

УДК 004.925

**Вяткін С. І.**

*к.т.н., с.н.с., Інститут автоматики і електрометрії СО РАН*

**Романюк О. Н.**

*д.т.н., професор кафедри програмного забезпечення,  
Вінницький національний технічний університет, м. Вінниця, Україна*

**Кирилащук С. А.**

*к. пед.н., доцент кафедри вищої математики,  
Вінницький національний технічний університет, м. Вінниця, Україна*

**Лисенко Є. С.**

*студент, Вінницький національний технічний університет, м. Вінниця, Україна*

## **МЕТОД ОБ'ЄМНО-ОРІЄНТОВАНОЇ ВІЗУАЛІЗАЦІЇ РЕАЛЬНОГО ЧАСУ**

З появою систем віртуальної реальності, в яких спостерігач занурюється в світ моделі, виникає потреба візуалізації віртуального середовища, максимально наближеною до того, що людина спостерігає в природних умовах. Для цього потрібні потужні системи візуалізації. Незважаючи на те, що системи візуалізації, що використовують полігональне завдання, набули широкого поширення в силу наявності відповідних архітектур і алгоритмів, актуальним залишається питання про ефективність використання подібних систем для подання об'ємів. Пропонується метод відображення поверхонь довільних форм і об'ємів, що дозволяє досягти якості зображення, кращого порівняно з полігональним завданням з великою кількістю багатокутників.

Запропонований метод візуалізації базується на двох основних аспектах для створення систем візуалізації, що забезпечують як універсальність, так і високу якість сцен:

1) новий метод завдання поверхонь вільних форм і математичний апарат опису тривимірних об'єктів, що не вимагає для відображення поверхонь завдання їх поліномами високих ступенів або апроксимації примітивами нижчого рівня (багатокутники, патчі), а також завдання об'ємних областей масивами скалярних даних [1, 2];

2) ефективний метод растрівання з мінімальною кількістю обчислень [3-5].

Характерною особливістю запропонованого методу завдання поверхонь є те, що основними примітивами обрані поверхні другого порядку - квадрат. На цій основі будується клас вільних форм.

Реалізована можливість описувати складні геометричні об'єкти, задаючи функцію відхилення (другого порядку) від базової поверхні теж другого порядку. Композиція базової поверхні та функції відхилення, в свою чергу, може бути новою функцією збурення, тобто, похідною для іншої базової поверхні. Отримана поверхня буде гладкою, і буде потрібно невелику кількість функцій збурення для створення складних форм поверхонь. Таким чином, вільні форми можна відображати без попередньої апроксимації їх великою кількістю патчів, не маючи проблем із завданням границь, кривих напівпросторів і т.д., властивих параметричному за-

вданню патчами.

Рельєф місцевості можна задати картою (сіткою) висот і без попередньої тріангуляції його відобразити, отримуючи при цьому високу реалістичність. Проблеми, які запропонований алгоритм дозволяє вирішити, звичайним способом вирішуються набагато складніше. Дійсно, звичайний метод представлення рельєфу гранями вимагає величезної кількості граней. Крім того, відразу виникають проблеми високої глибинної складності, відкидання задніх граней, проблема пріоритетів, визначення та зміни рівнів детальності, кліпування багатокутників пірамідою видимості та ін. У запропонованому методі цих проблем немає. У геометричному процесорі обробляється всього одна площина. Правильний пріоритетний порядок забезпечується відповідним обходом дерева і системою масок. Зворотна поверхня рельєфу відкидається автоматично. Кліпування рельєфу пірамідою видимості стає непотрібним, так як в процесі растрування автоматично забезпечується вибірка тільки потрібних висот з карти висот. Для зміни рівнів детальності використовується той же механізм, що і для звичайної текстури. Крім того, реалізована можливість описувати складні геометричні об'єкти, задаючи функцію відхилення (скалярне поле - текстуру форми) від базової поверхні

Метод растрування включає розроблений алгоритм багаторівневого рейкастинга (Ray Casting), який здійснює ефективний пошук елементів об'єму - вокселів, що беруть участь у формуванні зображення [4]. Пошук проводиться в просторі всередині куба, від -1 до +1 по кожній координаті, так що центр куба відповідає початку координат. На першому кроці рекурсії вихідний об'єм розбивається на чотири підоб'єми в екранній площині. Для кожного підоб'єму виконується тест на перетин з об'єктом. Якщо перетин має місце, то підоб'єм передається наступному рівню рекурсії. Підоб'єми, що не перетинаються з об'єктом, подальшої передачі в рекурсію не підлягають, що відповідає виключенню з розгляду квадратних ділянок екрану. В процесі пошуку вокселів, що містять в собі ділянки поверхні об'єкта, що формують зображення, алгоритм здійснює обхід кубічного простору по четвертинному дереві, листя якого є корінням двійкових піддерев. У процесі обходу дерева використовується механізм багаторівневого маскування в разі наявності непрозорих об'єктів.

Розв'язана задача простого (як в кубі) ділення перспективного простору (при цьому пірамідаліне простір може бути довільної форми, що дуже важливо для компенсації дисторсії) без збільшення порядку поверхонь за рахунок трансформації простору в простір. Застосування проєктивного перетворення узагальнює наведений алгоритм на пірамідаліні об'єми, що дозволяє синтезувати зображення з перспективою. Цікавою є матриця перетворення, що діє на однорідні координати, в такий спосіб

$$: \begin{pmatrix} C11 & C12 & C13 & C14 \\ C21 & C22 & C23 & C24 \\ C31 & C32 & C33 & C34 \\ C41 & C42 & C43 & C44 \end{pmatrix} \begin{pmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{pmatrix} = \begin{pmatrix} X_p \\ Y_p \\ Z_p \\ A_p \end{pmatrix}, \text{ або } (C)(M) = (P),$$

де  $(C)$  - матриця перетворення,  $(M)$  - однорідні координати точки простору  $M$ , и  $(P)$

– відповідні по відображенню координати в  $P$ .

Запропоновані алгоритмічні рішення для обробки тривимірних даних, призначених для візуального представлення в реальному масштабі часу, лягли в основу методу об'ємно-орієнтованої візуалізації реального часу.

Незважаючи на те, що функціональне подання об'єктів відкрило багаті можливості для подальшого розвитку в моделюванні твердих тіл - це інтеграція декількох різних моделей в рамках єдиного підходу, руйнування бар'єру між твердими і деформованими тілами, універсальність в обробці статичних і динамічних об'єктів, існують проблеми, які потребують подальшому дослідженню. Для реалізації була використана Compute Unified Device Architecture (CUDA) від компанії NVIDIA. CUDA - це модель паралельного програмування разом з набором програмних засобів, яка дозволяє реалізовувати програми на мові C++ для виконання на графічному акселераторі [3, 6].

### ЛІТЕРАТУРА

1. Vyatkin S. I., Romanyuk A. N., Pavlov S.V., Kotyra A., Mussabekova A. Offsetting and Blending with Perturbation Functions // Proceedings SPIE 10808, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2018. Vol. 10808. P. 108082Y. DOI: 10.1117/12.2501694
2. Vyatkin S. I., Romanyuk A.N., Voit B.L. Perturbation functions and operations in geometric modeling // International scientific-technical magazine Measuring and Computing Devices in Technological Processes.- Khmelnytsky, Ukraine.- 2017.- № 3 (59). - P. 117-120
3. Vyatkin S. I. Using GPU for Interactive Ray Casting 3D Models Based on Perturbation Functions // Journal of Algorithms, Computer Network, and Security – V.1. – No.2. – March 2016.
4. Vyatkin S. I., Romanyuk A. N., Savitska L. A. Multi-level ray casting of function-based surfaces // Journal of Physics: Conference Series, 803, № 1.
5. Романюк О. Н. Високопродуктивні методи та засоби зафарбовування тривимірних графічних об'єктів. Монографія. / О. Н. Романюк, А. В. Чорний. - Вінниця : УНІВЕСУМ-Вінниця, 2006. — 190 с.
6. Романюк О. Н. Класифікація графічних відеоадаптерів / О. Н. Романюк, Р. Ю. Довгалюк, С. В. Олійник // Наукові праці Донецького національного технічного університету. Сер. : Інформатика, кібернетика та обчислювальна техніка. - 2011. - Вип. 14. - С. 211-215