canvas.drawBitmap will cause native crash

DESCRIPTION wa...@gmail.com created issue #1

I known I am using canvas in an incorrect way, but I think this should not crash in native c++, it should crash by java runtime-exception.

demo in onDraw

```
Thread {
    Thread.sleep(1000)
    (context as Activity).runOnUiThread {
        canvas.drawBitmap(
            bmp,
            Rect(0, 0, bmp.width, bmp.height),
            Rect(
                (elementStart + borderPadding.left).toInt(), 0,
                ((elementStart + borderPadding.left) + textLen).toInt(),
                height
            ),
            borderPaint
        )
    }
}.start()
```

```
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG: Process uptime: 11719s
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG: Cmdline: com.example.hello
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG: pid: 11023, tid: 11023, name: m.example.hello  >>> com.example.hello <<<
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG: uid: 10514
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG: tagged_addr_ctrl: 0000000000000001 (PR_TAGGED_ADDR_ENABLE)
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG: pac_enabled_keys: 000000000000000f (PR_PAC_APIAKEY, PR_PAC_APIBKEY, PR_PAC_APDAKEY, PR_PAC_APDBKEY)
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG: signal 11 (SIGSEGV), code 1 (SEGV_MAPERR), fault addr 0x0000000000000120
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG: Cause: null pointer dereference
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:     x0  0000007c8fbc30e0  x1  0000000000000438  x2  0000000000000026  x3  0000000000000000
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:     x4  0000007c8f8d6d80  x5  0000000000000000  x6  00000000000001b8  x7  0000000000000062
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:     x8  0000000000000000  x9  50b758580f73d9e8  x10 0000000000000200  x11 0000000000003000
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:     x12 0000000000000000  x13 0000007d3565a2fc  x14 0000007d358de284  x15 0000000000000000
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:     x16 0000007d358e5240  x17 0000007d5fef2574  x18 0000007d69ad2000  x19 0000007c2688ec00
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:     x20 0000007c8fbc30e0  x21 0000007c2688ec48  x22 0000000000000438  x23 0000007d68c21000
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:     x24 0000007d68c21000  x25 0000007fe4e3dd98  x26 0000007fe4e3de0c  x27 0000007fe4e3dd98
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:     x28 0000007fe4e3de80  x29 0000007fe4e3db00
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:     lr  0000007d358399c8  sp  0000007fe4e3d940  pc  0000007d358399d8  pst 0000000060001000
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG: backtrace:
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:       #00 pc 00000000002399d8  /system/lib64/libhwui.so (android::uirenderer::skiapipeline::SkiaRecordingCanvas:
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:       #01 pc 0000000000203188  /system/lib64/libhwui.so (android::CanvasJNI::drawBitmapRect(_JNIEnv*, _jobject*,
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:       #02 pc 00000000001c11ec  /system/framework/arm64/boot-framework.oat (art_jni_trampoline+108) (BuildId: 0ae
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:       #03 pc 000000000020a20c  /apex/com.android.art/lib64/libart.so (nterp_helper+3852) (BuildId: 4ecc70344e180
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:       #04 pc 000000000043a398  /system/framework/framework.jar (android.graphics.BaseRecordingCanvas.drawBitmap+
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:       #05 pc 000000000020a254  /apex/com.android.art/lib64/libart.so (nterp_helper+3924) (BuildId: 4ecc70344e180
2023-10-08 14:40:36.633 7977-7977/? A/DEBUG:       #06 pc 0000000000004282  /data/data/com.example.hello/code_cache/.overlay/base.apk/classes3.dex (com.examp
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #07 pc 0000000000209334  /apex/com.android.art/lib64/libart.so (nterp_helper+52) (BuildId: 4ecc70344e180e7
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #08 pc 0000000000003c80  /data/data/com.example.hello/code_cache/.overlay/base.apk/classes3.dex (com.examp
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #09 pc 0000000000209334  /apex/com.android.art/lib64/libart.so (nterp_helper+52) (BuildId: 4ecc70344e180e7
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #10 pc 000000000000315c  /data/data/com.example.hello/code_cache/.overlay/base.apk/classes3.dex (com.examp
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #11 pc 000000000020b074  /apex/com.android.art/lib64/libart.so (nterp_helper+7540) (BuildId: 4ecc70344e180
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #12 pc 00000000001cae14  /system/framework/framework.jar (android.os.Handler.handleCallback+4)
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #13 pc 0000000000209334  /apex/com.android.art/lib64/libart.so (nterp_helper+52) (BuildId: 4ecc70344e180e7
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #14 pc 00000000001cac60  /system/framework/framework.jar (android.os.Handler.dispatchMessage+8)
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #15 pc 000000000020a254  /apex/com.android.art/lib64/libart.so (nterp_helper+3924) (BuildId: 4ecc70344e180
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #16 pc 00000000001ee1f0  /system/framework/framework.jar (android.os.Looper.loopOnce+364)
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #17 pc 0000000000209334  /apex/com.android.art/lib64/libart.so (nterp_helper+52) (BuildId: 4ecc70344e180e7
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #18 pc 00000000001ee918  /system/framework/framework.jar (android.os.Looper.loop+164)
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #19 pc 0000000000209334  /apex/com.android.art/lib64/libart.so (nterp_helper+52) (BuildId: 4ecc70344e180e7
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #20 pc 00000000001c8806  /system/framework/framework.jar (android.app.ActivityThread.main+246)
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #21 pc 000000000210c00  /apex/com.android.art/lib64/libart.so (art_quick_invoke_static_stub+576) (BuildId
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #22 pc 000000000027b4ac  /apex/com.android.art/lib64/libart.so (art::ArtMethod::Invoke(art::Thread*, unsig
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #23 pc 0000000000616f84  /apex/com.android.art/lib64/libart.so (_jobject* art::InvokeMethod<(art::PointerS
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #24 pc 0000000000596aa0  /apex/com.android.art/lib64/libart.so (art::Method_invoke(_JNIEnv*, _jobject*, _j
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #25 pc 0000000000099148  /system/framework/arm64/boot.oat (art_jni_trampoline+120) (BuildId: 8866718553fbb
```

```
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #26 pc 0000000000020a2b0  /apex/com.android.art/lib64/libart.so (nterp_helper+4016) (BuildId: 4ecc70344e180
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #27 pc 00000000004f7a36  /system/framework/framework.jar (com.android.internal.os.RuntimeInit$MethodAndArg
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #28 pc 00000000007384fc  /system/framework/arm64/boot-framework.oat (com.android.internal.os.ZygoteInit.ma
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #29 pc 0000000000210c00  /apex/com.android.art/lib64/libart.so (art_quick_invoke_static_stub+576) (BuildId
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #30 pc 0000000000027b4ac  /apex/com.android.art/lib64/libart.so (art::ArtMethod::Invoke(art::Thread*, unsig
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #31 pc 000000000061770c  /apex/com.android.art/lib64/libart.so (art::JValue art::InvokeWithVarArgs<art::Ar
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #32 pc 0000000000617bf8  /apex/com.android.art/lib64/libart.so (art::JValue art::InvokeWithVarArgs<_jmetho
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #33 pc 00000000004ff01c  /apex/com.android.art/lib64/libart.so (art::JNI<true>::CallStaticVoidMethodV(_JNI
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #34 pc 00000000000c0c04  /system/lib64/libandroid_runtime.so (_JNIEnv::CallStaticVoidMethod(_jclass*, _jme
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #35 pc 00000000000cd80c  /system/lib64/libandroid_runtime.so (android::AndroidRuntime::start(char const*,
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #36 pc 0000000000002610  /system/bin/app_process64 (main+1464) (BuildId: 6363373b68588b10c93602a28a32bb57)
2023-10-08 14:40:36.634 7977-7977/? A/DEBUG:       #37 pc 00000000000859b8  /apex/com.android.runtime/lib64/bionic/libc.so (__libc_init+100) (BuildId: 449f78
2023-10-08 14:40:36.635 7780-27971/com.ss.android.ugc.aweme E/Timon-InspectorEntry: ApiWatchdogSystem: traceInfo_id: 110000
```

COMMENTS

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

Assigned to an...@google.com.

---

jr...@google.com <jr...@google.com> #2

Status: Won't Fix (Infeasible)

Canvas' are recycled, so it cannot be feasibly converted to an exception