Master's thesis

Master's Programme in Computer Science

A Camera API Comparison

Viktor Horsmanheimo

September 6, 2024

Faculty of Science

University of Helsinki

Contact information

P. O. Box 68 (Pietari Kalmin katu 5)
00014 University of Helsinki,Finland

Email address: info@cs.helsinki.fi
URL: http://www.cs.helsinki.fi/

HELSINGIN YLIOPISTO – HELSINGFORS UNIVERSITET – UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty	Koulutusohjelma — Utbildningsprogram — Study programme
Faculty of Science	Master's Programme in Computer Science

Tekijä — Författare — Author

Viktor Horsmanheimo

Työn nimi — Arbetets titel — Title

A Camera API Comparison

Ohjaajat — Handledare — Supervisors

Keijo Helijanko

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
Master's thesis	September 6, 2024	8 pages, 1 appendix pages

Tiivistelmä — Referat — Abstract

Write your abstract here.

In addition, make sure that all the entries in this form are completed.

Finally, specify 1–3 ACM Computing Classification System (CCS) topics, as per https://dl.acm.org/ccs. Each topic is specified with one path, as shown in the example below, and elements of the path separated with an arrow. Emphasis of each element individually can be indicated by the use of bold face for high importance or italics for intermediate level.

ACM Computing Classification System (CCS)
General and reference → Document types → Surveys and overviews
Applied computing → Document management and text processing → Document management → Text editing

Avainsanat — Nyckelord — Keywords

embedded cameras, camera api

Säilytyspaikka — Förvaringsställe — Where deposited

Helsinki University Library

Muita tietoja — övriga uppgifter — Additional information

Software study track

Contents

1 Introduction

Views expressed in this thesis are my own and do not represent anyone else.

This thesis will give an overview of the history of camera API's, where it all began, the current situation and the direction we're going in.

2 History and Background

2.1 Background

This section will introduce the necessary concepts needed for understanding the thesis.

2.1.1 CMOS Sensors

2.2 History

In the early days cameras were quite simple, there effectively was just the "press button" followed by receiving a picture. This process did not allow for much customization. This was the case with most cameras such as webcams where the camera sensors were "smart sensors". This meant that the . Even the integrated laptop ones were just USB devices. Linux had initially had lacking support for this, but in the early 2000s Video For Linux (V4L, later V4L2 for version 2) was developed. This covered most of the use cases at the time.

In 2009 the Nokia N900 was released, this was a Linux based phone that was not like most cameras at the time. It provided interfaces for customizing just about everything. From ISPs to Image Processing Algorithms (IPA's), this meant that the current way camera API's worked was no longer sufficient. Enter FrankencameraAdams et al., 2010; this was an effort at the time to create an API that allowed the user to express the different options that cameras needed.

3 Current API's

3.1 Argus

Argus is a camera API that's developed by Nvidia. Argus runs only on Nvidia hardware, such as the Jetson Orin Nano Developer Kit.

Argus is roughly based on V4L2 though effectively a re-implementation. At the time Argus was created, V4L2 was still in an early stage. It di not supporting a variety of features such as multicamera setups etc. hence Argus was created in order to add the missing features easily. These days V4L2 and Argus are very close though in later chapters we'll give an overview of what pros and cons each has.

The reason Argus doesn't use V4L2 is largely a historical one at this point. At the time Argus was created V4L2 was in an early stage, it had very limited features. Nvidia created Argus in order to leviate these issues but ended up with an API that is very similar to V4L2. Automotive for example needed support for multi camera use cases, this was something that wasn't supported in V4L2 at the time.

Over time, the API has aged a little though. Today it's still very dependent on EGL. EGL is a very dated API, like OpenGL it's largely in maintenance mode today.

3.2 libcamera

libcamera is an open-source C++ embedded camera framework that supports a large number of complex cameras such as the IMX 219, 477 and many more. It supports multiple encoders to receive images in for example PNGs/raw images. The primary target for libcamera is Arm processors in the form of Raspberry PI's, Chrome OS and Android though many other architectures are also supportedPinchart, 2021. Because image processing algorithms are often proprietary and very secret, libcamera also allows for binary blobs to be supplied by the vendors. The vendors do need to submit the "base case" where the camera can take a still image with decent quality to get the blob approved.

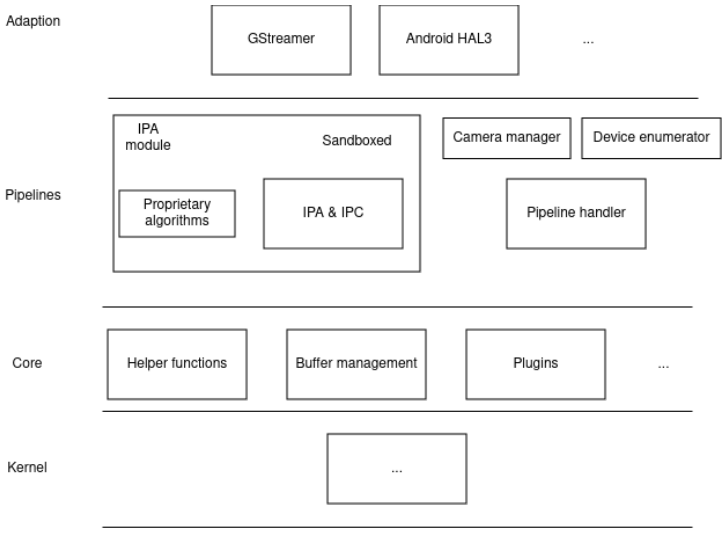

Figure 3.1: High level overview of how the libcamera architecture looks like

3.3 HAL3

Hardware Abstraction Layer 3 (HAL3) is the camera API that's included in each Android phone, it has remarkable features for a mobile phone camera. It was created to bridge the gap between the higher level the API camera2 and the lower level hardware API's. It allows for more modification than camera2, while requiring more work to manage.

3.4 Kamaros

Kamaros is a new API currently (2024) being designed. It's goal is to standardize the embedded camera frameworks. It's under development, not much can be mentioned about the API itself due to a spec not being out yet. It hopes to be easy to use and ability to leverage Vulkan for post processing.

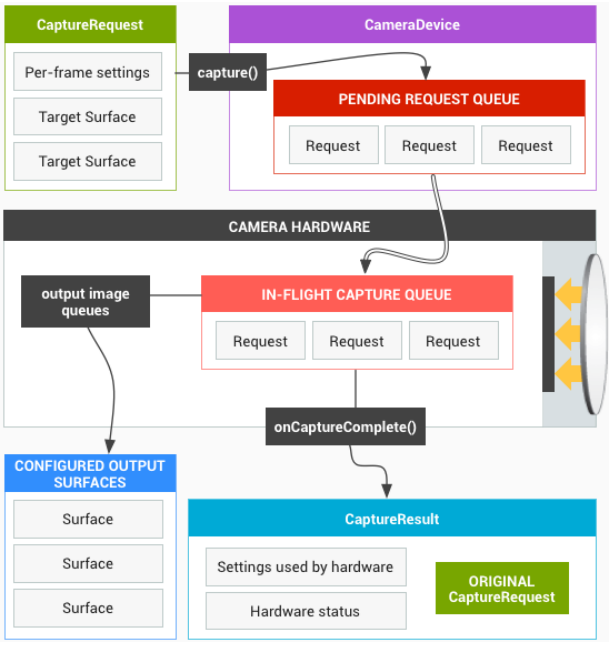

Figure 3.2: HAL3 architecture *Camera Hal* 2024

4 Discussion

5 Conclusions

Bibliography

Adams, A., Talvala, E.-V., Park, S. H., Jacobs, D. E., Ajdin, B., Gelfand, N., Dolson, J., Vaquero, D., Baek, J., Tico, M., et al. (2010). "The Frankencamera: an experimental platform for computational photography". In: *ACM SIGGRAPH 2010 papers*, pp. 1–12.

Camera Hal (Sept. 2024). https://source.android.com/docs/core/camera/camera3.

Pinchart, L. (Nov. 2021). "Giving Linux a Camera Stack: libcamera's 3 Years Journey and Exciting Future". URL: https://www.youtube.com/watch?v=gOusavlAlOE.

Appendix A Sample Appendix

You can add one or more appendices to your thesis.