

**В. В. Войтко, к. т. н., доц.; О. В. Романюк**

## **АНАЛІЗ МЕТОДІВ НОРМАЛІЗАЦІЇ ВЕКТОРІВ НОРМАЛЕЙ ДЛЯ ЗАДАЧ ФОРМУВАННЯ ТРИВИМІРНИХ ЗОБРАЖЕНЬ**

*Проведено аналіз ефективних методів нормалізації векторів нормалей для задач комп'ютерної графіки. Наведено основні методи визначення векторів нормалей для поверхонь, що задані аналітично та у вигляді полігональних даних.*

**Ключові слова:** вектор нормалі, нормалізація векторів нормалі, сферично-кутова інтерполяція.

### **Вступ**

Для досягнення фотореалістичності в комп'ютерній графіці необхідно точно відтворити властивості поверхні та фізично правильно описати ефекти освітлення на сцені. Так, однією з базових операцій, яка дозволяє здійснити це, є знаходження векторів нормалі. Розрахунок векторів нормалі є невід'ємною складовою більшості методів формування шорстких та рельєфних поверхонь. У методі бамп-меппінгу [1], наприклад, вектори нормалі розраховуються для обчислення попиксельної освітленості поверхні. Ступінь освітленості точки залежить від кута між нормаллю і променем світла: чим менший кут, тим більша освітленість в цій точці поверхні. Для нерівних поверхонь нормалі в кожній точці будуть різні. Саме цей принцип є основою методу бамп-меппінгу. А метод нормал-меппінг базується на використанні карт нормалей. Колір зображення говорить про те, як зорієнтована нормаль поверхні в цьому пікселі зображення. Фактично карта нормалі задає геометрію поверхні віртуальним обертанням напрямку нормалі в цьому пікселі. На відміну від карт нерівностей (бамп-карт) ефект карт нормалей гарно видно навіть на не дуже блискучих матеріалах. Що стосується методів формування шорстких поверхонь, які використовують карти зміщення [2], то вони також вимагають прорахунку нормалей, оскільки задають зміщення точок поверхні вздовж нормалі.

Векторами нормалей задаються позиції спостерігача і джерела світла. А також вектор нормалі до поверхні задає її локальну кривизну. Так, при зафарбовуванні методом Фонга вздовж стрічки сканування інтерполюються значення векторів нормалей, які потім використовуються в моделі освітлення Фонга (яка все ще залишається дуже популярною та широко використовується в комп'ютерній графіці) для обчислення дифузної та спекулярної складової освітленості точки поверхні. При цьому досягається краща локальна апроксимація кривизни поверхні і, відповідно, отримується більш реалістичне зображення.

Згідно з формулою зафарбовування необхідна нормалізація векторів нормалі. Нормалізація вектора нормалі [3] потребує виконання трьох операцій ділення, трьох операцій множення, двох операцій додавання й операцію знаходження квадратного кореня. У зв'язку з цим можна констатувати, що векторні операції займають вагомий частину обчислювального процесу. Таким чином, актуальним питанням є спрощення процедури нормалізації з метою її апаратної реалізації.

### **Мета дослідження**

Метою цього дослідження є аналіз методів нормалізації векторів нормалей для використання в задачах комп'ютерної графіки.

## Вектор нормалі. Знаходження вектора нормалі

Вектор нормалі, або нормаль до плоскої поверхні – це вектор, який перпендикулярний до цієї поверхні. Нормаль до непласкої поверхні в деякій точці  $P$  є вектором, який перпендикулярний до дотичної площини до цієї поверхні в точці  $P$ , див. рис. 1. Напрямок нормалі визначає орієнтацію поверхні в просторі. У комп'ютерній графіці вектор нормалі використовують для імітації геометричних деталей на плоских поверхнях. У цьому випадку функція буде визначати найменше відхилення від дійсного напрямку нормалі в кожній точці поверхні з метою створення відблисків та затінених ділянок.

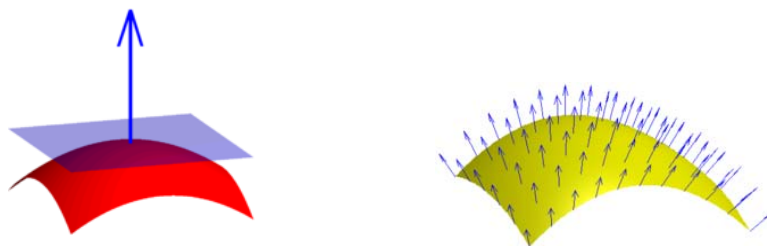


Рис. 1. Визначення вектора нормалі для непласких поверхонь

На ідеальній сфері, наприклад, нормаль до точки поверхні має той же напрям, що й вектор з центру сфери в цю точку. Для інших типів поверхонь існують інші кращі способи знаходження нормалей, які залежать від того, яким чином задається поверхня.

Варто пам'ятати, що гладкі поверхні апроксимуються великою кількістю невеликих плоских полігонів. Якщо вектори, перпендикулярні до цих полігонів, використовуються в якості нормалей апроксимованих поверхонь, то поверхні виглядають сегментованими, оскільки простір векторів не є неперервним за межами полігонів. Однак, у багатьох випадках для моделі існує точний математичний опис і в кожній точці може бути обчислений вектор істинної нормалі. Використання істинних нормалей суттєво покращує результат візуалізації (рис. 2).

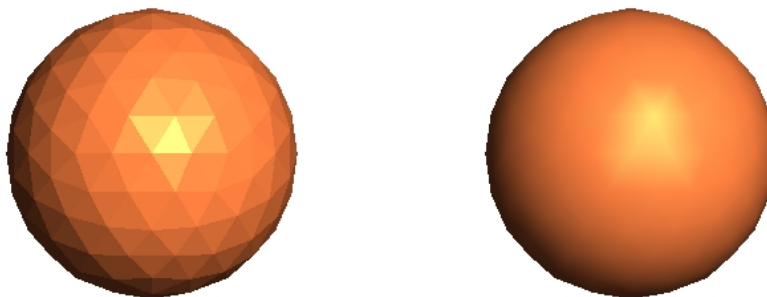


Рис. 2. Істинні нормалі (справа) проти полігональних нормалей (зліва)

### *Розрахунок нормалей для аналітичних поверхонь*

Аналітичні поверхні – це плавні поверхні, які описуються математичним рівнянням або деяким набором рівнянь. У більшості випадків нормалі простіше знаходити для аналітичних поверхонь, для яких наявний вичерпний опис у такій формі:

$$\mathbf{V}(s, t) = [X(s, t) \ Y(s, t) \ Z(s, t)],$$

де  $s$  і  $t$  визначені в деякому просторі, а  $X$ ,  $Y$  і  $Z$  – диференційовані функції двох змінних.

Для того, щоб знайти нормаль, необхідно обчислити часткові похідні  $\frac{\partial V}{\partial s}$  і  $\frac{\partial V}{\partial t}$ , які є

векторами, дотичними до поверхні у напрямках  $s$  і  $t$ . Їхній векторний добуток  $\frac{\partial V}{\partial s} \times \frac{\partial V}{\partial t}$  перпендикулярний до них обох і, як наслідок, перпендикулярний до поверхні. Наступна формула відображає процес обчислення векторного добутку двох векторів:

$$[v_x v_y v_z] \times [w_x w_y w_z] = [(v_y w_z - w_y v_z) (v_z w_x - w_z v_x) (v_x w_y - w_x v_y)]. \quad (1)$$

Для нормалізації результуючого вектора  $[x, y, z]$  необхідно знайти його довжину:

$$l = \sqrt{x^2 + y^2 + z^2} \quad (2)$$

і поділити на неї кожен з компонентів вектора.

Як приклад для таких розрахунків візьмемо аналітичну поверхню  $V(s, t) = [2s \quad t^2 \quad 4 + 2st]$ . Вирахуємо часткові похідні і за допомогою формули (1) знайдемо векторний добуток отриманих векторів:

$$\frac{\partial V}{\partial s} = [2 \quad 0 \quad 2t], \quad \frac{\partial V}{\partial t} = [0 \quad 2t \quad 2s], \quad \frac{\partial V}{\partial s} \times \frac{\partial V}{\partial t} = [-4t^2 \quad -4s \quad 4t].$$

Таким чином, якщо в цьому прикладі  $s=2$  і  $t=1$ , то відповідною точкою поверхні є точка з координатами (4, 1, 8), а вектором нормалі буде вектор (-4, -8, 4). Довжина цього вектора дорівнює 9,8, тоді вектор нормалі одиничної довжини дорівнює:

$$(-4/9.8, -8/9.8, 4/9.8) = (-0.408163, -0.816326, 0.408163).$$

#### *Розрахунок нормалей за полігональними даними*

Дуже часто поверхні задані у вигляді полігональних даних. Для того, щоб вони виглядали гладкими, а не сегментованими, потрібно обчислювати нормалі до поверхні. У більшості випадків найпростіше це можна зробити за допомогою обчислення вектору нормалі для кожного полігонального фрагмента поверхні, а потім усереднити нормалі сусідніх фрагментів. На рис. 3 зображена поверхня та її полігональна апроксимація.

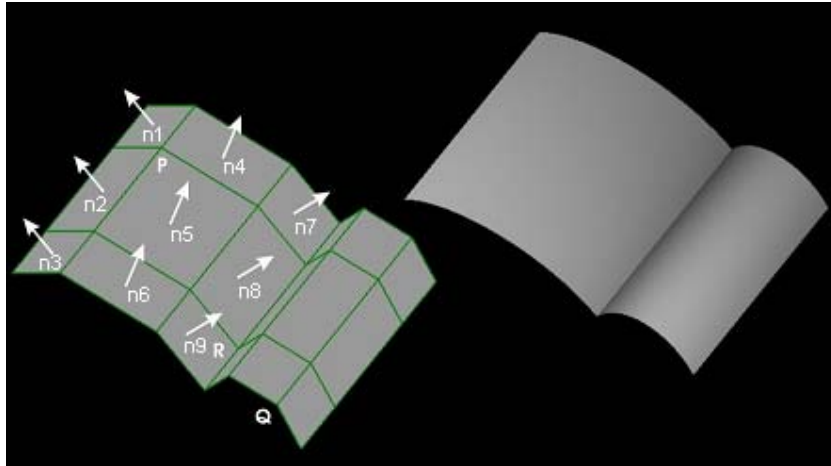


Рис. 3. Усереднення векторів нормалі

Для того, щоб знайти вектор нормалі для плоского полігона потрібні три будь-які вершини полігона  $v_1, v_2$  і  $v_3$ , які не лежать на одній прямій. Векторний добуток

$$[v_1 - v_2] \times [v_2 - v_3] \quad (3)$$

і буде нормаллю до полігона. Як правило, результуючий вектор потрібно нормалізувати. Потім потрібно усереднити нормалі сусідніх полігонів, щоб не надавати більше ваги одному з них. Наприклад, якщо у випадку показаному на рис. 3,  $n_1, n_2, n_4$  і  $n_5$  – нормалі полігонів, які

з'єднуються в точці Р, то потрібно знайти суму векторів  $n_1+n_2+n_4+n_5$ , а потім нормалізувати отриманий вектор. Результуючий вектор може слугувати нормаллю в точці Р.

У деяких моделях бувають плавні частини і гострі кути (точка R на рис. 3 розміщена на такому ребрі). У такому випадку не потрібно усереднювати нормалі сусідніх полігонів, а навпаки – полігони з однієї сторони ребра повинні бути нарисовані з використанням однієї нормалі, а з другої сторони – з використанням іншої нормалі.

### Аналіз методів нормалізації векторів нормалі

Науковцями фірми VIDIA було отримано формулу [4] для наближеної нормалізації векторів нормалей з використанням одного кроку ітерації Ньютона – Рафсона:

$$\vec{N}_n = \frac{\vec{N}}{3} (3 - \vec{N} \cdot \vec{N}). \quad (4)$$

Розрахунок згідно з наведеною формулою вимагає однієї операції ділення, 6 операцій множення та 3 операції додавання. Великі абсолютні похибки визначення ортогональних складових вектора обмежують використання формули для задач зафарбовування.

Згідно з підходом Р. Ліона вираз  $1/\sqrt{\vec{N} \cdot \vec{N}}$ , який використовується для нормалізації вектора нормалі  $\vec{N}$ , розкладають у ряд Тейлора і обмежуються першими трьома членами. Тоді [5]:

$$\vec{N}_n = \vec{N} \left( 1 - \frac{1}{2} ((\vec{N} \cdot \vec{N}) - 1) + \frac{3}{8} ((\vec{N} \cdot \vec{N}) - 1)^2 \right). \quad (5)$$

Формула хоча і придатна для апаратної реалізації, але не забезпечує прийнятної точності.

Інтерполяцію одиничних векторів нормалей між початковим  $\vec{N}_a$  та кінцевим  $\vec{N}_b$  векторами, які мають одиничну довжину, можна виконати за формулою [6]:

$$\vec{N}(w) = \vec{N}_a \frac{\sin((1-w)\psi)}{\sin\psi} + \vec{N}_b \frac{\sin(w\psi)}{\sin\psi}, \quad (6)$$

де  $w \in [0, 1]$ , а  $\psi$  – кут між векторами нормалей  $\vec{N}_a$  та  $\vec{N}_b$ .

Однак, суттєвим недоліком методу є необхідність розрахунку тригонометричних функцій  $\sin$ ,  $\arccos$  як для знаходження конкретного вектора нормалі, так і для розрахунку невідомих параметрів  $w$  та  $\psi$ . Крім того, розрахунок  $\vec{N}(w)$  передбачає виконання в циклі нормалізації операції ділення.

У роботі [7] запропоновано використовувати принцип дихотомії для проведення нормалізації векторів нормалей шляхом послідовного поділу навпіл кута між векторами нормалей у початковій та кінцевій точках рядка растеризації трикутника. Загальна формула має вигляд

$$\vec{N}_{\left(\frac{1}{2^{n+1}}\right)} = \frac{\vec{N}_a + \vec{N}_{\frac{1}{2^n}}}{\sqrt{2 + z_{2^n}}}, \quad (7)$$

де  $\vec{N}_a$  – початковий вектор; позначення  $\left(\frac{1}{2^n}\right)$  означає кількість сегментів, які отримано на  $n$  – ітерації при послідовному поділі навпіл сегментів рядка растеризації трикутника за умови, що при  $n = 0$  сегмент дорівнює рядку растеризації;  $z_{2^n} = \sqrt{2(1 + \cos \frac{\psi}{2^{n-1}})}$ ;  $\psi$  – кут між векторами нормалей  $\vec{N}_a$  (початковим) та  $\vec{N}_b$  (кінцевим).

На кожній ітерації для знаходження знаменника виконується лише одна операція

додавання та одна операція визначення квадратного кореня.

Перевагою цього методу є те, що вираз  $\frac{1}{\sqrt{2+z_{2^n}}}$  можна апроксимувати рядом Чебишева, що забезпечує незначні похибки апроксимації. При використанні багаточлена першої степені

$$\frac{1}{\sqrt{2+z_{2^n}}} \approx -0,07 \cdot z_{2^n} + 0,64 \quad (8)$$

максимальна абсолютна похибка апроксимації не перевищує 0,0005, а відносна 0,12 %. Цю формулу доцільно використовувати для екранів із невеликою роздільною здатністю для яких трикутники, які складають поверхню тривимірного об'єкту, мають незначні розміри.

При використанні багаточлену другої степені

$$\frac{1}{\sqrt{2+z_{2^n}}} \approx 0,014 \cdot z_{2^n}^2 - 0,119 \cdot z_{2^n} + 0,681 \quad (9)$$

максимальна абсолютна похибка апроксимації не перевищує  $2 \cdot 10^{-5}$ , а відносна 0,004 %. Аналіз показав, що при використанні цієї апроксимаційної формули час розрахунку вектору  $\vec{N}_{(\frac{1}{2^n})}$  зменшується в 2,5 разів порівняно з класичною реалізацією.

Інший метод нормалізації векторів нормалі, запропонований в роботі [8], передбачає використання квадратичної інтерполяції за умови, що відомо одиничні вектори нормалей у початковій та кінцевій точках  $i$ -того рядка растеризації трикутника. Проміжні значення векторів нормалей у рядку растеризації трикутника знаходяться за формулою:

$$\vec{N}_{i,t} = \vec{G}_i \cdot t^2 + \vec{P}_i \cdot t + \vec{Q}_i \quad (10)$$

Нехай  $\vec{N}_{i,l}$ ,  $\vec{N}_{i,p}$ ,  $\vec{N}_{i,c}$  – відповідно вектори нормалей у лівій, правій та середній точках рядка растеризації трикутника. При  $t=0$   $\vec{N}_{i,l} = \vec{Q}_i$ . У правій точці рядка растеризації  $t=1$ , тому

$$\vec{N}_{i,p} = \vec{G}_i + \vec{P}_i + \vec{Q}_i. \text{ Оскільки в середній точці рядка растеризації } t=1/2, \text{ то } \vec{N}_{i,c} = \frac{\vec{G}_i}{4} + \frac{\vec{P}_i}{2} + \vec{Q}_i.$$

Після ряду перетворень можна знайти наступний вираз для знаходження вектору нормалі в середній точці:

$$\vec{N}_{i,c} = \frac{1}{\sqrt{2(1 + \vec{N}_{i,l} \cdot \vec{N}_{i,p})}} (\vec{N}_{i,l} + \vec{N}_{i,p}) \quad (11)$$

Перевагою цього методу є можливість апроксимувати вираз  $\frac{1}{\sqrt{2(1 + \vec{N}_{i,l} \cdot \vec{N}_{i,p})}}$  поліномом

Чебишева другої степені, що дозволяє зменшити час розрахунку вектора нормалі в середній точці рядка растеризації більш як в 2,5 рази. Аналіз показав, що при програмній реалізації запропонованого методу час розрахунку векторів нормалей для середнього трикутника зменшився в 2,8 разів порівняно з класичною реалізацією.

Для формування векторів нормалі можна використовувати і сферично-кутову інтерполяцію [9] (рис. 4). Цей метод доцільно використовувати за умови оперування з кутами над векторами нормалей. Проміжні значення векторів нормалі за цим методом визначаються за формулою:

$$\vec{N}(t) = \vec{N}_a \cdot \cos(t \cdot \varphi) + \vec{N}_k \cdot \sin(t \cdot \varphi) \quad (12)$$

де  $t$  – номер піксела вздовж рядка rasterизації,  $t \in [0, l]$ ;  $\vec{N}_a$  – початковий вектор;  $\vec{N}_k = \frac{\vec{N}_b - \vec{N}_a(\vec{N}_b \cdot \vec{N}_a)}{\sqrt{1 - (\vec{N}_b \cdot \vec{N}_a)^2}}$  – нормований вектор нормалі;  $\vec{N}_b$  – кінцевий вектор;  $\varphi = \psi / m$  – кут між двома сусідніми векторами нормалей, де  $\psi = \arccos(\vec{N}_a \cdot \vec{N}_b)$  і  $m$  – довжина рядка rasterизації.

Цю формулу після ряду перетворень можна записати в ітераційній формі:

$$\vec{N}(t+1) = 2\vec{N}(t) \cdot \cos \varphi - \vec{N}(t-1) \quad (13)$$

З останньої формули можна зробити висновок, що одиничний вектор нормалі при сферично-кутовій інтерполяції можна знайти через два попередніх значення.

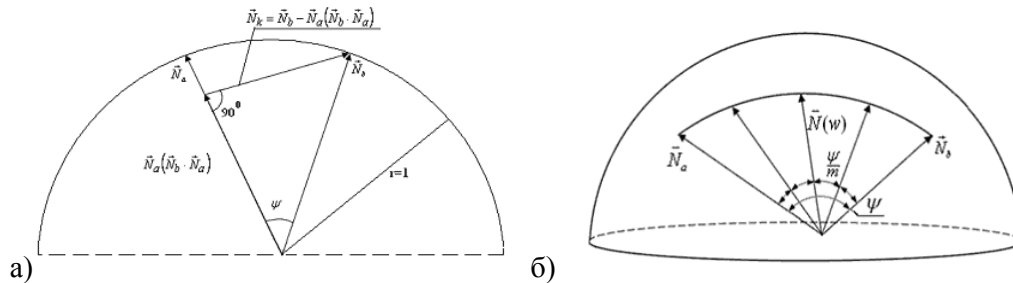


Рис. 4. Визначення вектору  $\vec{N}_k$  (а) та сферично-кутова інтерполяція векторів нормалі (б)

Аналіз показав, що при програмній реалізації час розрахунку згідно з формулою (13) векторів нормалей середнього трикутника зменшився в 1,7 разів порівняно з обчисленням одиничних векторів нормалей шляхом кутової інтерполяції. Таким чином, можна констатувати, що досягається суттєве підвищення швидкодії зафарбовування поверхні.

### Висновки

Знаходження векторів нормалі та здійснення їхньої нормалізації займає вагомую частину процесу зафарбовування тривимірних зображень в задачах комп'ютерної графіки. Значна кількість науковців присвятили свої дослідження спрощенню процедури нормалізації векторів нормалей. Аналіз, проведений у цій роботі, показав, що на сьогодні найбільш ефективними методами нормалізації є методи, що використовують квадратичну інтерполяцію та сферично-кутову інтерполяцію за умови оперування з кутами над векторами нормалей, а також принцип дихотомії. Ці методи дозволяють покращити час розрахунку векторів нормалей від 1,7 до 2,8 разів у порівнянні з класичною нормалізацією, що в свою чергу суттєво підвищує швидкодію зафарбовування тривимірних зображень.

### СПИСОК ЛІТЕРАТУРИ

1. Blinn J. F. Simulation of wrinkled surfaces. // Computer Graphics (SIGGRAPH '78 Proceedings). – 1978. – № 12. – P. 286 – 292.
2. Donnelly W. Per-pixel displacement mapping with distance functions // In GPU Gems II, M. P., (Ed.). Addison-Wesley. – 2005. – P. 123 – 136.
3. Романюк О. Н. Новий підхід до підвищення реалістичності зафарбовування тривимірних об'єктів за методом Гуро // Інформаційні технології та комп'ютерна інженерія. – 2005. – №2. – С. 106 – 109.
4. Barrera T., Hast A., Bengtsson E. Fast Near Phong-Quality Software // Shading SCCG'06 full paper. – 2006. – P. 44 – 50.
5. Lyon R. F. Phong Shading Reformulation for Hardware Renderer Simplification // Apple Technical Report. – 1993. – № 43. – 34 p.
6. Hast A. Improved Algorithms for Fast Shading and Lighting // Acta Universitatis Upsaliensis. Uppsala. – 2004. – 116 p.
7. Романюк О. Н. Комбіноване використання бінарної та кодової лінійної інтерполяції для нормалізації

векторів нормалей при зафарбовуванні тривимірних об'єктів // Вестник Херсонського національного технічного університету. — 2006. — № 25. — С. 408 — 411.

8. Романюк О. Н. Використання квадратичної інтерполяції для зафарбовування тривимірних графічних об'єктів // Реєстрація, зберігання і обробка даних. — 2006. — Т. 8. — № 4. — С. 31 — 37.

9. Романюк О. Н. Реалізація рендерингу Фонга з використанням сферично-кутової інтерполяції / О. Н. Романюк, А. В. Чорний // Вісник Вінницького політехнічного інституту. — 2004. — № 3. — С. 66 — 71.

**Войтко Вікторія Володимирівна** — к. т. н., доцент кафедри програмного забезпечення; тел. 8 (0432) 59-84-83.

**Романюк Оксана Володимирівна** — аспірант кафедри програмного забезпечення; тел. 8 (0432) 65-85-06.

Вінницький національний технічний університет.