



**ЕЛЕКТРОННІ ІНФОРМАЦІЙНІ
РЕСУРСИ: СТВОРЕННЯ, ВИКОРИСТАННЯ,
ДОСТУП ТА УПРАВЛІННЯ**

ЗБІРНИК МАТЕРІАЛІВ

Міжнародної науково-практичної Інтернет-конференції

20-21 листопада 2024 р.

Міністерство освіти і науки України
Вінницький національний технічний університет
Національна академія Державної прикордонної служби України ім. Богдана
Хмельницького
Одеський національний технологічний університет
Вінницький національний медичний університет ім. М.І. Пирогова
КЗВО «Вінницька академія безперервної освіти»
Сумський обласний інститут післядипломної педагогічної освіти
Університет Бельсько-Бяльський (Польща)

**«ЕЛЕКТРОННІ ІНФОРМАЦІЙНІ
РЕСУРСИ: СТВОРЕННЯ, ВИКОРИСТАННЯ,
ДОСТУП ТА УПРАВЛІННЯ»**

ЗБІРНИК МАТЕРІАЛІВ

Міжнародної науково-практичної Інтернет-конференції
20-21 листопада 2024 р.

Суми/Вінниця
НІКО/КЗВО «Вінницька академія безперервної освіти»
2024

УДК 004
ББК 32.97
Е50

Рекомендовано до видання Вченою радою КЗВО «Вінницька академія безперервної освіти» (протокол № 8 від 20.11.2024 р.)

Електронні інформаційні ресурси: створення, використання, доступ та управління. Збірник матеріалів Міжнародної науково-практичної Інтернет конференції 20-21 листопада 2024 р. – Суми/Вінниця: НІКО / КЗВО «Вінницька академія безперервної освіти», 2024. – 220 с.

ISBN 978-617-7422-24-1

Збірник містить матеріали Міжнародної науково-практичної Інтернет конференції «Електронні інформаційні ресурси: створення, використання, доступ та управління. Матеріали збірника подано у авторській редакції. Автори опублікованих матеріалів несуть повну відповідальність за підбір, точність наведених фактів, цитат, статистичних даних, власних імен та інших відомостей, Матеріали відтворюються зі збереженням змісту, орфографії та синтаксису текстів, наданих авторами.

УДК 004
ISBN 978-617-7422-24-1

© Вінницький національний технічний університет 2024

© КЗВО «Вінницька академія безперервної освіти», 2024

© Видавництво Суми, НІКО, 2024

ЗМІСТ

Аксьонов І.Е., Торяник Л.О.	Цифрові технології в науці, освіті та промисловості	7
Андренко К.В., Єрмакова Н.А.	Роль освітніх курсів з опанування мобільної грамотності для дорослих в умовах війни в Україні	10
Андрійчук М. Д. Павлюк Д. В. Лазаренко В. О.	Новітні цифрові технології для ефективної трансформації освітнього процесу	12
Бабенко І. О., Десятнюк Л. Б.	Цифровізація медицини: шлях до ефективнішої охорони здоров'я в Україні	14
Бабюк Н.П., Панасюк Б.Ю.	Аналіз методів моделювання архітектури програмних систем	15
Бідник Т.В.	Організаційно-правовий механізм цифровізації територіальних громад	17
Біла, В.М., Каленіченко, Л.І.	Юридичні аспекти використання електронних доказів	19
Бойчук В. О.	Інформаційна система спортивного комплексу	21
Василенко Н. С., Романюк О. Н.	Аналіз методу згладжування SRAA	22
Величко Н. П., Романюк О. Н.	Інформаційне забезпечення процесу вивчення математики в школі	23
Виниченко Є.О., Торяник Л. О.	Основні підходи та технології комп'ютерної візуалізації та віртуальна, доповнена реальність	25
Войтко В.В., Малініч П.П.	Використання карти висот у сфері комерційної доставки в межах міста	28
Войтко В.В., Позур М.Ю.	Метапрограмування з використанням REFLECTION.EMIT в .NET	30
Войтко В.В., Черноволик Г.О., Барчук Н.С., Гаврилюк О.В., Осипенко К.С.	Удосконалення методу "острови" для підвищення швидкості роботи програмних застосунків у браузері	31
Вуйчак Є. Д.	Розробка комп'ютерної системи управління SMART – холодильником	35
Граняк В. Ф.	Особливості виявлення аномалій технічних параметрів асинхронного двигуна на основі аналізу його статорних струмів	36
Грицишин В. О., Майданюк В. П.	Використання стеганографії для захисту рентгенівських знімків	41
Губіна С.І.	Формування емоційного інтелекту майбутніх учителів в умовах дистанційного навчання	42
Дрижук О.А	Використання цифрових технологій в освітньому процесі	45
Дудукало Н.С., Романюк О.Н.	Особливості методу трасування шляху	47
Дудукало Н.С., Романюк О.Н.	Аналіз розподільних здатностей екранів	50
Завальнюк, Є. К., Романюк, О. Н.	Аналіз вимог до графічних мов програмування	52
Зігунов, О.М, Козленко В.О.	Упровадження інформаційно-комунікаційних технологій в освітній процес у ВСП "Сумський фаховий коледж національного університету харчових технологій"	54
Зьора І. Є., Хошаба О. М.	Вирішення проблеми непрозорості письмових атестаційних робіт студентів при дистанційній формі навчання	59

Римар П.В.	Використання бібліотеки OPENGL для створення мобільних додатків з іграми	163
Рожицький М. Є., Кожем'яко А. В.	Сучасні тенденції розвитку автомобільного транспорту в Україні: порівняльний аналіз і регіональні програми модернізації	164
Романюк О. Н., Бобко О. Л., Мельник А. В., Шевчук Р.П. Романюк О. Н., Новосельцев О. О., Мельник А. В., Майданюк В. П., Шевчук Р.П.	Хмарні сервіси для рендерингу тривимірних зображень	166
Романюк О. Н., Новосельцев О.О., Станіславенко С.Г., Майданюк В.П., Романюк С.О.	Аналіз найпопулярніших пакетів прикладних програм для текстуровання в галузі комп'ютерної графіки	168
Романюк О. Н., Снігур А. В.	Аналіз нових методів текстуровання	171
Романюк О.Н., Безсмертний О. Ю., Романюк О.В., Мельник А.	Часові математичні моделі процесу читання	174
Сандрацький Р.В., Рейда О. М.	Методи прискореної нормалізації векторів	177
Сацюк І. А., Романюк О.Н.	Методи та засоби контролю даних "розумного годинника" для управління фізичними навантаженнями під час тренувань	180
Складанюк, О. О., Майданюк, В. П.	Аналіз платформи NVIDIA RUBIN	182
Соболь О.О.,	Цифрові технології і гейміфікація у сучасних навчальних процесах і освіті	185
Сотніков В. А.	Цифрова грамотність для молодших школярів: базові навички	186
Стахов О. Я., Шклярук М. Б., Сентюрін Є. Є.	Інтеграція фінансових технологій у навчальні програми	188
Стяглик Н.І.	Cuda, як платформа для високопродуктивних обчислень на GPU у науці та технологіях	189
Суліма Ю. О., Ткаченко О. М.	Медіаграмотність як засіб протидії дезінформації: роль освіти у формуванні критичного мислення	191
Теренчук А.Т.	Дослідження методів розв'язання NP-повних задач на прикладі задачі про перекриття точок колами заданого радіуса	193
Торяник Л. О.	Інформаційне забезпечення медичної освіти в умовах діджиталізації	195
Фоменко Є.В.	Візуальні інструменти та методи покращення засвоєння матеріалу з дискретної математики	196
Чепіга І.С.	Цифровізація документообігу в територіальних громадах в умовах електронного урядування	199
Шевченко Д.Г., Городецький О.В.	Науково-практичні рекомендації щодо вдосконалення управління цифровізацією бізнес-процесів підприємства	201
Шевчук П Г	Захист держави засобами цифрових технологій	212
Шеншин О. О., Романюк, О. Н.	Вразливість людини поряд з мовними нейромережами	213
Яремко С.А., Шевчук В.В.	Фізично обґрунтована анімація в комп'ютерній графіці	215
	Сучасні тенденції в розробці додатків для мобільних пристроїв	218

МЕТОДИ ПРИСКОРЕНОЇ НОРМАЛІЗАЦІЇ ВЕКТОРІВ

Анотація. Проаналізовано методи прискореної нормалізації векторів вплив комп'ютерних ігор на психічний та фізичний стан людини.

Ключові слова. геймінг, кіберспорт, комп'ютерні ігри

Нормалізація векторів — це процес перетворення векторів таким чином, щоб їх довжина (або норма) дорівнювала одиниці. Це дуже корисна операція, яка застосовується в багатьох галузях. У комп'ютерній графіці нормалізація векторів необхідна при обчисленні напрямків світла, нормалей поверхонь для коректного відображення світлотінів та реалістичної візуалізації. В алгоритмах класифікації та кластеризації Машинного навчання нормалізація допомагає уніфікувати масштаби даних, що поліпшує збіжність навчальних процесів та підвищує їх ефективність. У галузі обробки зображень використання нормалізованих векторів дозволяє здійснювати точніші порівняння та аналіз характеристик зображень, таких як текстури та контури. У симуляціях динаміки руху, таких як моделювання польотів або віртуальні реальності, нормалізація векторів використовується для обрахунку напрямків сил та їх взаємодій. У навігації та управлінні рухом роботів нормалізація векторів важлива для точного визначення напрямків та відстаней. Нормалізовані вектори часто використовуються для обчислення кутів між векторами та інших геометричних параметрів.

Ці приклади демонструють універсальність нормалізації векторів та її значення у різних наукових та технічних застосуваннях.

Наближена нормалізація векторів може бути корисною в областях, де висока точність не є критичною, але важливою є швидкість обчислень, наприклад, у комп'ютерних іграх або в реальному часі графічних додатках. Ось декілька методів, які зазвичай використовуються для наближеної нормалізації векторів:

Нормалізація [1-5] вектора вимагає виконання трьох ділень, трьох множень, двох додавань та обчислення квадратного кореня. Це свідчить про те, що векторні операції займають значну частину обчислювального процесу. Тому актуальним є питання спрощення процедури нормалізації для її апаратної реалізації.

Апроксимація нормалізації векторів може бути корисною в різних галузях, таких як формування комп'ютерних зображень, машинне навчання та цифрова обробка сигналів, де потрібно ефективно обробляти великі масиви даних.

Метод Лоутонера використовується для обчислення оберненого квадратного кореня довжини вектора. Звичайно це включає в себе використання кількох ітерацій методу Ньютон-Рафсона (або методу дихотомії) для знаходження оберненого квадратного кореня суми квадратів компонент вектора. Метод швидких інверсійних квадратних коренів (Fast Inverse Square Root) відомий своїм використанням у графічному API Quake III Arena. Він використовує магічне число та бітові операції для швидкого наближення оберненого квадратного кореня числа. Хоча метод менш точний, він значно швидший за традиційні методи. Можна використовувати ряд Тейлора або інші математичні розклади для наближеного обчислення функцій, потрібних для нормалізації, наприклад, оберненого квадратного кореня.

Для додаткового прискорення можна використовувати таблиці пошуку для зберігання попередньо обчислених значень обернених квадратних коренів. Це особливо корисно в умовах, де обчислювальна потужність обмежена, але доступ до пам'яті є швидким.

Часто використовують ітеративні методи з передчасною зупинкою, коли досягнута точність вважається достатньою для конкретної задачі, що дозволяє економити час.

Розглянемо більш детально кожний з цих методів.

Метод Лоутонера — це метод апроксимації, який використовується для швидкого обчислення оберненого квадратного кореня числа, що є особливо корисним у комп'ютерній графіці та інших обчислювальних застосуваннях. Цей метод часто асоціюється з оптимізованим алгоритмом, відомим як "швидкий обернений квадратний корінь" (Fast Inverse Square Root), який був популяризований

завдяки його використанню у відеоіграх, таких як Quake III Arena. Метод Лоутонера дозволяє швидко обчислювати обернений квадратний корінь, використовуючи менше операцій, ніж традиційні методи, такі як ті, що базуються на ітеративному алгоритмі Ньютона-Рафсона. У випадку з методом швидкого оберненого квадратного кореня, алгоритм використовує так зване "магічне число", яке допомагає досягти наближення шляхом бітових операцій. Це число було вибрано емпірично та ефективно використовується для ініціалізації процесу апроксимації.

Цей метод, незважаючи на свою апроксимативну природу, залишається популярним вибором там, де швидкість важливіша за абсолютну точність, особливо в індустрії відеоігор та візуалізації.

Алгоритм швидкого оберненого квадратного кореня включає ряд етапів. Виконується початкова апроксимація. Для початкового наближення використовується бітові операції. Число перетворюється з плаваючою комою у ціле число, а потім модифікується за допомогою магічного

числа $\frac{1}{\sqrt{x}}$.

Для 32-бітного числа з плаваючою комою: $i=0x5f3759df-(i>>1)$. Тут I є бітовим представленням вхідного числа x , перетвореного до цілого типу, а магічне число $0x5f3759df$ використовується для створення початкової апроксимації.

Після отримання початкової апроксимації, застосовується одна або кілька ітерацій методу Ньютона (також відомого як метод Ньютона-Рафсона) для покращення точності:

$$y = y \cdot (1.5 - 0.5 \cdot x \cdot y^2)$$

Ця ітерація використовується для покращення початкової апроксимації, зводячи розбіжність та підвищуючи точність.

Цей алгоритм широко відомий своєю здатністю швидко знаходити обернений квадратний корінь, використовуючи мінімальні ресурси. Це особливо корисно у графічних обчисленнях, де необхідно нормалізувати вектори для обчислення освітлення та відбивання у 3D сцені.

Тейлорівський ряд та інші асимптотичні розклади можуть бути корисними для апроксимації функцій, які зустрічаються в нормалізації векторів, таких як обчислення оберненого квадратного кореня. Ці методи забезпечують аналітичні апроксимації, які можуть бути використані для швидкого обчислення в умовах обмежених ресурсів або коли необхідна висока швидкість обчислень. Ось як можна використовувати Тейлорівський ряд для апроксимації оберненого квадратного кореня, що є ключовим в нормалізації векторів.

Розглянемо тейлорівський ряд для $f(x) = \frac{1}{\sqrt{x}}$. Функцію $\frac{1}{\sqrt{x}}$ можна розкласти у

Тейлорівський ряд навколо точки a , якщо вона диференційована в цій точці і в її околі. Якщо розкласти цю функцію в ряд біля точки $a=1$, отримаємо:

$$\frac{1}{\sqrt{x}} = \frac{1}{\sqrt{1+(x-1)}} = \sum_{n=0}^{\infty} (-1)^n \binom{-1/2}{n} (x-1)^n,$$

де $\binom{-1/2}{n}$ є біноміальним коефіцієнтом, який може бути виражений через факторіали:

$$\binom{-1/2}{n} = \frac{(1/2)(-3/2)(-5/2)\dots((-2n+1)/2)}{n!} =$$

Це розклад може бути використаний для апроксимації $\frac{1}{\sqrt{x}}$ при невеликих значеннях $x - 1$, що

є корисним для нормалізації векторів, коли x близьке до 1.

Для практичних обчислень, зазвичай достатньо взяти перші кілька членів ряду для досягнення необхідної точності. Наприклад, перші два члени ряду дадуть наступну апроксимацію:

$$\frac{1}{\sqrt{x}} \approx 1 - \frac{1}{2}(x-1)$$

Ця формула може бути використана для нормалізації векторів, якщо довжина вектора x досить близька до 1, що зменшує обчислювальне навантаження порівняно з точними методами.

Таблиці пошуку (lookup tables, LUTs) є ефективним засобом оптимізації обчислень, особливо коли потрібно швидко виконувати повторювані або складні математичні операції, як нормалізація векторів. У контексті нормалізації векторів, таблиці пошуку можуть бути використані для швидкого доступу до попередньо обчислених значень, що знижує обчислювальне навантаження на процесор. Ось як це можна реалізувати:

Спочатку потрібно визначити діапазон значень, які вектор може приймати, та рівень точності, який вам потрібен. Наприклад, якщо вектори зазвичай мають довжину від 0 до 1, ви можете вирішити створити таблицю для цього діапазону з кроком 0.01.

Для кожного значення в обраному діапазоні обчислюється обернений квадратний корінь. Ці значення зберігаються в таблиці пошуку. Наприклад, для довжини вектора 1, обчислює $\frac{1}{l}$ і зберігає результат.

Припустимо, необхідно нормалізувати 3D вектори. Необхідно сформувати створити LUT для довжин від 0.01 до 1.00 з кроком 0.01. Кожен елемент у таблиці буде відповідати оберненому квадратному кореню відповідної довжини. Коли потрібно нормалізувати вектор виконують такі дії :

Обчислюють довжину $l = \sqrt{x^2 + y^2 + z^2}$; визначає найближче значення в LUT; множите кожен компоненту вектора на значення з таблиці.

Обчислення нормалізації векторів стає значно швидшим, оскільки замінюються складні математичні операції простим пошуком в таблиці. Зменшується кількість вимогливих до процесора обчислень.

Точність залежить від розміру та кроку таблиці пошуку. Таблиці пошуку можуть вимагати значний обсяг пам'яті, особливо якщо ви хочете забезпечити високу точність для великого діапазону значень.

Ітеративні методи з зупинкою раніше — це підходи, які застосовуються для оптимізації обчислювальних процесів, де важливий баланс між швидкістю виконання та точністю результату. У контексті нормалізації векторів, такі методи можуть допомогти швидко досягти прийнятної точності без потреби в повному обчисленні до найточнішого результату. Це особливо корисно в областях, де обчислення мають виконуватися в реальному часі, таких як комп'ютерні ігри, реальні фізичні симуляції, чи обробка потокових даних.

Ітеративні методи з зупинкою раніше для нормалізації векторів зазвичай починають з грубої апроксимації нормалізаційного множника (наприклад, оберненого квадратного кореня від довжини вектора) і потім послідовно покращують цю апроксимацію з кожним ітераційним кроком. Процес ітерацій зупиняється, коли досягнуто заданого порога точності.

Один з найпопулярніших ітеративних методів для обчислення оберненого квадратного кореня — метод Ньютона (або Ньютона-Рафсона). Розглянемо основні етапи.

Задається початкове наближення y_0 , яке може бути взято з таблиці пошуку або вираховано за допомогою швидкого алгоритму.

Використовується формула Ньютона для уточнення значення: $y_{n+1} = y_n(1.5 - \frac{1}{2}xy_n^2)$, де

x — довжина вектора, яка обчислюється як $x = \sqrt{v_x^2 + v_y^2 + v_z^2}$

Ітерації продовжуються до тих пір, поки не буде досягнуто бажаної точності або поки зміна між послідовними ітераціями не стане нижче заданого порогу.

У реальних системах можна задати поріг зупинки на основі вимог до точності та обчислювальної потужності. Наприклад, у відеоіграх може бути прийнятно використовувати менш точний, але швидший метод нормалізації, оскільки малі помилки в нормалізації векторів зазвичай не впливають критично на візуальну якість.

Метод дозволяє швидко досягти "достатньої" точності. Зменшення кількості ітерацій знижує загальне навантаження на процесор. Ітеративні методи з зупинкою раніше добре підходять для застосувань, де важлива швидкість, а не абсолютна точність.

Ось декілька нових методів, які можуть бути використані для апроксимації нормалізації векторів: використання нейронних мереж для апроксимації нормалізації векторів. моделі можуть бути навчені на великих обсягах даних для швидкого отримання нормалізованих векторів; заміна точних обчислень на швидкі апроксимації, наприклад, використання розширених бітових операцій для швидшого обчислення квадратного кореня; використання адаптивних фільтрів для нормалізації, що дозволяє динамічно змінювати алгоритм в залежності від характеристик вектора; використання гомоморфних функцій для нормалізації векторів, що дозволяє виконувати обчислення без необхідності відновлення вихідних даних; використання просторових структур, таких як kd-дерева або octrees, для зменшення кількості необхідних нормалізацій векторів під час рендерингу; вибір нормалей на основі локальних характеристик поверхні, що дозволяє зменшити кількість векторів, які потребують нормалізації; застосування унітарних матриць для швидкої нормалізації векторів, що зменшує кількість обчислень, необхідних для отримання нормалізованого результату.

Ці методи можуть зменшити обчислювальну складність та підвищити ефективність нормалізації векторів у різних застосуваннях, включаючи комп'ютерну графіку та обробку зображень.

Список використаних джерел

1. Завальнюк Є. К., Романюк О. Н. Методи нормалізації нормалей для зафарбовування поверхонь об'єктів. Integration of Education, Science and Business in Modern Environment: Winter Debates: Proceedings of the 5th International Scientific and Practical Internet Conference. Дніпро, 2024. С. 73-76
2. Романюк О.Н., Дудник О.О., Снігур А.В., Рейда О.М., Романюк О.В. Особливості нормалізації векторів при перспективно-коректному відтворенні кольорів. Наукові праці ДонНТУ. Серія "Інформатика, кібернетика та обчислювальна техніка", № 1(32), 2021, -с.11-17
3. Романюк О. Н. Метод спрощеного визначення векторів для задач рендерингу [Електронний ресурс] / О. Н. Романюк, О. В. Романюк, О. О. Яковенко // Тези доповідей Всеукраїнської науково-практичної Інтернет-конференції студентів, аспірантів та молодих науковців «Молодь в науці: дослідження, проблеми, перспективи» (МН-2020), м. Вінниця, 18-29 травня 2020 р. – Електрон. текст. дані. – 2020.
4. Романюк О.В. Один із апаратних підходів до нормалізації векторів у системах комп'ютерної графіки [Текст] / О. В. Романюк, О. Н. Романюк, Т. М. Павлик // Вимірювальна та обчислювальна техніка в технологічних процесах. – 2010. – № 2. – С. 140-144.
5. Обідник М. Д. Прискорена нормалізація векторів для формування зображень високополігональних сцен [Текст] / М. Д. Обідник, О. Н. Романюк // Інформаційні технології та комп'ютерна інженерія. - 2013. - № 1.

УДК 004.514:004.4:793.7:004.896

**САНДРАЦЬКИЙ Р.В.,
РЕЙДА О. М.,**

Вінницький національний технічний університет

МЕТОДИ ТА ЗАСОБИ КОНТРОЛЯ ДАНИХ "РОЗУМНОГО ГОДИННИКА" ДЛЯ УПРАВЛІННЯ ФІЗИЧНИМИ НАВАНТАЖЕННЯМИ ПІД ЧАС ТРЕНУВАНЬ

Анотація. Розглянуто сучасні підходи до розробки програмної інтерактивної системи аналізу та контролю даних "розумного годинника" для управління фізичними навантаженнями під час тренувань. Основна увага приділяється методам адаптації інтерфейсу та алгоритмів обробки даних, що реагують на фізичний стан і поведінку користувача під час фізичних навантажень. Вплив таких змін на загальний тренувальний досвід розглядається в контексті оптимізації фізичних навантажень, персоналізації тренувань і підвищення мотивації користувачів. Дослідження базується на аналізі існуючих технологічних рішень та пропонує нові підходи до інтеграції розумних годинників для покращення ефективності та безпеки тренувального процесу.

Ключові слова: аналіз даних, контроль даних, розумний годинник, тренування, оптимізація фізичних навантажень, персоналізація тренувань, безпека тренувань.

Abstract. Modern approaches to the development of a software interactive system for analysis and control of "smart watch" data for managing physical loads during training are considered. The main

**ЕЛЕКТРОННІ ІНФОРМАЦІЙНІ РЕСУРСИ:
СТВОРЕННЯ, ВИКОРИСТАННЯ, ДОСТУП ТА УПРАВЛІННЯ**

Збірник матеріалів
Міжнародної науково-практичної Інтернет-конференції
20-21 листопада 2023 р.

Редактор С.А.Пойда, М.С. Ніколаєнко
Комп'ютерне верстання С.А.Пойда, М.С. Ніколаєнко

Підписано до друку 15.11.2024 Гарнітура Times New Roman
Формат 60x84/16 Папір офсетний
Друк цифровий Ум. друк. арк. 12,8
Тираж 300 пр. Зам. № 2/24

Видавництво НІКО
м.Суми, вул.Харківська, 54
Свідоцтво про внесення до Державного реєстру
суб'єктів видавничої справи України
серія СМв № 044
від 15.10.2012
E-mail: ms.niko@i.ua
Телефон для замовлень: +38(066) 270-64-68