



Infeasible Bug P2 + Add Hotlist [AOSP] assigned adexe s nau

 STATUS UPDATE No update yet. Edit

 DESCRIPTION

lb...@toogoodtogo.com created issue [#1](#)

Nov 15, 2018 07:14PM

Devices/Android versions reproduced on:

122 crashes

86% from Android 5

6% from Android 8

4% from Android 6

2% from Android 7

2% from other versions

42% of crashes come from devices manufactured by Archos.

I've had the most luck reproducing on Android 5.

Stacktrace:

Fatal Exception: androidx.renderscript.RSRuntimeException: Error loading RS jni library: java.lang.UnsatisfiedLinkError: JNI_ERR returned from JNI_OnLoad in "/data/app/com.app.tgtg-1/lib/arm64/librsjni_androidx.so" Support lib API: 2301

at androidx.renderscript.RenderScript.internalCreate(RenderScript.java:1414)

at androidx.renderscript.RenderScript.create(RenderScript.java:1599)

at androidx.renderscript.RenderScript.create(RenderScript.java:1549)

at androidx.renderscript.RenderScript.create(RenderScript.java:1525)

at androidx.renderscript.RenderScript.create(RenderScript.java:1512)

at com.app.tgtg.util.BlurBuilder.blur(BlurBuilder.java:29)

build.gradle:

renderscriptTargetApi 15

renderscriptSupportModeEnabled true

Imports:

import androidx.renderscript.Allocation;

import androidx.renderscript.Element;

import androidx.renderscript.RenderScript;

import androidx.renderscript.ScriptIntrinsicBlur;

Line where it crashes:

RenderScript rs = RenderScript.create(context);

Report

Type

Priorit

Severi

Status

Acces

Expan

Assign

Verifie

Collab

CC

AOSP

Estim:

Found

Target

Verifie


In Pro

Show


✓ Mentioned issues (1)

✓ Links (6)

Hide all

 Mentioned issues (1)

P3 librsjni_androidx.so SIGSEGV "This is likely a dup of [b/133169129](#) . It would be great if you can try the updated librsjni_androidx prebuilt in comment#12 in that thread." mi...@ [#40](#)

 Links (6)

"[https://drive.google.com/file/d/0BxWOVCO-Ab55aHRHbHIJMjVMaGpWaVnLTUF5VC1UX20w...](#)"

lb...@ [#3](#)

"I have test project to reproduce issue: [https://github.com/ernest-p/BlurView](#)"

po...@ [#9](#)

"[https://www.guardsquare.com/en/products/proguard/manual/...](#)"

mu...@ [#16](#)

"[http://proguard-rules.pro](#)"

mu...@ [#16](#)

"[https://github.com/dwi336/Eleven/blob/master/app/src/main/java/org/lineageos/eleven/cache/BI...](#)"

dw...@ [#25](#)


"[http://d.a.Na](#)"

se...@ [#35](#)

COMMENTS


All comments

↓ Oldest first

 ve...@google.com <ve...@google.com>

Assigned to ve...@google.com.

Nov 16, 2018 07:07PM

 ve...@google.com <ve...@google.com> [#2](#)

Nov 16, 2018 10:40PM

Thank you for reporting this issue. For us to further investigate this issue, please provide the following additional information:

Android build

Which Android build are you using? (e.g. OPP1.170223.012)

Device used

Which device did you use to reproduce this issue?

Steps to reproduce

What steps are needed to reproduce this issue?

Please provide sample project or apk to reproduce the issue. Also mention the steps to be followed for reproducing the issue with the given sample project or apk.

Frequency

How frequently does this issue occur? (e.g 100% of the time, 10% of the time)

Expected output

What is the expected output?

Current output

What is the current output?

Android bug report capturing

After reproducing the issue, press the volume up, volume down, and power button simultaneously. This will capture a bug report on your device in the "bug reports" directory. Attach the bug report file to this issue.

Alternate method

After reproducing the issue, navigate to "developer settings", ensure "USB debugging" is enabled, then enable "Bug report shortcut". Capture bug report by holding the power button and selecting the "Take bug report" option.

Screen capture of the issue

Press the volume down and power buttons simultaneously. The image will appear in your gallery. Attach the screenshot file to this issue.

Screen record of the issue, for clarity

Please capture screen record or video of the issue using following steps:

adb shell screenrecord /sdcard/video.mp4

Subsequently use following command to pull the recorded file:

adb pull /sdcard/video.mp4

Attach the file to this issue.

Note: Please upload the files to google drive and share the folder to android-bugreport@google.com, then share the link here.

lb...@toogoodtogo.com <lb...@toogoodtogo.com> #3

Nov 23, 2018 01:24AM

Android build

7.19.401.51 CL482424 release-keys

Device used

HTC One

(There are definitely many more devices this occurs on, but I don't personally have them. It seems to mainly occur on Android 5.

Steps to reproduce

Attempt to use androidx.renderscript.RenderScript.create(context)

Do I really need to provide a sample project for such a simple to reproduce issue?

Frequency

100% of the time (For any given device, it either happens always or never)

Expected output

A RenderScript object should be created.

Current output

Something funky and unpredictable will happen. I became aware of this issue because I was getting stacktraces through crashlytics. On this test device, it does different things (black screen, frozen app, or returning to splash screen). But there's always A/libc: Fatal signal 11 (SIGSEGV), code 1, fault addr 0x0 in tid 22720. On the HTC One at least. There's no segfault in the crashlytics stack traces.

Android bug report capturing

<https://drive.google.com/file/d/0BxWOVCO-Ab55aHRHbHIUMjVMaGpWaVNLtUF5VC1UX20wSHR3/view?usp=sharing>

And some relevant stuff from the logcat

A/libc: Fatal signal 11 (SIGSEGV), code 1, fault addr 0x0 in tid 28262 (com.app.tgtg)

W/libc: Security Level: (1), Debug information is controlled by the DUMPABLE flag.

I/DEBUG: *** **

I/DEBUG: Build fingerprint: 'htc/htc_europe/m7:5.0.2/LRX22G/482424.51:user/release-keys'

I/DEBUG: Revision: '3'

I/DEBUG: ABI: 'arm'

I/DEBUG: pid: 28262, tid: 28262, name: com.app.tgtg >>> com.app.tgtg <<<

I/DEBUG: signal 11 (SIGSEGV), code 1 (SEGV_MAPERR), fault addr 0x0

I/DEBUG: r0 b84786b0 r1 a1252144 r2 00010004 r3 00000001

I/DEBUG: r4 00000043 r5 00000000 r6 00010004 r7 a1245145

I/DEBUG: r8 b8478958 r9 bec26de8 sl bec26dd8 fp b52e5c74

I/DEBUG: ip b846dca0 sp bec26bf8 lr b51ca13b pc a1245156 cpsr 200b0030
























I/DEBUG: backtrace:

I/DEBUG: #00 pc 00000156 /data/app/com.app.tgtg-1/lib/arm/librsjni_androidx.so (JNI_OnLoad+17)










I/DEBUG: #01 pc 001d5139 /system/lib/libart.so (art::JavaVMExt::LoadNativeLibrary(std::__1::basic_string<char, std::__1::char_traits<char>, std::__1::allocator<char> >*)+1580)

I/DEBUG: #02 pc 001fa1e1 /system/lib/libart.so (art::Runtime::nativeLoad(_JNIEnv*, _jclass*, _jstring*, _jobject*, _jstring*)+548)

I/DEBUG: #03 pc 00078861 /data/dalvik-cache/arm/system@framework@boot.oat

	ve...@google.com <ve...@google.com> #4 <p>Could you please confirm, if you have faced similar issue on google devices.</p>	Nov 27, 2018 06:05PM	
	la...@gmail.com <la...@gmail.com> #5 <p>I don't have any google devices running Android 5</p>	Nov 28, 2018 03:31AM	
	lb...@toogoodtogo.com <lb...@toogoodtogo.com> #6 <p>I don't have any google devices running Android 5</p>	Nov 28, 2018 03:32AM	
	lb...@toogoodtogo.com <lb...@toogoodtogo.com> #7 <p>In the project with the issue, I can import androidx.renderscript.RenderScript, but in the sample project I tried to create for you to make it easier to reproduce, you can't. It only lets you import android.renderscript.RenderScript;</p> <p>If I switch the project with the issue to use android.renderscript instead of androidx renderscript, I get an warning from android studio that ScriptIntrinsicBlur.create() requires API 17.</p> <p>I can't figure out why the sample project won't let me import android.renderscript. Both projects use android.support.v4.util.Jetifier android.enableJetifier=true android.useAndroidX=true</p>	Nov 29, 2018 08:49PM	
 Thu Nov 29 2018 10:44:18 GMT+0100 (Central European Standard Time).png 352 KB View Download			
	ve...@google.com <ve...@google.com> #8 <p>We have passed this to the development team and will update this issue with more information as it becomes available.</p>	Nov 30, 2018 06:57PM	
	po...@gmail.com <po...@gmail.com> #9 <p>I'm facing this issue as well on Android Simulator, Android API 19.</p> <p>I have test project to reproduce issue: https://github.com/ernest-p/BlurView Project was working fine till migrated to AndroidX</p>	Dec 1, 2018 05:14PM	
	[Deleted User] <[Deleted User]> #10 <p>We faced with the same problem on Huawei devices:</p> <p>Model Name: P8 Lite RAM: 2048 MB Manufacturer: Huawei Android Version: 5.0</p>	Dec 12, 2018 02:42AM	
	[Deleted User] <[Deleted User]> #11 <p>I encounter this error on my Huawei device.</p> <p>Manufacturer: Huawei Model name: P8 Lite RAM: 2048 MB Android Version: 5.0.1</p>	Dec 12, 2018 03:22AM	
	[Deleted User] <[Deleted User]> #12 <p>deleted</p> <p>Message last modified on Dec 14, 2018 10:57AM</p>	Dec 13, 2018 11:32PM	
	[Deleted User] <[Deleted User]> #13 <p>If someone is interested in as temporary workaround we changed the package from X to old support ones because RenderScript class is still shipped into sdk also. But we would like to have things in normal way. Do you have any information when these issues will be fixed?</p>	Dec 17, 2018 08:25PM	
	lb...@toogoodtogo.com <lb...@toogoodtogo.com> #14 <p>Yeah, and the problem with that workaround is it only works on API 17+. I still need to support API 16.</p>	Dec 19, 2018 06:56PM	

	mu...@gmail.com <mu...@gmail.com> #15	Dec 19, 2018 08:51PM
	<p>I have the same problem. Solved by porting back to standard android lib RenderScript as I don't have problems with only 17+.</p> <p>This pretty easy to reproduce, just try following (Kotlin): `val rs = androidx.renderscript.RenderScript.create(appContext)` on API lvl 22 device and you'll get a crash that you can't even catch with try catch block!</p> <p>Haven't tried with APIs lower than 22, but I'm sure that you'll get similar results.</p>	
	mu...@gmail.com <mu...@gmail.com> #16	Dec 19, 2018 09:01PM
	<p>The original question seems to be more related to pro guard obfuscation of JNI bridge methods. That can be fixed as described there: https://www.guardsquare.com/en/products/proguard/manual/examples#native</p> <p>or just by adding following to proguard-rules.pro file:</p> <pre>-keepclasseswithmembernames,includedescriptorclasses class * { native <methods>; } # RenderScript to be safe on "native callback" side -keep class androidx.renderscript.** { *; }</pre> <p>But then you'll end up with the issue mentioned above.</p>	
	ve...@google.com <ve...@google.com> #17	Dec 20, 2018 08:28PM
	<p>Is this reproducible on a API 21 emulator (x86 and arm)? Please confirm.</p>	
	ve...@google.com <ve...@google.com> #18	Dec 27, 2018 05:14PM
	<p>Please share the details requested in comment #17, to proceed further with the bug.</p>	
	lb...@toogoodtogo.com <lb...@toogoodtogo.com> #19	Dec 27, 2018 07:15PM
	<p>Yes, it is reproducing on the emulator for me. And it is the Segfault variant of the crash.</p>	
	<div>  Thu Dec 27 2018 09:16:08 GMT+0100 (Central European Standard Time).png 123 KB View Download </div>	
	lb...@toogoodtogo.com <lb...@toogoodtogo.com> #20	Dec 27, 2018 07:17PM
	<p>API 21 x86 Android 5.0 (Google APIs)</p>	
	ve...@google.com <ve...@google.com> #21	Dec 27, 2018 09:23PM
	<p>Thanks for the details. Please provide the below information for further investigation.</p> <p>Android bug report capturing After reproducing the issue, press the volume up, volume down, and power button simultaneously. This will capture a bug report on your device in the "bug reports" directory. Attach the bug report file to this issue.</p> <p>Alternate method After reproducing the issue, navigate to "developer settings", ensure "USB debugging" is enabled, then enable "Bug report shortcut". Capture bug report by holding the power button and selecting the "Take bug report" option.</p> <p>Note: Please upload the files to google drive and share the folder to android-bugreport@google.com, then share the link here.</p>	
	lb...@toogoodtogo.com <lb...@toogoodtogo.com> #22	Dec 27, 2018 10:06PM
	<p>I did that already</p>	
	lb...@toogoodtogo.com <lb...@toogoodtogo.com> #23	Dec 27, 2018 10:07PM
	<p>See #3</p>	
	lb...@toogoodtogo.com <lb...@toogoodtogo.com> #24	Dec 27, 2018 10:08PM
	<p>https://issuetracker.google.com/issues/119582492#comment3</p>	

	<p>dw...@gmail.com <dw...@gmail.com> #25</p> <p>I have also this issue. The workaround</p> <p>https://issuetracker.google.com/issues/119582492#comment13</p> <p>works for me:</p> <p>https://github.com/dwi336/Eleven/blob/master/app/src/main/java/org/lineageos/eleven/cache/BlurBitmapWorkerTask.java</p> <p>The bug was reproduced under Emulator API 14, 16, 19, 21, 23, 25, 27, 28 and physical devices with API 14, 19, 25.</p>	<p>Jan 1, 2019 10:44PM ⋮</p>
	<p>[Deleted User] <[Deleted User]> #26</p> <p>Any information about in which month it will be fixed?</p>	<p>Jan 14, 2019 10:49PM ⋮</p>
	<p>an...@gmail.com <an...@gmail.com> #27</p> <p>It's reproducable on LGE Nexus 5. Android 5.0.1. The same with AndroidX</p>	<p>Jan 22, 2019 06:48PM ⋮</p>
	<p>kr...@gmail.com <kr...@gmail.com> #28</p> <p>Hey, ve...@google.com, do you need any additional information to proceed with the fix?</p>	<p>Feb 5, 2019 06:01PM ⋮</p>
	<p>we...@gmail.com <we...@gmail.com> #29</p> <p>Are there any updates on this issue? Because this is blocking me from updating to Android X</p>	<p>Mar 6, 2019 08:52PM ⋮</p>
	<p>sv...@img.ly <sv...@img.ly> #30</p> <p>Hi, I am the lead developer from the PhotoEditor SDK for Android. We and many of our customers are also waiting for a bug fix.</p>	<p>Mar 15, 2019 12:32AM ⋮</p>
	<p>pe...@gmail.com <pe...@gmail.com> #31</p> <p>So i use system Renderscript for api >= 18, and java fast blur for others. also to reduce the apk size.</p>	<p>Mar 15, 2019 01:06PM ⋮</p>
	<p>pa...@outlook.com <pa...@outlook.com> #32</p> <p>Just encountered this bug today on Android 5.1.1 after migrating to AndroidX and running an extensive break point debugging chain until the culprit was found since the log only showed the following:</p> <p>A/libc: Fatal signal 11 (SIGSEGV), code 1, fault addr 0x0 in tid 10270</p> <p>Proguard rules do contain the new package prefix as pointed out by comment #16</p> <p>Reverting to the previous package of Renderscript is not an ideal solution since those packages are no longer receiving updates since September 2018</p> <p>Would appreciate an update regarding this issue whenever is possible.</p>	<p>Mar 21, 2019 05:50AM ⋮</p>
	<p>pe...@gmail.com <pe...@gmail.com> #33</p> <p>Same problem. When migrated to AndroidX started to get some crashes.</p> <p>It happens right at the point of instantiating RenderScript with: <code>RenderScript.create(context)</code></p> <p>Between our test devices we can reproduce it in a: Motorola XT1058 - API 5.1</p> <p>The stacktrace we could pull is next:</p> <pre> 04-24 18:51:21.630 17345-17751/com.testpkg.test W/linker: librsjni_androidx.so: unused DT entry: type 0x60000010 arg 0x35 04-24 18:51:21.630 17345-17751/com.testpkg.test W/linker: librsjni_androidx.so: unused DT entry: type 0x6ffffef5 arg 0xad78 04-24 18:51:21.630 17345-17751/com.testpkg.test W/linker: librsjni_androidx.so: unused DT entry: type 0x6ffffffe arg 0xad38 04-24 18:51:21.630 17345-17751/com.testpkg.test W/linker: librsjni_androidx.so: unused DT entry: type 0x6ffffffe arg 0x2 ----- beginning of crash 04-24 18:51:21.631 17345-17751/com.testpkg.test A/libc: Fatal signal 11 (SIGSEGV), code 1, fault addr 0x0 in tid 17751 (ad_renderscript) 04-24 18:51:21.734 374-374/? I/DEBUG: *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** 04-24 18:51:21.734 374-374/? I/DEBUG: Build fingerprint: 'motorola/ghost_att/ghost:5.1/LPAS23.12-21.7-1/1:user/release-keys' 04-24 18:51:21.734 374-374/? I/DEBUG: Revision: 'p300' 04-24 18:51:21.734 374-374/? I/DEBUG: ABI: 'arm' </pre>	<p>Apr 25, 2019 03:12AM ⋮</p>

04-24 18:51:21.734 374-374/? I/DEBUG: pid: 17345, tid: 17751, name: ad_rendercript >>> com.testpkg.test <<<
04-24 18:51:21.734 374-374/? I/DEBUG: signal 11 (SIGSEGV), code 1 (SEGV_MAPERR), fault addr 0x0
04-24 18:51:21.745 374-374/? I/DEBUG: r0 b489d200 r1 a42ab144 r2 00010004 r3 00000001
04-24 18:51:21.745 374-374/? I/DEBUG: r4 00000043 r5 00000000 r6 00010004 r7 a429e145
04-24 18:51:21.745 374-374/? I/DEBUG: r8 a1e24400 r9 a38e3840 sl a38e3830 fp b47faa6c
04-24 18:51:21.745 374-374/? I/DEBUG: ip b4838218 sp a38e3658 lr b46d5665 pc a429e156 cpsr 200b0030
04-24 18:51:21.745 374-374/? I/DEBUG: backtrace:
04-24 18:51:21.746 374-374/? I/DEBUG: #00 pc 00000156 /data/app/com.testpkg.test-1/lib/arm/librsjni_androidx.so (JNI_OnLoad+17)
04-24 18:51:21.746 374-374/? I/DEBUG: #01 pc 001e1663 /system/lib/libart.so (art::JavaVMExt::LoadNativeLibrary(std::__1::basic_string<char, std::__1::char_traits<char>, std::__1::allocator<char> > const&, art::Handle<art::mirror::ClassLoader>, std::__1::basic_string<char, std::__1::char_traits<char>, std::__1::allocator<char> >*)+1614)
04-24 18:51:21.746 374-374/? I/DEBUG: #02 pc 002081db /system/lib/libart.so (art::Runtime_nativeLoad(_JNIEnv*, _jclass*, _jstring*, _jobject*, _jstring*)+514)
04-24 18:51:21.746 374-374/? I/DEBUG: #03 pc 0007a925 /system/framework/arm/boot.oat
04-24 18:51:22.627 374-374/? I/DEBUG: Tombstone written to: /data/tombstones/tombstone_02
04-24 18:51:22.628 983-1009/? I/BootReceiver: Copying /data/tombstones/tombstone_02 to DropBox (SYSTEM_TOMBSTONE)

[Deleted User] <[Deleted User]> [#34](#)

Apr 27, 2019 04:08AM ⋮

Get the same error :: after migrating to AndroidX

Android 5.1.1 Samsug J2
Android 7.1 Samsung A6

A/libc: Fatal signal 11 (SIGSEGV), code 1, fault addr 0x0 in tid 28262

se...@gmail.com <se...@gmail.com> [#35](#)

May 10, 2019 10:14PM ⋮

Seeing this error too now, for a Samsung Galaxy A7, arm64-v8a, Android 8.0.0,

Error loading RS jni library: java.lang.UnsatisfiedLinkError: JNI_ERR returned from JNI_OnLoad in "/data/app/...=/lib/arm64/librsjni_androidx.so" Support lib API: 2301
at androidx.rendercript.RenderScript.b(:2)
at androidx.rendercript.RenderScript.a(Unknown Source:35)
at androidx.rendercript.RenderScript.a(:1)
at d.a.Na.d(:1)
at d.a.Na\$a.run(:9)

co...@zuricate.net <co...@zuricate.net> [#36](#)

May 14, 2019 05:21AM ⋮

This error on Huawei Y530, armeabi-v7a, Android 4.3:

libc: Fatal signal 11 (SIGSEGV) at 0x00000000 (code=1), thread 14599
#00 pc 00000156 /data/app-lib/testapp/librsjni_androidx.so (JNI_OnLoad+17)
beb92564 575cdb14 /data/app-lib/testapp/librsjni_androidx.so
beb92588 575c3145 /data/app-lib/testapp/librsjni_androidx.so (JNI_OnLoad)

Not a problem on other devices and worked fine with android.support.v8.rendercript, before migrating to androidx.rendercript.

This is blocking our migration to Androidx.

ad...@gmail.com <ad...@gmail.com> [#37](#)

May 14, 2019 03:17PM ⋮

Encountered same issue, app crashing
on line: "RenderScript rs = RenderScript.create(context);"
in device "galaxy J2",
android version "5.1.1"
But working well only on google devices like "pixel 2" android version "9"
Blocking migration to AndroidX in my case too.

se...@gmail.com <se...@gmail.com> [#38](#)

May 17, 2019 09:18PM ⋮

New crash reports from Sky Devices PLATINUM 4.0 (Platinum_4_0), Android 6.0
and Xiaomi MI MAX 2 (oxygen), Android 7.1. Identical stack trace from both devices:

androidx.rendercript.RSRuntimeException:
at androidx.rendercript.RenderScript.internalCreate (RenderScript.java)
at androidx.rendercript.RenderScript.create (RenderScript.java)
or .helpDestroy (RenderScript.java)
or .nAllocationCopyToBitmap (RenderScript.java)
or .nAllocationCreateBitmapBackedAllocation (RenderScript.java)
or .nAllocationCreateTyped (RenderScript.java)
or .nContextCreate (RenderScript.java)
or .nElementCreate (RenderScript.java)
or .nIncAllocationCreateTyped (RenderScript.java)
or .nIncContextCreate (RenderScript.java)
or .nIncTypeCreate (RenderScript.java)
or .nScriptForEach (RenderScript.java)
or .nScriptIntrinsicCreate (RenderScript.java)
or .nScriptSetVarF (RenderScript.java)
or .nScriptSetVarObj (RenderScript.java)
at androidx.rendercript.RenderScript.create (RenderScript.java)
at androidx.rendercript.RenderScript.create (RenderScript.java)
at androidx.rendercript.RenderScript.create (RenderScript.java)

Any news? Recommendations? Should we drop androidx?

Thanks

jo...@gmail.com <jo...@gmail.com> [#39](#)

Jun 17, 2019 09:52PM ⋮

Also having this issue on all android 5 versions using androidx.renderscript.

Crashlytics log:

Crashed: Thread

```
0 librsjni_androidx.so 0xa24f4156 (Missing)
1 libart.so 0xb5029801 (Missing)
2 libc.so 0xb6dc1651 (Missing)
3 libart.so 0xb510a8ce (Missing)
4 libart.so 0xb5109e22 (Missing)
5 libart.so 0xb5109e36 (Missing)
6 libart.so 0xb510a8ce (Missing)
7 libart.so 0xb5109e22 (Missing)
8 libart.so 0xb5109e36 (Missing)
9 libart.so 0xb50d0615 (Missing)
10 libart.so 0xb4f16543 (Missing)
11 libart.so 0xb510a8ce (Missing)
12 libart.so 0xb5109f3a (Missing)
13 libart.so 0xb510a8ce (Missing)
14 libart.so 0xb5109e22 (Missing)
15 (Missing) 0x7421f7be (Missing)
16 libc.so 0xb6dc1651 (Missing)
```

Play developer console log:

```
/lib/arm/librsjni_androidx.so (JNI_OnLoad+17)
#01 pc 00000000001e32e9 /system/lib/libart.so
(_ZN3art9JavaVMExt17LoadNativeLibraryERKNS1_11char_traitsIcEENS1_9allocatorIcEEEEENS1_6HandleINS1_6mirror11ClassLoaderEEEEPS7_+1512)
#02 pc 000000000020bee7 /system/lib/libart.so (_ZN3artL18Runtime_nativeLoadEP7_JNIEnvP7_jclassP8_jstringP8_jobjectS5_+534)
#03 pc 0000000000198c9 /system/framework/arm/boot.oat
```

Any updates on this?

mi...@google.com <mi...@google.com> [#40](#)

Jun 21, 2019 07:40AM ⋮

This is likely a dup of [b/133169129](#). It would be great if you can try the updated librsjni_androidx prebuilt in [comment#12](#) in that thread.

ys...@gmail.com <ys...@gmail.com> [#41](#)

Aug 10, 2019 04:41AM ⋮

Hello any update ? I am having same problem on android 5.1 version

```
androidx.renderscript.RSRuntimeException
com.xxx.manager.BlurTransformation.transform
```

```
RenderScript renderScript = RenderScript.create(context);
```

ad...@google.com <ad...@google.com>

Nov 28, 2019 05:52PM

Reassigned to vk...@google.com.

si...@gmail.com <si...@gmail.com> [#42](#)

Dec 2, 2019 06:14PM ⋮

Same issue on my app while run on 5.0 device.

vi...@google.com <vi...@google.com>

Dec 9, 2019 08:03PM

Reassigned to cc...@google.com.

th...@gmail.com <th...@gmail.com> [#43](#)

Feb 1, 2020 10:58AM ⋮

Please fix this issue because for now RenderScript is unusable or provide workaround as usual...

cc...@google.com <cc...@google.com>

Aug 11, 2020 05:34PM

Reassigned to ni...@google.com.

cc...@google.com <cc...@google.com>

Aug 12, 2020 02:25AM

Reassigned to an...@google.com.



yu...@gmail.com <yu...@gmail.com> [#44](#)

Feb 9, 2021 07:47PM ⋮

please fix issue



pa...@outlook.com <pa...@outlook.com> [#45](#)

May 27, 2021 12:10AM ⋮

It has been left unfixed for 4 years now...



pa...@outlook.com <pa...@outlook.com> [#46](#)

May 27, 2022 03:01AM ⋮

This was left unfixed for so long (5years) the library itself even became deprecated.

Outstanding example of what developers expect most of the times they file a new bug report.

This is an embarrassing commitment from the Android team, and it is a trend plaguing all over these issues.



al...@google.com <al...@google.com>

Feb 9, 2023 05:56AM

Status: Won't Fix (Infeasible)