
The OpenCV Manager Manual

Release 2.4.6.1

December 31, 2014

CONTENTS

1	Android OpenCV Manager	1
1.1	Introduction	1
1.2	Manager Workflow	3
1.3	Java OpenCV Loader	5
1.4	Base Loader Callback Interface Implementation	6
1.5	Loader Callback Interface	8
1.6	Install Callback Interface	9
2	Java API	11

ANDROID OPENCV MANAGER

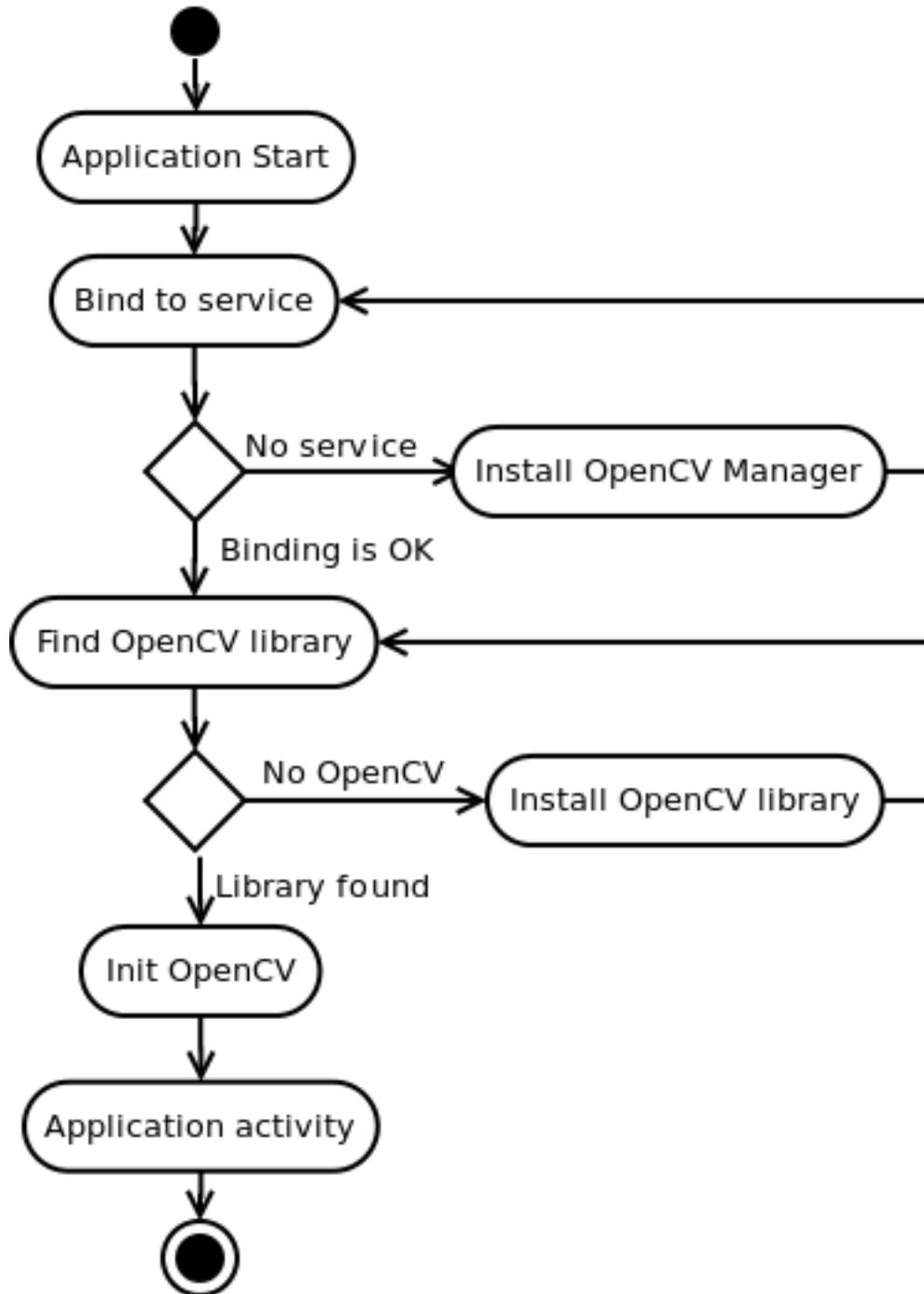
Contents:

1.1 Introduction

OpenCV Manager is an Android service targeted to manage OpenCV library binaries on end users devices. It allows sharing the OpenCV dynamic libraries between applications on the same device. The Manager provides the following benefits:

1. Less memory usage. All apps use the same binaries from service and do not keep native libs inside themselves;
2. Hardware specific optimizations for all supported platforms;
3. Trusted OpenCV library source. All packages with OpenCV are published on Google Play market;
4. Regular updates and bug fixes;

Usage model for end user



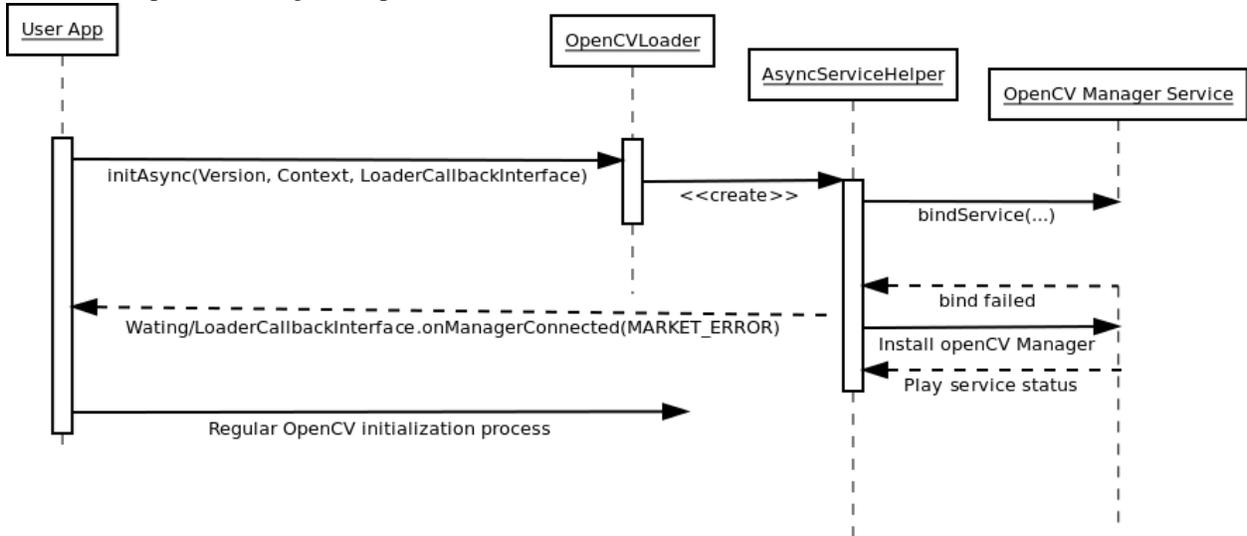
First OpenCV app:

1. Any OpenCV-dependent app is installed from Google Play marketplace or manually;
2. At the first launch, it suggests installation of OpenCV Manager;
3. Then OpenCV Manager is downloaded and installed, using the Google Play application.

Hardware Platform	Android ver.	Package name
armeabi-v7a (ARMv7-A + NEON)	>= 2.3	OpenCV_2.4.6_Manager_2.9_armv7a-neon.apk
armeabi-v7a (ARMv7-A + NEON)	= 2.2	OpenCV_2.4.6_Manager_2.9_armv7a-neon-android8.apk
armeabi (ARMv5, ARMv6)	>= 2.3	OpenCV_2.4.6_Manager_2.9_armeabi.apk
Intel x86	>= 2.3	OpenCV_2.4.6_Manager_2.9_x86.apk
MIPS	>= 2.3	OpenCV_2.4.6_Manager_2.9_mips.apk

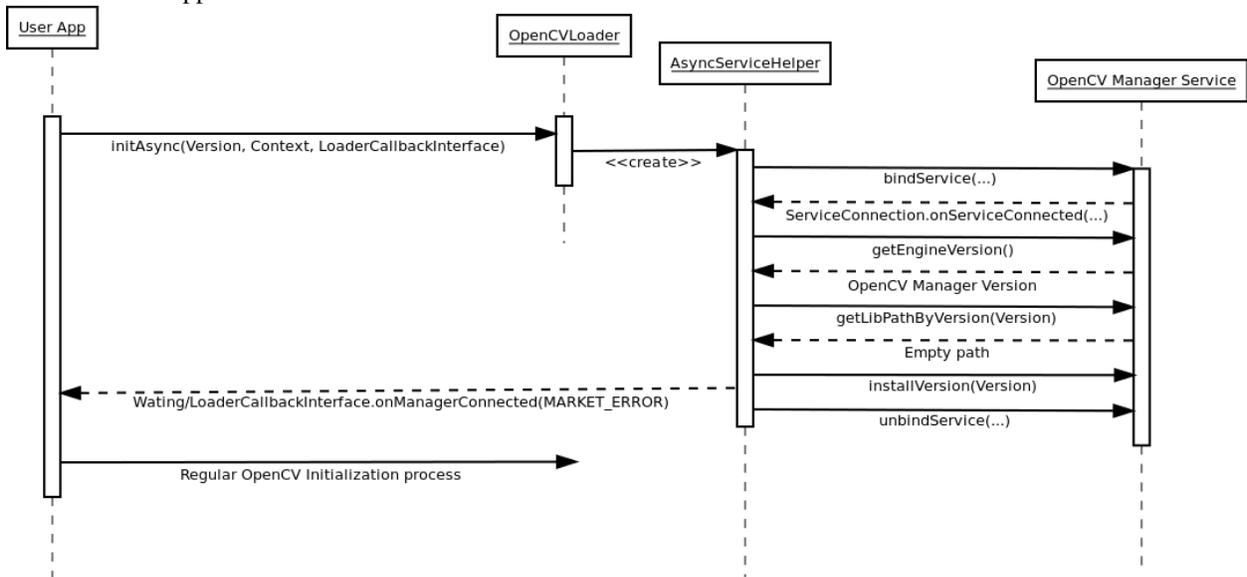
First application start

There is no OpenCV Manager or OpenCV libraries:

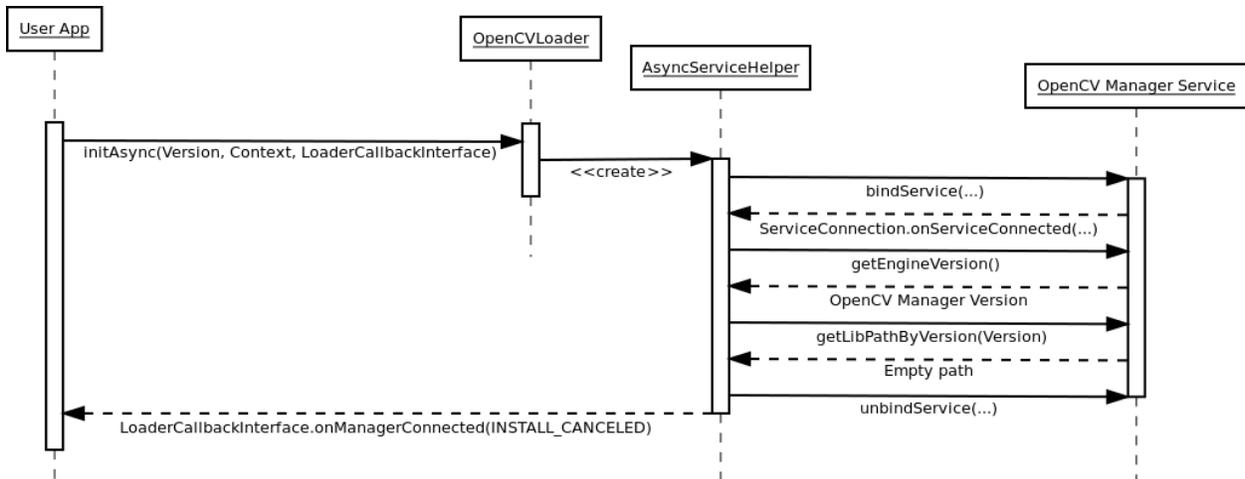


Additional library package installation

There is an OpenCV Manager service, but it does not contain appropriate OpenCV library. If OpenCV library installation has been approved:

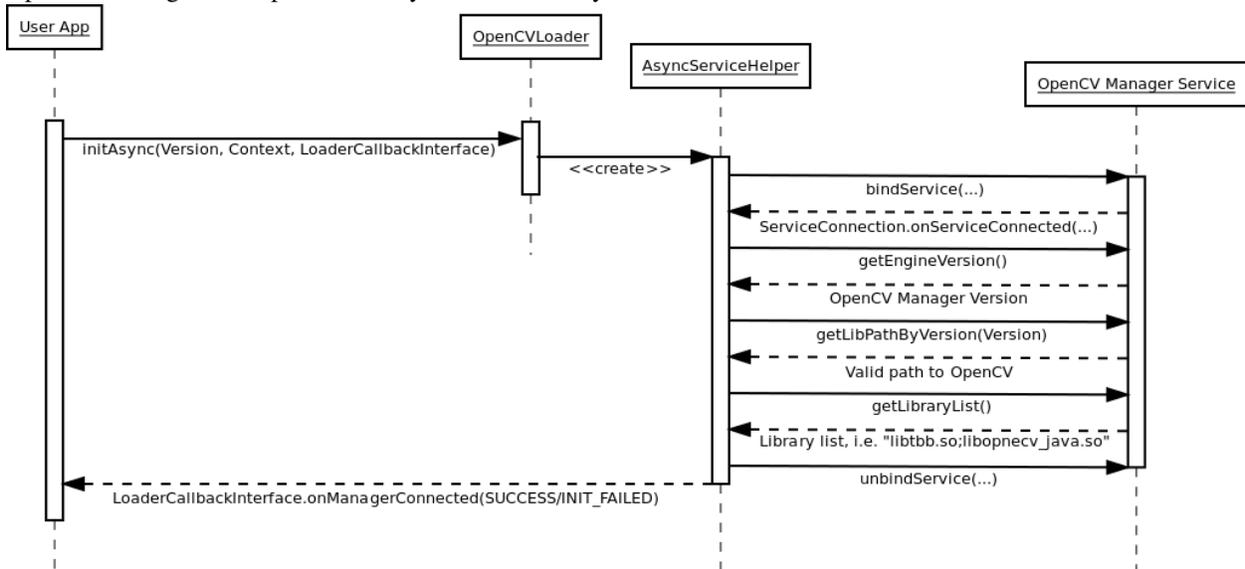


If OpenCV library installation has been cancelled:



Regular application start

OpenCV Manager and OpenCV library has been already installed.



1.3 Java OpenCV Loader

class `OpenCVLoader`

Helper class provides common initialization methods for OpenCV library.

boolean `initDebug()`

static boolean `initDebug()`

Loads and initializes OpenCV library from within current application package. Roughly it is analog of `system.loadLibrary("opencv_java")`.

Return type boolean;

Returns returns true if initialization of OpenCV was successful.

Note: This method is deprecated for production code. It is designed for experimental and local development purposes only. If you want to publish your app use approach with async initialization.

boolean initAsync()

static boolean initAsync(String Version, Context AppContext, LoaderCallbackInterface Callback)

Loads and initializes OpenCV library using OpenCV Manager.

Parameters

Version – OpenCV Library version.

AppContext – application context for connecting to the service.

Callback – object, that implements LoaderCallbackInterface for handling connection status (see BaseLoaderCallback).

Return type boolean;

Returns returns true if initialization of OpenCV starts successfully.

OpenCV version constants

OPENCV_VERSION_2_4_2

OpenCV Library version 2.4.2

OPENCV_VERSION_2_4_3

OpenCV Library version 2.4.3

OPENCV_VERSION_2_4_4

OpenCV Library version 2.4.4

OPENCV_VERSION_2_4_5

OpenCV Library version 2.4.5

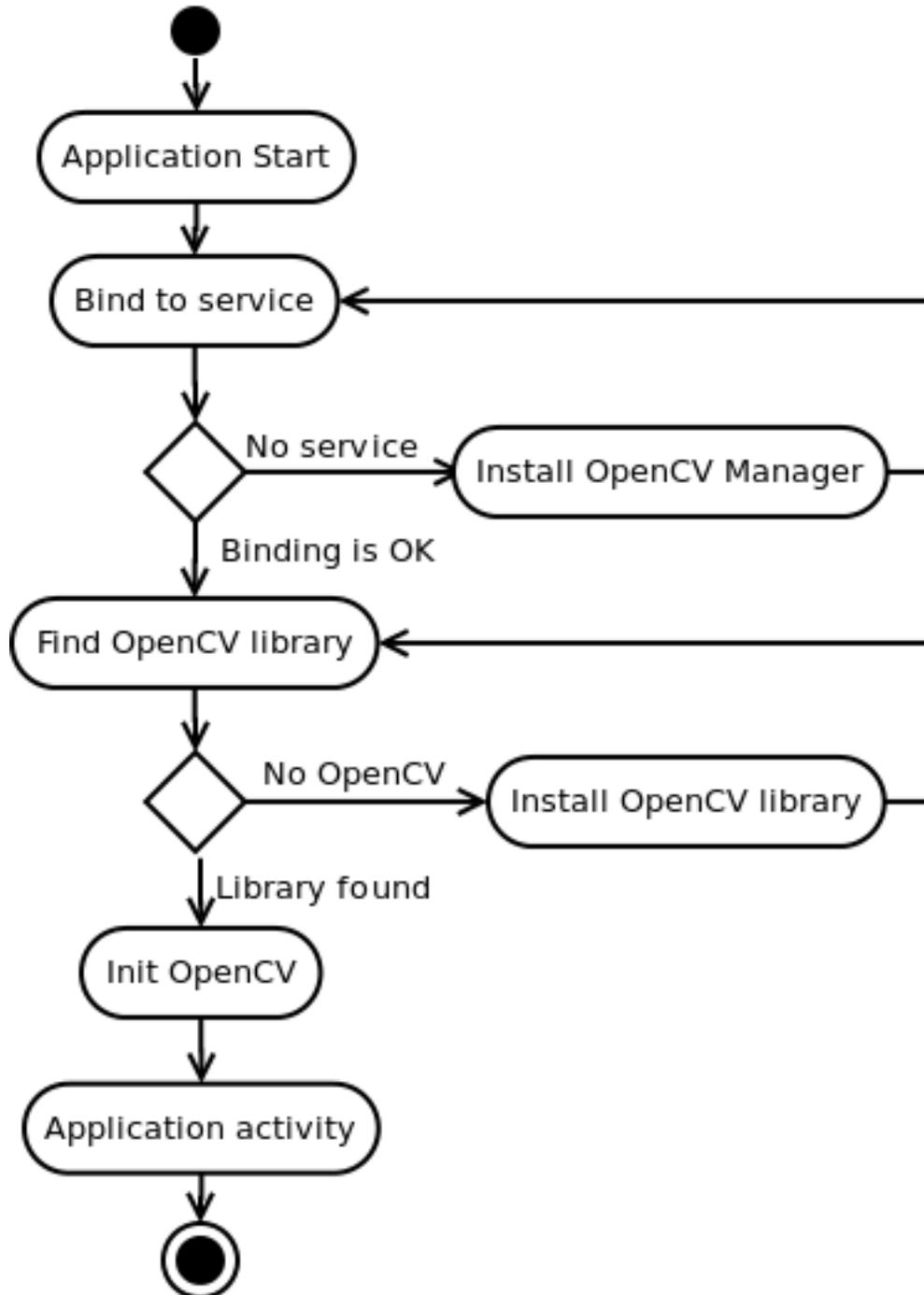
OPENCV_VERSION_2_4_6

OpenCV Library version 2.4.6

1.4 Base Loader Callback Interface Implementation

class BaseLoaderCallback

Basic implementation of LoaderCallbackInterface. Logic of this implementation is well-described by the following scheme:



Using in Java Activity

There is a very base code snippet implementing the async initialization with `BaseLoaderCallback`. See the “15-puzzle” OpenCV sample for details.

```

1 public class MyActivity extends Activity implements HelperCallbackInterface
2 {
3 private BaseLoaderCallback mOpenCVCallBack = new BaseLoaderCallback(this) {

```

```
4  @Override
5  public void onManagerConnected(int status) {
6      switch (status) {
7          case LoaderCallbackInterface.SUCCESS:
8              {
9                  Log.i(TAG, "OpenCV loaded successfully");
10                 // Create and set View
11                 mView = new puzzle15View(mAppContext);
12                 setContentView(mView);
13             } break;
14             default:
15                 {
16                     super.onManagerConnected(status);
17                 } break;
18         }
19     }
20 };
21
22 /** Call on every application resume */
23 @Override
24 protected void onResume()
25 {
26     Log.i(TAG, "Called onResume");
27     super.onResume();
28
29     Log.i(TAG, "Trying to load OpenCV library");
30     if (!OpenCVLoader.initAsync(OpenCVLoader.OPENCV_VERSION_2_4_6, this, mOpenCVCallBack))
31     {
32         Log.e(TAG, "Cannot connect to OpenCV Manager");
33     }
34 }
```

Using in Service

Default `BaseLoaderCallback` implementation treats application context as `Activity` and calls `Activity.finish()` method to exit in case of initialization failure. To override this behavior you need to override `finish()` method of `BaseLoaderCallback` class and implement your own finalization method.

1.5 Loader Callback Interface

class `LoaderCallbackInterface`

Interface for a callback object in case of asynchronous initialization of OpenCV.

void `onManagerConnected()`

void `onManagerConnected(int status)`

Callback method that is called after OpenCV library initialization.

Parameters

status – status of initialization (see “Initialization Status Constants” section below).

void onPackageInstall()

void onPackageInstall(InstallCallbackInterface Callback)

Callback method that is called in case when package installation is needed.

Parameters

callback – answer object with `install` and `cancel` methods and package description.

Initialization status constants

SUCCESS

OpenCV initialization finished successfully

MARKET_ERROR

Google Play (Android Market) application cannot be invoked

INSTALL_CANCELED

OpenCV library installation was cancelled by user

INCOMPATIBLE_MANAGER_VERSION

Version of OpenCV Manager is incompatible with this app. Manager update is needed.

INIT_FAILED

OpenCV library initialization failed

1.6 Install Callback Interface

class InstallCallbackInterface

Callback interface for package installation or update.

String getPackageName()

String getPackageName()

Get name of a package to be installed.

Return type string;

Returns returns package name, i.e. “OpenCV Manager Service” or “OpenCV library”.

void install()

void install()

Installation of package has been approved.

void cancel()

void cancel()

Installation of package has been cancelled.

void wait_install()

void wait_install()

Wait for package installation.

JAVA API

Java API reference (JavaDoc): external [link](#).