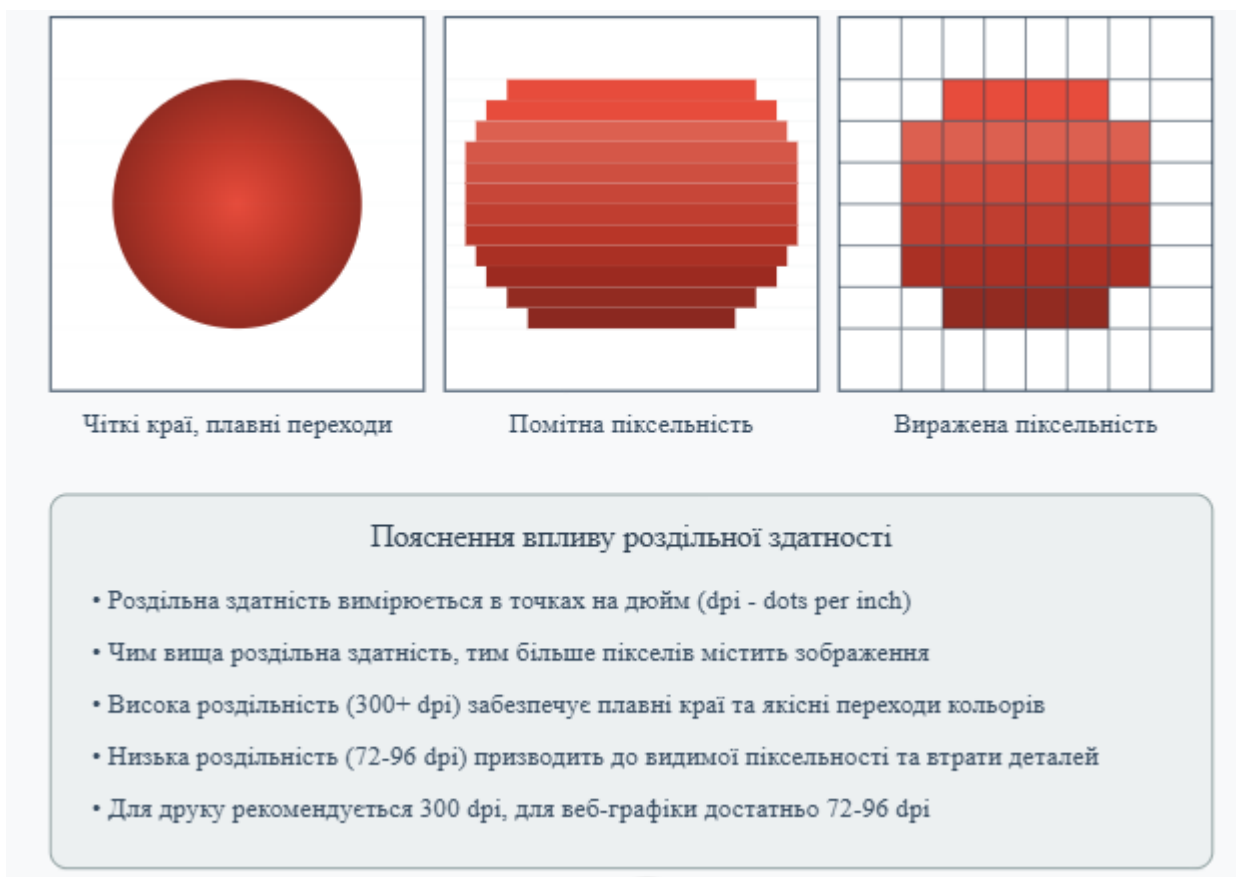


Рисунок 2.1 - Приклад растрового зображення з різною роздільною здатністю: вплив на якість

Шари є потужною концепцією в растровій графіці, яка дозволяє розділяти різні елементи зображення на окремі площини, які можна редагувати незалежно.

Кожен шар може мати власний режим накладання, прозорість, ефекти. Шари дозволяють створювати складні композиції, експериментувати з різними варіантами без ризику втратити оригінальне зображення, працювати нелінійно, змінюючи порядок елементів.

Растрова графіка є основним форматом для цифрової фотографії, оскільки фотографічні зображення містять складні градієнти, текстури та деталі, які природно відображаються піксельною сіткою. Сучасні цифрові камери створюють зображення з роздільною здатністю від кількох мегапікселів до сотень мегапікселів у професійних моделях. Обробка сирих даних з матриці камери вимагає демозаїкінгу – інтерполяції повноколірного зображення з фільтру Баєра, який реєструє лише один колірний канал на кожному пікселі.

Сканування фізичних зображень перетворює аналогову інформацію у цифрову растрову форму. Процес сканування включає дискретизацію просторової інформації з певною роздільною здатністю та квантування кольору з певною глибиною. Якість сканованого зображення залежить від оптичної роздільної здатності сканера, динамічного діапазону, точності кольоропередачі.

Растрові зображення широко використовуються у веб-дизайні, де важливо знаходити баланс між якістю та розміром файлу для швидкого завантаження. Адаптивні зображення, які автоматично підлаштовуються під розмір екрана та щільність пікселів пристрою, стають стандартом сучасної веб-розробки. Формати нового покоління, такі як WebP, AVIF, пропонують краще стиснення порівняно з традиційними форматами при збереженні якості.



Рисунок 2.2 - Порівняння різних форматів стиснення растрових зображень: JPEG, PNG, WebP

Незважаючи на обмеження, растрова графіка залишається домінуючим форматом для представлення візуальної інформації завдяки своїй універсальності, здатності відтворювати складні візуальні ефекти та широкий підтримці апаратного та програмного забезпечення. Розуміння принципів растрової графіки, форматів збереження, методів обробки є необхідною базою для будь-якого фахівця у галузі комп'ютерної графіки.

ЛЕКЦІЯ 3

ВЕКТОРНА ГРАФІКА: ПРИНЦИПИ ПОБУДОВИ ТА ПЕРЕВАГИ

Векторна графіка представляє собою фундаментально інший підхід до створення та збереження цифрових зображень порівняно з растровою графікою. Замість зберігання інформації про кожен окремий піксель, векторна графіка використовує математичний опис геометричних об'єктів, таких як точки, лінії, криві, багатокутники та інші примітиви. Кожен об'єкт у векторному зображенні визначається набором координат, параметрів форми, атрибутів заливки та обведення, що дозволяє зберігати зображення у компактному вигляді незалежно від його візуальної складності.

Основним елементом зображення у векторній графіці є контур, який описується за допомогою математичних рівнянь. Найпростішим прикладом є пряма лінія, яка визначається координатами двох точок. Коло описується координатами центру та радіусом. Більш складні форми, такі як криві Без'є, використовують контрольні точки для визначення форми кривої. Цей математичний підхід дозволяє точно відтворювати геометричні форми на будь-якому пристрої виведення незалежно від його роздільної здатності.

Переваги векторної графіки є численними та визначають її широке застосування у різних сферах. Найважливішою перевагою є масштабованість без втрати якості. Оскільки зображення описується математичними формулами, воно може бути відтворене з будь-яким рівнем деталізації. Це робить векторну графіку ідеальною для створення логотипів, іконок, шрифтів, технічних креслень, де важливо зберігати чіткість контурів при будь-якому масштабі.

Іншою важливою перевагою є компактність файлів. Векторні зображення, що складаються з простих геометричних форм, займають значно менше місця порівняно з растровими аналогами аналогічної візуальної складності. Наприклад, логотип, що складається з кількох кривих та заливок, може бути описаний

кількома кілобайтами даних, тоді як його растровий еквівалент високої роздільної здатності може займати мегабайти.

Векторна графіка підтримує об'єктно-орієнтований підхід до редагування. Кожен елемент зображення залишається окремим об'єктом, який можна виділити, перемістити, змінити його атрибути без впливу на інші елементи. Це дозволяє легко вносити зміни до складних композицій, експериментувати з різними варіантами дизайну, створювати бібліотеки багаторазово використовуваних елементів.

Колір у векторній графіці застосовується до об'єктів у вигляді заливки внутрішньої області та обведення контуру. Заливка може бути суцільним кольором, градієнтом, візерунком або навіть іншим векторним об'єктом. Обведення характеризується товщиною, типом лінії, кольором, стилем кінцівок та з'єднань. Такі атрибути дозволяють створювати візуально багаті зображення з мінімальними витратами пам'яті.

Векторні формати файлів, такі як SVG, AI, EPS, CDR, мають різне призначення та сумісність. SVG є відкритим стандартом, що підтримується веб-браузерами, що робить його ідеальним для веб-графіки. Формати AI та EPS є стандартами у поліграфії та професійному дизайні, підтримують складні ефекти та сумісні з програмним забезпеченням Adobe. CDR є рідним форматом CorelDRAW, популярного у технічному дизайні.

Створення векторної графіки здійснюється за допомогою спеціалізованих редакторів, які надають інструменти для малювання примітивів, роботи з кривими, маніпулювання об'єктами. Інструмент "перо" дозволяє створювати довільні контури шляхом розміщення опорних точок та налаштування дотичних. Інструменти геометричних фігур спрощують створення стандартних форм. Функції вирівнювання, розподілу, групування полегшують роботу зі складними композиціями.

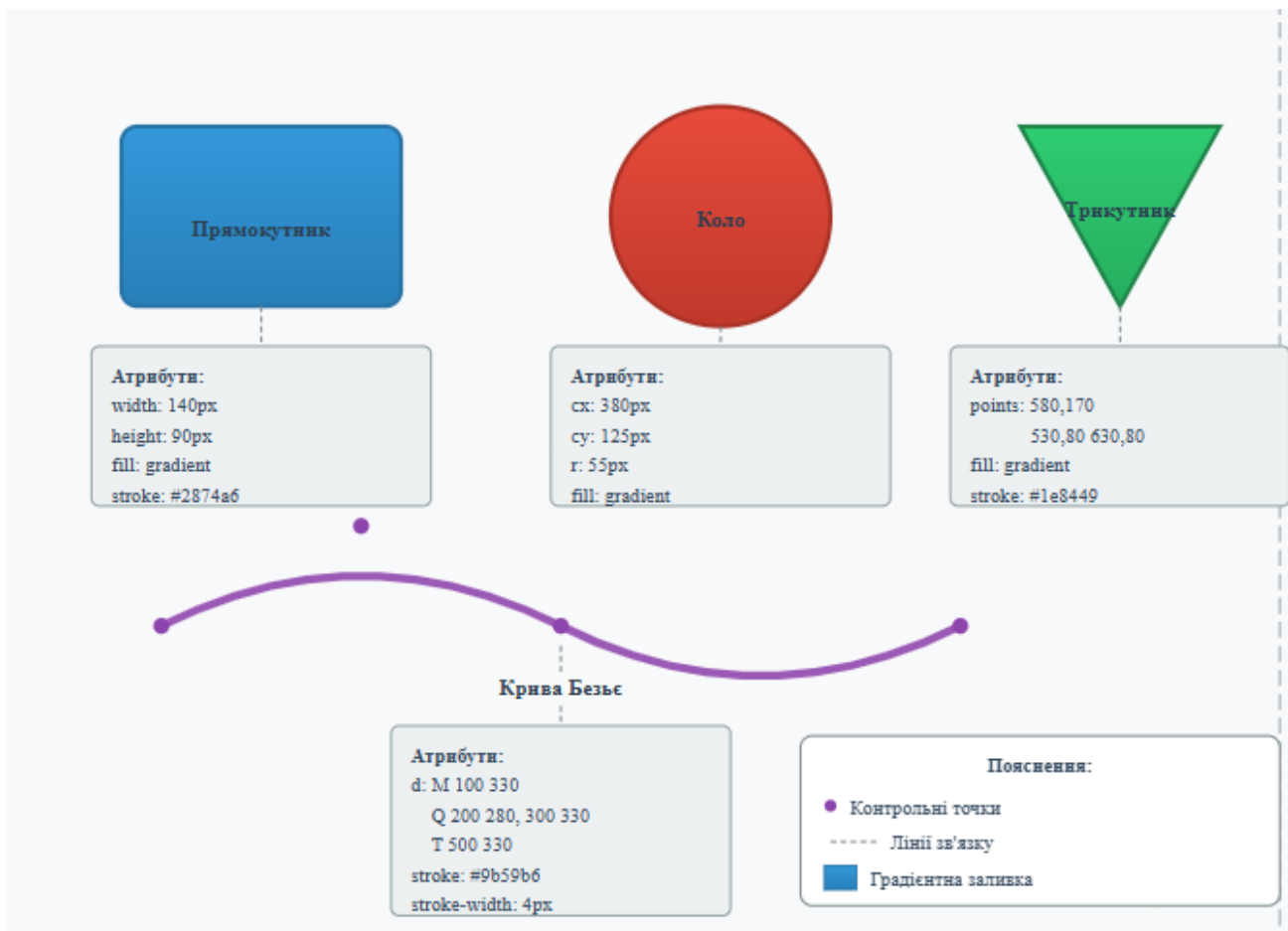


Рисунок 3.1 - Приклад векторного зображення з виділеними об'єктами та їхніми атрибутами

Трансформації є потужним інструментом векторної графіки. Об'єкти можуть бути переміщені, повернуті, масштабовані, віддзеркалені, скошені за допомогою афінних перетворень. Ці операції виконуються шляхом множення координат на матриці перетворень, що забезпечує математичну точність та оборотність. Комбінування трансформацій дозволяє створювати складні композиції з простих елементів.

Булеві операції над об'єктами дозволяють створювати нові форми шляхом логічних операцій об'єднання, перетину, віднімання, виключного АБО. Ці

операції є особливо корисними при створенні складних логотипів, іконок, технічних ілюстрацій, де потрібно точно контролювати форму кінцевого об'єкта. Результат булевих операцій залишається векторним об'єктом, зберігаючи всі переваги масштабованості.

Векторна графіка широко використовується у веб-дизайні завдяки формату SVG, який підтримує анімацію, інтерактивність, стилізацію за допомогою CSS. Векторні іконки та інтерфейсні елементи виглядають чітко на екранах з будь-якою щільністю пікселів, включаючи дисплеї високої роздільної здатності. Адаптивна векторна графіка автоматично підлаштовується під розмір контейнера, забезпечуючи оптимальне відображення на різних пристроях.



Рисунок 3.2 - Приклад використання векторної графіки у веб-дизайні:

У поліграфії векторна графіка є стандартом для створення макетів, логотипів, шрифтів, ілюстрацій. Можливість експорту у формат з будь-якою роздільною здатністю забезпечує високу якість друку на будь-якому носії. Векторні файли легко інтегруються у системи верстки, підтримують управління кольором за допомогою профілів ICC, що гарантує точність кольоропередачі.

Незважаючи на переваги, векторна графіка має обмеження. Вона не підходить для відтворення фотографічних зображень зі складними градієнтами та текстурами, де растровий підхід є більш ефективним. Складні векторні зображення з тисячами об'єктів можуть вимагати значних обчислювальних ресурсів для відображення та редагування. Деякі візуальні ефекти, такі як розмиття, тіні, прозорість, можуть бути складними для реалізації у векторному форматі.

Сучасні векторні редактори поєднують можливості векторної та растрової графіки, дозволяючи імпортувати растрові зображення як текстури або фони, застосовувати растрові ефекти до векторних об'єктів, експортувати композиції у різних форматах. Така гібридна модель розширює творчі можливості дизайнерів та інженерів, дозволяючи обирати оптимальний підхід для кожного завдання.

Розуміння принципів векторної графіки є критично важливим для розробників програмного забезпечення, оскільки багато графічних бібліотек та форматів базуються на векторних концепціях. Бібліотеки, такі як Cairo, Skia, Qt Graphics View, надають API для програмування векторної графіки. Формати, такі як PDF, SVG, PostScript, використовують векторний опис для забезпечення масштабованості та сумісності.

ЛЕКЦІЯ 4

ФРАКТАЛЬНА ТА ТРИВИМІРНА ГРАФІКА

Фрактальна графіка займає особливе місце у світі комп'ютерної візуалізації, базуючись на математичних об'єктах, які демонструють властивість самоподібності. Фрактал – це геометрична фігура, окремі частини якої подібні до фігури в цілому. Ця властивість дозволяє описувати надзвичайно складні форми за допомогою відносно простих рекурсивних алгоритмів, що робить фрактальну графіку потужним інструментом для моделювання природних явищ та створення художніх візуалізацій.

Математичною основою фрактальної графіки є ітеративні процеси та рекурсивні функції. Найвідомішим прикладом є множина Мандельброта, яка визначається ітерацією комплексної функції $f(z) = z^2 + c$. Для кожної точки комплексної площини виконується послідовність ітерацій, і колір точки визначається кількістю ітерацій, необхідних для досягнення певного порогу. Результатом є нескінченно складна межа з фрактальною розмірністю, яка демонструє самоподібність на різних масштабах.

Фрактальна графіка знаходить широке застосування у моделюванні природних об'єктів, які важко описати за допомогою класичної геометрії. Гори, хмари, берегові лінії, дерева, листя – всі ці форми мають фрактальну природу. Алгоритми, такі як система ітерованих функцій, L-системи, метод зсуву середини, дозволяють генерувати реалістичні природні ландшафти та об'єкти за допомогою невеликої кількості параметрів.

У комп'ютерній графіці фрактали використовуються для генерації процедурних текстур, які можуть бути масштабовані без втрати деталізації. Фрактальний шум, такий як шум Перліна або симплекс-шум, застосовується для створення природних поверхонь, хмар, вогню, води. Процедурна генерація на

основі фракталів дозволяє створювати унікальні візуальні ефекти без необхідності зберігати великі обсяги даних.

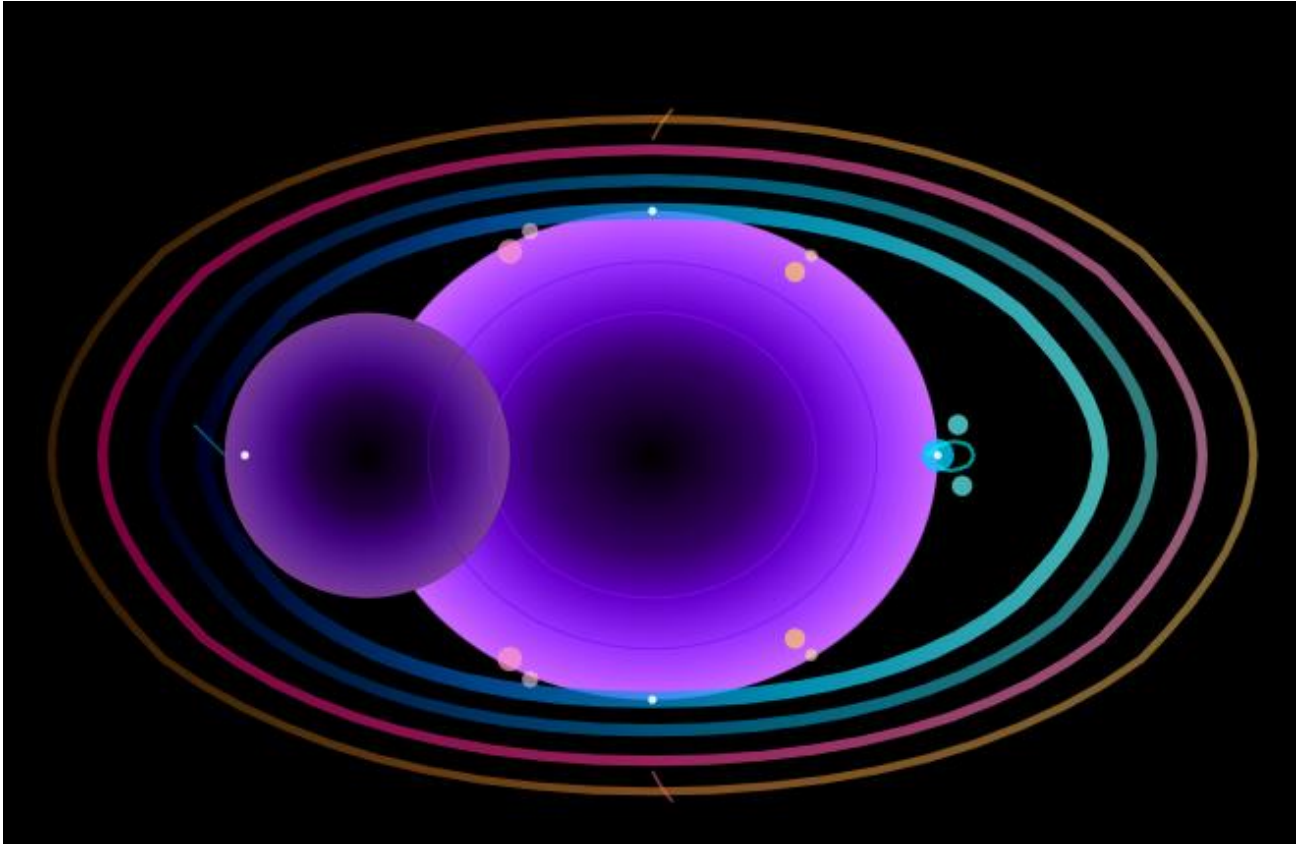


Рисунок 4.1 - Множина Мандельброта: приклад фрактального зображення з нескінченною деталізацією

Тривимірна графіка представляє собою найбільш складний та потужний напрямок комп'ютерної візуалізації, який дозволяє створювати та маніпулювати об'єктами у віртуальному тривимірному просторі. Процес створення тривимірного зображення включає кілька етапів: моделювання геометричної форми, визначення матеріалів та текстур, налаштування освітлення та камери, і нарешті, рендеринг – перетворення тривимірної сцени у двовимірне зображення.

Моделювання тривимірних об'єктів здійснюється за допомогою різних підходів. Полігональне моделювання базується на представленні поверхонь у вигляді сітки трикутників або чотирикутників. Цей підхід є найбільш поширеним завдяки простоті реалізації та ефективній підтримці апаратним забезпеченням. Параметричне моделювання використовує математичні поверхні, такі як поверхні Без'є або NURBS, для створення гладких форм з точною геометрією. Скульптурне моделювання імітує процес ліплення, дозволяючи художникам інтуїтивно формувати об'єкти.

Координатні системи є фундаментальним поняттям тривимірної графіки. Локальна система координат об'єкта визначає його геометрію відносно власного початку. Світова система координат визначає положення всіх об'єктів сцени відносно спільного початку. Система координат камери визначає, як сцена проектується на двовимірну площину зображення. Перетворення між цими системами виконуються за допомогою матриць 4×4 , які включають обертання, масштабування та перенесення.

Проекція є процесом перетворення тривимірних координат у двовимірні координати екрана. Ортографічна проекція зберігає паралельність ліній та використовується у технічних кресленнях. Перспективна проекція імітує сприйняття людським оком, де віддалені об'єкти виглядають меншими. Матриця проекції визначає параметри камери, такі як кут огляду, співвідношення сторін, ближня та дальня площини відсічення.

Відсічення невидимих поверхонь є критично важливим етапом рендерингу, який визначає, які частини об'єктів видимі з точки зору камери. Алгоритм Z-буфера зберігає глибину кожного пікселя та порівнює нові фрагменти з існуючими, відкидаючи ті, що знаходяться позаду. Алгоритм художника сортує полігони за відстанню та малює їх від задніх до передніх, забезпечуючи правильне перекриття.

Освітлення у тривимірній графіці моделює взаємодію світла з поверхнями об'єктів. Модель Фонга розраховує інтенсивність світла як суму фонового, дифузного та спекулярного компонентів. Дифузне відбиття залежить від кута між нормаллю поверхні та напрямком світла, створюючи основне освітлення об'єкта. Спекулярне відбиття імітує блики на глянцеvih поверхнях, залежачи від кута між відбитим променем та напрямком на спостерігача.

Тривимірна графіка є основою сучасної комп'ютерної анімації, візуальних ефектів у кіно, відеоігор, віртуальної та доповненої реальності. Розвиток апаратного прискорення графічних обчислень дозволяє рендерити складні сцени у реальному часі, забезпечуючи інтерактивність та іммерсивність. Сучасні графічні процесори підтримують трасування променів апаратно, що дозволяє досягти фотореалістичної якості зображення.

Програмне забезпечення для тривимірної графіки включає системи моделювання, такі як Blender, Autodesk Maya, 3ds Max, Cinema 4D, які надають потужні інструменти для створення та анімації 3D-об'єктів. Ігрові рушії, такі як Unity, Unreal Engine, Godot, інтегрують графічний конвеєр з фізикою, штучним інтелектом, мережевими можливостями для створення інтерактивних застосунків.

Розуміння принципів тривимірної графіки є необхідним для розробників програмного забезпечення, оскільки багато застосунків вимагають інтеграції 3D-візуалізації. Графічні бібліотеки, такі як OpenGL, DirectX, Vulkan, надають низькорівневий доступ до графічного конвеєра. Бібліотеки високого рівня, такі як Three.js, Babylon.js, спрощують створення 3D-графіки у веб-середовищі.

ЛЕКЦІЯ 5

КОЛОРИМЕТРІЯ. АДИТИВНА МОДЕЛЬ RGB

Колориметрія є науковою дисципліною, яка вивчає методи кількісного вимірювання та опису кольору. Розуміння фізичних та фізіологічних аспектів сприйняття кольору є фундаментальним для комп'ютерної графіки, оскільки дозволяє точно відтворювати візуальну інформацію на різних пристроях. Людське око сприймає колір за допомогою трьох типів колбочок, чутливих до різних діапазонів спектру, що лежить в основі трихроматичної теорії кольору.

Адитивна колірна модель RGB є основною моделлю для пристроїв, що випромінюють світло, таких як монітори, телевізори, проектори. Назва моделі походить від англійських слів Red, Green, Blue – червоний, зелений, синій, які є основними кольорами цієї системи. Принцип адитивного змішування полягає в тому, що колір утворюється шляхом додавання світлових потоків різних кольорів.

У моделі RGB кожен колір представлений трьома компонентами, значення яких зазвичай змінюються від 0 до 255 у 8-бітному представленні. Чорний колір відповідає відсутності світла (0, 0, 0), білий колір – максимальній інтенсивності всіх трьох компонентів (255, 255, 255). Проміжні значення створюють мільйони відтінків шляхом різного співвідношення червоного, зеленого та синього каналів.

Адитивне змішування демонструє цікаві властивості. Додавання червоного та зеленого світла дає жовтий колір, червоного та синього – пурпуровий, зеленого та синього – блакитний. Рівномірне змішування всіх трьох основних кольорів у різних пропорціях дозволяє отримати будь-який колір у видимому спектрі. Це пояснює, чому екрани використовують лише три типи світлодіодів або люмінофорів для відтворення повноколірного зображення.

Глибина кольору визначає кількість бітів, що використовуються для кодування кожного каналу. 8-бітне представлення на канал забезпечує 256 рівнів

яскравості, що дає понад 16 мільйонів можливих кольорів. Професійні дисплеї можуть використовувати 10- або 12-бітне представлення, забезпечуючи більшу точність кольоропередачі та плавніші градієнти.

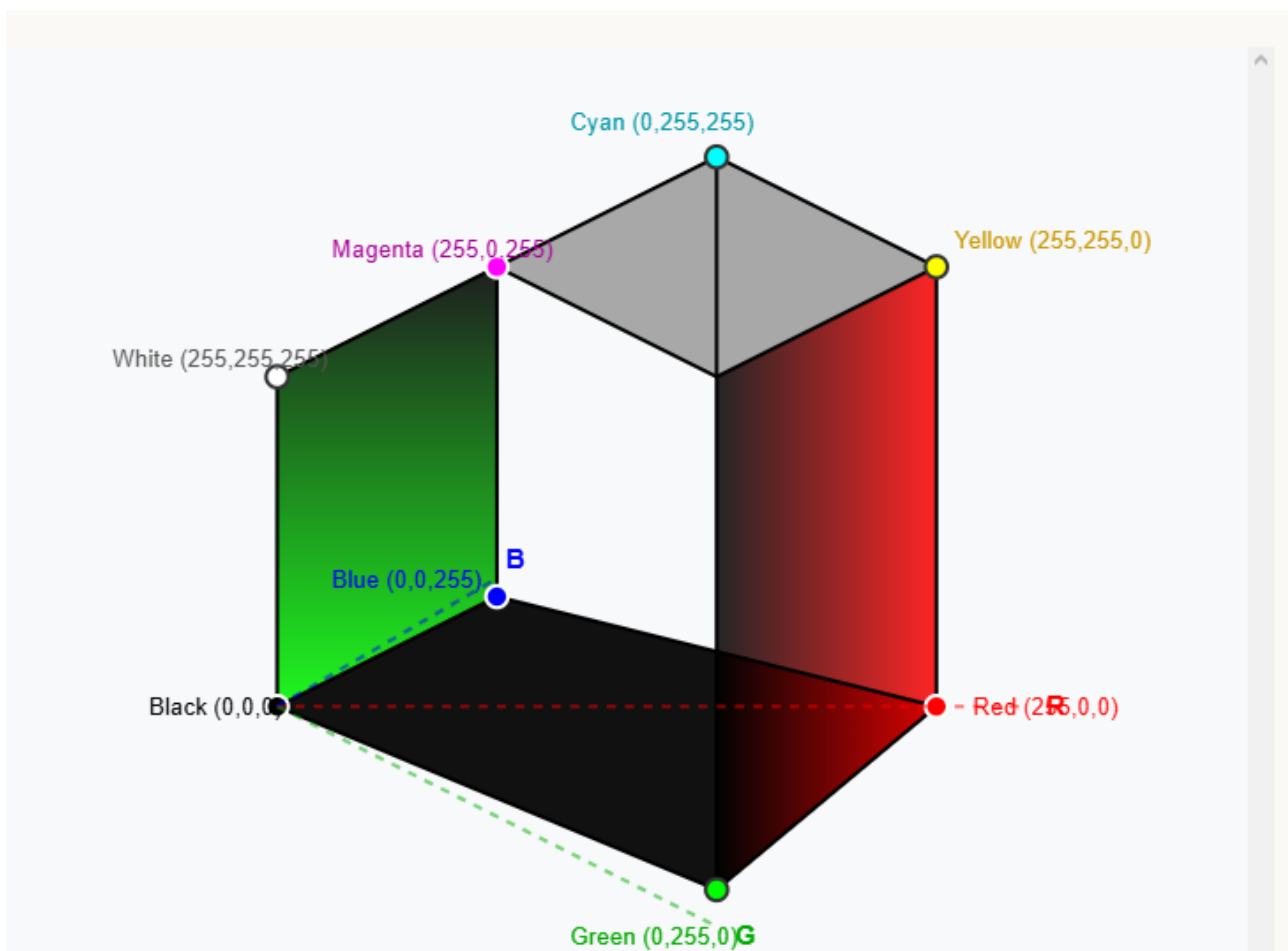


Рисунок 5.1 - Куб кольорів моделі RGB: тривимірне представлення простору кольорів

Колірний простір RGB не є уніфікованим – різні пристрої можуть інтерпретувати однакові числові значення по-різному. Стандартизовані колірні простори, такі як sRGB, Adobe RGB, ProPhoto RGB, визначають конкретні хроматичності основних кольорів та функцію передачі тону. sRGB є найпоширенішим стандартом для веб-контенту та споживчої електроніки, забезпечуючи сумісність між різними пристроями.

Управління кольором є критично важливим для забезпечення консистентності візуалізації на різних пристроях. Профілі ICC містять інформацію про колірні характеристики пристрою та дозволяють перетворювати кольори між різними просторами. Калібрування моніторів за допомогою колориметрів забезпечує точне відтворення кольорів, що є особливо важливим у професійній фотографії, дизайні, поліграфії.

Програмування з використанням моделі RGB є основою багатьох графічних застосунків. У мовах програмування, таких як Python з бібліотекою PIL або OpenCV, колір пікселя задається кортежем із трьох значень. Графічні бібліотеки, такі як OpenGL, використовують нормалізовані значення від 0.0 до 1.0 для кожного каналу. Розуміння цих особливостей дозволяє ефективно працювати з кольором на програмному рівні.

Адитивна модель має обмеження, пов'язані з фізичними можливостями пристроїв відображення. Жоден реальний дисплей не може відтворити весь діапазон кольорів, видимих людським оком. Гамма-корекція компенсує нелінійність сприйняття яскравості людським оком, забезпечуючи більш природне відображення градієнтів.

Сучасні технології дисплеїв, такі як OLED, Mini-LED, Quantum Dot, розширюють можливості відтворення кольору, забезпечуючи більшу яскравість, контрастність та ширший колірний охоплення. Підтримка стандартів, таких як HDR10, Dolby Vision, дозволяє передавати зображення з розширеним динамічним діапазоном, зберігаючи деталі як у тінях, так і у світлих ділянках.

Розуміння адитивної моделі RGB є необхідним для розробників, які створюють графічні інтерфейси, обробляють зображення, працюють з відео. Правильне використання кольорових каналів дозволяє реалізувати ефекти, такі як змішування шарів, корекція кольору, створення фільтрів. Знання принципів колориметрії забезпечує професійний підхід до візуалізації даних та створення користувацьких інтерфейсів.

ЛЕКЦІЯ 6

СУБТРАКТИВНА МОДЕЛЬ СМУК ТА МОДЕЛЬ LAB

Субтрактивна колірна модель СМУК є основою поліграфічного виробництва та друку, базуючись на принципі віднімання світла. На відміну від адитивної моделі RGB, де колір утворюється додаванням світлових потоків, субтрактивна модель працює за рахунок поглинання частин спектру білого світла пігментами або барвниками. Назва моделі походить від Cyan, Magenta, Yellow, Key (black) – блакитний, пурпуровий, жовтий, чорний.

Принцип субтрактивного змішування полягає в тому, що кожен барвник поглинає певну частину видимого спектру. Блакитний пігмент поглинає червоне світло, пурпуровий – зелене, жовтий – синє. При накладанні всіх трьох кольорів теоретично має виходити чорний колір, оскільки поглинається весь спектр. Однак через недосконалість реальних пігментів суміш дає брудно-коричневий відтінок, тому додається чорний компонент для отримання глибокого чорного та економії кольорових барвників.

У моделі СМУК кожен канал представляє кількість барвника, що наноситься на папір, і зазвичай вимірюється у відсотках від 0% до 100%. Нульове значення означає відсутність барвника, 100% – максимальне покриття. Процес друку передбачає послідовне накладання чотирьох фарб через окремі друкарські форми, що вимагає точної реєстрації для уникнення розмиття та кольорових спотворень.

Конвертація між моделями RGB та СМУК є необхідним етапом підготовки зображень до друку. Оскільки колірний охоплення СМУК є вужчим за RGB, деякі яскраві кольори, відтворювані на екрані, не можуть бути точно надруковані. Професійне програмне забезпечення надає інструменти для попереднього перегляду друкованого результату та корекції кольорів для мінімізації втрат.

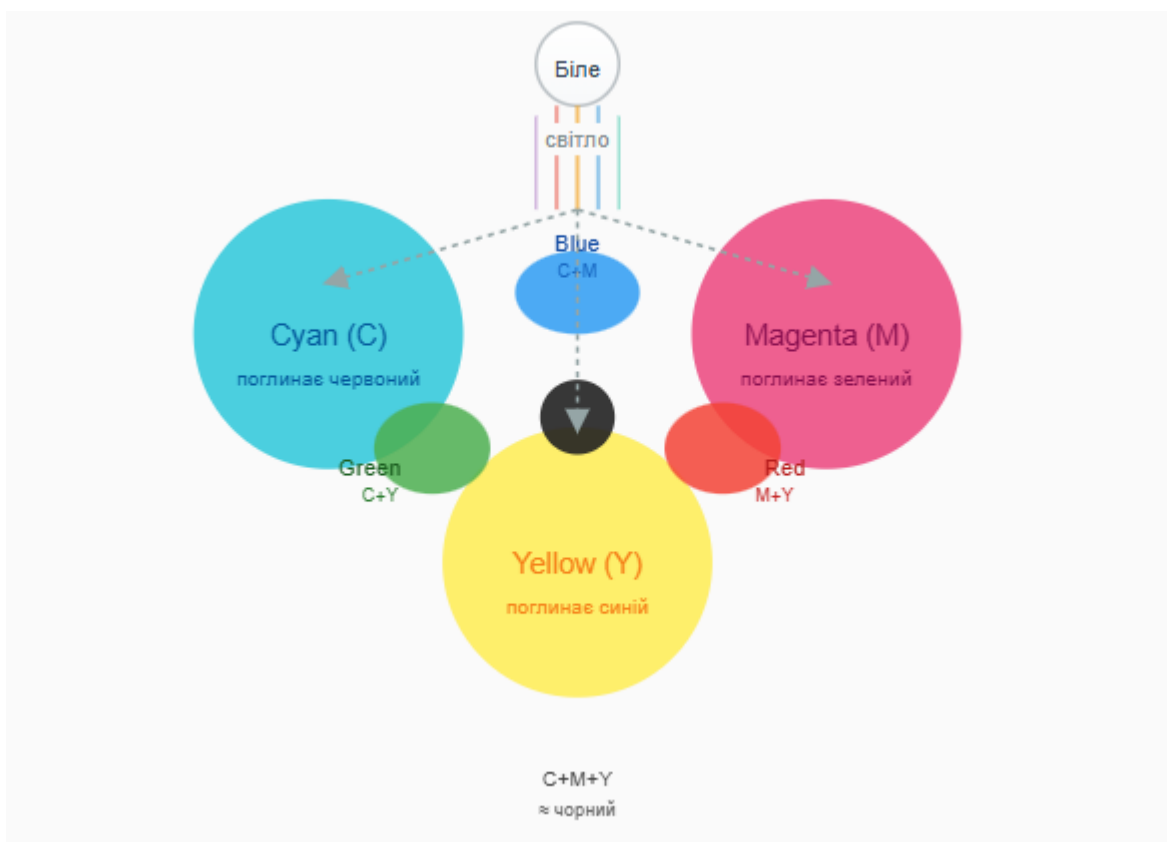


Рисунок 6.1 - Принцип субтрактивного змішування кольорів: поглинання світла пігментами

Модель LAB, також відома як CIELAB, є пристрій-незалежною колірною моделлю, розробленою Міжнародною комісією з освітлення для опису всіх кольорів, видимих людським оком. Вона базується на трьох компонентах: L – світлота (від 0 для чорного до 100 для білого), а – вісь від зеленого до червоного, b – вісь від синього до жовтого. Така структура відображає спосіб сприйняття кольору людською зоровою системою.

Перевагою моделі LAB є рівномірність сприйняття – однакові числові відмінності відповідають приблизно однаковим візуальним відмінностям. Це робить LAB ідеальною для кольорних вимірювань, порівняння кольорів, корекції

зображень. У просторі LAB можна виконувати корекцію контрастності та насиченості незалежно від кольорового тону, що важко реалізувати в RGB або CMYK.

Модель LAB широко використовується у професійній обробці зображень для складних корекцій. Наприклад, підвищення різкості можна застосовувати лише до каналу світлоти, уникаючи посилення кольорового шуму. Селективна корекція кольору дозволяє змінювати конкретні відтінки без впливу на інші. Конвертація між LAB та іншими моделями дозволяє використовувати переваги кожного простору для різних етапів обробки.

Колірне управління в сучасних робочих процесах базується на стандартизованих профілях ICC, які описують характеристики пристроїв у пристрій-незалежному просторі, такому як LAB або XYZ. Це дозволяє точно перетворювати кольори між сканерами, моніторами, принтерами, забезпечуючи консистентність від початку до кінця виробничого ланцюжка.

Розуміння субтрактивної моделі CMYK та моделі LAB є критично важливим для розробників, які працюють з поліграфічними застосунками, системами управління кольором, професійними графічними редакторами. Правильна обробка кольору забезпечує високу якість кінцевого продукту та задоволення вимог клієнтів у різних галузях виробництва.

ЛЕКЦІЯ 7

МОДЕЛІ HSB І HLS. ПРАКТИЧНЕ ВИКОРИСТАННЯ КОЛІРНИХ МОДЕЛЕЙ

Моделі HSB (Hue, Saturation, Brightness) та HLS (Hue, Lightness, Saturation) є інтуїтивно зрозумілими колірними просторами, розробленими для зручності вибору та корекції кольорів людиною. На відміну від технічних моделей, таких як RGB або CMYK, ці моделі організують колір відповідно до способу мислення художників та дизайнерів, що робить їх ідеальними для інтерфейсів вибору кольору та інструментів корекції.

Модель HSB представляє колір за трьома параметрами. Відтінок (Hue) визначає основний колір у спектрі та вимірюється у градусах від 0° до 360°, де 0° – червоний, 120° – зелений, 240° – синій. Насиченість (Saturation) визначає інтенсивність кольору від 0% (відтінок сірого) до 100% (максимально насичений колір). Яскравість (Brightness) визначає кількість світла від 0% (чорний) до 100% (максимальна яскравість).

Модель HLS має схожу структуру, але використовує параметр світлоти (Lightness) замість яскравості. Світлота 0% відповідає чорному, 100% – білому, а 50% – "чистому" кольору. Таке представлення дозволяє легко отримувати пастельні відтінки зменшенням насиченості та темні варіанти зменшенням світлоти, що є зручним для створення колірних схем та палітр.

Практичне використання моделей HSB/HLS проявляється у багатьох аспектах комп'ютерної графіки. Інтерфейси вибору кольору часто використовують двовимірне поле для насиченості та яскравості з кольоровою смугою для відтінку, що дозволяє інтуїтивно знаходити потрібний колір. Інструменти корекції зображень застосовують ці моделі для селективної зміни кольорових характеристик без впливу на інші параметри.

У веб-дизайні моделі HSB/HLS використовуються для генерації колірних варіацій. Наприклад, створення станів кнопки (normal, hover, active) може

базуватися на зміні світлоти або насиченості базового кольору. Адаптивні теми інтерфейсів можуть автоматично генерувати темні варіанти шляхом корекції параметрів світлоти.

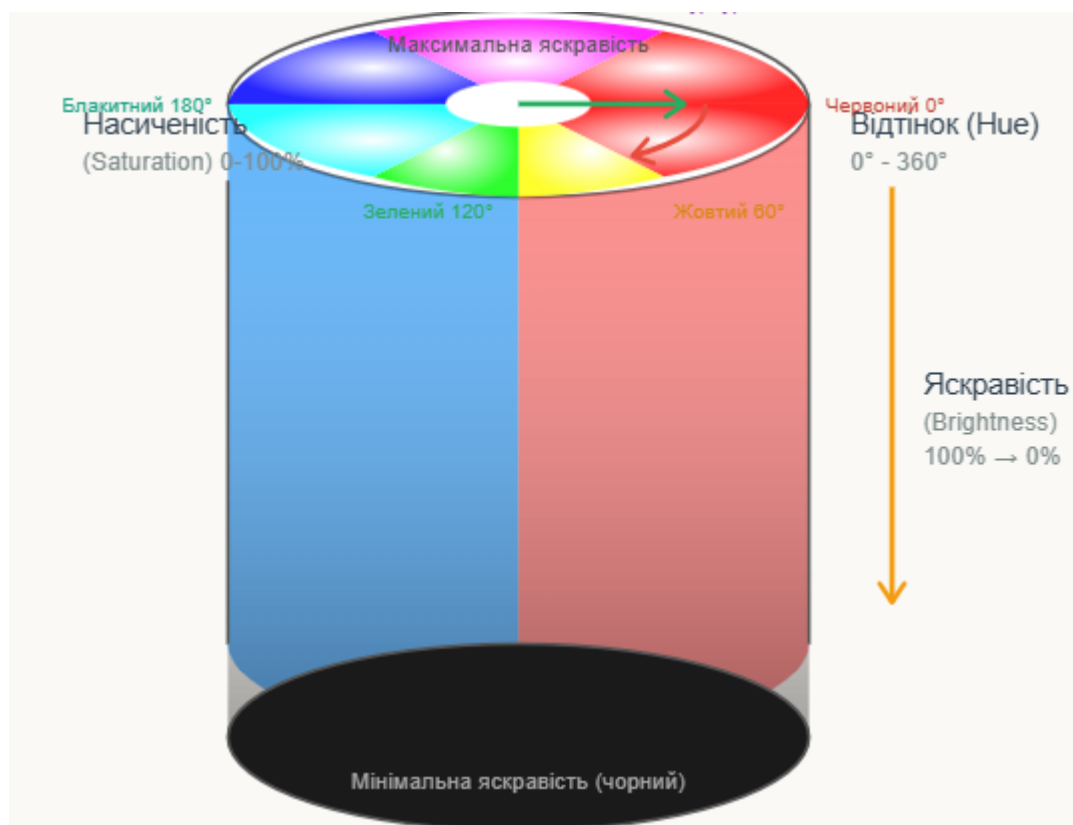


Рисунок 7.1 - Циліндричне представлення колірної моделі HSB: відтінок по колу, насиченість по радіусу, яскравість по вертикалі

Програмування з використанням HSB/HLS спрощує реалізацію колірних алгоритмів. Багато графічних бібліотек надають функції конвертації між RGB та HSB/HLS, що дозволяє виконувати операції у зручному просторі та повертати результат у стандартний формат. Наприклад, створення градієнту між двома кольорами може бути реалізоване шляхом інтерполяції параметрів відтінку, насиченості та яскравості.

Вибір колірної моделі залежить від конкретного завдання. RGB є оптимальним для роботи з екранним відображенням та програмуванням графіки.

СМУК необхідний для підготовки до друку. LAB використовується для професійної колірної корекції та вимірювань. HSB/HLS ідеальні для інтуїтивного вибору кольору та створення гармонійних палітр. Розуміння переваг та обмежень кожної моделі дозволяє обирати оптимальний інструмент для кожного етапу роботи.

Сучасні системи управління кольором підтримують автоматичну конвертацію між різними моделями, забезпечуючи гнучкість робочого процесу. Однак важливо пам'ятати, що конвертації можуть призводити до втрати точності, особливо при переході між просторами з різним колірним охопленням. Професійний підхід передбачає роботу у пристрій-незалежному просторі якомога довше та виконання фінальної конвертації безпосередньо перед виведенням.

Розуміння практичного використання колірних моделей є необхідним для розробників інтерфейсів, графічних застосунків, систем візуалізації даних. Правильний вибір моделі та методів корекції забезпечує високу якість візуального контенту та покращує користувацький досвід.

ЛЕКЦІЯ 8

РОБОТА З ДВОВИМІРНОЮ ГРАФІКОЮ. ШАРИ ТА ДОПОМІЖНІ ЗАСОБИ

Робота з двовимірною графікою вимагає використання потужних інструментів організації та точного контролю над елементами зображення. Шари є фундаментальною концепцією сучасних графічних редакторів, яка дозволяє розділяти складне зображення на окремі площини, що можна редагувати незалежно. Кожен шар може містити растрові зображення, векторні об'єкти, текст, ефекти, і при цьому залишатися ізольованим від інших елементів композиції.

Шари працюють за принципом прозорості плівки, накладеної одна на одну. Верхні шари перекривають нижні, але прозорі ділянки дозволяють бачити контент нижчих рівнів. Така організація дозволяє створювати складні композиції, експериментувати з різними варіантами без ризику втратити оригінальні дані, застосовувати ефекти до окремих елементів. Порядок шарів можна змінювати, впливаючи на візуальне відображення сцени.

Режими накладання шарів визначають, як пікселі верхнього шару взаємодіють з пікселями нижніх шарів. Режим "Нормальний" просто замінює пікселі, "Помноження" затемнює зображення, "Екран" освітлює, "Перекриття" підвищує контраст. Ці режими базуються на математичних операціях над значеннями кольорів і дозволяють створювати складні візуальні ефекти без модифікації вихідних даних.

Допоміжні засоби для точного моделювання включають сітки, напрямні, прив'язки, які полегшують вирівнювання та позиціонування об'єктів. Сітка – це регулярна система ліній, що відображається на робочому полотні та допомагає дотримуватися рівномірних інтервалів. Напрямні – це лінії, які можна вільно розміщувати для вирівнювання конкретних елементів. Прив'язка автоматично

"притягує" об'єкти до сітки, напрямних або країв інших об'єктів, забезпечуючи точність розміщення.

Трансформаційні інструменти дозволяють змінювати положення, розмір, орієнтацію об'єктів. Переміщення здійснюється перетягуванням або введенням точних координат. Масштабування зберігає пропорції при затиснутій клавіші модифікатора. Обертання навколо довільної точки дозволяє створювати динамічні композиції. Всі трансформації можуть бути застосовані до окремих шарів або груп шарів.

Виділення є основним інструментом селективної роботи з елементами зображення. Прямокутне та еліптичне виділення дозволяє ізолювати прості області. Інструмент "Ласо" дає можливість створювати довільні контури виділення. "Чарівна паличка" автоматично виділяє області схожого кольору. Маски виділення зберігають складні контури для повторного використання та дозволяють створювати плавні переходи за допомогою розмиття країв.

Історія дій та відміни операцій є критично важливою функцією графічних редакторів. Багаторівнева історія дозволяє відкатитися до будь-якого попереднього стану зображення, експериментувати з різними підходами без страху втратити роботу. Знімки станів дозволяють зберігати ключові етапи роботи для швидкого порівняння варіантів.

Робочі простори та налаштування інтерфейсу дозволяють адаптувати середовище під конкретні завдання. Панелі інструментів, палітри, вікна властивостей можна групувати, закріплювати, приховувати для оптимізації робочого процесу. Профілі налаштувань дозволяють швидко перемикатися між різними режимами роботи, такими як малювання, ретушування, підготовка до друку.

Розуміння принципів роботи з шарами та допоміжними засобами є необхідним для ефективної роботи з двовимірною графікою. Ці інструменти дозволяють організувати складні проекти, забезпечити точність виконання,

прискорити процес створення та редагування зображень. Для розробників програмного забезпечення знання цих концепцій важливе при створенні графічних інтерфейсів, редакторів, систем візуалізації.

ЛЕКЦІЯ 9

РОБОТА З ТЕКСТОМ. ІНСТРУМЕНТИ МОДЕЛЮВАННЯ ТА РЕДАГУВАННЯ

Робота з текстом у комп'ютерній графіці поєднує типографіку, дизайн та технічні аспекти відображення символів. Текст є не лише носієм інформації, але й важливим візуальним елементом композиції, який впливає на сприйняття всього зображення. Сучасні графічні системи надають потужні інструменти для створення, форматування та інтеграції тексту у графічні проекти.

Шрифти є цифровими представленнями наборів символів, які визначають форму кожної літери, цифри, знака пунктуації. Векторні шрифти, такі як TrueType та OpenType, описують контури символів за допомогою кривих Без'є, що забезпечує масштабованість без втрати якості. Кожний шрифт має набір характеристик: сімейство, накреслення, кегль, міжрядковий інтервал, міжбуквений інтервал, які впливають на візуальне сприйняття тексту.

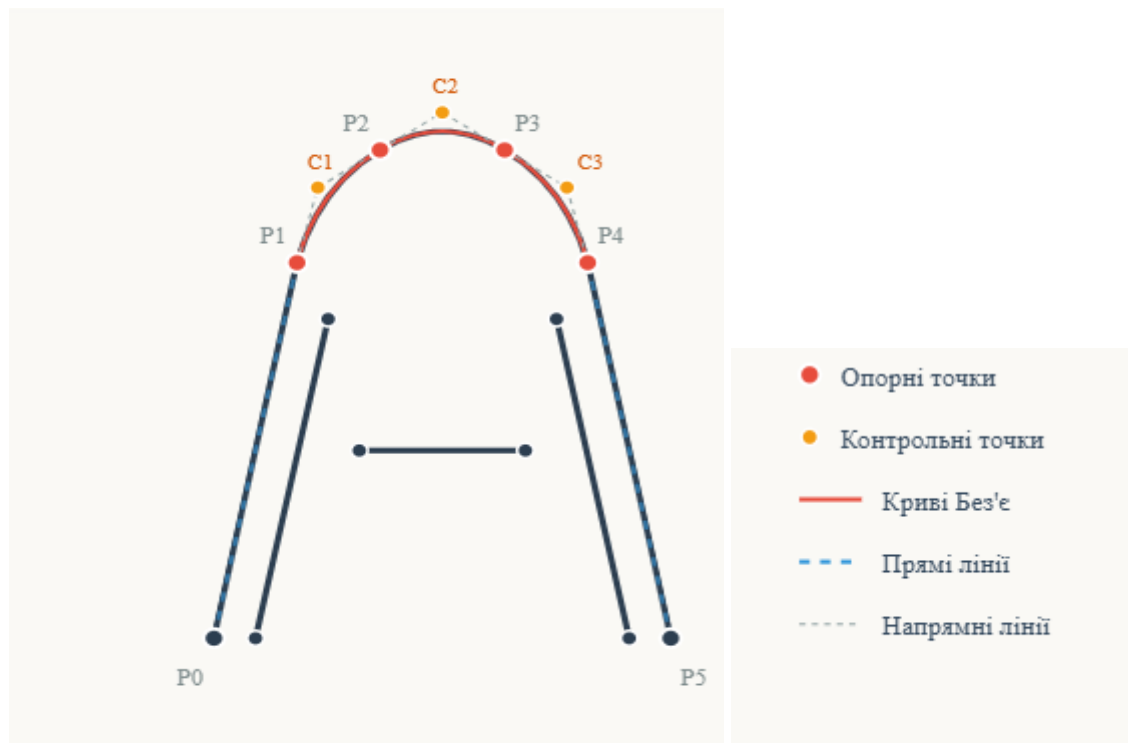


Рисунок 9.1 - Приклад векторного опису символу шрифту: контури Без'є та контрольні точки

Форматування тексту включає вибір шрифту, розміру, кольору, вирівнювання, інтервалів. Вирівнювання може бути лівим, правим, по центру, по ширині, що впливає на ритм та читабельність тексту. Кернінг – регулювання відстані між конкретними парами символів для візуально рівномірного розподілу. Трекінг – загальне збільшення або зменшення інтервалів між усіма символами рядка.

Текстові шари у графічних редакторах зберігають текст у редагованому вигляді, дозволяючи змінювати контент та форматування у будь-який момент. Растеризація тексту перетворює його у піксельне зображення, що дозволяє застосовувати растрові ефекти, але втрачає можливість редагування тексту. Векторизація тексту створює контури символів як графічні об'єкти, що корисно для логотипів та декоративних елементів.

Інструменти моделювання та редагування двовимірної графіки охоплюють широкий спектр функцій для створення та модифікації зображень. Інструмент "Перо" дозволяє створювати довільні контури шляхом розміщення опорних точок та налаштування дотичних, що є основою векторного малювання. Інструменти геометричних фігур спрощують створення прямокутників, кіл, багатокутників з точними параметрами.

Інструменти ретушування призначені для корекції растрових зображень. "Штамп" копіює текстуру з однієї ділянки в іншу для видалення дефектів. "Відновлююча пензель" автоматично підбирає текстуру та колір для безшовного заповнення. "Розмиття" та "Різкість" змінюють локальну контрастність для акцентування або пом'якшення деталей.

Фільтри та ефекти застосовують алгоритмічні перетворення до зображень. Розмиття за Гаусом усереднює значення пікселів у заданому радіусі, створюючи плавний перехід. Ефект "Тінь" додає імітацію відкидання тіні об'єктом, підвищуючи об'ємність. "Гرادієнтна заливка" створює плавний перехід між кольорами, що корисно для фонів та об'ємних ефектів.

Автоматизація процесів через скрипти та дії дозволяє прискорити рутинні операції. Запис дій фіксує послідовність команд для повторного виконання на інших зображеннях. Скриптові мови, такі як JavaScript у Adobe Photoshop або Python у GIMP, надають програмний доступ до функцій редактора для складної обробки та пакетної конвертації.

Розуміння інструментів роботи з текстом та моделювання є необхідним для створення професійних графічних проектів. Для розробників програмного забезпечення знання цих концепцій важливе при реалізації текстових рушіїв, графічних інтерфейсів, систем верстки та візуалізації.

ЛЕКЦІЯ 10

КРИВІ БЕЗ'Є ТА СПЛАЙНИ ДЛЯ СТВОРЕННЯ ДОВІЛЬНИХ ФІГУР

Криві Без'є та сплайни є математичним фундаментом векторної графіки, дозволяючи створювати гладкі, точні та масштабовані контури довільної форми. Ці інструменти базуються на параметричних рівняннях, які визначають положення точок кривої через контрольні точки, що забезпечує інтуїтивне керування формою при математичній точності.

Крива Без'є першого порядку є простою прямою лінією між двома точками. Крива другого порядку використовує одну контрольну точку, яка визначає напрямок дотичної, створюючи параболічну дугу. Крива третього порядку, найпоширеніша у графіці, має дві контрольні точки, що дозволяє створювати складніші форми з контролем кривизни на обох кінцях. Математично крива Без'є третього порядку описується рівнянням $B(t) = (1-t)^3P_0 + 3(1-t)^2tP_1 + 3(1-t)t^2P_2 + t^3P_3$, де t змінюється від 0 до 1.

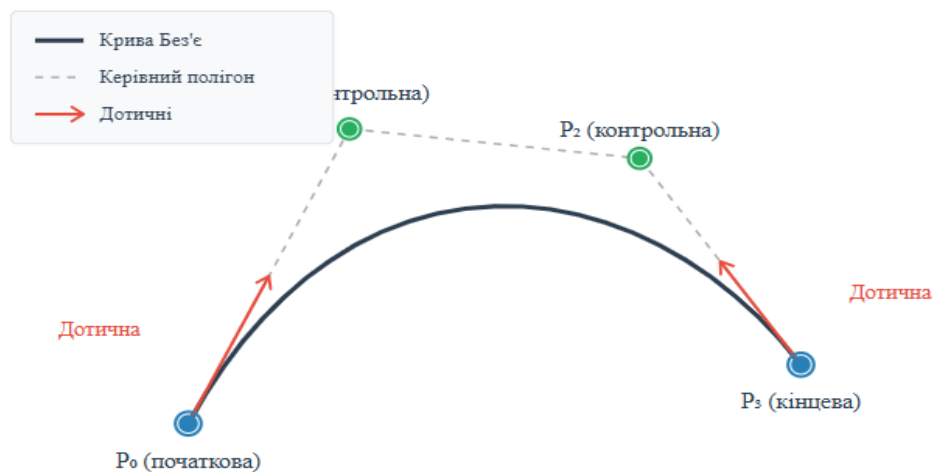


Рисунок 10. - Крива Без'є третього порядку: початкова та кінцева точки, контрольні точки та дотичні

Контрольні точки кривої Без'є не лежать на самій кривій, а визначають її форму через дотичні вектори. Переміщення контрольної точки змінює напрямок та довжину дотичної, впливаючи на кривизну відповідної ділянки. Це дозволяє інтуїтивно формувати складні контури шляхом маніпуляції обмеженою кількістю точок.

Сплайни є узагальненням кривих Без'є, що складаються з кількох сегментів, з'єднаних з певною гладкістю. Кубічний сплайн забезпечує неперервність першої та другої похідних у точках з'єднання, що створює візуально гладку криву без різких зламів. В-сплайни та NURBS (Non-Uniform Rational B-Splines) додають можливість локального контролю та точного представлення кінцевих перерізів.

Інструмент "Перо" у графічних редакторах реалізує інтерактивне створення кривих Без'є. Клік створює кутову точку з різким переходом, перетягування створює гладку точку з дотичними. Комбінування кутових та гладких точок дозволяє створювати контури будь-якої складності – від простих іконок до складних ілюстрацій.

Редагування кривих включає додавання та видалення точок, зміну типу точок, налаштування дотичних. Інструмент "Пряме виділення" дозволяє переміщувати окремі точки та контрольні ручки для тонкої корекції форми. Функції спрощення контуру зменшують кількість точок при збереженні візуальної форми, що корисно для оптимізації файлів.

Замкнені контури, створені кривими Без'є, можуть бути заповнені кольором, градієнтом або візерунком. Комбінування кількох контурів за допомогою булевих операцій дозволяє створювати складні форми з вирізами, отворами, вкладеними областями. Такий підхід є основою векторного дизайну логотипів, іконок, шрифтів.

У тривимірному моделюванні криві Без'є та сплайни використовуються для створення профілів, траєкторій, поверхонь обертання та витягування. Поверхня Без'є визначається сіткою контрольних точок у двох параметричних напрямках,

що дозволяє моделювати складні гладкі форми. NURBS поверхні є стандартом у CAD-системах для точного інженерного моделювання.

Програмування з кривими Без'є вимагає розуміння параметричних рівнянь та алгоритмів обчислення точок. Бібліотеки, такі як Cairo, Skia, Path2D у Canvas API, надають функції для створення та рендерингу кривих. Алгоритми де Кастельжо та де Бора дозволяють ефективно обчислювати точки кривих та сплайнів.

Розуміння математичних основ кривих Без'є та сплайнів є необхідним для розробників графічних застосунків, CAD-систем, анімаційних рушіїв. Ці інструменти забезпечують точність, гнучкість та ефективність представлення геометричних форм у цифровому середовищі.

ЛЕКЦІЯ 11

ОСНОВИ ТРИВИМІРНОГО МОДЕЛЮВАННЯ. БУЛЕВІ ОПЕРАЦІЇ

Тривимірне моделювання є процесом створення математичного представлення об'єктів у віртуальному просторі, який слугує основою для візуалізації, анімації, симуляції та виробництва. Сучасні системи 3D-моделювання надають різноманітні підходи та інструменти для формування геометрії, кожен з яких має свої переваги та сфери застосування.

Полігональне моделювання є найпоширенішим методом, який представляє поверхні об'єктів у вигляді сітки трикутників або чотирикутників. Кожен полігон визначається вершинами з координатами у тривимірному просторі, ребрами, що з'єднують вершини, та гранями, обмеженими ребрами. Таке представлення є простим для обробки графічним конвеєром та ефективно підтримується апаратним прискоренням.

Булеві операції є потужним інструментом комбінування простих геометричних тіл для створення складних форм. Операція об'єднання (Union) створює новий об'єкт, що включає об'єм обох вихідних тіл. Операція перетину (Intersection) залишає лише спільну частину об'єктів. Операція віднімання (Difference) видаляє з першого об'єкта частину, що перекривається другим. Ці операції базуються на теорії множин та дозволяють інтуїтивно конструювати складну геометрію.

Процес виконання булевих операцій включає визначення перетинів поверхонь, розрізання полігонів у точках перетину, класифікацію отриманих фрагментів як внутрішніх або зовнішніх, та формування нової топології. Цей алгоритмічно складний процес вимагає точних обчислень для уникнення артефактів, таких як неводонепроникні сітки або вироджені полігони.

Обведення контуру (Extrude) є фундаментальною операцією створення тривимірних об'єктів з двовимірних профілів. Двовимірний контур

переміщується вздовж заданого напрямку, утворюючи бічні грані, а початковий та кінцевий контури стають основами об'єкта. Ця операція дозволяє швидко створювати призми, циліндри, складні профільовані форми на основі векторних контурів.

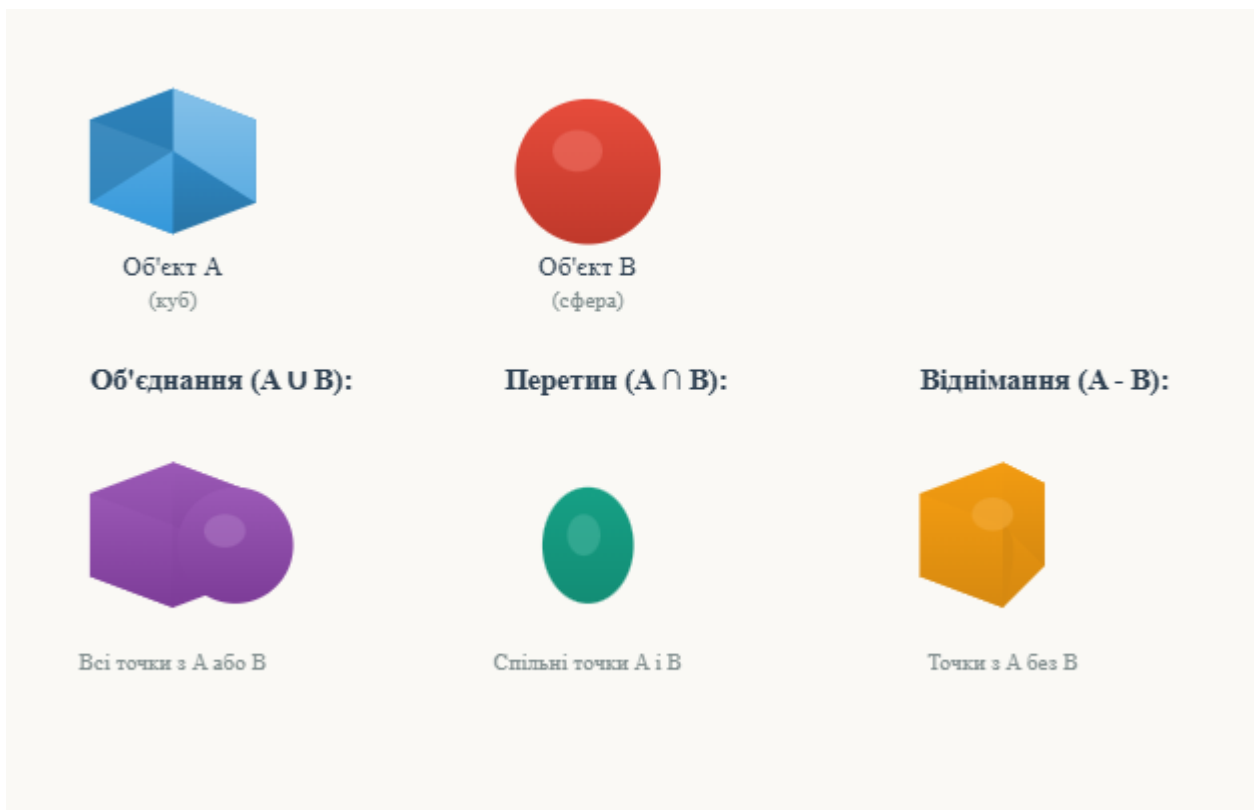


Рисунок 11.1 - Булеві операції над 3D-об'єктами: об'єднання, перетин, віднімання

Створення внутрішніх контурів дозволяє формувати отвори, порожнини, вкладені структури всередині об'єкта. Внутрішній контур при екструзії утворює порожнисту структуру, що корисно для моделювання труб, корпусів, архітектурних елементів. Комбінування зовнішніх та внутрішніх контурів за допомогою булевих операцій розширює можливості конструювання.

Поділ групи вершин по окремих мешах є важливою операцією для організації складних моделей. Меш – це незалежна сітка полігонів, яка може мати

власні матеріали, трансформації, анімації. Розбиття складного об'єкта на логічні меши полегшує редагування, оптимізацію, експорт у ігрові рушії або системи візуалізації.

Створення граней меша включає з'єднання існуючих вершин та ребер для формування нових полігонів. Інструменти "Міст", "Заповнення", "Трикутнування" автоматизують цей процес, аналізуючи геометрію та створюючи оптимальну топологію. Ручне створення граней дозволяє точно контролювати структуру сітки, що важливо для анімації та субдивізіону.

Каркасний спосіб відображення (Wireframe) показує лише ребра полігонів без заповнення граней, що дозволяє бачити внутрішню структуру моделі та топологію сітки. Цей режим корисний для редагування геометрії, перевірки щільності полігонів, виявлення артефактів. Перемикання між режимами відображення (каркас, суцільний, з текстурою) дозволяє працювати на різних рівнях деталізації.

Оптимізація полігональної сітки є критично важливою для продуктивності рендерингу, особливо у застосунках реального часу, таких як відеоігри. Зменшення кількості полігонів (ретопологія) при збереженні візуальної форми дозволяє прискорити обробку без втрати якості. Автоматичні алгоритми ретопології аналізують кривизну поверхні та розподіляють полігони відповідно до візуальної важливості ділянок.

Розуміння основ тривимірного моделювання та булевих операцій є необхідним для розробників, які працюють з 3D-графікою, САД-системами, ігровими рушіями. Ці інструменти дозволяють ефективно створювати та маніпулювати складною геометрією, забезпечуючи точність та гнучкість у процесі проектування.

ЛЕКЦІЯ 12

МАТЕРІАЛИ І ТЕКСТУРИ В 3D ГРАФІЦІ

Матеріали і текстури є ключовими компонентами тривимірної графіки, які визначають візуальні властивості поверхонь об'єктів. Матеріал описує фізичні характеристики поверхні: колір, блискучість, прозорість, рельєфність, тоді як текстура надає детальну візуальну інформацію, таку як візерунки, зображення, процедурні візерунки, які накладаються на геометрію.

Матеріали у сучасних графічних конвеєрах базуються на моделях освітлення, таких як PBR (Physically Based Rendering), які імітують реальну взаємодію світла з матеріалами. Основні параметри PBR-матеріалу включають альбедо (базовий колір), металевість (ступінь відбиття як метал), шорсткість (мікрорельєф поверхні), нормаль (деталізація рельєфу), висоту (паралакс-ефект), оклюзію (затінення у западинах).

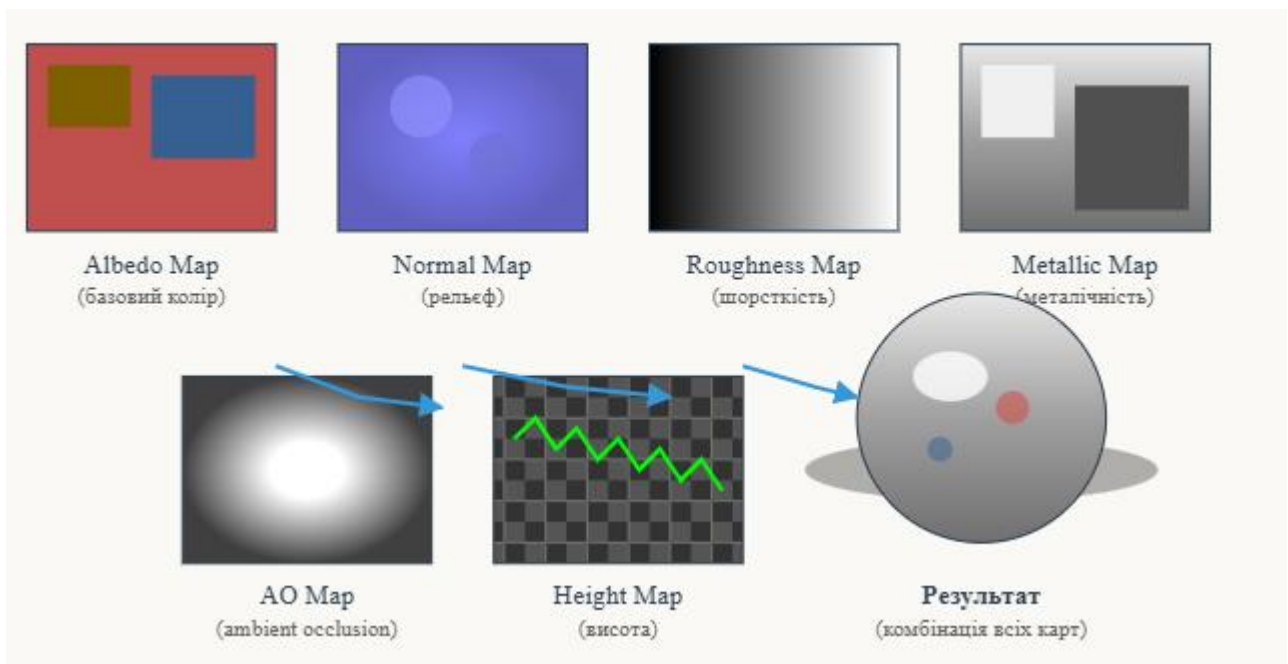


Рисунок 12.1 - PBR-матеріал: карти параметрів та їх вплив на візуальне відображення поверхні

Текстури є двовимірними зображеннями, які відображаються на тривимірних поверхнях за допомогою координат текстурування (UV-координат). Кожна вершина меша має пару координат (U, V), що визначає відповідну точку на текстурі. Правильне розгортання UV-розгортки забезпечує рівномірне нанесення текстури без спотворень та швів.

Використання зображень та відео як текстури розширює візуальні можливості 3-сцен. Статичні зображення застосовуються для дифузних карт, карт нормалей, карт шорсткості. Відеотекстури дозволяють створювати динамічні поверхні, такі як екрани моніторів, водна поверхня, вогонь. Процедурні текстури, згенеровані алгоритмічно, забезпечують нескінченну деталізацію без збільшення розміру файлу.

Накладання текстур включає кілька технік для досягнення реалізму. Трипланарне текстурування проектує текстуру з трьох напрямків, уникаючи спотворень на складних поверхнях. Tileable-текстури можуть нескінченно повторюватися без видимих швів, що корисно для великих поверхонь, таких як земля, стіни, вода. Масштабування, обертання, зміщення координат текстури дозволяють точно позиціонувати візерунок.

Карти нормалей імітують мікрорельєф поверхні без збільшення геометричної складності. Вони зберігають відхилення нормалей у RGB-каналах зображення, що змінює освітлення пікселів при рендерингу. Це дозволяє створювати враження детальної геометрії при мінімальній кількості полігонів, що критично важливо для продуктивності.

Паралакс-мапінг та рельєфне текстурування додають глибину текстурі за рахунок зміщення текстурних координат залежно від кута огляду. Це створює ілюзію об'ємної поверхні з виступами та заглибленнями. Сучасні методи, такі як віртуальне текстурування та мегатекстури, дозволяють працювати з текстурами надзвичайно високої роздільної здатності, завантажуючи лише необхідні ділянки в пам'ять.

Анімація текстур дозволяє створювати динамічні ефекти, такі як течія води, мерехтіння вогню, рух хмар. Це реалізується шляхом зміни координат текстури у часі, переключення між кадрами анімації, або використання процедурних функцій, залежних від часу. Синхронізація анімації текстур з анімацією геометрії забезпечує цілісність візуального ефекту.

Оптимізація текстур включає стиснення форматів, таких як BC/DXT, ASTC, ETC, які зменшують обсяг пам'яті та пропускну здатність без значної втрати якості. Міпмапінг – попередньо обчислені зменшені копії текстури – покращує якість при віддаленому відображенні та зменшує аліасинг. Атласи текстур об'єднують кілька маленьких текстур в одну велику, зменшуючи кількість перемикачів текстур при рендерингу.

Розуміння принципів роботи з матеріалами та текстурами є необхідним для створення реалістичної та продуктивної 3D-графіки. Для розробників програмного забезпечення знання цих концепцій важливе при реалізації графічних рушіїв, редакторів матеріалів, систем візуалізації.

ЛЕКЦІЯ 13

НАЛАШТУВАННЯ РЕНДЕРА. ВІЗУАЛІЗАЦІЯ ТРИВИМІРНИХ СЦЕН

Рендеринг є фінальним етапом створення тривимірного зображення, процесом перетворення математичного опису сцени у двовимірне зображення, придатне для відображення на екрані або друку. Налаштування рендера визначає якість, реалізм та час обробки зображення, що вимагає балансу між візуальними вимогами та обчислювальними ресурсами.

Конвеєр рендерингу складається з кількох етапів. На етапі геометрії виконуються перетворення координат, відсічення невидимих частин, проектування на екран. На етапі растеризації полігони перетворюються у фрагменти (потенційні пікселі) з інтерполяцією атрибутів. На етапі освітлення обчислюється колір кожного фрагмента з урахуванням матеріалів, текстур, джерел світла. Фінальний етап включає тест глибини, змішування, пост-обробку.

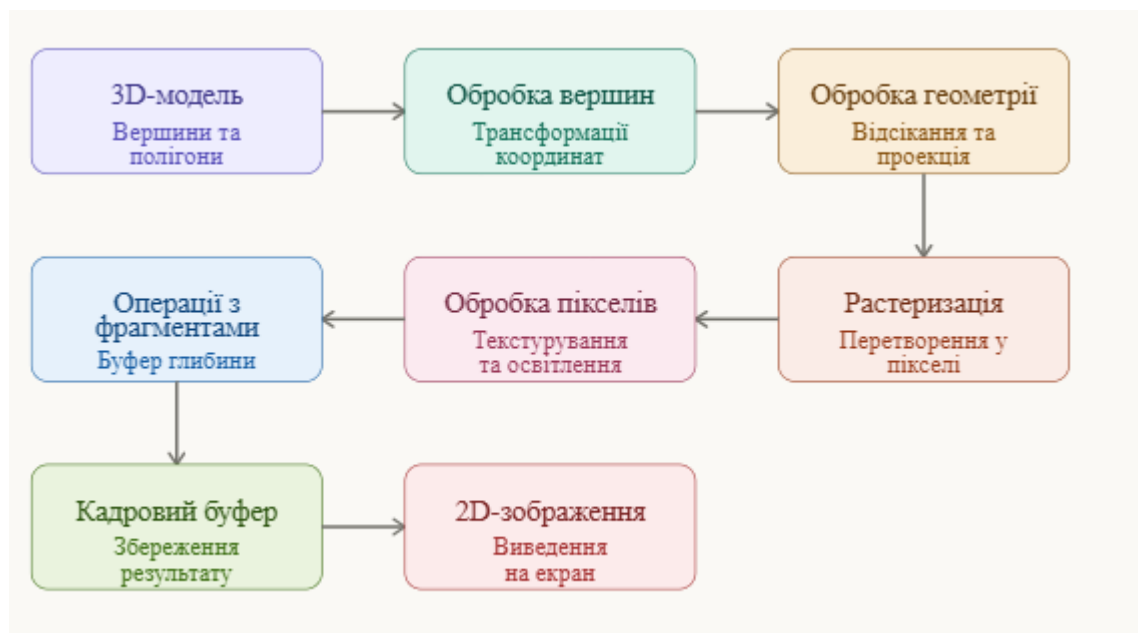


Рисунок 13.1 - Графічний конвеєр: послідовність етапів від 3D-моделі до 2D-зображення

Методи рендерингу поділяються на растеризацію та трасування променів. Растеризація є швидким методом, який проектує полігони на екран та заповнює їх пікселями. Вона домінує у застосунках реального часу, таких як відеоігри, завдяки ефективній апаратній підтримці. Трасування променів моделює фізичне поширення світла, відстежуючи шлях променів від камери через сцену до джерел світла, що забезпечує фотореалістичні результати з точними тінями, відбиттями, заломленнями.

Гібридні підходи, такі як трасування променів у реальному часі (Ray Tracing), поєднують переваги обох методів. Основна сцена рендериться растеризацією, а складні ефекти, такі як відбиття, тіні, глобальне освітлення, обчислюються трасуванням променів. Сучасні графічні процесори з апаратною підтримкою RTX дозволяють досягти інтерактивної частоти кадрів з високою якістю зображення.

Налаштування освітлення є критично важливим для реалізму сцени. Точкові джерела випромінюють світло рівномірно у всіх напрямках, прожектори мають спрямований конус, площинні джерела створюють м'які тіні. Глобальне освітлення (Global Illumination) моделює багаторазове відбиття світла між поверхнями, створюючи природне розсіяне освітлення та колірні відблиски.

Пост-обробка застосовує ефекти до вже відрендереного зображення для покращення візуальної якості. Антиаліасинг згладжує ступінчасті краї об'єктів. Глибина різкості імітує фокусування камери, розмиваючи фон або передній план. Bloom додає сяйво яскравим ділянкам. Tone mapping перетворює HDR-значення у діапазон, придатний для відображення на SDR-екранах.

Оптимізація рендерингу включає рівень деталізації (LOD), який зменшує складність об'єктів на відстані, відсічення невидимих об'єктів (Occlusion Culling), інстанціювання для багаторазового відображення однакових об'єктів. Ці техніки дозволяють підтримувати високу частоту кадрів у складних сценах без втрати візуальної якості.

Формати виведення рендеру залежать від призначення зображення. Для екранного відображення використовуються формати PNG, JPEG, WebP. Для поліграфії – TIFF, PSD з підтримкою високої глибини кольору. Для анімації – послідовності зображень або відеоформати, такі як ProRes, DNxHD. HDR-формати, такі як OpenEXR, зберігають розширений динамічний діапазон для подальшої обробки.

Розуміння процесу рендерингу та налаштувань є необхідним для створення якісної тривимірної візуалізації. Для розробників програмного забезпечення знання цих концепцій важливе при реалізації графічних рушіїв, систем візуалізації, інструментів для створення контенту. Сучасні тенденції, такі як трасування променів у реальному часі, машинне навчання для апскейлінгу, процедурна генерація, відкривають нові можливості для комп'ютерної графіки.

СПИСОК ДЖЕРЕЛ ЛІТЕРАТУРИ

1. Журавчак Л. М. Програмування комп'ютерної графіки та мультимедійні засоби: навч. посібник / Л. М. Журавчак, О. М. Левченко. Львів: Вид-во Львівської політехніки, 2019. 276 с.
2. Основи комп'ютерної графіки у Figma. Навчальний посібник / В.В. Дворжак, М.В. Талах, Ю.О. Ушенко. – Чернівці: Чернівецький нац. ун-т ім. Федьковича, 2022 р. 258 с.
3. Jason Hoppe. Adobe Illustrator: A Complete Course and Compendium of Features Paperback – Rocky Nook; Illustrated Edition (June 23, 2020). 388 p.
4. Комп'ютерна графіка: конспект лекцій для студентів усіх форм навчання спеціальностей 122 «Комп'ютерні науки» та 123 «Комп'ютерна інженерія» з курсу «Комп'ютерна графіка» / Укладач: Скиба О. П. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2019. 88 с.
5. Тмєнова Н.П. Комп'ютерна графіка: навч.-метод. посіб. / Н. П. Тмєнова. К.: ВПЦ «Київський університет», 2017. - 111 с.
6. Морозенко О.П., Малишко Г.В. Комп'ютерні методи нарисної геометрії та інженерної графіки: Конспект лекцій. Частина 3 .-Дніпро: НМетАУ, 2019. - 48с.
7. Комп'ютерна графіка: конспект лекцій для студентів усіх форм навчання спеціальностей 122 «Комп'ютерні науки» та 123 «Комп'ютерна інженерія» з курсу «Комп'ютерна графіка» / Укладач: Скиба О.П. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2019. 88 с.
8. Тмєнова Н.П. Комп'ютерна графіка. Навчально-методичний посібник. Київ: ВПЦ «Київський університет», 2017. 111 с.

9. Василюк А. С. Комп'ютерна графіка: навчальний посібник / А. С. Василюк, Н. І. Мельникова. Львів: Видавництво Львівської політехніки, 2016. 308 с.
10. Мельник О.С. Комп'ютерна анімація та 3D-моделювання: Навчальний посібник / Укладач: О.С. Мельник.- Умань: УДПУ ім. Павла Тичини, 2018. – 141 с.
11. Troy Berge. DESIGNING AND PROTOTYPING WITH FIGMA: Simplified beginner's guide to Figma web- design tool. Paperback – September 25, 2022. 134 p.
12. Mildred W. Estep. Figma (software): All you need to know about figma (software) Paperback – November 2, 2022. 71p