

Comments (2)DependenciesDuplicates (0)Blocking (0)Resources (0)

InfeasibleBugP4+ Add Hotlist

👤

STATUS UPDATENo update yet.

Edit

📄

DESCRIPTIONma...@gmail.com created issue #1

Feb 27, 2021 03:33AM⋮

Hi!

I've written an image analysis layer to analyze frames using OpenCV. This layer is fully written in C++, and can easily be used from the iOS Camera APIs (Objective-C++ -> C++ is a trivial call).

Now I'm trying to re-use this code-base for Android, and while it technically *works* to use the Image Analysis API like this:

```
imageAnalysis.setAnalyzer(executor, ImageAnalysis.Analyzer { image ->
    // I'm calling the C++ code using JNI from here
})
```

... I'm trying to find out if there is a direct way of calling the C++ code with the camera frame data without using Java. (essentially skipping the "Java" in C++ -> Java -> C++)

In Camera2 I could achieve that fairly easily, since it is much closer to the HAL than CameraX. Since CameraX is using Camera2 under the hood, I assume CameraX also receives the "camera frames" from C++, then dispatches those to Java, and I, in my code above, call C++ again (which is a waste of processing power, instead I want to avoid the Java part).

Is that possible? Could you somehow expose the raw APIs?

Thanks

Reporter

ma...@gmail.com

Type

Bug

Priority

P4

Severity

S4

Status

Won't fix (Infeasible)

Access

Default accessView

Expanded Access?

Assignee

ca...@google.com

Verifier

--

Collaborators

👤

CC

🔔

er...@google.com  
fu...@google.com  
ma...@gmail.com  
tr...@google.com

AOSP ID

--

Estimate

--

InReview

--

Pending

--

Targeted To

--

TF Estimate

--

Found In

--

Targeted To

--

Verified In

--

In Prod

Show 1 additional field

▼

COMMENTS

All comments⌵↓ Oldest first

er...@google.com

<er...@google.com>

#2

Feb 27, 2021 10:06AM⋮

Status: Won't Fix (Infeasible)

Hi There,

Thanks again for filing this feature request.

The team discussed this request and the summary is there are a few assumptions that are not correct, but essentially to answer your questions - this is currently not possible in CameraX and we don't have plans to expose this.

The team also mentioned that while it is an extra cost (C++ -> Java -> C++), it's not too high of an overhead.

I've cc'd a couple folks from the eng team if they have any other details, but unfortunately we cannot accommodate this request. Thanks!