

ISSN 1996-1588

Міністерство освіти і науки України



НАУКОВІ ПРАЦІ
ДОНЕЦЬКОГО НАЦІОНАЛЬНОГО
ТЕХНІЧНОГО УНІВЕРСИТЕТУ

Серія: Інформатика, кібернетика
та обчислювальна техніка

№2(41) 2025



Дрогобич 2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Наукові праці
Донецького національного технічного
університету

**Серія: “Інформатика, кібернетика
та обчислювальна техніка”**

Всеукраїнський науковий збірник

Заснований у травні 1996 року

Виходить 2 рази на рік

№ 2 (41) ’ 2025

Дрогобич – 2025

Публікується згідно з рішенням Вченої ради ДВНЗ «Донецький національний технічний університет» (протокол № 11 від 23.10.2025).

Збірник містить наукові статті співробітників ДонНТУ та інших навчальних і наукових закладів України, які є науковими партнерами ДонНТУ. Публікації висвітлюють результати наукових досліджень і розробок в таких напрямках, як інформатика, чисельні методи, паралельні обчислення, програмування, розробка засобів обчислювальної техніки, дослідження комп'ютерних мереж, машинна графіка і обробка зображень, математичне моделювання в різних галузях. Матеріали збірника призначені для наукових співробітників, викладачів, інженерно-технічних працівників, аспірантів та студентів.

Засновник та видавець – Донецький національний технічний університет (ДонНТУ)

РЕДАКЦІЙНА КОЛЕГІЯ:

Д-р техн. наук, проф. Є.О. Башков (головний редактор); д-р техн. наук, проф. Я.Ю. Дорогий, (заступник головного редактора); член-кореспондент НАН України, д-р техн. наук, проф. В.П. Боюн; д-р техн. наук, проф. О.А. Дмитрієва, д-р техн. наук, проф. О.О. Баркалов; д-р техн. наук, проф. О.В. Вовна; д-р техн. наук, проф. С.Д. Погорілий; д-р техн. наук, проф. О.Н. Романюк; д-р техн. наук, проф. В.А. Святний; д-р техн. наук, проф. Г.Г. Швачич; д-р техн. наук, доц. І.С. Лактіонов; канд. техн. наук, доц. І.Я. Зеленцова; канд. техн. наук, доц. Н.О. Маслова; канд. техн. наук, доц. І.А. Назарова (відп. секретар випуску), Є.О. Єжова (технічний редактор).

Адреса редакції:

Юридична адреса: пл. Шибанкова, 2, м. Покровськ, Донецька область, 85300

Фактична адреса: вул. Самбірська, 76, м. Дрогобич, Львівська обл., 82111

Адреса для листування: вул. Шевченка, 9, м. Дрогобич, Львівська обл., 82100

E-mail: yevhen.bashkov@donntu.edu.ua

Збірник зареєстровано в Державному комітеті інформаційної політики, телебачення та радіомовлення України. Свідоцтво: серія КВ, №7374 від 03.06.2003.

Збірник включено до переліку наукових фахових видань України, в яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора наук, кандидата наук та ступеня доктора філософії за спеціальностями 121 Інженерія програмного забезпечення, 122 Комп'ютерні науки, 123 Комп'ютерна інженерія (наказ Міністерства освіти і науки України №409 від 17 березня 2020 р.)

Збірник "Наукові праці ДонНТУ, серія "Інформатика, кібернетика та обчислювальна техніка" за наказом № 409 МОНУ від 17. 03.2020 отримав категорію Б.

ЗМІСТ

О.А. Дмитрієва Адаптивні блокові інтегратори з ермітовою інтерполяцією	4
Б.О. Семьонов, С.Д. Погорілий Застосування трансформерних моделей для генерації повідомлень про внесені зміни для систем контролю версій	17
А.О. Храпко, І.Я. Зеленцова, А.В. Пархоменко, Т.В. Голуб, О.М. Гладкова Алгоритм системи візуалізації для контролю рухів робота-маніпулятора	30
R. Chekhmestruk, O. Romanyuk An Algorithmic Quasi-Realtime Model for Cloth Projection onto a Parametric Body in Geodesic Coordinate Space	39
Є.О. Єжова Мультимодальна біометрична аутентифікація користувачів на основі зображення обличчя та динаміки клавіатурного почерку	52
С.М. Бабчук Вдосконалена класифікація великих мовних моделей	65
T. Starovoit Integration of Fuzzy Graph Neural Networks and Physics-Informed Models for Multi-Criteria Metaheuristic Optimization of Hydraulic Modelling Processes	73
О. А. Поздняков, А. В. Пархоменко Міграція користувацького коду на нові версії складних комп'ютерних систем з використанням методів і моделей інтелектуального реінжинірингу	86
С.В. Зікратий, В.Я. Піх, М.М. Піх, М.О. Слабінога, Ю.Й. Стрілецький, В.І. Шекета Порівняльний аналіз методів реалізації інтелектуальності на основі ПЛІС	99
І.А. Назарова, О.А. Тихонова Моделювання та аналіз паралельних обчислень на базі теорії графів	109
В.І. Манжула, Б.С. Тимчишин Семантичне моделювання гетерогенних джерел шкідливих викидів в атмосферу для проактивного управління екосистемою міста	118
А.О. Попова, Д.Ю. Хома Інтеграція механізмів уваги до глибоких нейронних мереж стегоаналізу цифрових зображень	128
Н.О. Маслова, Ю.І. Шоломинський Інтеграція комп'ютерної логіки у DevSecOps для перевірки безпеки	140
Є. К. Завальнюк, О. Н. Романюк Сферично-кутова інтерполяція векторів на основі квадратичних поліномів	150

УДК 004.92

¹Є.К. Завальнюк, аспірант²О.Н. Романюк, д-р техн. наук, проф.^{1,2}Вінницький національний технічний університет, м. Вінниця, Україна¹qq9272627@gmail.com²rom8591@gmail.com

Сферично-кутова інтерполяція векторів на основі квадратичних поліномів

У статті запропоновано нові вирази для підвищення продуктивності розрахунку проміжних векторів рядка растеризації на основі паралельної сферично-кутової інтерполяції. Проаналізовано особливості використання векторів нормалей, падіння світла, відбиття світла для формування високореалістичних тривимірних зображень. Обґрунтовано необхідність високопродуктивного обчислення векторів для більш швидкого формування зображень. Проаналізовано недоліки існуючих методів лінійної інтерполяції векторів, сферично-лінійної інтерполяції векторів, сферично-кутової інтерполяції векторів. Розглянуто переваги методу ітеративної сферично-кутової інтерполяції векторів, а також перспективи паралельного застосування даного методу. Досліджено зростання обчислювальної складності коефіцієнтів виразу паралельної сферично-кутової інтерполяції векторів у кожній подальшій точці. Обґрунтовано доцільність розробки спрощених коефіцієнтних виразів для підвищення продуктивності паралельної сферично-кутової інтерполяції векторів. Розроблено спрощені коефіцієнтні квадратичні поліноми шляхом застосування поліноміальної регресії. Порівняно продуктивність застосування розроблених виразів і оригінальних повних виразів. Залежно від номера точки рядка растеризації, запропоновано комбіноване використання даних виразів. Додатково, порівняно продуктивність комбінованого застосування повних і спрощених виразів для паралельної інтерполяції векторів та традиційних непаралельних методів інтерполяції векторів. Обчислено виграш у продуктивності від використання нових виразів. Проаналізовано відхилення довжин векторів, обчислених на основі спрощених квадратичних коефіцієнтів, від одиничної довжини. Наведено візуалізовані приклади розрахованих проміжних векторів рядка растеризації. У програмному засобі 3D-рендерингу Idx3d сформовано зображення тестової фігури на основі використання запропонованих виразів. Запропоновані коефіцієнтні квадратичні вирази забезпечують високопродуктивне формування реалістичних зображень у системах рендерингу, що функціонують у режимі реального часу.

Ключові слова: рендеринг, комп'ютерна графіка, нормалізація векторів, сферично-кутова інтерполяція векторів, розпаралелення

DOI: 10.31474/1996-1588-2025-2-41-150-158

Вступ

Під час рендерингу тривимірних зображень важливим є як і детальне відтворення поверхонь об'єктів сцени, так і висока продуктивність здійснення обчислень. Якість відтворення особливостей поверхонь об'єктів визначається способом подання їх геометрії, а також використаною моделлю відбиття світла. Локальні геометричні особливості у точці поверхні об'єкта задаються вектором нормалі до неї, для моделювання відбиття світла від точки поверхні застосовуються вектори до джерела світла та до камери (спостерігача). Для високореалістичного формування зображень, як правило, дані вектори повинні бути визначені у кожній точці поверхні об'єкта та мають бути нормалізовані, що зумовлює значні обчислювальні затрати на етапі

зафарбовування.

Для підвищення продуктивності 3D-рендерингу застосовуються високопродуктивні вирази й методи розрахунку векторів. Зокрема, використовуються методи на основі розпаралелення обчислення векторів. Однак, дані методи мають обмежене застосування через зростання обчислювальної складності у кожній наступній точці рядка растеризації. Тому необхідною є розробка нових високопродуктивних виразів і методів обчислення векторів рядків растеризації.

Аналіз літератури

Інтенсивність кольору пікселя I може бути розрахована з урахуванням внеску розсіяного освітлення та відбитого від поверхні світла.

Причому, обчислюються складова відбиття світла від поверхні у різних напрямках (дифузна складова) і складова відбиття світла у сконцентрованому напрямку (спекулярна складова). Як правило, використовується вираз [1-3]:

$$I = k_a \cdot I_a + k_d \cdot (\vec{L} \cdot \vec{N}) \cdot I_s + k_s \cdot f_r(\vec{L}, \vec{V}) \cdot I_s,$$

де k_a – коефіцієнт розсіяного освітлення, I_a – інтенсивність розсіяного освітлення, k_d – коефіцієнт дифузного відбиття світла, k_s – коефіцієнт спекулярного відбиття світла, $\vec{N}$ – вектор нормалі до точки поверхні, $\vec{L}$ – вектор падіння світла до точки поверхні, $\vec{V}$ – вектор відбиття світла від точки поверхні, I_s – інтенсивність джерела світла, f_r – двопроменева функція відбивної здатності (ДФВЗ) [4-7], що визначає відбивні властивості матеріалу.

Точність обчислення дифузної та спекулярної складових відбиття світла, а також продуктивність зафарбовування поверхні об'єкта, суттєво залежать від особливостей обчислення векторів $\vec{N}$, $\vec{L}$, $\vec{V}$. Зокрема, поширений метод зафарбовування Фонга [8,9] передбачає інтерполяцію векторів між вершинами трикутника у кожній його точці, далі вектори повинні бути нормалізовані, тобто, приведеними до одиничної довжини. У випадку спрощеного розрахунку даних векторів, втрати у якості візуалізації сцени повинні бути мінімальні.

Відомим є метод лінійної інтерполяції векторів (LERP), що полягає у використанні виразу [10,11]:

$$\vec{N}(k) = (1 - k) \cdot \vec{N}_A + k \cdot \vec{N}_B,$$

де $\vec{N}_A, \vec{N}_B$ – крайні вектори рядка растеризації, $k \in [0,1]$.

Вираз є простим, однак подальша нормалізація розрахованих векторів є трудомісткою, оскільки включає операцію знаходження оберненого квадратного кореня [12]:

$$\frac{\vec{N}(k)}{|\vec{N}(k)|} = \frac{\vec{N}(k)}{\sqrt{\vec{N}(k)_x \vec{N}(k)_x + \vec{N}(k)_y \vec{N}(k)_y + \vec{N}(k)_z \vec{N}(k)_z}}.$$

Авторами було запропоновано ряд виразів для пришвидшеного обчислення оберненого квадратного кореня під час нормалізації векторів. Наприклад, може бути застосований спрощений вираз [13], отриманий розкладом оригінального виразу у ряд Тейлора:

$$\frac{1}{\sqrt{z}} \approx \begin{cases} 1.291 - 1.076(z - 0.6) + 1.345(z - 0.6)^2, & z \leq 0.68, \\ 1 + ((1 - z) + (1 - z)^2)/2, & z > 0.68, \end{cases}$$

де $z = \vec{N} \cdot \vec{N}$.

На відміну від LERP, метод сферично-

лінійної інтерполяції (SLERP) [14-17] забезпечує прямий розрахунок проміжних одиничних векторів на основі виразу

$$\vec{N}(k) = \frac{\sin((1-k)\psi)}{\sin(\psi)} \cdot \vec{N}_A + \frac{\sin(k\psi)}{\sin(\psi)} \cdot \vec{N}_B,$$

де ψ – кут між $\vec{N}_A, \vec{N}_B$.

Недоліком SLERP є здійснення 2 трудомістких операцій ділення. Основний вираз методу сферично-кутової інтерполяції [18] векторів передбачає здійснення лише операцій додавання та множення

$$\vec{N}(t) = \cos(t \cdot \phi) \cdot \vec{N}_A + \sin(t \cdot \phi) \cdot \vec{N}_K,$$

де ϕ – кут між двома сусідніми векторами у рядку растеризації, $\vec{N}_K = \frac{\vec{N}_B - \vec{N}_A(\vec{N}_B \cdot \vec{N}_A)}{\sqrt{1 - (\vec{N}_B \cdot \vec{N}_A)^2}}$, t – номер точки у рядку растеризації.

Недоліком виразу сферично-кутової інтерполяції векторів є необхідність визначення кута $t \cdot \phi$. Більшу продуктивність визначення проміжних векторів забезпечує ітеративна форма сферично-кутової інтерполяції [18], що полягає у розрахунку вектора на основі двох попередніх:

$$\vec{N}(t + 1) = 2\vec{N}(t) \cdot \cos(\phi) - \vec{N}(t - 1).$$

Перспективним є застосування даного виразу для паралельного розрахунку проміжних векторів вздовж рядка растеризації. Тоді, вектор обчислюється у кожній новій точці рядка на основі перших двох векторів. Зокрема, відомими є вирази для паралельного розрахунку векторів у точках $\vec{N}(t + 1), \vec{N}(t + 2), \vec{N}(t + 3), \vec{N}(t + 4)$. Наприклад, вираз обчислення $\vec{N}(t + 2)$ має вид [19]:

$$\vec{N}(t + 2) = \vec{N}(t) \cdot (2 \cos(2\phi) + 1) + \vec{N}(t - 1) \cdot (-2 \cos(\phi)).$$

Водночас, середній трикутник полігональної моделі може містити до 100 точок [20], а рядок растеризації середнього трикутника вміщує не більше 16 точок [20]. При паралельному обчисленні векторів на основі перших двох векторів рядка растеризації обчислювальна складність коефіцієнтів біля $\vec{N}(t), \vec{N}(t - 1)$ значно зростає у кожній новій точці. Тому актуальною є розробка високопродуктивних виразів паралельного обчислення проміжних векторів вздовж рядка растеризації.

Метою роботи є підвищення продуктивності паралельного обчислення векторів вздовж рядка растеризації за рахунок використання спрощених поліноміальних виразів.

Спрощені квадратичні коефіцієнти для розпаралеленої сферично-кутової інтерполяції векторів

Вираз паралельного обчислення нормалізованих векторів на основі перших двох векторів рядка растеризації у виді:

$$\vec{N}(t) = \vec{N}(2) \cdot coef1(t) + \vec{N}(1) \cdot coef2(t),$$

де $t \in [3, 15]$ – номер точки у рядку rasterизації, $coef1(t), coef2(t)$ – поліноміальні коефіцієнти $\bar{N}(2)$ і $\bar{N}(1)$ відповідно, вирази яких залежать від t . У таблиці 1 наведено повні вирази поліноміальних коефіцієнтів $coef1(t), coef2(t)$ залежно від номера точки t . Вирази сформовані на основі ітеративної сферично-кутової інтерполяції векторів.

Таблиця 1 – Повні вирази коефіцієнтів $coef1(t), coef2(t)$ залежно від t

t	Вирази $coef1(t)/coef2(t)$
3	$2 \cdot \cos(\varphi);$
4	-1 $4 \cdot \cos(\varphi)^2 - 1;$
5	$-2 \cdot \cos(\varphi)$ $8 \cdot \cos(\varphi)^3 - 4 \cdot \cos(\varphi);$
6	$1 - 4 \cos(\varphi)^2$ $16 \cos(\varphi)^4 - 12 \cos(\varphi)^2 + 1;$
7	$4 \cos(\varphi) - 8 \cos(\varphi)^3$ $32 \cos(\varphi)^5 - 32 \cos(\varphi)^3 + 6 \cos(\varphi);$
8	$-1 + 12 \cos(\varphi)^2 - 16 \cos(\varphi)^4$ $64 \cos(\varphi)^6 - 80 \cos(\varphi)^4 + 24 \cos(\varphi)^2 - 1;$
9	$-6 \cos(\varphi) + 32 \cos(\varphi)^3 - 32 \cos(\varphi)^5$ $128 \cos(\varphi)^7 - 192 \cos(\varphi)^5 + 80 \cos(\varphi)^3 -$
10	$-8 \cos(\varphi);$ $1 - 24 \cos(\varphi)^2 + 80 \cos(\varphi)^4 - 64 \cos(\varphi)^6$ $256 \cos(\varphi)^8 - 448 \cos(\varphi)^6 + 240 \cos(\varphi)^4 -$
11	$-40 \cos(\varphi)^2 + 1;$ $8 \cos(\varphi) - 80 \cos(\varphi)^3 + 192 \cos(\varphi)^5 - 128 \cos(\varphi)^7$ $512 \cos(\varphi)^9 - 1024 \cos(\varphi)^7 + 672 \cos(\varphi)^5 -$
12	$-160 \cos(\varphi)^3 + 10 \cos(\varphi);$ $-1 + 40 \cos(\varphi)^2 - 240 \cos(\varphi)^4 + 448 \cos(\varphi)^6 -$
13	$-256 \cos(\varphi)^8$ $1024 \cos(\varphi)^{10} - 2304 \cos(\varphi)^8 + 1792 \cos(\varphi)^6 -$
14	$-560 \cos(\varphi)^4 + 60 \cos(\varphi)^2 - 1;$ $-10 \cos(\varphi) + 160 \cos(\varphi)^3 - 672 \cos(\varphi)^5 +$
15	$+1024 \cos(\varphi)^7 - 512 \cos(\varphi)^9$ $2048 \cos(\varphi)^{11} - 5120 \cos(\varphi)^9 + 4608 \cos(\varphi)^7 -$
	$-1792 \cos(\varphi)^5 + 280 \cos(\varphi)^3 - 12 \cos(\varphi);$ $1 - 60 \cos(\varphi)^2 + 560 \cos(\varphi)^4 - 1792 \cos(\varphi)^6 +$
	$+2304 \cos(\varphi)^8 - 1024 \cos(\varphi)^{10}$ $4096 \cos(\varphi)^{12} - 11264 \cos(\varphi)^{10} + 11520 \cos(\varphi)^8 -$
	$-5376 \cos(\varphi)^6 + 1120 \cos(\varphi)^4 - 84 \cos(\varphi)^2 + 1;$ $12 \cos(\varphi) - 280 \cos(\varphi)^3 + 1792 \cos(\varphi)^5 -$
	$-4608 \cos(\varphi)^7 + 5120 \cos(\varphi)^9 - 2048 \cos(\varphi)^{11}$

Продовження таблиці 1

15	$8192 \cos(\varphi)^{13} - 24576 \cos(\varphi)^{11} + 28160 \cos(\varphi)^9 -$ $-15360 \cos(\varphi)^7 + 4032 \cos(\varphi)^5 - 448 \cos(\varphi)^3 +$ $+14 \cos(\varphi);$ $-1 + 84 \cos(\varphi)^2 - 1120 \cos(\varphi)^4 + 5376 \cos(\varphi)^6 -$ $-11520 \cos(\varphi)^8 + 11264 \cos(\varphi)^{10} - 4096 \cos(\varphi)^{12}$
----	---

Як показано у таблиці 1, зі збільшенням номера точки t обчислювальна складність поліномів суттєво зростає. Зазначимо, що у кожній новій точці рядка rasterизації вирази $coef1(t), coef2(t)$ оновлюються згідно з формулами

$$coef1(t+1) = coef1(t) \cdot \cos(\varphi) + \cos((t+1)\varphi);$$

$$coef2(t+1) = \frac{coef2(t)}{\cos(\varphi)} + \frac{\cos((t+1)\varphi)}{-\cos(\varphi)}.$$

Вирази $coef1(t), coef2(t)$ можна дещо спростити, якщо використати відношення $\cos(2\varphi) = 2 \cos(\varphi)^2 - 1$. Позначимо дані коефіцієнти на основі $\cos(2\varphi)$ як $coef2_1(t), coef2_2(t)$. У таблиці 2 наведено вирази $coef2_1(t), coef2_2(t)$ залежно від номера t .

Таблиця 2 – Вирази коефіцієнтів $coef1(t), coef2(t)$ відносно t

t	Вирази $coef1(t)/coef2(t)$
3	$2 \cdot \cos(\varphi);$
4	-1 $2 \cos(2\varphi) + 1;$
5	$-2 \cdot \cos(\varphi)$ $\cos(2\varphi) \cdot 4 \cos(\varphi);$
6	$-2 \cos(2\varphi) - 1$ $4 \cos(2\varphi)^2 + 2 \cos(2\varphi) - 1;$
7	$-4 \cos(\varphi) \cos(2\varphi)$ $2 \cos(\varphi)(4 \cos(2\varphi)^2 - 1);$
8	$-4 \cos(2\varphi)^2 - 2 \cos(2\varphi) + 1$ $8 \cos(2\varphi)^3 + 4 \cos(2\varphi)^2 - 4 \cos(2\varphi) - 1;$
9	$2 \cos(\varphi)(1 - 4 \cos(2\varphi)^2)$ $8 \cos(\varphi)(2 \cos(2\varphi)^3 - \cos(2\varphi));$
10	$-8 \cos(2\varphi)^3 - 4 \cos(2\varphi)^2 + 4 \cos(2\varphi) + 1$ $16 \cos(2\varphi)^4 + 8 \cos(2\varphi)^3 - 12 \cos(2\varphi)^2 -$
11	$-4 \cos(2\varphi) + 1;$ $8 \cos(\varphi)(\cos(2\varphi) - 2 \cos(2\varphi)^3)$
	$2 \cos(\varphi)(16 \cos(2\varphi)^4 - 12 \cos(2\varphi)^2 + 1);$ $-16 \cos(2\varphi)^4 - 8 \cos(2\varphi)^3 + 12 \cos(2\varphi)^2 +$ $+4 \cos(2\varphi) - 1$

Продовження таблиці 2

12	$32 \cos(2\varphi)^5 + 16 \cos(2\varphi)^4 - 32 \cos(2\varphi)^3 -$ $-12 \cos(2\varphi)^2 + 6 \cos(2\varphi) + 1;$ $-2 \cos(\varphi)(16 \cos(2\varphi)^4 - 12 \cos(2\varphi)^2 + 1)$
13	$4 \cos(\varphi)(16 \cos(2\varphi)^5 - 16 \cos(2\varphi)^3 + 3 \cos(2\varphi));$ $-32 \cos(2\varphi)^5 - 16 \cos(2\varphi)^4 + 32 \cos(2\varphi)^3 +$ $+12 \cos(2\varphi)^2 - 6 \cos(2\varphi) - 1$
14	$64 \cos(2\varphi)^6 + 32 \cos(2\varphi)^5 - 80 \cos(2\varphi)^4 -$ $-32 \cos(2\varphi)^3 + 24 \cos(2\varphi)^2 + 6 \cos(2\varphi) - 1;$ $-4 \cos(\varphi) \left(16 \cos(2\varphi)^5 - 16 \cos(2\varphi)^3 + \right.$ $\left. + 3 \cos(2\varphi) \right)$
15	$2 \cos(\varphi)(64 \cos(2\varphi)^6 - 80 \cos(2\varphi)^4 + 24 \cos(2\varphi)^2 -$ $-1);$ $-64 \cos(2\varphi)^6 - 32 \cos(2\varphi)^5 + 80 \cos(2\varphi)^4 +$ $+32 \cos(2\varphi)^3 - 24 \cos(2\varphi)^2 - 6 \cos(2\varphi) + 1$

Як показано у таблиці 2, вирази $coef2_1(t), coef2_2(t)$ є більш простими, ніж $coef1(t), coef2(t)$. Однак, $coef2_1(t), coef2_2(t)$ аналогічно характеризуються суттєвим зростанням обчислювальної складності при збільшенні t . Тому, знайдемо наближені квадратичні вирази розрахунку $coef1(t), coef2(t)$.

Розраховуємо значення $coef1(t), coef2(t)$ для точок $t \in [5, 15]$ (для $t \in [3, 4]$ степінь $coef1(t), coef2(t)$ уже не перевищує 2) відносно значень $\cos(\varphi)$ 0.5 – 0.999.

Зазначені значення відповідають кутам між двома сусідніми векторами $\frac{\pi}{3} - \frac{\pi}{64}$. Причому, враховуємо, що для n точок у рядку растеризації максимальний кут між двома сусідніми векторами становить $\frac{\pi}{n-1}$.

На рис. 1 наведено приклад розрахованих значень $coef1(t)$ відносно $t \in [5, 15]$, $\cos(\varphi) \in [0.5, 0.999]$ у MS Excel.

Розраховані значення $coef1(t), coef2(t)$ було проаналізовано з використанням поліноміальної регресії. У результаті, відносно кожного номера точки рядка растеризації $t \in [5, 15]$ отримано спрощені квадратичні вирази $coef1(t), coef2(t)$.

Позначимо отримані коефіцієнти як $s_coef1(t), s_coef2(t)$. У таблиці 3 наведено вирази $s_coef1(t), s_coef2(t)$ відносно $t \in [3, 15]$. Для $t \in [3, 4]$ залишено оригінальні вирази. Оскільки коефіцієнти $s_coef1(t), s_coef2(t)$ розраховуються лише на основі косинуса кута між сусідніми векторами, забезпечується можливість їх паралельного обчислення у 13-и точках рядка. Окремо визначається лише другий вектор рядка.

 $\cos(\varphi)$

0,5					
0,707	0				
0,809	1	0			
0,866	1,732	1	0		
0,901	2,247	1,802	1		
0,924	2,613	2,414	1,848	1	0
0,94	2,879	2,879	2,532	1,879	1
0,951	3,078	3,236	3,078	2,618	1,902
0,959	3,229	3,513	3,513	3,229	2,683
0,966	3,346	3,732	3,864	3,732	3,346
0,971	3,439	3,907	4,148	4,148	3,907
0,975	3,514	4,049	4,381	4,494	4,381
0,977	3,545	4,11	4,482	4,645	4,591
0,978	3,574	4,165	4,574	4,783	4,783
0,981					5,126
0,983					5,419
0,985					5,671
0,991					6,635
0,997					
0,999					

t 5 6 7 8 9

 $\cos(\varphi)$

0,5					
0,707					
0,809					
0,866					
0,901					
0,924					
0,94	0				
0,951	1	0			
0,959	1,919	1	0		
0,966	2,732	1,932	1	0	
0,971	3,439	2,771	1,942	1	0
0,975	4,049	3,514	2,802	1,95	1
0,977	4,321	3,85	3,199	2,398	1,485
0,978	4,574	4,165	3,574	2,827	1,956
0,981	5,027	4,736	4,262	3,625	2,848
0,983	5,419	5,234	4,872	4,343	3,666
0,985	5,759	5,671	5,411	4,987	4,411
0,991	7,078	7,4	7,596	7,661	7,596
0,997				9,809	10,462
0,999				11,231	12,025

t 10 11 12 13 14 15

Рисунок 1 – Розраховані значення $coef1(t)$ відносно $t \in [5, 15]$, $\cos(\varphi) \in [0.5, 0.999]$ Таблиця 3 – Вирази коефіцієнтів $s_coef1(t), s_coef2(t)$ залежно від t

t	Вирази $s_coef1(t), s_coef2(t)$
3	$2 \cdot \cos(\varphi);$ -1
4	$4 \cdot \cos(\varphi)^2 - 1;$ $-2 \cdot \cos(\varphi)$
5	$\cos(\varphi)(20.474 \cdot \cos(\varphi) - 21.339) + 4.8551;$ $\cos(\varphi)(-4.0508 \cdot \cos(\varphi) + 0.087) + 0.9631$
6	$\cos(\varphi)(65.763 \cdot \cos(\varphi) - 92.964) + 32.173;$ $\cos(\varphi)(-21.804 \cdot \cos(\varphi) + 23.751) - 5.9458$
7	$\cos(\varphi)(165.44 \cdot \cos(\varphi) - 264.4) + 104.91;$ $\cos(\varphi)(-71.106 \cdot \cos(\varphi) + 102.91) - 36.798$
8	$\cos(\varphi)(352.4 \cdot \cos(\varphi) - 600.31) + 254.81;$ $\cos(\varphi)(-176.83 \cdot \cos(\varphi) + 285.95) - 115.09$
9	$\cos(\varphi)(739.03 \cdot \cos(\varphi) - 1317.3) + 586.22;$ $\cos(\varphi)(-404 \cdot \cos(\varphi) + 698.47) - 301.47$
10	$\cos(\varphi)(1222.7 \cdot \cos(\varphi) - 2222.7) + 1008.9;$ $\cos(\varphi)(-739.03 \cdot \cos(\varphi) + 1317.3) - 586.22$
11	$\cos(\varphi)(2261.4 \cdot \cos(\varphi) - 4209.8) + 1958.4;$ $\cos(\varphi)(-1421.8 \cdot \cos(\varphi) + 2611.6) - 1198.8$

Продовження таблиці 3

12	$\cos(\varphi)(3637 \cdot \cos(\varphi) - 6858.1) + 3232;$ $\cos(\varphi)(-2418.2 \cdot \cos(\varphi) + 4518.1) - 2109.8$
13	$\cos(\varphi)(5158.2 \cdot \cos(\varphi) - 9791.5) + 4645.2;$ $\cos(\varphi)(-3548.3 \cdot \cos(\varphi) + 6681.7) - 3144.3$
14	$\cos(\varphi)(7718.6 \cdot \cos(\varphi) - 14773) + 7067.3;$ $\cos(\varphi)(-5392.2 \cdot \cos(\varphi) + 10254) - 4873.7$
15	$\cos(\varphi)(10658 \cdot \cos(\varphi) - 20501) + 9856.6;$ $\cos(\varphi)(-7601.1 \cdot \cos(\varphi) + 14540) - 6951.5$

Проаналізуємо виграш у продуктивності від застосування $s_coef1(t), s_coef2(t)$. Порівняємо продуктивність паралельного розрахунку векторів вздовж рядків растеризації з різною кількістю точок на основі $coef1(t), coef2(t), coef2_1(t), coef2_2(t), s_coef1(t), s_coef2(t)$. Порівняння продуктивності здійснено шляхом розрахунку кількості тактів процесора AMD Zen 4 (2022) [21], необхідних для обчислення проміжних векторів рядка растеризації. Результати порівняння наведено у таблиці 4. При розрахунку враховано використання зсувів замість множення, повторне використання множників. Другий вектор обчислюється на основі методу SLERP.

Таблиця 4 – Сумарні кількості тактів для обчислення проміжних векторів рядка растеризації на основі паралельного використання $coef1(t), coef2(t), coef2_1(t), coef2_2(t), s_coef1(t), s_coef2(t)$

Всього точок	Кількість тактів при паралельному використанні		
	$coef1(t), coef2(t)$	$coef2_1(t), coef2_2(t)$	$s_coef1(t), s_coef2(t)$
4	82	82	82
5	95	96	95
6	99	99	
7	104	103	
8	109	104	
9	114	108	
10	119	108	
11	124	113	107
12	130	114	
13	135	119	
14	141	120	
15	152	125	
16	157	125	

Отже, використання наближених квадратичних виразів $s_coef1(t), s_coef2(t)$ забезпечує виграш у продуктивності до 17% порівняно з використанням оригінальних повних виразів, якщо у рядку растеризації (8,16] точок.

Для меншої кількості точок доцільно застосовувати $coef1(t), coef2(t)$,

$coef2_1(t), coef2_2(t)$, коефіцієнти яких є числами степеня 2.

Тоді, обґрунтованим є комбіноване використання виразів $coef1(t), coef2(t), coef2_1(t), coef2_2(t), s_coef1(t), s_coef2(t)$. Позначимо комбіновані набори виразів як $comb_coef1(t), comb_coef2(t)$:

$$\begin{aligned} comb_coef1(3) &= coef1(3), \\ comb_coef1(4) &= coef1(4), \\ comb_coef1(5) &= coef1(5), \\ comb_coef1(6) &= coef2_1(6), \\ comb_coef1(7) &= coef2_1(7), \\ comb_coef1(8) &= s_coef1(8), \\ comb_coef1(9) &= s_coef1(9), \\ comb_coef1(10) &= s_coef1(10), \\ comb_coef1(11) &= s_coef1(11), \\ comb_coef1(12) &= s_coef1(12), \\ comb_coef1(13) &= s_coef1(13), \\ comb_coef1(14) &= s_coef1(14), \\ comb_coef1(15) &= s_coef1(15). \end{aligned}$$

Аналогічно визначаються вирази $comb_coef2(t)$.

Далі порівняємо продуктивність розрахунку проміжних векторів рядка растеризації на основі паралельного використання виразів $comb_coef1(t), comb_coef2(t)$ і на основі традиційних методів LERP (з подальшою нормалізацією), SLERP, сферично-кутової інтерполяції, ітеративної сферично-кутової інтерполяції (таблиця 5).

Таблиця 5 – Сумарні кількості тактів для обчислення проміжних векторів на основі методів LERP, SLERP, видів сферично-кутової інтерполяції залежно від кількості точок рядка растеризації

Всього точок	Кількість тактів при використанні			
	LERP	SLERP	Сферично-кутова інтерполяція	
			Звичайна	На основі ітерацій
4	116	108	86	82
5	169	157	111	94
6	222	206	136	106
7	275	255	161	118
8	328	304	186	130
9	381	353	211	142
10	434	402	236	154
11	487	451	261	166
12	540	500	286	178
13	593	549	311	190
14	646	598	336	202
15	699	647	361	214
16	752	696	386	226

На рис. 2 наведено виграш у продуктивності від паралельного застосування коефіцієнтів $comb_coef1(t), comb_coef2(t)$ порівняно з

використанням ітеративної сферично-кутової нормалізації, що є найбільш продуктивним методом із розглянутих у таблиці 5.

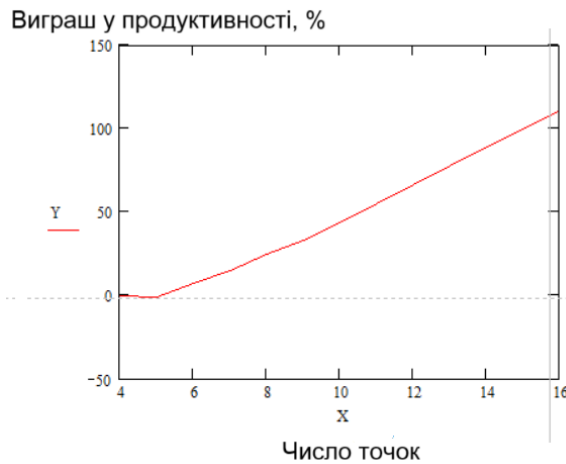


Рисунок 2 – Виграш у продуктивності від паралельного застосування $comb_coef1(t), comb_coef2(t)$ порівняно з використанням ітеративної сферично-кутової інтерполяції

Отже, відносно кількостей точок у рядку rasterизації [6,16] виграш у продуктивності порівняно зі застосуванням ітеративної сферично-кутової інтерполяції становить від 7 до 111%. Виграш у продуктивності відносно кількостей точок [4,16] порівняно з використанням LERP становить від 41 до 603%.

Далі, проаналізуємо відхилення векторів, розрахованих на основі наближених квадратичних виразів, від одиничної довжини.

Відхилення векторів від одиничної довжини знайдемо з виразом

$$1 - \sqrt{(\vec{N}(2) \cdot s_coef1(t) + \vec{N}(1) \cdot s_coef2(t))^2} =$$

$$= 1 - \sqrt{s_coef1(t)^2 + s_coef2(t)^2 + 2 \cdot s_coef1(t) \cdot s_coef2(t) \cdot \cos(\phi)}$$

У таблиці 6 наведено середні й максимальні відносні відхилення довжин векторів, розрахованих на основі $s_coef1(t), s_coef2(t)$, від 1. Значення відносних відхилень наведено для кутів між крайніми векторами рядка rasterизації $\psi \in \{5, 10, 30, 60, 90, 120, 179\}$. Середні значення відносного відхилення обчислено відносно різних сумарних кількостей векторів у рядку rasterизації.

Отже, вирази коефіцієнтів $s_coef1(t), s_coef2(t)$ при $t \in [8, 15]$, забезпечують розрахунок векторів з відносно невеликим відхиленням від одиничної довжини, що не перевищує 2.5-10%.

Для виразів $s_coef1(t), s_coef2(t)$ при $t \in [5, 7]$, що не використовуються у якості виразів $comb_coef1(t), comb_coef2(t)$, середнє й

максимальне відносні відхилення від одиничної довжини не перевищують 4.4%.

Таблиця 6 – Середні та максимальні відносні відхилення довжин векторів, обчислених на основі $s_coef1(t), s_coef2(t)$, від 1

Індекс у рядку	Середнє та максимальне відхилення вектора від одиничної довжини, %						
	$\psi = 179$	$\psi = 120$	$\psi = 90$	$\psi = 60$	$\psi = 30$	$\psi = 10$	$\psi = 5$
8	0.2/0.3	0.8/2.0	1.9/3.7	4.2/5.3	6.2/6.6	6.9/6.9	7.0/7.0
9	1.1/3.0	0.3/0.8	1.0/2.0	2.4/3.2	4.3/4.6	5.2/6.9	5.0/5.0
10	0.7/1.1	0.6/0.9	0.5/0.9	1.6/2.5	2.8/4.0	4.9/4.9	5.0/5.0
11	0.8/1.1	1.2/1.9	2.4/2.5	2.0/2.3	0.7/0.9	0.1/0.1	0.0/0.0
12	2.9/3.1	0.8/1.7	1.6/1.9	1.8/2.0	0.7/0.8	0.1/0.1	0.0/0.0
13	9.2/10.0	1.0/1.5	3.4/3.8	3.4/3.6	1.2/1.4	0.2/0.2	0.0/0.0
14	4.3/4.7	0.5/0.9	3.1/3.2	3.1/3.2	1.2/1.2	0.2/0.2	0.0/0.0
15	8.3/8.3	3.4/3.4	0.9/0.9	2.2/2.2	1.0/1.0	0.1/0.1	0.0/0.0

На рис. 3 наведено візуалізовані приклади розрахованих на основі $s_coef1(t), s_coef2(t)$ проміжних векторів рядка rasterизації. У всіх випадках сумарна кількість векторів рядка становить 16, кут між крайніми векторами рядка дорівнює 25° (рис. 3 а), 5° (рис. 3 б).

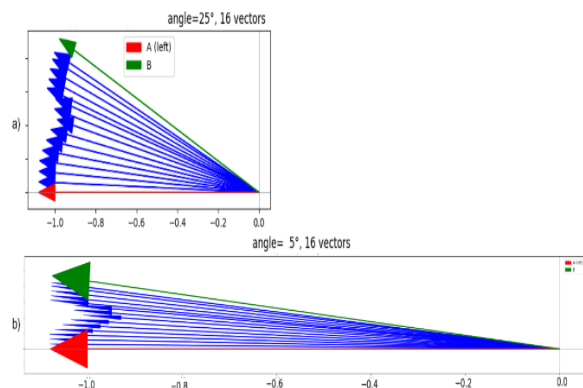


Рисунок 3 – Візуалізація обчислених проміжних векторів на основі коефіцієнтів $s_coef1(t), s_coef2(t)$

На рис. 3 показано, що обчислені вектори мають довжину, наближену до одиничної та є достатньо рівномірно віддаленими один від одного.

У програмному засобі Idx3d було візуалізовано тестову фігуру «Голова» на основі запропонованих виразів. Результати візуалізації наведено на рисунку 4.

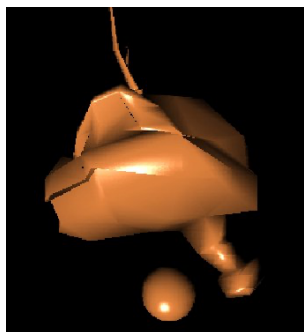


Рисунок 4 – Сформоване зображення тестової фігури «Голова» на основі $s_coef1(t)$, $s_coef2(t)$

Отже, запропоновані вирази забезпечують більш продуктивну нормалізацію векторів при збереженні високої якості візуалізації 3D-сцени.

Висновки

У статті запропоновано нові квадратичні поліноміальні коефіцієнти для підвищення продуктивності паралельного розрахунку векторів вздовж рядка растеризації.

У результаті, продуктивність розрахунку векторів вздовж рядка растеризації підвищено на 7-121% порівняно з використанням ітеративної сферично-кутової інтерполяції векторів. Причому, відхилення довжин векторів від одиничної довжини є прийнятними й не перевищують 2.5-10%.

Розроблені вирази призначені для формування реалістичних зображень у системах 3D-рендерингу реального часу.

Література

1. Symeonidou A. Colour computer-generated holography for point clouds utilizing the Phong illumination model / A. Symeonidou, D. Blinder, P. Schelkens // Optics Express. 2018. Vol. 26, no. 8. pp. 10282-10298. URL: <https://doi.org/10.1364/OE.26.010282> (date of access: 05.10.2025).
2. Detecting Shadow of Moving Object based on Phong Illumination Model / F. Peng Li, Z. Song, B. Li, M. Jie Wu, C. Shen // International Conference on Information Sciences, Machinery, Materials and Energy (ICISMME 2015), Chongqing, China, 11–13 April 2015. 2015. pp. 2019-2022. URL: <https://doi.org/10.2991/icismme-15.2015.413> (date of access: 05.10.2025).
3. Morgand A. A Geometric Model for Specularity Prediction on Planar Surfaces with Multiple Light Sources / A. Morgand, M. Tamaazousti, A. Bartoli // IEEE Transactions on Visualization and Computer Graphics. 2018. Vol. 24, no. 5. pp. 1691-1704. URL: <https://doi.org/10.1109/TVCG.2017.2677445> (date of access: 05.10.2025).
4. BRDF Representation and Acquisition / D. Guarnera, G. C. Guarnera, A. Ghosh, C. Denk, M. Glencross // Computer graphics forum. 2016. Vol. 35, no. 2. pp. 625-650. URL: <https://doi.org/10.1111/cgf.12867> (date of access: 05.10.2025).
5. BXDF material acquisition, representation, and rendering for VR and design / G. C. Guarnera, D. Guarnera, G. J. Ward, M. Glencross, I. Hall // SIGGRAPH Asia 2019 Courses SA 2019, Brisbane, Australia, 17–20 November 2019. 2019. Art. no. 121. URL: <https://doi.org/10.1145/3355047.3362092> (date of access: 05.10.2025).
6. Renhorn I. G. E. Developing a generalized BRDF model from experimental data / I. G. E. Renhorn, G. D. Boreman // Optics Express. 2018. Vol. 26, no. 13. pp. 17099-17114. URL: <https://doi.org/10.1364/OE.26.017099> (date of access: 05.10.2025).
7. Research on the illumination model based on light scattering properties of steel surface / Y. Liu, J. Kong, P. Xu, C. Liu, G. Zheng // MIPPR 2015: Remote Sensing Image Processing, Geographic Information Systems, and Other Applications, Enshi, China, 27–30 June 2024. 2024. Art. no. 98151Q. URL: <https://doi.org/10.1117/12.2205637> (date of access: 05.10.2025).
8. Phong B. T. Illumination for Computer Generated Pictures / B. T. Phong // Communications of the ACM. 1975. Vol. 18, no. 6. pp. 311-317. URL: <https://doi.org/10.1145/360825.360839> (date of access: 05.10.2025).
9. Phong shading approximation of computer-generated holography based on fully analytical triangle meshes / X. Zou, Q. Fu, Y. Liu, M. Yang, P. Wang, Y. Zhang, T.-C. Poon // Chinese Optics Letters. 2025. Vol. 23, no. 4. Art. no. 040501. URL: <https://doi.org/10.3788/COL202523.040501> (date of access: 05.10.2025).
10. Marschner S. Fundamentals of Computer Graphics / S. Marschner, P. Shirley. Boca Raton: A K Peters / CRC Press, 2021. 700 p.
11. Pharr M. Physically based rendering: from theory to implementation: Third edition / M. Pharr, W. Jacob, G. Humphreys. Elsevier Inc., 2016. 1233 p.
12. Method of combined vector normalization of 3D-objects / O. Romanyuk, Y. Zavalniuk, O. Bobko, I. Arsenyuk, O. Romanyuk, N. Sachaniuk-Kavets'ka, L. Nykyforova, A. Kotyra, A. Kozbakova, A. Kalizhanova // Photonics Applications in Astronomy, Communications, Industry, and High Energy Physics Experiments 2024, Lublin, Poland, 27–30 June 2024. 2024. Art. no. 134000O. URL: <https://doi.org/10.1117/12.3054900> (date of access: 05.10.2025).

13. Accelerated Vector Normalization for Rendering Tasks / O. Romanyuk, Y. Zavalniuk, P. Mykhaylov, R. Chekhmestruk, N. Titova, S. Romanyuk // 2024 14th International Conference on Advanced Computer Information Technologies (ACIT), Ceske Budejovice, Czech Republic, 19–21 September 2024. 2024. pp. 640-644. URL: <https://doi.org/10.1109/acit62333.2024.10712602> (date of access: 05.10.2025).
14. Real-Time Rendering. Fourth edition / T. Akenine-Möller, E. Haines, N. Hoffman, A. Pesce, M. Iwanicki, S. Hillaire. Boca Raton: A K Peters / CRC Press, 2018. 1199 p.
15. Szeliski R. Computer Vision: Algorithms and Applications 2nd Edition / R. Szeliski. Springer, 2022. 1206 p.
16. Bhooshan S. Morph & Slerp: Shape description for 3D printing of concrete / S. Bhooshan, T. van Mele, P. Block // Proceedings - Scf 2020 ACM Symposium on Computational Fabrication, USA, 5–6 November 2020. 2020. Art. no. 1. URL: <https://doi.org/10.1145/3424630.3425413> (date of access: 05.10.2025).
17. Wang D. Dual-Quaternion-Based SLERP MPC Local Controller for Safe Self-Driving of Robotic Wheelchairs / D. Wang, W. Cao, A. Takanishi // Robotics. 2023. Vol. 12, no. 6. Article no. 153. URL: <https://doi.org/10.3390/robotics12060153> (date of access: 05.10.2025).
18. Войтко В. В. Аналіз методів нормалізації нормалей для задач формування тривимірних зображень / В. В. Войтко, О. В. Романюк // Наукові праці ВНТУ. 2009. № 1. URL: <https://ir.lib.vntu.edu.ua/bitstream/handle/123456789/4386/109.pdf?sequence=3> (дата звернення: 05.10.2025).
19. Романюк О. Н. Реалізація редерингу Фонга з використанням сферично-кутової інтерполяції / О. Н. Романюк, А. В. Чорний // Вісник Вінницького політехнічного інституту. 2004. № 1. С. 66-71.
20. Романюк О. Н. Один із підходів до підвищення швидкодії зафарбовування / О. Н. Романюк, М. Д. Обідник // Наукові праці Донецького національного технічного університету. Серія : Інформатика, кібернетика та обчислювальна техніка. 2011. Випуск 21 (183). С. 116–121.
21. Fog A. Instruction tables / A. Fog // Agner. URL: https://www.agner.org/optimize/instruction_tables.pdf (date of access: 05.10.2025).

References

1. Symeonidou, A., Blinder, D., and Schelkens, P. (2018), "Colour computer-generated holography for point clouds utilizing the Phong illumination model", *Optics Express*, 26 (8), pp. 10282-10298, available at: <https://doi.org/10.1364/OE.26.010282>.
2. Peng Li, F., Song, Z., Li, B., Jie Wu, M., and Shen, C. (2015), "Detecting Shadow of Moving Object based on Phong Illumination Model", *International Conference on Information Sciences, Machinery, Materials and Energy (ICISMME 2015), April 11-13, 2015, Chongqing, China*, Atlantis Press, pp. 2019-2022, available at: <https://doi.org/10.2991/icismme-15.2015.413>.
3. Morgand, A., Tamaazousti, M., and Bartol, A. (2018), "A Geometric Model for Specularity Prediction on Planar Surfaces with Multiple Light Sources", *IEEE Transactions on Visualization and Computer Graphics*, 24(5), pp. 1691-1704, available at: <https://doi.org/10.1109/TVCG.2017.267744>.
4. Guarnera, D., Guarnera, G. C., Ghosh, A., Denk, C., and Glencross, M. (2016), "BRDF Representation and Acquisition", *Computer graphics forum*, 35(2), pp. 625-650, available at: <https://doi.org/10.1111/cgf.12867>.
5. Guarnera, G. C., Guarnera, D., Ward, G. J., Glencross, M., and Hall, I. (2019), "BXDF material acquisition, representation, and rendering for VR and design", *SIGGRAPH Asia 2019 Courses SA 2019, November 17-20, 2019, Brisbane, Australia*, ACM, Art. no. 121, available at: <https://doi.org/10.1145/3355047.3362092>.
6. Renhorn, I. G. E., and Boreman, G. D. (2018), "Developing a generalized BRDF model from experimental data", *Optics Express*, 26(13), pp. 17099-17114, available at: <https://doi.org/10.1364/OE.26.017099>.
7. Liu, Y., Kong, J., Xu, P., Liu, C., and Zheng, G. (2015), "Research on the illumination model based on light scattering properties of steel surface", *MIPPR 2015: Remote Sensing Image Processing, Geographic Information Systems, and Other Applications, June 27-30, 2024, Enshi, China*, SPIE, Art. no. 98151Q, available at: <https://doi.org/10.1117/12.2205637>.
8. Phong, B. T. (1975), "Illumination for Computer Generated Pictures", *Communications of the ACM*, 18(6), pp. 311-317, available at: <https://doi.org/10.1145/360825.360839>.
9. Zou, X., Fu, Q., Liu, Y., Yang, M., Wang, P., Zhang, Y., and Poon, T.-C. (2025), "Phong shading approximation of computer-generated holography based on fully analytical triangle meshes", *Chinese Optics Letters*, 23(4), Art. no. 040501, available at: <https://doi.org/10.3788/COL202523.040501>.
10. Marschner, S., and Shirley, P. (2021), *Fundamentals of Computer Graphics. Fifth edition*, Boca Raton: A K Peters / CRC Press. 700 p.
11. Pharr, M., Jacob, W., and Humphreys, G. (2016), *Physically based rendering: from theory to implementation: Third edition*, Elsevier Inc. 1233 p.
12. Romanyuk, O., Zavalniuk, Y., Bobko, O., Arsenyuk, I., Romanyuk, O., Sachaniuk-Kavets'ka, N., Nykyforova, L., Kotyra, A., Kozbakova, A., and Kalizhanova, A. (2024), "Method of combined vector

- normalization of 3D-objects", *Photonics Applications in Astronomy, Communications, Industry, and High Energy Physics Experiments 2024, June 27-30, 2024, Lublin, Poland*, SPIE, Art. no. 134000O, available at: URL: <https://doi.org/10.1117/12.3054900> (date of access: 05.10.2025).
13. Romanyuk, O., Zavalniuk, Y., Mykhaylov, P., Chekhmestruk, R., Titova, N., and Romanyuk, S. (2024), "Accelerated Vector Normalization for Rendering Tasks", *2024 14th International Conference on Advanced Computer Information Technologies (ACIT), September 19–21, 2024, Ceske Budejovice, Czech Republic*, IEEE, pp. 640-644, available at: <https://doi.org/10.1109/acit62333.2024.10712602>.
 14. Akenine-Möller, T., Haines, E., Hoffman, N., Pesce, A., Iwanicki, M., and Hillaire, S. (2018), *Real-Time Rendering. Fourth edition*, Boca Raton: A K Peters / CRC Press. 1199 p.
 15. Szeliski, R. (2022), *Computer Vision: Algorithms and Applications 2nd Edition*, Springer. 1206 p.
 16. Bhooshan, S., van Mele, T., and Block, P. (2020), "Morph & Slerp: Shape description for 3D printing of concrete", *Proceedings - Scf 2020 ACM Symposium on Computational Fabrication, November 5-6, 2020, USA*, ACM, art. no. 1, available at: <https://doi.org/10.1145/3424630.3425413>.
 17. Wang, D., Cao, W., and Takanishi, A. (2023), "Dual-Quaternion-Based SLERP MPC Local Controller for Safe Self-Driving of Robotic Wheelchairs", *Robotics*, 12(6), Article no. 153, available at: <https://doi.org/10.3390/robotics12060153>.
 18. Voitko, V. V., and Romanyuk, O. V. (2009), "Analysis of methods for normalization of vectors of normals for the tasks of formation of three dimensional images" ["Analiz metodiv normalizatsii normaliei dlia zadach formuvannia tryvymirnykh zobrazhen"], *Scientific works of Vinnytsia National Technical University*, № 1, available at: <https://ir.lib.vntu.edu.ua/bitstream/handle/123456789/4386/109.pdf?sequence=3>.
 19. Romanyuk, O. N., and Chornyi, A. V. (2004), "Implementation of the Phong rendering with the usage of spherical-angular interpolation" ["Realizatsiia rederynhu Fonha z vykorystanniam sferychno-kutovoi interpolatsii"], *Visnyk of Vinnytsia Polytechnical Institute*, № 1, pp. 66-71.
 20. Romanyuk, O. N., and Obidnyk, M. D. (2011), "One of the approaches to increasing the productivity of shading" ["Odyn iz pidkhodiv do pidvyshchennia shvydkodii zafarbovuvannia"], *Scientific Papers of Donetsk National Technical University. Informatics, Cybernetics and Computer Science*, 21 (183), pp. 116–121.
 21. Fog, A. (2022), "Instruction tables", *Agner*, available at: https://www.agner.org/optimize/instruction_tables.pdf.

Надійшла до редакції 19.10.2025

¹Y. K. ZAVALNIUK, ²O. N. ROMANYUK

^{1,2}Vinnytsia National Technical University, Vinnytsia, Ukraine

¹qq9272627@gmail.com, ²rom8591@gmail.com

SPHERICAL-ANGULAR INTERPOLATION OF VECTORS BASED ON QUADRATIC POLYNOMIALS

In the article, it is proposed new quadratic expressions to increase the productivity of calculating intermediate vectors of a rasterization line based on parallel spherical-angular interpolation. The key features of using normal vectors, light incidence, and light reflection vectors to form highly realistic three-dimensional images are analyzed. The need for high-performance vector calculation for faster image formation is substantiated. The shortcomings of existing methods of linear interpolation of vectors, spherical-linear interpolation of vectors, and spherical-angular interpolation of vectors are analyzed. The advantages of the iterative spherical-angular interpolation method of vectors are considered, as well as the prospects for parallel application of this method. The increase in the computational complexity of the coefficients of the expression of parallel spherical-angular interpolation of vectors at each subsequent point is investigated. The feasibility of developing simplified coefficient expressions to increase the productivity of parallel spherical-angular interpolation of vectors is substantiated. Simplified coefficient quadratic polynomials are developed by using polynomial regression. The performance of applying the developed expressions and the original full expressions is compared. Depending on the rasterization line point number, the combined use of these expressions is proposed. Additionally, the performance of the combined use of full and simplified expressions for parallel vector interpolation and traditional non-parallel vector interpolation methods is compared. The performance gain from the use of new expressions is calculated. The deviation of the lengths of vectors calculated on the basis of simplified quadratic coefficients from the unit length is analyzed. Visualized examples of the calculated intermediate vectors of the rasterization line are provided. An image of a test figure is formed in the Idx3d 3D rendering software based on the use of the proposed expressions. The proposed coefficient quadratic expressions provide high-performance formation of realistic images in rendering systems operating in real time.

Keywords: rendering, computer graphics, vector normalization, spherical-angular vector interpolation, parallelization

№2(41) 2025

