

УДК 004.94+001:372.82

doi:10.20998/2413-4295.2016.42.23

ВІКОРИСТАННЯ RENDERSCRIPT НА ANDROID-ПРИСТРОЯХ

І. А. КУЧЕРЕНКО

Кафедра спеціалізованих комп'ютерних систем (СКС), Українського державного хіміко-технологічного університету, Дніпро, УКРАЇНА.

АНОТАЦІЯ. Сьогодні більшість смартфонів і планшетів виробляються на базі Android. Основними причинами поширення цієї операційної системи полягають в наступному.

Android підтримує велику кількість пристроїв від різних виробників. Android характеризується високою доступністю коштів розробки. Інструментальні засоби розробки для Android безкоштовні, в той час як розробка, наприклад, iPhone (від компанії Apple) вимагає значних початкових інвестицій.

Головні переваги портативних комп'ютерів наступному: розмір, тривалість роботи без підзарядки і, звичайно ж, багатфункціональність. Таким чином, ми повинні визнати, що майбутнє ПК - в портативній, легкій та функціональній техніці - планшети, електронні книги, нетбуки і смартфони – і все це в основному працює на операційній системі Android.

Ключові слова: Android; Renderscript; Java; компілятори; мова програмування; Android

APPLICATION FOR ANDROID-RENDERSRIPT DEVICES

І. А. KUCHERENKO

Department of Specialized Computer Systems (SCS), Chemical Engineering University, Dnipro, UKRAINE

ABSTRACT The article mentions the spread of Android on most portable devices. Among the reasons for this distribution is select, most manufacturers support platform, free distribution tool for development and a large community of supporters of free software. The main advantages of smartphones are size, duration of work without recharging and, of course, multi-functionality. Therefore, we must recognize that the future of PC - a portable, light and functional technology - tablets, e-books, netbooks and smartphones - all mostly working on operating system Android.

Apps for smartphones should work quickly so as not to violate ergonomic interface. Use basic package software developer Android interface does not use additional computing resources that a modern device. Therefore, to perform some operations to the more traditional low-level tools than Java. One solution is RenderScript, allowing better use of hardware capabilities without losing cross-platform.

For the purpose of the study was create by a software module that performs monochromaticism.

image. To perform this operation using different tools: Java and RenderScript. The article contains the results of the research, reflecting a significant increase in speed using a low-level tools for specific tasks.

Keywords: Android; Renderscript; Java; compilers; programming language; Android.

Вступ

За даними International Data Corporation (IDC), міжнародної компанії маркетингових досліджень, консультативних послуг, організатор конференцій в області інформаційних технологій, телекомунікацій і споживчої електроніки, світовий ринок смартфонів виріс на 0.7% у 2016 році. Пристрої під керуванням операційної систем (далі ОС) Android домінують за кількістю з часткою 87,6%. (рис. 1.) Це зростання відбувається в першу чергу за рахунок розвинених ринків, таких як США, Канада, Японія і Західна Європа.

Смартфони (з англ. *Smart* — розумний, і англ. *phone* — телефон) — окрема категорія телефонів, які — на відміну від простих стільникових телефонів — мають більше оперативної пам'яті і власний потужний, як для кишенькових пристроїв процесор, працюють під операційною системою Symbian 6.1 і вище, операційними системами платформи Windows Mobile 5 і вище або Palm OS, операційною системою

iOS, Android, Bada. Завдяки таким даним підтримують багато програм написаних на C++ та java, зокрема 3D-ігри. [1]

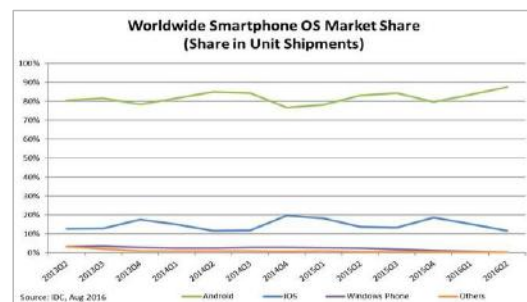


Рис. 1.— Частки пристроїв з різними ОС

Смартфони відрізняються від звичайних телефонів наявністю розвинутою операційною системою, можливістю встановлювати додаткове програмне забезпечення, найчастіше від сторонніх

розробників, для додавання нових можливостей. Розширення функціональності можливо за рахунок J2ME-програм, які підтримуються практично всіма мобільними телефонами, смартфонами і комунікаторами. Екран цілого ряду мобільних телефонів не поступається більшості смартфонів (останнім часом телефони оснащуються і сенсорними екранами), багато моделей оснащені роз'ємом для картки пам'яті. Смартфони в очах користувачів виглядають привабливішими за рахунок багатьох факторів (частково зумовлені просуванням смартфонів і комунікаторів виробниками), таких як: сучасні мультимедійні функції (якісніша камера, розширені можливості відтворення відеофайлів, поліпшені музичні здібності), Wi-Fi, GPS і інших.

Важливо відзначити, що програми, написані спеціально для операційної системи смартфона або комунікатора є повноцінними скомпільованими в двійковий код послідовностями низькорівневих мікропроцесорних команд. А з урахуванням того, що у всіх смартфонах встановлені потужніші процесори, ніж у мобільних телефонах, можливості програміста зі створення таких програм практично не обмежені. Спеціалізовані програми функціональніші й раціональніше використовують ресурси процесора, ніж «універсальні» J2ME-програми. Тому смартфони користуються популярністю серед розробників програмного забезпечення і ентузіастів. [2]

Користуватися смартфонами дуже зручно, адже вони дозволяють будь-де та будь-коли виконувати різноманітні операції: відправляти електронного листа, переглядати новини, створювати документи та навіть редагувати зображення чи відео. Проте розробникам мобільних додатків потрібно пам'ятати: перевагою ОС Android для таких пристроїв є націленість на пристрої з обмеженими ресурсами. Програми повинні виконувати свої функції швидко, не зловживаючи оперативною пам'яттю та запасом акумулятора.

Мета роботи

Метою даної роботи є дослідження компіляторів та мов програмування, які використовуються для створення мобільних додатків для ОС Android. Використовуючи інтегроване середовище розробки Android Studio в роботі створюється модуль, який містить функції, котрі виконують операцію створення монохромного зображення з кольорового та реалізовані засобами мов програмування RenderScript та Java.

Таким чином, проводиться порівняння продуктивності традиційних мов програмування та компілятора з більш низькорівневими засобами, які дають можливості для роботи з апаратним забезпеченням, для редагування зображень.

Виклад основного матеріалу

Основною мовою програмування для ОС Android є Java. Це об'єктно-орієнтована мова програмування, яка компілюється у байт-код, який при виконанні інтерпретується віртуальною машиною для конкретної платформи. Передусім Java розроблялась так, щоб працювати незалежно від цільової платформи, тому вона має менше низькорівневих можливостей для роботи з апаратним забезпеченням, що у певних ситуаціях дало б значний приріст продуктивності. Сучасні смартфони мають графічні та багатоядерні центральні процесори, але пакет програмного забезпечення Android не надає інтерфейсу для їх використання. Існують певні задачі такі як: обробка зображень чи великих масивів даних, що потребують багато обчислювальних ресурсів. За необхідності таких дій Java дозволяє викликати підпрограми, написані іншими мовами програмування, наприклад: C, C++, Assembler, Render Script.

RenderScript - це прикладний програмний інтерфейс (API), який включає функції для 2D/3D рендеринга і математичних обчислень з високою продуктивністю. Він дозволяє описати будь-яку задачу з однотипними незалежними обчисленнями над великим об'ємом даних і розбити її на однорідні підзадачі, які можуть бути виконані швидко і паралельно на багатоядерних Android-платформах [6].

Така технологія може підвищити продуктивність ряду Android-додатків, пов'язаних з обробкою зображень, розпізнаванням образів, фізичним моделюванням, клітинно-автоматної моделлю і ін., котрі, в свою чергу, не втраять апаратної незалежності [6].

RenderScript почав підтримуватися в версії Honeycomb/Android 3.0 (API 11). А саме, в Android SDK в директорії платформи-інструментів llvm-rs-cc (автономний компілятор) для компіляції RenderScript (*.rs файл) в байт-код (*.bc файл) і генерації Java класів об'єктів (*.java файли) для структур та глобальних змінних. Основною llvm-rs-cc є Clang — компілятор з невеликими змінами під Android, який являє собою front-end для LLVM компілятора [7].

В ОС Android з'явився фреймворк, побудований на базі LLVM back-end, який відповідає за компіляцію байт-коду під час виконання, лінковку з потрібними бібліотеками, запуск і контроль виконання RenderScript.(рис.2.) Цей фреймворк складається з наступних частин: libbcc, котрий займається ініціалізацією LLVM контексту відповідно до зазначених метаданих в байт-коді, компіляцією байт-коду і динамічної лінковкою з потрібними бібліотеками з libRS; libRS містить реалізацію бібліотек (math, time, drawing, ref-counting,...), структур і типів (Script, Type, Element, Allocation, Mesh, various matrices,...)(рис.3.) [7].

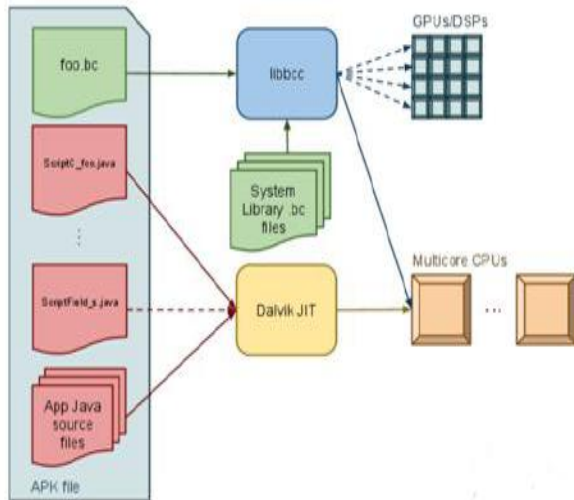


Рис. 2. – Схема back-end компіляції Render Script коду

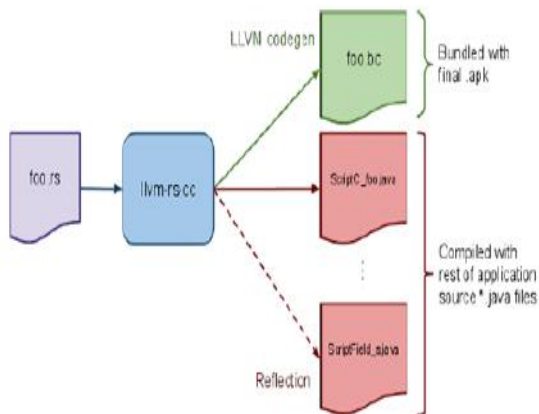


Рис. 3. – Схема front-end компіляції Render Script коду

Обговорення результатів

В роботі створюється монохромне зображення з кольорового. (рис.4.) Для цього було написано дві функції: javaMonoChromeFilter та renderScriptMonoChromeFilter. Перша була реалізована засобами мови програмування Java, а друга — Render Script. Для дослідження використовувався смартфон Meizu Pro 6 з наступними характеристиками:

- Центральний процесор Helio X25: 10 ядер: ARM Cortex-A53 1.4GHz x4 + ARM Cortex-A53 2.0GHz x4 + ARM Cortex-A72 2.5GHz x2
- GPU: Mali-T880
- Hi-Fi ЦАП: Cirrus Logic CS43L36
- Чип NXP Smart PA
- RAM: 4 Гб, LPDDR3



Рис. 4. – Графічний інтерфейс користувача програмного модулю

В таблиці 1 наведено результати тестування Android-додатку при різних розмірах оригінального зображення.

Таблиця 1– Результати тестування програмного модуля

Розмір зображення	960x54	700x39	500x28	300x16
1	1	4	1	9
Render Script	156 мс	98 мс	52 мс	30 мс
Java	1083 мс	672 мс	366 мс	150 мс

З таблиці видно, що функція реалізована засобами Render Script працює швидше, ніж функція написана на Java. Поліпшення продуктивності досягається шляхом виконання нативного коду на пристрої. Однак, на відміну від NDK, це рішення кроссплатформенно. Мова розробки Renderscript - це C99 з деякими розширеннями, під час процесу розробки, він компілюється в апаратно-незалежному проміжному форматі і розміщується в пакеті програми. Слід зауважити, що час роботи Render Script-реалізації включає створення Render Script-контексту, виділення і ініціалізацію необхідної пам'яті, створення і прив'язку Render Script до контексту, а також що продуктивність можна підвищити, відмовившись від

точності арифметичних операцій з числами, що не принципово для даної задачі.

Переваги:

Апаратно-незалежний додаток. Renderscript байт-код, що знаходиться в APK файлі, під час виконання буде скомпільовано в машинний код того апаратно-обчислювального модуля (CPU) платформи, де буде запущений;

Швидкість виконання досягається завдяки паралельним обчисленням;

Недоліки:

Відсутність докладної документації для роботи з Renderscript ускладнює розробку додатків.

Відсутність підтримки виконання Renderscript потоків на GPU, DSP. Можливі проблеми під час виконання з балансуванням потоків в гетерогенному запуску, управлінням спільною пам'яттю.

Висновки

В результаті проведення дослідження виявлено, що використання Render Script дає певні переваги у продуктивності програмного модуля. У роботі були розглянуті ART(dalvik якщо запускати програму на пристрої з версією ОС Android < 5.0) і Render Script реалізації монохромної обробки зображень різних розмірів. За рахунок розпаралелювання, роботи компілятора і нативного виконання коду Render Script значно перевершує Java в продуктивності для зображень середніх розмірів. Даний дослід виявляє випадки, коли його застосування може дати підвищення продуктивності додатків, які при цьому залишаються апаратно-незалежними.

Список літератури

1. **Медник, З.** Программирование под Android / **З. Медник, Л. Дорнин.** – СПб.: Питер. – 2012. – 460 с.
2. <https://uk.wikipedia.org/wiki/%D0%A1%D0%BC%D0%B0%D1%80%D1%82%D1%84%D0%BE%D0%BD>
3. **Казанцев, Р.** Использование RenderScript на android-устройствах с процессорами Intel [Web]. – Назва з титул. екрана.
4. **Марчеті, А.** RenderScript: parallel computing on Android, the easy way: підручник / **А. Марчеті.** – Paperbook. – 2016. – 106 с.
5. **Васильев, А. Н.** Java. Объектно-ориентированное программирование: Учебное пособие / **А. Н. Васильев.** – СПб.: Питер. – 2011. – 400 с.
6. <https://en.wikipedia.org/wiki/RenderScript>
7. <https://developer.android.com/guide/topics/renderscript/compute.html>
8. **Griffiths, D.** HeadFirst Android Development / **D. Griffiths, D. Griffiths.** – 2016. – 704 p.

Відомості про авторів (About authors)

Кучеренко Ігор Андрійович – студент кафедри спеціалізованих комп'ютерних систем (СКС), Українського державного хіміко-технологічного університету. Дніпро (Дніпропетровськ), Україна; e-mail: kuchernkoigor@gmail.com.

Igor Andreevich Kucherenko – student of the Department of Specialized Computer Systems (SCS), Ukrainian State Chemical Technology University. Dnipro (Dnipropetrovsk), Ukraine; e-mail: kuchernkoigor@gmail.com.

9. **Gutierrez, F.** Pro Spring Boot / **Felipe Gutierrez.** – 2016. – 365 p.
10. **Deitel, P.** Android 6 for Programmers: An App-Driven Approach / **P. Deitel, H. Deitel, A. Wald.** – 2015. – 480 p.
11. **Kendal, S.** Object Oriented Programming using Java / **Simon Kendal.** – 2016. – 216 p.
12. **Tanko, M.** A cloud-based Java compiler for smart devices / **M. Tanko, M. Hamada** // *International Conference on Information Technology Based Higher Education and Training (ITHET).* – 2016. – №15. – P. 1–6. – doi: 10.1109/ITHET.2016.7760742.
13. **Golhar, R.** Design and implementation of android base mobile app for an institute / **R. Golhar, P. Vyawahare, P. Borghare** // *IEEE.* – 2016. – №24. – P. 3–5. – doi: 10.1109/ICEEOT.2016.7755391.
14. **Radhakrishnan, J.** Huffman coding and decoding using Android / **J. Radhakrishnan, D. Alluri** // *IEEE.* – 2016. – P. 0361 – 0365. – doi: 10.1109/ICCSP.2016.7754156.

Bibliography (transliterated)

1. **Medniks, Z., Dornin, L.** Programmirovanie pod Android [Programming for Android]. SanktPeterburb: Peter, 2012, 460 p.
2. <https://uk.wikipedia.org/wiki/%D0%A1%D0%BC%D0%B0%D1%80%D1%82%D1%84%D0%BE%D0%BD>
3. **Kazantsev, R.** Ispolzovanie RenderScript na android-ustroystvakh s protsessorami Intel [Using RenderScript on android-devices with Intel processors] [Web]. Hosting Project s title. this function the.
4. **Marcheti, A.** RenderScript: parallel computing on Android, the easy way: pidruchnik. Paperbook, 2016, 106 p.
5. **Vasilev, A. N.** Java. Ob"ektно-orientirovanное programmirovanie: Uchebnoe posobie [Java. Object-oriented programming]. Textbook. SanktPeterburb: Peter, 2011, 400 p.
6. <https://en.wikipedia.org/wiki/RenderScript>
7. <https://developer.android.com/guide/topics/renderscript/compute.html>
8. **Griffiths, D., Griffiths, D.** HeadFirst Android Development, 2016, 704, p.
9. **Gutierrez, F.** Pro Spring Boot, 2016, 365 p.
10. **Deitel, P., Deitel, H., Wald, A.** Android 6 for Programmers: An App-Driven Approach, 2015, 480 p.
11. **Kendal, S.** Object Oriented Programming using Java, 2016, 216 p.
12. **Tanko, M., Hamada, M.** A cloud-based Java compiler for smart devices. *International Conference on Information Technology Based Higher Education and Training (ITHET).* 2016, 15, 1- 6, doi: 10.1109/ITHET.2016.7760742.
13. **Golhar, R., Vyawahare, P., Borghare, P.** Design and implementation of android base mobile app for an institute. *IEEE.* 2016, 24, 3-5, doi: 10.1109/ICEEOT.2016.7755391.
14. **Radhakrishnan, J., Alluri, D.** Huffman coding and decoding using Android. *IEEE.* 2016, 0361 - 0365, doi: 10.1109 / ICCSP.2016.7754156.

Будь ласка посилайтесь на цю статтю наступним чином:

Кучеренко, І. А. Використання renderscript на android-пристроях / **І. А. Кучеренко** // Вісник НТУ «ХПІ», Серія: Нові рішення в сучасних технологіях. – Харків: НТУ «ХПІ». – 2016. – № 42 (1214). – С. 140-144. – doi:10.20998/2413-4295.2016.42.23.

Please cite this article as:

Kucherenko, I. A. Use renderscript on android-devices / **I. A. Kucherenko** // Bulletin of NTU "KhPI". Series: New solutions in modern technologies. – Kharkiv: NTU "KhPI", 2016, **42** (1214), 140–144, doi:10.20998/2413-4295.2016.42.23.

Пожалуйста ссылайтесь на эту статью следующим образом:

Кучеренко, И. А. Использование renderscript на android-устройствах / **И. А. Кучеренко** // Вестник НТУ «ХПИ», Серія: Новые решения в современных технологиях. – Харьков: НТУ «ХПИ». – 2016. – № 42 (1214). – С. 140-144. – doi:10.20998/2413-4295.2016.42.23.

АННОТАЦИЯ. Сегодня большинство смартфонов и планшетов производятся на базе Android. Основными причинами распространения этой операционной системы являются в следующем.

Android поддерживает большое количество устройств от различных производителей. Android характеризуется высокой доступности средств разработки. Инструментальные средства разработки для Android бесплатны, в то время как разработка, например, iPhone (от компании Apple) требует значительных начальных инвестиций.

Главные преимущества портативных компьютеров следующем: размер, продолжительность работы без подзарядки и, конечно же, многофункциональность. Таким образом, мы должны признать, что будущее ПК - в портативной, легкой и функциональной технике - планшеты, электронные книги, нетбуки и смартфоны - и все это в основном работает на операционной системе Android.

Ключевые слова: Android; Renderscript; Java; компиляторы; язык программирования; Android.

Надійшла (received) 14.12.2016