TONY KWOK

Male • August 14th, 1984 • tony_kwok@126.com • +86 134 888 62189

Education

2007.9-2009.4, Dept. of Computer Science, Beijing Institute of Technology, China, M.E.
2003.9-2007.6, Dept. of Computer Science, QUFU Normal University, China, B.E.

Work Experience

#01: June 2011 ~ present, Sony Mobile Communications Inc.

- Camera application new feature development:

Voice-guided self-portrait: according to the face location calculated by face detection engine in the camera viewfinder, guide the user how to move the phone (e.g. left, up, right, and down) via voice and trigger a capture when face locates in a proper position automatically.

Sound photo: capture a much more playful photo with sound embedded in it. The sound data is stored in JPEG file's EXIF APP2 section, not a separating file, so it is much easier to transfer and share among different devices.

Real-time face beauty: apply various effects like Distortion, Vignette, Kaleidoscope, etc. on camera's preview frame buffer (YUV format) before sending it to graphics module to complete rendering and compositing. There are mainly 2 different solutions to achieve this: GPU or CPU. Both of them are implemented, compared and finally, CPU solution was chosen since it is more performant and less power consumption.

- GPU This solution is much power consuming and the device becomes hot quickly because it uses GL Shader to do the YUV-RGB conversion first and then uses OpenGL to render the RGB frame on GLSurfaceView.

- CPU Just need to get the backend graphics buffer of the SurfaceView and then copy the post-processed YUV frame to it directly without color conversion. Since this solution uses CPU to do compositing, it consumes less power than GPU solution, and thermal won't increase quickly.

- Camera performance improvement:

Camera View layout was so complicated that impacted launch speed. By investigating the camera view layout hierarchies, I found that there were so many redundant/unnecessary views: one kind of them could be removed, and another kind of them could be merged, and the rest of them should use lazy loading approach. It's not an easy job since any changes should not cause degrades, but it's worth it. Finally, our camera app launches much more quickly than before even for cold boot case.

#02: July 2009-June 2011, Kingdee Software Co., Ltd.

EAS (Enterprise Application Suite based on Kingdee Business Operating System) is a Client-Server based ERP software. On the client side, the UI is built with Java Swing, we enrich the existing Swing both with extensions as well as brand new ones to meet our special requirement, such as KingdeeTable, F7BizPromptBox and so on; On the server side, we adopt Java EE technologies and our own application server – Apusic, the fourth application

server that passes Java EE 5.0 certification in the world. During this period, I was responsible for Manufacture Module design and development, including Production Line Scheduling, Manufacture Order Split and Reform, and so on.

Projects

#01: Java Bytecode Class File Encryption Based on Java Native Interface

Most Obfuscators only rename the identifiers in java class file. As such, it does nothing but makes the files decompiled by Java Decompilers much more difficult to read. Considering these limits, I overwrite the ClassLoader and use Java Native Interface to convert the byte array into Java class, because the main part is written in C, this approach puts the security on a level comparable to software written in a native language.

#02: Content-Based Image Retrieval System Based on Java

A CBIR library which I implemented in Java based on the following thesis: "Fast Multi-Resolution Image Querying, Charles E. Jacobs, Adam Frinkelstein, David H. Salesin" for Material Management System to help users finding the most similar materials corresponding to the image they provided efficiently..

#03: Expression Evaluation on Android Platform

*A tool for evaluating expressions of type string, e.g. double value = eval("3.14 * (3 ^ 2) - (3 / 2) * 2"); As we all know there is an eval function in JavaScript and Android supports JavaScript as well, but unfortunately, WebView.loadUrl() has no return value, so we used a customized WebChromeClient and got the result through the boolean onJsAlert (WebView view, String url, String message, JsResult result) callback..*

#04: Alloy Look and Feel Improvement and Enhancement

Alloy Look and Feel is a pluggable Java look and feel that can be used by Java developers to enhance the appearance and atmosphere of their Java Swing applications. It is developed by the INCORS® Corporation but they have stopped supporting it since version 1.4.4, I fixed some known bugs and customized it to meet the aesthetic standard of mine.

*#05: Regex4j**

Regex4j is a Regular Expression Engine ported from the System.Text.RegularExpressions module of Microsoft's .Net Core Project for Java platform. As a replacement of the Java built-in Pattern class, with one of its advantages, you can avoid the StackOverflowError issues (JDK-6337993, JDK-6882582, and JDK-8078476), which have existed in the java.util.regex package since Java 1.4 and still not fixed in the latest JDK.

*#06: Primitive**

Primitive is a Java Swing desktop app and open source library for geometrizing images into shapes. Using only geometric primitives like rectangles, triangles, and ellipses, the software recreates images as abstract arrangements of shapes. This project was ported from the popular and excellent work of Michael Fogleman - Reproducing images with geometric primitives. The original implementation is written in Go, I ported it to Java and implemented an antialiasing rasterizer based on the same area/coverage accumulation algorithm as the Freetype "smooth" module and the Anti-Grain Geometry library.

** Visit https://github.com/tonykwok for more details.*

简历

个人简介

- 姓名：郭鹏
- 性别：男
- 出生年月：*1984* 年 *08* 月
- 籍贯：山东省枣庄市
- 学历：北京理工大学计算机科学与技术学院 - 计算机科学与技术 - 硕士研究生
- 联系电话：*13488862189*
- 电子邮件：*tony_kwok@126.com*

工作经历

| *2011* 年 *6* 月 ~ 今 索尼移动通讯产品（中国）有限公司

从事 *Android* 平台 *Camera* 模块的开发维护工作，包括：

- *Camera application* 开发与维护：
  - 基于人脸检测技术（根据当前人脸位置，语音提示用户需要向哪个方向移动手机）实现使用后置摄像头进行自拍；
  - 使用 *JPEG EXIF* 规范中的 *APP2* 数据段保存语音数据，实现可以拍摄包含语音信息的图片的相机应用（类似三星 *GalaxyS4* 中的 *Sound & Shot* 拍照模式）；
  - 实时美妆应用，用户可以从美妆模板列表中选择不同的美妆效果，比如：美白，瘦脸，腮红，睫毛，烟熏等，之后便可以在取景框（*Viewfinder*）中实时预览最终的效果；
- *Camera framework* 功能扩展：
  - 设计并实现 *CameraExtensionService* 模块，其主要用来实现索尼 *Camera* 所特有的功能，比如快速连拍（*BurstShot*），目标跟踪对焦（*Object Tracking*），以及自动场景识别（*Scene Detection*）等；因为 *CameraExtensionService* 是独立于 *CameraService* 运行在自己的进程空间的系统服务，该设计很好的减少了对 *AOSP* 代码的耦合；
  - *Android 8.0 Oreo* 引入了 *Treble* 架构来实现 *System* 与 *Vendor* 的解耦，由于 *CameraExtensionService* 运行在 *System* 分区，因此需要实现 *HIDL*，负责对 *Treble* 架构进行调研并将 *CameraExtensionService* 对底层 *HAL* 的调用 *HIDL* 化；

| *2009* 年 *7* 月 ~ *2011* 年 *6* 月 金蝶软件有限公司

从事基于 *Java EE* 架构和 *Java Swing* 的 *EAS* 企业管理应用套件的开发工作。该软件所使用的应用服务器为金蝶自己研发的 *Apusic*，该应用服务器目前已通过 *SUN* 公司的 *Java EE 5.0* 认证。软件开发环境为 *Kingdee Business Operating System*，*BOS* 是金蝶多年积累的用于快速开发企业级应用程序的开发环境，通过可视化的配置和基于元数据的描述，发布策略实现绝大部分业务对象、业务逻辑（如：增、删、改、查）代码的自动生成。在此期间，我主要负责生产计划层次的开发任务，主要包括：供料计算、生产线日能力计划维护、需求计划维护、生产订单分割改制、物料替代以及相关业务单据的套打、报表设计等工作。

学习与研究

| Java Swing Alloy Look and Feel 研究与改进

Alloy 是 INCORS®公司开发的一款面向 Java Swing 的 Pluggable Look and Feel 产品，无论是在 Windows 平台还是 MAC 平台都有很好的用户体验，比如默认按钮的“脉动（pulse）”效果以及玻璃风格的进度条等。美中不足的是该软件是在基于 Java SE 1.3 版本开发的，目前最新版本为 1.4.4 并且已经停止更新。在对这款软件进行研究与改进的过程中对 Java Swing 体系中的 MVC 结构设计有了更好的认识，同时也对如何利用 Java 2D 绘制漂亮的用户界面，比如：渐变（Gradient）、动画（Animation）等，有了一定的实践经验。

| 基于 JNI 的 Java Bytecode Class 文件加密

目前，对于 Java 软件的加密，通常采用的都是使用混淆软件（Obfuscator）对编译生成的 Class 文件进行混淆，比如开源的 ProGuard（目前该软件已经在 Google 发布 Android 2.3 Gingerbread 的时候，作为一个插件添加到 ADT 中）。混淆降低了的程序的可读性，提高了循环结构的复杂度，比如：混淆后的代码较多地使用了 Break 语句等等，但是使用反编译工具（比如：JAD）还是能够看出个所以然来。使用 JNI 技术结合自定义的 ClassLoader 可以避免这个问题，因为 Class 文件本身已经被加密了所以反编译软件无法识别（一般反编译软件会首先查看所读取的 Class 文件是否符合 Java 规范，即是否以 CAFEBABE 字节码开头），只有在加载该 Class 到 JVM 的时候，自定义的 ClassLoader 才负责解密并将对应的 Byte 数组转化成相应的 Class 对象。

| 基于纹理特征的图像检索系统

该项目的客户是一家从事布料销售的外贸公司，为了能及时快速地查询到目前数据库中是否有某一类花纹的布料，系统需要根据操作员输入的样本图片，对数据库进行查询，返回按相似度有大到小的匹配结果。显然，颜色不是客户关心的重点，图像纹理特征才是图像是否匹配的判定标准，同时考虑到特征提取效率等问题，决定采用相对简单 Haar 小波特征，最终采用 Java 语言并参考网上的一篇论文（Fast Multi-resolution Image Querying）实现了该算法。

| Android 平台上的字符串类型数学运算表达式求值

在 JDK 5.0 版本之后，Java 平台开始支持对脚本语言的支持（比如：JavaScript），因此，在 Java 平台上解决上述问题还是比较简单的，直接调用 JavaScript 中的 eval 方法即可。虽然 Android 平台上已经有用于浏览 Web 页面的 WebView 控件，并且该控件也支持 JavaScript 脚本，但是它在执行 JavaScript 脚本的时候是没有返回值的。最后通过参考 API 文档，使用自定义的 WebChromeClient 解决了该问题。

| Regex4j

Regex4j 是一个改写自 微软 .Net 开源项目，运行在 Java 平台上的正则表达式引擎。该引擎能够避免 JDK 内置 java.util.regex.Pattern 类在处理复杂模式匹配串时所导致的 StackOverflowError 异常（JDK-6337993、JDK-6882582、JDK-8078476），该异常自 JDK 1.4 便存在，并且在最新的 JDK 里面依旧存在。

开源项目

https://www.github.com/tonykwok