

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Одеська національна академія харчових технологій**  
**Університет Інформатики і прикладних знань, м.Лодзь, Польща**  
**Національний технічний університет України «Київський**  
**політехнічний інститут»**  
**Навчально-науковий інститут комп'ютерних систем і технологій**  
**«Індустрія 4.0» ім. П.М. Платонова**

**XXI Всеукраїнська науково-технічна конференція**  
**молодих вчених, аспірантів та студентів**

**«СТАН, ДОСЯГНЕННЯ ТА ПЕРСПЕКТИВИ**  
**ІНФОРМАЦІЙНИХ СИСТЕМ І ТЕХНОЛОГІЙ»**

*Матеріали конференції*



Одеса

22-23 квітня 2021 р.

Стан, досягнення та перспективи інформаційних систем і технологій / Матеріали XXI Всеукраїнської науково-технічної конференції молодих вчених, аспірантів та студентів. Одеса, 22-23 квітня 2021 р. - Одеса, Видавництво ОНАХТ, 2021 р. – 229 с.

Збірник включає матеріали доповідей учасників конференції, які об'єднані за тематичними напрямками конференції.

## ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

Голова - д.т.н., проф., Єгоров Б.В., ректор ОНАХТ.

### Співголови:

**Поварова Н.М.** – к.т.н., доц., проректор з наукової роботи ОНАХТ,  
**Котлик С.В.** – к.т.н., доц., директор ННІКСіТ "Індустрія 4.0" ОНАХТ,  
**Даріуш Долива**, д.математичн.наук, уповноважений декана факультету Інформатики УІтаПЗ, м.Лодзь, Польща,  
**Ковалюк Т.В.** - к.т.н., доц. кафедри АСОІтаУ НТУУ «Київський політехнічний інститут»

### Члени оргкомітету:

**Плотніков В. М.** – д.т.н., проф., завідувач кафедри ІТтаКБ ОНАХТ,  
**Артеменко С.В.** – д.т.н., проф., завідувач кафедри КІ ОНАХТ,  
**Хобін В.А.** – д.т.н., проф., завідувач кафедри АТПтаРС ОНАХТ,  
**Тарасенко В.П.** – д.т.н., проф., завідувач кафедри СКС НТУУ «Київський політехнічний інститут»,  
**Невлюдов І.Ш.** – д.т.н., проф., завідувач кафедри КІТАМ ХНУРЕ,  
**Мельник А.О.** – д.т.н., проф., завідувач кафедри ЕОМ НУ “Львівська політехніка”,  
**Жуков І.А.** – д.т.н., проф., завідувач кафедри КСтаМ НАУ.

Матеріали подано українською, російською та англійською мовами.  
Редактор збірника Котлик С.В.

технологій)	
ПЕРСПЕКТИВИ РОЗРОБКИ ТА ВПРОВАДЖЕННЯ WEB-ДОДАТКІВ. <b>ГАФІЯК А.М., ДЯЧЕНКО-БОГУН А.О., ЧЕПІГА Р.В.</b> (Національний університет «Полтавська політехніка імені Юрія Кондратюка»)	212
КІБЕРСПОРТ ЯК ІНВЕСТИЦІЙНО ПРИВАБЛИВА ГАЛУЗЬ ДЛЯ УКРАЇНИ. <b>ЖМАЙ О.В.</b> (Громадська організація «Молодіжна організація “Енектус” при Одеському національному університеті імені І.І.Мечникова)	214
ГЕНЕРАЦІЯ МЕШУ НА ОСНОВІ ІГРОВОГО РУШІЯ <b>UNITY</b> . <b>КУЛАКОВ В.А., ЖУКОВЕЦЬКА С.Л.</b> (Одеська національна академія харчових технологій)	216
РОЗВИТОК КОМП'ЮТЕРНИХ ІГОР ЗА МОТИВАМИ КНИГ. <b>ПИЛИПЕНКО С.А., СІРЕНКО О.І.</b> (Одеська національна академія харчових технологій)	217
АНАЛІЗ ТРИВИМІРНИХ ДИСПЛЕЇВ. <b>РОМАНЮК О.Н., ДЕДА В.П., ХОШАБА О.М.</b> (Вінницький національний технічний університет)	218
РОЗРОБКА ІГРОВОГО ЕКОНОМІЧНОГО СИМУЛЯТОРА. <b>БОДЮЛ О.С., СІРОМЛЯ Д.С.</b> (Одеська національна академія харчових технологій)	220
PLANING THE OPTIONAL WAY OF MODILE WORK. <b>TROFIMENKO M.S.</b> (National aviation university)	222
АНАЛІЗ ШЕЙДЕРІВ. <b>РОМАНЮК О.Н., ІВАХА О.А., ДУДНИК О.О.</b> (Вінницький національний технічний університет)	223
ОГЛЯД СУЧАСНИХ ТЕХНОЛОГІЙ СТВОРЕННЯ ВЕБ-СТОРИНОК ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ. <b>ЧУБАРОВ Є.Є., ЗИБІНА К.В.</b> (Харківський Національний Університет Радіоелектроніки)	225
ВИКОРИСТАННЯ ФРАКТАЛІВ ДЛЯ КОМПАКТНОГО КОДУВАННЯ ПАТЕРНІВ В ГРАФІЧНОМУ ДИЗАЙНІ. <b>ШЕВЧЕНКО В.В., ШЕВЧЕНКО О.В.</b> (Київський національний університет імені Тараса Шевченка)	227

path from which to the desired vertex is the shortest. It is used to find the shortest path on the map in an unfamiliar area.

As a result of the analysis, it was concluded that when developing a mobile robot for movement and orientation in confined spaces, the IDA \* search algorithm has advantages, and for the development of combat mobile robot - it is better to use the Lee Algorithm - wave tracing.

#### **List of literature sources**

1. K. K. Tomas, R. L. Rivest, and K. F. Shtajn, "Algoritmi: pobudova i analiz," M.: «Vilyams», pp. 296-391, Aug. 2006.

2 Y.G. Martynenko, "Upravlenie dvizheniem mobilnykh kolesnykh robotov," pp. 1590-1594, Sept. 2005.

УДК 004.925

### **АНАЛІЗ ШЕЙДЕРІВ**

РОМАНЮК О. Н., ІВАХА О. А., ДУДНИК О.О.  
Вінницький національний технічний університет

*Комп'ютерна графіка широко використовується в різних галузях діяльності людини, оскільки забезпечує реалістичне відтворення предметів і процесів [1]. При формуванні графічних зображень широко використовуються шейдери.*

Шейдером [1-7] називається програма для візуального визначення поверхні об'єкта. Це може бути опис розсіювання та поглинання світла, накладення текстури.

**Шейдер** – це комп'ютерна програма, яку виконують процесори відеокарти (GPU). Відеокарта обробляє потік даних, які мають точку входу та точку виходу, код команд, які виконують обробку. Під «потокм даних» мається на увазі інформація про координати вершин полігонів і текстурних координат, які передаються в програму шейдера.

Існує три типи рейдерів [2-7]: вершинні (вертексні), геометричні та фрагментні (піксельні). Основна перевага використання шейдерів – їх гнучкість, що спрощує і здешевлює цикл розробки програми, і при тому що підвищує складність і достовірність сцен, що візуалізуються.

**Вертексний (вершинний) шейдер** – це найбільш усталений і найпоширеніший тип серед 3D-шейдерів. Метою таких шейдерів є перетворення 3D-положення кожної вершини у віртуальному просторі до 2D-координати, при якій вона відображається на екрані.

Вершинний процесор може виконувати різні види операцій з графікою: маніпулювання координатами положення кольору або текстури, перетворення вершин і нормалі. Вершинні шейдери забезпечують чіткий контроль над деталями положення, руху, освітлення та кольору.

**Піксельний шейдер**, також відомий як фрагментний, обчислює колір, текстурні координати, глибину та інші параметри кожного фрагмента зображення. Піксельний шейдер використовується на кінцевому етапі графічного конвеєра для формування фрагмента зображення [1].

**Геометричний шейдер** – це шейдер, який здатний обробляти та генерувати нові графічні примітиви, такі як точки, лінії та трикутники. Крім того, геометричний шейдер здатний генерувати примітиви «на льоту», не залучаючи при цьому центрального процесора. Вперше почав використовуватися на відеокартах Nvidia серії 8 [8].

На даний момент існує близько шести різних мов програмування шейдерів, і всі вони так чи інакше транслюють в одні і ті ж машинні коди, які виконуються на GPU [6].

**Шейдерні мови програмування** є C-подібними, але в той же час більш гнучкі в роботі зі змінними. У цілому, мова для шейдерів націлена під конкретне завдання, а саме: вираховування з плаваючою комою, матрицями чи векторами.

Максимальна ефективність і швидкість шейдерної програми залежить від ряду умов :

- програма повинна бути простою та короткою;
- чим менше обчислень, тим швидше буде працювати шейдер;
- чим більш лінійна програма, тим більш оптимізовано буде працювати шейдер.

Типи даних, котрі використовуються при побудові програм шейдерів, діляться на категорії:

- цілочисельні та логічні (bool, int, half);
- з плаваючою комою (float, double);
- вектори (float2, float3, float4, double3, bool2 і т.п., або vector <float, 3>);
- матриці (float2x2, float3x3, float4x4, double2x2, int4x3, або matrix <float, 2, 2> і т. п.).

У шейдерних мовах програмування [5-7] використовуються також арифметичні оператори, але зі своєю специфікою. Наприклад, операція додавання та віднімання по-різному реалізується для float, вектору та матриці. Тим не менш, це не заважає перемножити змінні різних типів: наприклад, вектор можна помножити на число, а матрицю – на вектор.

Шейдерна мова високого рівня HLSL надає розробнику множинну функцій. Ось лише деякі з них, які досить часто застосовують на практиці:

- sin, cos, tan, arctan, arcsin, arccos;
- mul – множить між собою вектори, матриці і числа;
- length (vector) – розраховує довжину вектору;
- distance (a, b) – відстань між двома точками, заданими векторами;
- dot - вираховує dot product;
- lerp - лінійна інтерполяція;
- tex1D, tex2D, tex3D - отримання пікселя з текстурного семплера для одномірної, двомірної та тривимірної текстури [1].

Проведені дослідження показують, що використання шейдерів дозволяє оптимізувати роботу графічних систем.

Підвищення продуктивності та реалістичності можна досягти за рахунок використання нових підходів для формування тривимірних зображень, підвищення функціональних можливостей та продуктивності шейдерних процесорів.

Шляхом апаратної підтримки найбільш поширених інструкцій по реалізації рендеринга можна підвищити продуктивність шейдерних процесорів і зменшити їх кількість. Розширення функціональності шейдерних процесорів дасть можливість реалізувати більш реалістичні зображення, а також зменшить навантаження на центральний процесор і шину обміну даними між системною платою й відеокартою.

#### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- [1] О. Н. Романюк, та А. В. Чорний, *Високопродуктивні методи та засоби зафарбовування тривимірних графічних об'єктів*. Вінниця, Україна: УНІВЕСУМ-Вінниця, 2006.
- [2] О. Н. Романюк, О. О. Дудник, та Н. С. Костюкова. “Реалізація альтернативного конвеєра рендерингу на GPU з використанням обчислювальних шейдерів“, Наукові праці ДонНТУ Серія “Інформатика, кібернетика та обчислювальна техніка”. -№2 (25), с.103-108, 2017.
- [3] О. Н. Романюк “Підвищення ефективності шейдерних моделей графічних відеокарт за рахунок попереднього аналізу спекулярної складової кольору“, Нові технології. - № 2(20). с. 295- 230. 2008.
- [4] О. Н. Романюк “Розподілення обчислювального процесу зафарбовування між вершинними, піксельними і геометричними шейдерами в графічному процесорі”, Праці дев'ятої Міжнародної науково-практичної конференції «Сучасні інформаційні та електронні технології», Одеса, 19–23 травня 2008.с. 23.
- [5] О. Н. Романюк, А. В. Марущак, та В.А Шмалюк. “Шейдери та їх мови програмування”. The 18 th International scientific and practical conference « MODERN SCIENCE, PRACTICE, SOCIETY » (25-26 May 2020). Boston, USA 2020. с. 402-407
- [6] А. В. Боресков. *Разработка и отладка шейдеров*. СПб: БХВ-Петербург, 2006.

[7] “Шейдер, типи шейдерів, шейдерні мови програмування“. [Електронний ресурс]. Доступно:

<https://uk.wikipedia.org/wiki/%D0%A8%D0%B5%D0%B9%D0%B4%D0%B5%D1%80>.

УДК 004.42

## **ОГЛЯД СУЧАСНИХ ТЕХНОЛОГІЙ СТВОРЕННЯ ВЕБ-СТОРИНОК ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ**

ЧУБАРОВ Є. Е., ЗИБІНА К. В.

([yevhen.chubarov@nure.ua](mailto:yevhen.chubarov@nure.ua))

Харківський Національний Університет Радіоелектроніки

*Робота присвячена огляду технологій, які використовуються для забезпечення швидкої і коректної роботи web-сторінок на мобільних пристроях. Розглянуто технології PWA і AMP, їх переваги та недоліки, доцільність їх використання в конкретних ситуаціях.*

Станом на початок 2021-го року Інтернетом користуються 4,66 мільярда людей у всьому світі при тому, що загальна чисельність населення Землі досягла показника в 7,83 мільярда чоловік. Варто відзначити, що мобільними пристроями користуються 5,22 мільярда чоловік, тобто 2/3 світового населення [1]. Поширення смартфонів і технологій Wi-Fi дозволило людям користуватися Інтернетом за допомогою мобільних гаджетів. Це призвело до необхідності розробляти сайти, які могли б адекватно відображатися на невеликих екранах смартфонів або планшетів і були б комфортними у використанні. Також зростають вимоги до швидкості завантаження web-сторінок: за даними дослідників оптимальний час завантаження сторінки становить від 1,5 до 3 секунд. Більш тривале завантаження сторінки призводить до того, що користувачі просто залишають її, не дочекавшись завершення завантаження.

Сьогодні у розробників є великий арсенал засобів для створення веб-сторінок для мобільних пристроїв, наприклад технології PWA – Progressive Web Applications (прогресивні web-додатки) та AMP – Accelerated Mobile Pages (прискорені web-сторінки). При створенні web-сторінок проблемою є правильний вибір технології, який залежить від багатьох факторів, наприклад, від призначення додатка в цілому (інтернет-магазин або портал новин), від необхідності роботи конкретних платформах (стабільна робота в старих браузерах) і т.д.

У даній роботі мова піде про переваги і недоліки кожної з технологій, а також будуть зроблені висновки щодо доцільності використання цих технологій.

Технологія PWA дозволяє візуально перетворити сайт в мобільний додаток в браузері, зберігаючи такий функціонал мобільного прикладного ПО, як відправка push-повідомлень, робота в режимі офлайн завдяки кешуванню за допомогою Service Worker і частковий доступ до апаратного забезпечення. При цьому на робочий стіл смартфона або планшета встановлюється ярлик сайту, який візуально не відрізняється від іконки звичайного мобільного додатку. Також варто відзначити, що додатки PWA можна встановлювати в обхід Play Market або App Store, а передача даних відбувається по протоколу HTTPS, тому даний підхід до розробки є безпечним.

До недоліків PWA можна віднести обмежену підтримку пристроями та браузерами, обмежену роботу офлайн, обмежений доступ до апаратного забезпечення, відсутність можливості взаємодії з вбудованими функціями системи (наприклад, з голосовими помічниками, такими як Siri) і підвищена витрата заряду акумулятора гаджета.

Тепер перейдемо до AMP – технології прискорених мобільних сторінок, яка активно підтримується і просувається компанією Google. AMP дозволяє істотно прискорити завантаження web-сторінок на мобільних пристроях, що безпосередньо впливає на розташування сторінки в пошуковій видачі, тобто чим вище швидкість завантаження, тим вище сторінка у видачі. Варто відзначити, що Google кешує AMP-сторінки, відображаючи AMP-сторінки у вигляді каруселі на сторінці пошукової видачі. Таким чином, компанія,

**XXI Всеукраїнська науково-технічна конференція  
молодих вчених, аспірантів та студентів**

**«СТАН, ДОСЯГНЕННЯ ТА ПЕРСПЕКТИВИ  
ІНФОРМАЦІЙНИХ СИСТЕМ І ТЕХНОЛОГІЙ»**

Одеса

22-23 квітня 2021 р.

Збірник включає доповіді учасників конференції. Тези доповідей публікуються у вигляді, в якому вони були подані авторами.

Відповідальність за зміст і форму подачі матеріалу несуть автори статей.

**Редакційна колегія:** Котлик С.В., Корнієнко Ю.К.

**Комп'ютерний набір і верстка:** Соколова О.П.

**Відповідальний за випуск:** Котлик С.В.