

# OpenCV for Unity 2.0.7

iOS & Android support

WindowsStoreApps8.1 & WindowsPhone8.1 & Windows10 UWP(beta) support

Win & Mac & Linux Standalone support

Support for preview in the Editor

Work with Unity Free & Pro

## System Requirements

Build Win Standalone & Preview Editor : Windows7 or later

Build Mac Standalone & Preview Editor : OSX 10.8 or later

**OpenCV for Unity** is an Assets Plugin for using **OpenCV** from within **Unity**.

- Since this package is a **clone of OpenCV Java**, you are able to use the same API as OpenCV Java 3.1.0.
- You can image processing in **real-time** by using the **WebCamTexture** capabilities of Unity. (**real-time face detection works smoothly in iPhone 5**)
- Provides a method to interconversion of **Unity's Texture2D** and **OpenCV's Mat**.
- **IDisposable** is implemented in many classes. You can manage the resources with the “**using**” statement.

Please download [Demo Application](#) for Android and watch [Setup Tutorial Video](#).

## [API Reference OpenCV for Unity](#)

Please refer to [OpenCV official document](#) for the details of the argument of the method.

SampleCode using OpenCV for Unity is available.

- [MarkerBased AR Sample](#)
- [FaceTracker Sample](#)
- [Vuforia with OpenCV for Unity Sample](#)
- [Kinect with OpenCV for Unity Sample](#)

## Version changes

**2.0.7** [Common]Added aruco, structured\_light, xfeatures2d module. [Common]Added ArUcoSample, GrabCutSample, InpaintSample, MatchShapesSample, MSERSample.

**2.0.6** [WSA]Fixed an issue where Windows App Certification Kit fails.

**2.0.5** [Common]Added HOGDescriptorSample.

**2.0.4** [Android]Added Support for Split Application Binary (.OBB) [Android]Removed opencvforunity.jar.

**2.0.3** [Common]Added SVMSample. [Common]Fixed VideoCaptureSample and WebCamTextureAsyncDetectFaceSample. [UWP]Added

OpenCVForUnityUWP\_Beta2.zip.

**2.0.2** [Common]Fixed CS0618 warnings: 'UnityEngine.Application.LoadLevel(string)' is obsolete: 'Use SceneManager.LoadScene'.

**2.0.1** [OSX]Fixed SIGILL Exception. [Common]Added Utils.setDebugMode() method. [Common]Added MatchTemplateSample, StereoBMSample, SeamlessCloneSample and WebCamTextureDetectCirclesSample. [Common]Added flipVertical flag, flapHorizontal flag and GetWebCamDevice() method to WebCamTextureToMatHelper.cs.

**2.0.0** [Common]Updated to OpenCV3.1.0. [Common]Included Old Version based on "OpenCV2.4.11". [Common] Included Beta Version of Windows10 UWP Support.( This is beta version based on OpenCV3.0.0. opencv\_contrib modules is not supported.)

**Beta16** [iOS]Fixed libopencvforunity.a Bitcode Setting.

**Beta15** [Common]Fixed WebCamTextureToMatHelper.cs.(Add didUpdateThisFrame () method)

**Beta14** [Common]Fixed WebCamTextureToMatHelper.cs.( Bug of rotation conversion from WebCamTexture to Mat in Win,Mac StandAlone Build)

**Beta13** [Common]Added fastTexture2DToMat() and fastMatToTexture2D(). [Common] Renewed the samples using WebCamTextureToMatHelper.(Supports all screen orientation.)

**Beta12** [iOS]Fixed malloc\_error that occurs in Unity5.3.1p2.

**Beta11** [iOS]Enabled Jpeg format.(Added mjpeg format support in VideoCapture class)

**Beta10** [iOS]Enabled Bitcode.

**Beta9** [UWP]Added support for Windows10 UWP.( This is a test version. opencv\_contrib modules is not supported.)

**Beta8** [Common]Fix FaceRecognizerSample. [Common] Delete the method using Default parameter specifiers. [Android] Compile the library using "armabi-v7a with NEON" option.

**Beta7** [Common]Add WrapPerspectiveSample, HandPoseEstimationSample.

**Beta6** [iOS]Fix WebCamTexture bug of SampleScene in Unity5.2.

**Beta5** [Linux]Add Linux Support. [WindowsStoreApp8.1]Support for methods using Low-level Native Plugin Interface. [Common]Rewrite SampleScene.

**Beta4** [Common]Add Utils. getGraphicsDeviceType(). [Common]Add SampleScene Setup Tutorial Video for Unity5.

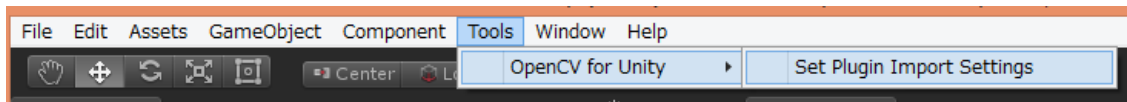
**Beta3** [Common]Add CamShiftSample.(Object Tracking) [Common]Add OpenCVForUnityMenuItem.cs.( This script set plugin import settings automatically from MenuItem.)

**Beta2** [iOS] Fix problem when working with Metaio(UnityAppController problem). [Common]Add [System.Serializable] to basic class. [Common] change folder name from “OpenCVForUnity/OpenCVForUnity\_Editor/” to “OpenCVForUnity/Editor/”. [iOS]Move “OpenCVForUnity/OpenCVForUnity\_Editor/opencv2.framework” to “OpenCVForUnity/Plugins/iOS”folder.

If you want to try Windows10 UWP Platform(Beta Version), Please unzip “OpenCVForUnityUWP\_Beta.zip”, create new Project, import “uwp\_beta.unitypackage”.

**Quick setup procedure to run the sample scene**([Setup Tutorial Video](#))

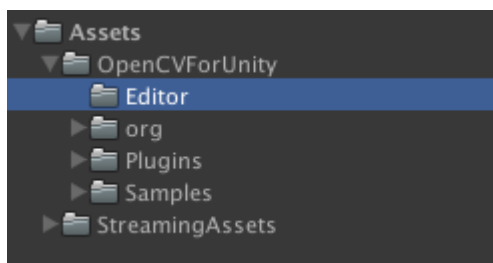
1. Select MenuItem[Tools/OpenCV for Unity/Set Plugin Import Settings].



2. Move “OpenCVForUnity/StreamingAssets/” folder to “Assets/” folder.
3. Please set [PlayerSettings]-[Resolution and Presentation]-[Orientation]-[Default Orientation : Landscape Left] when you build the sample scene.
4. Add all of the “\*\*\*.unity” in the “OpenCVForUnity” folder to [Build Settings] – [Scene In Build].

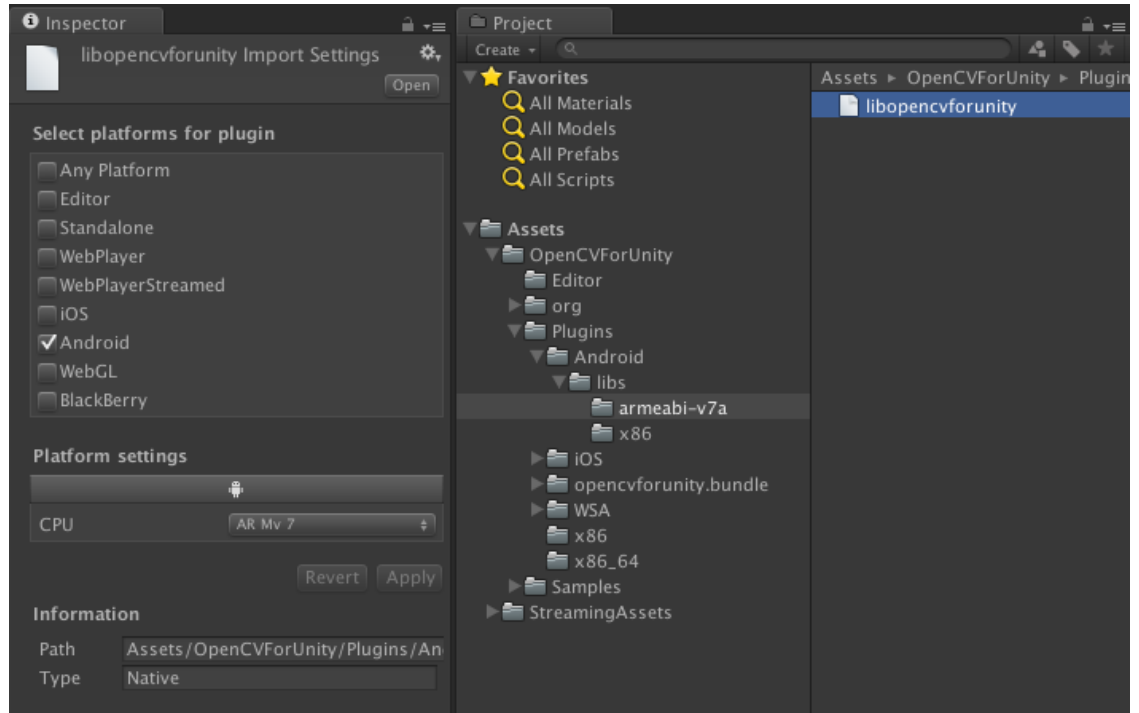
※Inspector Setting of “opencv2.framework” and “opencvforunity.bundle” might have been reset at the time of import. In that case, re-setup is required.

**Screenshot after the setup**

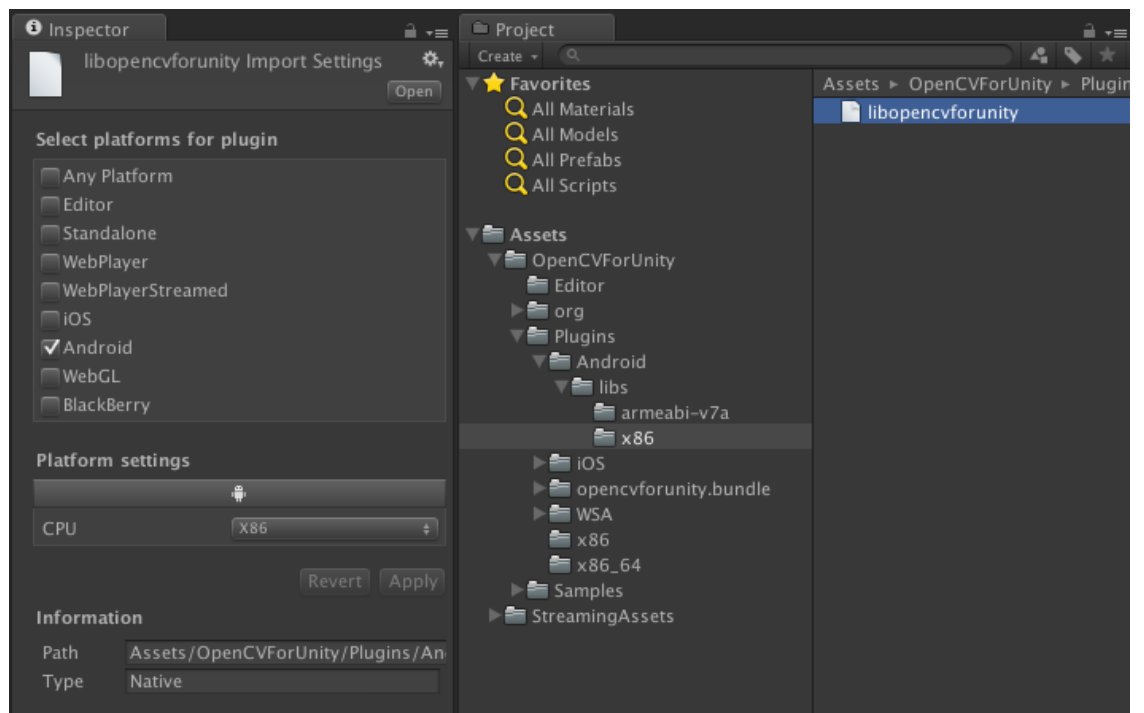


## Android Setup Procedure

- “OpenCVForUnity/Plugins/libs/armeabi-v7a/\*.so” - Select platform Android and CPU ARMv7 in Inspector.



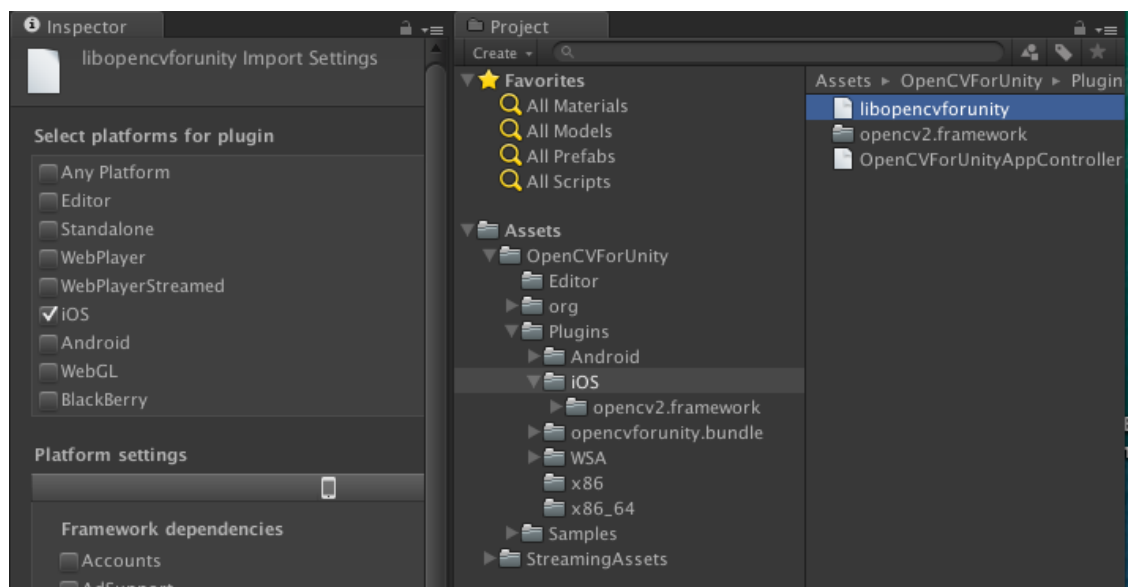
- “OpenCVForUnity/Plugins/libs/x86/\*.so” – Select platform Android and CPU x86 in Inspector.



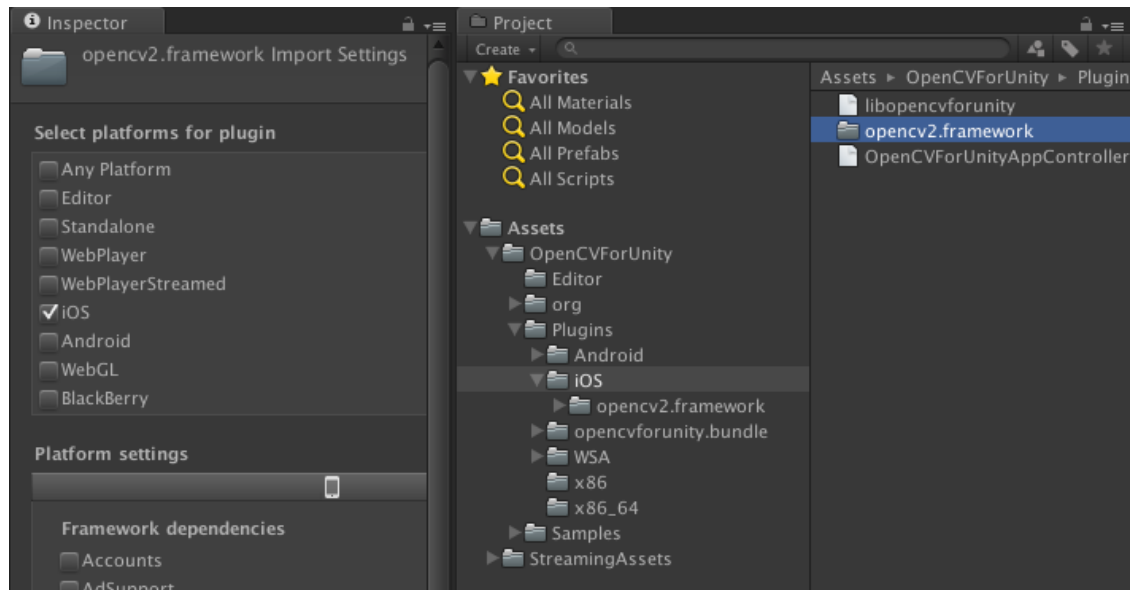
- Put the file that you want to use for `Utils.getFilePath()` in the “Assets/StreamingAssets/” folder. (haarcascade\_frontalface\_alt.xml etc is for `OpenCVForUnitySample.scene`. Please copy only when necessary.)

### iOS Setup Procedure

- “OpenCVForUnity/Plugins/iOS/libopencvforunity.a” – Select platform iOS in Inspector.



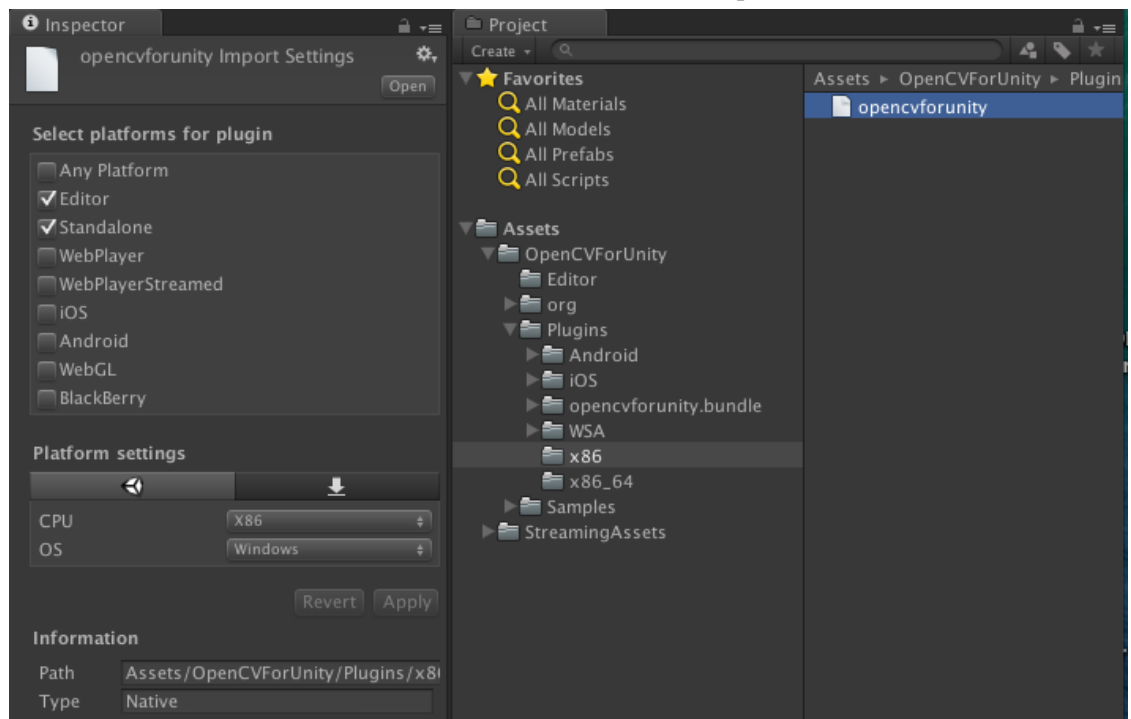
- “OpenCVForUnity/Plugins/iOS/opencv2.framework” – Select platform iOS in Inspector.



- Put the file that you want to use for `Utils.getPath()` in the “Assets/StreamingAssets/” folder. (haarcascade\_frontalface\_alt.xml etc is for OpenCVForUnitySample.scene. Please copy only when necessary.)

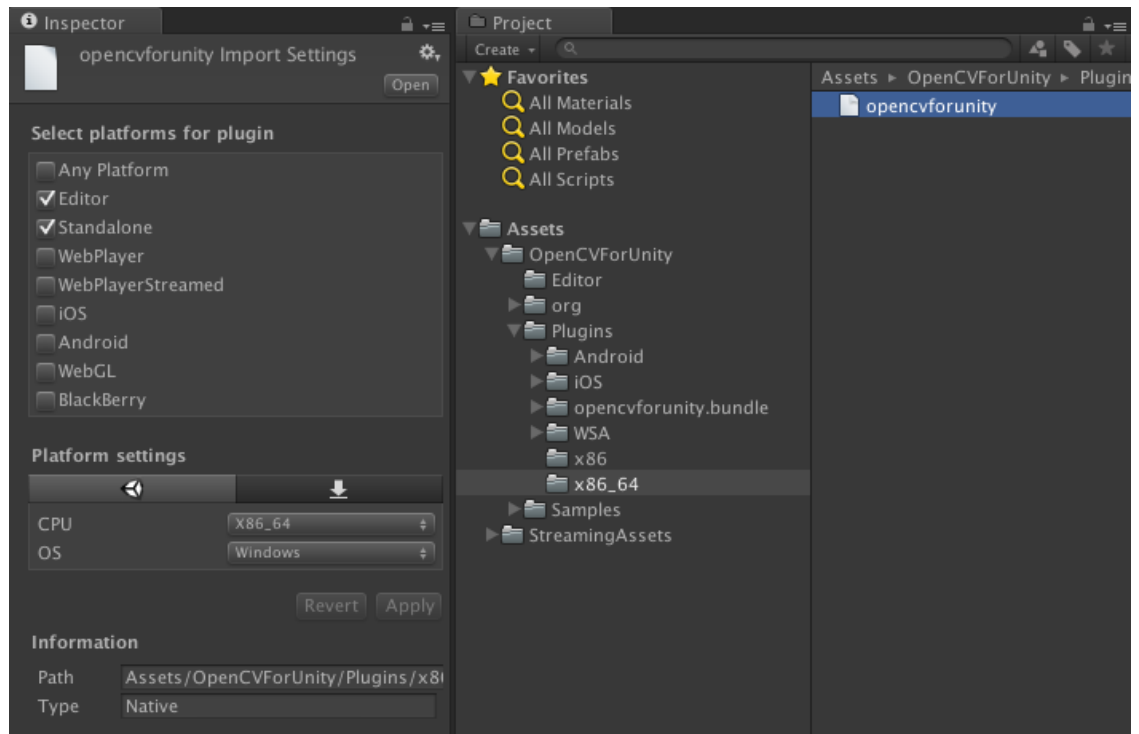
### Win Standalone Setup Procedure

- “OpenCVForUnity/Plugins/x86/opencvforunity.dll” – Select platform Editor, Standalone and CPU x86 and OS Windows in Inspector.
















- “OpenCVForUnity/Plugins/x86\_64/opencvforunity.dll” – Select platform Editor, Standalone and CPU x86\_64 and OS Windows in Inspector.

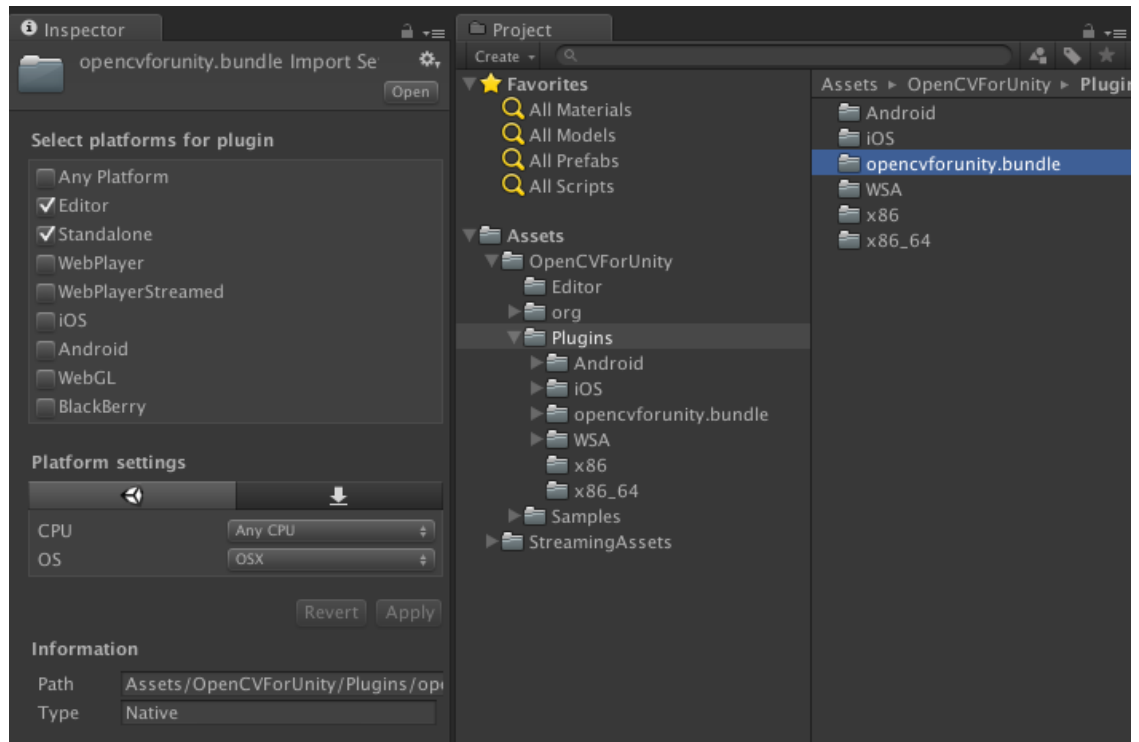


- Put the file that you want to use for `Utils.getFilePath()` in the “Assets/StreamingAssets/”. (haarcascade\_frontalface\_alt.xml etc is for OpenCVForUnitySample.scene. Please copy only when necessary.)
- If you use the “VideoCapture(string filename) or VideoWriter”, require setup.
  - 1)Download “OpenCV for Windows Version 3.1.0”(http://opencv.org/downloads.html).
  - 2)Set PATH variable to "opencv\_ffmpeg310.dll" or "opencv\_ffmpeg310\_64.dll".  
 if 32bit, "%path%to%opencv%build%x86%vc12%bin%".  
 if 64bit, "%path%to%opencv%build%x64%vc12%bin%".  
 Or  
 2)Copy to Project Folder.

-  Assets
-  Library
-  ProjectSettings
-  test\_Data
-  Assembly-CSharp.csproj
-  Assembly-CSharp-vs.csproj
-  opencv\_ffmpeg310\_64.dll
-  test.exe
-  TestProject.sln
-  TestProject.userprefs
-  TestProject-csharp.sln

## Mac Standalone Setup Procedure

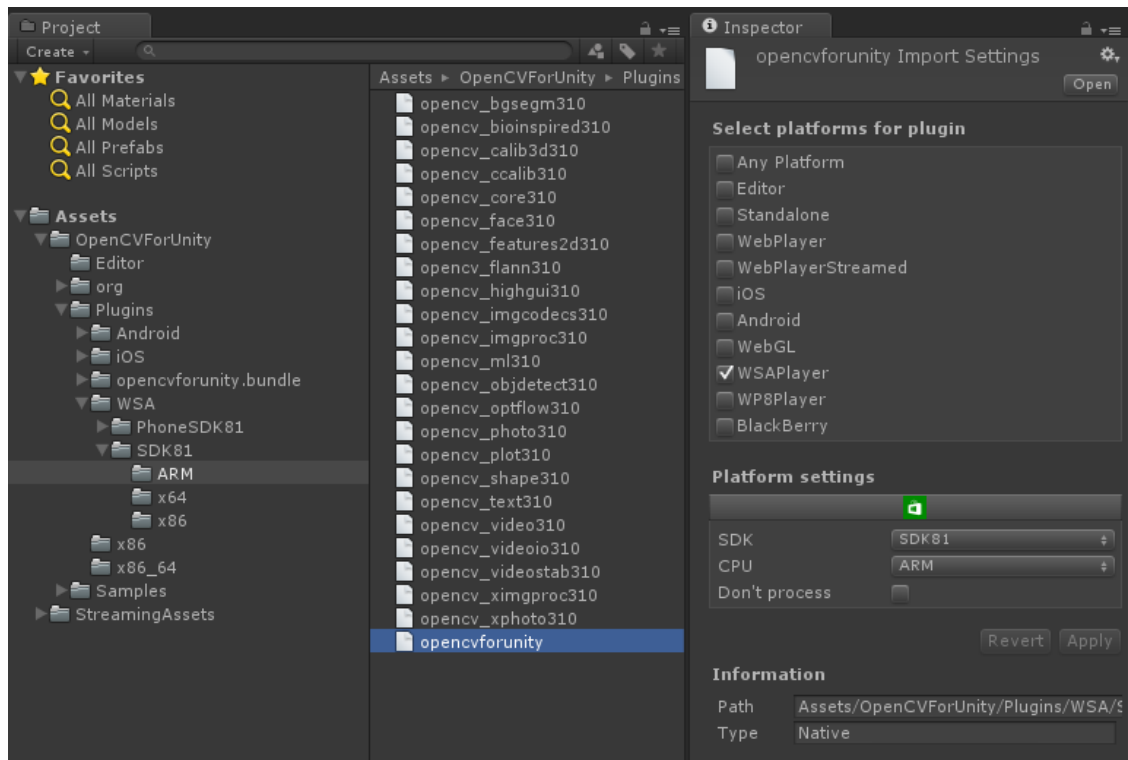
- “OpenCVForUnity/Plugins/opencvforunity.bundle” – Select platform Editor, Standalone and CPU x86\_64 and OS OSX in Inspector.



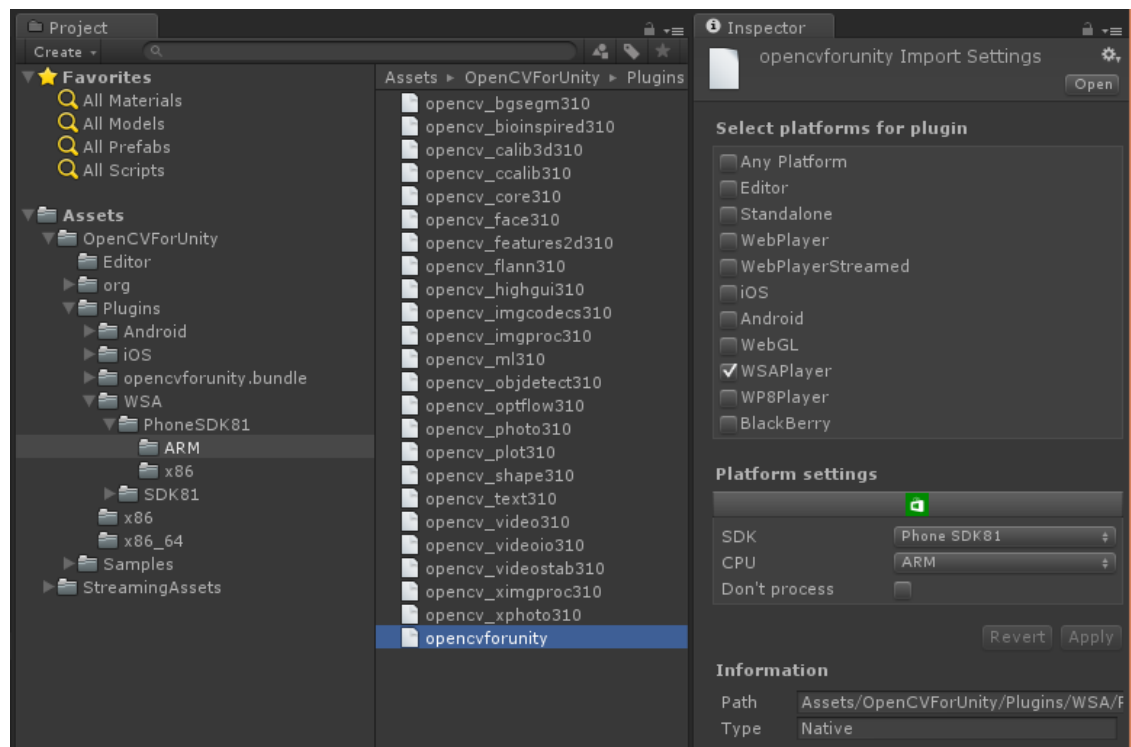
- Put the file that you want to use for `Utils.getFilePath()` in the “Assets/StreamingAssets/”. (haarcascade\_frontalface\_alt.xml etc is for `OpenCVForUnitySample.scene`. Please copy only when necessary.)

## WindowsStoreApps8.1 & WindowsPhone8.1 Setup Procedure

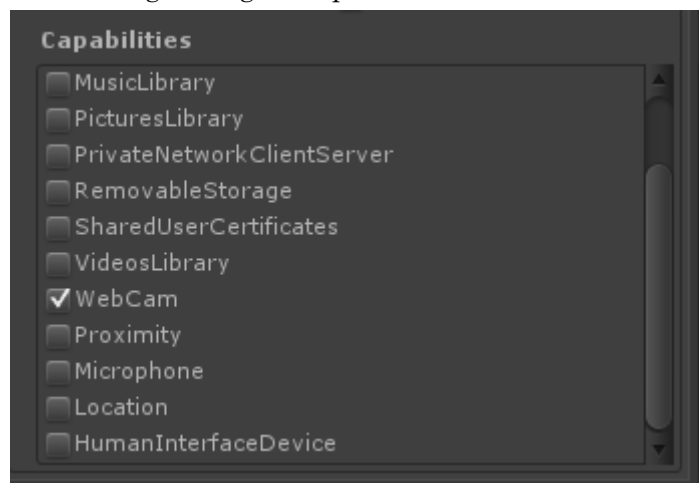
- “OpenCVForUnity/Plugins/WSA/SDK81/ARM/\*.dll” - Select platform WSAPlayer and SDK81 and CPU ARM in Inspector. Set “x86” and “x64” in the same way as “ARM”.



- “OpenCVForUnity/Plugins/WSA/PhoneSDK81/ARM/\*.dll” - Select platform WSAPlayer and PhoneSDK81 and CPU ARM in Inspector. Set “x86” in the same way as “ARM”.



- Put the file that you want to use for `Utils.getFilePath()` in the "Assets/StreamingAssets/". (haarcascade\_frontalface\_alt.xml etc is for `OpenCVForUnitySample.scene`. Please copy only when necessary.)
- If use `webCamTextue` class, Please choose "WebCam" in [PlayerSettings]-[PublishingSettings]-[Capabilities].



## Linux Setup Procedure

- Install OpenCV3.1.0  
<https://github.com/opencv/opencv/tree/d102ea96c0df38b2de96de5ca039ac3fe229967b>) with opencv-  
contrib([https://github.com/opencv/opencv\\_contrib/tree/a4135ceba18ea1080ed6e4dee0b512158d2ac558](https://github.com/opencv/opencv_contrib/tree/a4135ceba18ea1080ed6e4dee0b512158d2ac558))

### Example of Install command

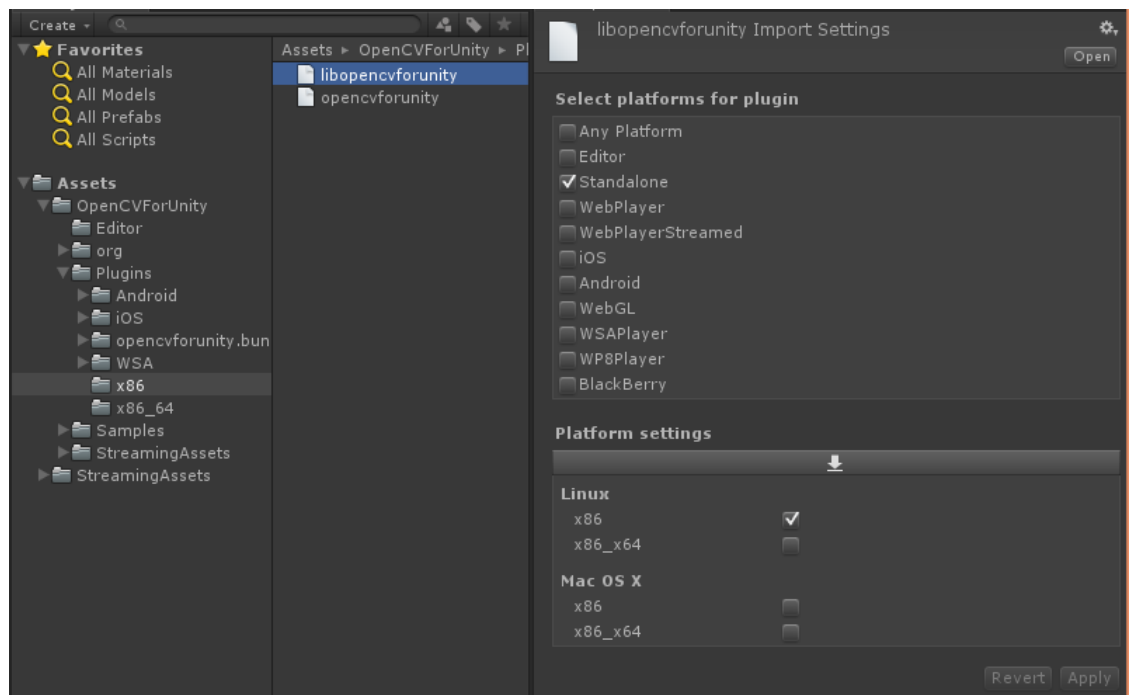
```

1.  #!/bin/sh
2.
3.  sudo apt-get -y install build-essential cmake git pkg-config libdc1394-22
libdc1394-22-dev libjpeg-dev libpng12-dev libtiff5-dev libjasper-dev
libavcodec-dev libavformat-dev libswscale-dev libxine2-dev
libgstreamer0.10-dev libgstreamer-plugins-base0.10-dev libv4l-dev libtbb-
dev libqt4-dev libfaac-dev libmp3lame-dev libopencore-amrnb-dev
libopencore-amrwb-dev libtheora-dev libvorbis-dev libxvidcore-dev x264
v4l-utils unzip
4.  mkdir opencv
5.  cd opencv
6.  wget
https://github.com/opencv/opencv/archive/d102ea96c0df38b2de96de5ca03
9ac3fe229967b.zip
7.  unzip d102ea96c0df38b2de96de5ca039ac3fe229967b.zip
8.  wget
https://github.com/opencv/opencv_contrib/archive/a4135ceba18ea1080ed6
e4dee0b512158d2ac558.zip
9.  unzip a4135ceba18ea1080ed6e4dee0b512158d2ac558.zip
10. cd opencv-d102ea96c0df38b2de96de5ca039ac3fe229967b
11. mkdir build
12. cd build
13. cmake -D CMAKE_BUILD_TYPE=RELEASE -D
CMAKE_INSTALL_PREFIX=/usr/local -D WITH_TBB=ON -D
BUILD_opencv_python2=OFF -D BUILD_opencv_python3=OFF -D
WITH_V4L=ON -D
OPENCV_EXTRA_MODULES_PATH=../../opencv_contrib-
a4135ceba18ea1080ed6e4dee0b512158d2ac558/modules ..
14. make -j $(nproc)

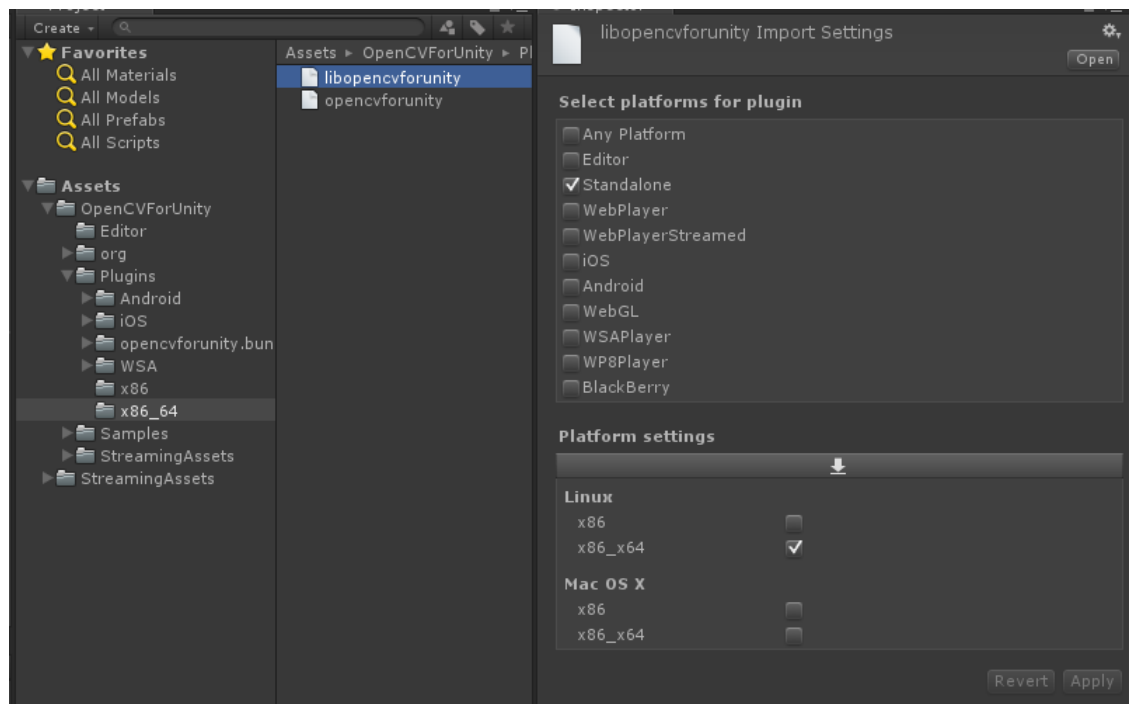
```

15. `sudo make install`
16. `sudo /bin/bash -c 'echo "/usr/local/lib" > /etc/ld.so.conf.d/opencv.conf'`
17. `sudo ldconfig`

- “OpenCVForUnity/Plugins/x86/libopencvforunity.so” – Select platform Editor, Standalone and CPU x86 and OS Linux in Inspector.



- “OpenCVForUnity/Plugins/x86\_64/libopencvforunity.so” – Select platform Editor, Standalone and CPU x86\_64 and OS Linux in Inspector.



- Put the file that you want to use for `Utils.getFilePath()` in the “Assets/StreamingAssets/”. (haarcascade\_frontalface\_alt.xml etc is for `OpenCVForUnitySample.scene`. Please copy only when necessary.)
- Additional Setting is required to run on the editor.  
<http://forum.unity3d.com/threads/native-plugin-in-editor-steam-specifically.384970/>



## Q & A

Q1.

“DllNotFoundException: opencvforunity” is displayed on the console when run the sample scene.

A1.

Plugin does not seem to be loaded correctly. Please check the setup procedure.

Q2.

“ArgumentException: The output Mat object has to be of the same size” is displayed on the console when run the sample scene.

A2.

After having set up Plugin, Plugin may work well when you reboot Unity.

Q3.

"Level 'Texture2DtoMatSample' (-1) could not be loaded because it has not been added to the build settings." is displayed on the console when run the sample scene.

A3.

Please Add all of the “\*\*\*.unity” in the “OpenCVForUnity” folder to [Build Settings] – [Scene In Build].

Q4.

In DetectFaceSample or WebCamTextureDetectFaceSample, red rectangle is not displayed around face.

A4.

you might have failed to read the "haarcascade\_frontalface\_alt.xml".Please confirm whether there is "OpenCVForUnity/StreamingAssets" folder at the right position.

Q5.

Support Web platform?

A5.

Since the Unity Web Player does not support the native plugin, "OpenCV for Unity" does not support "WebPlayer Platform".

Q6.

Support Windows10 UWP ?

A6.

If you want to try Windows10 UWP Platform(Beta Version), Please unzip “OpenCVForUnityUWP\_Beta.zip”, create new Project, import “uwp\_beta.unitypackage”.

IMPORTANT: READ BEFORE DOWNLOADING, COPYING, INSTALLING OR USING.

By downloading, copying, installing or using the software you agree to this license.  
If you do not agree to this license, do not download, install,  
copy or use the software.

License Agreement  
For Open Source Computer Vision Library

Copyright (C) 2000-2008, Intel Corporation, all rights reserved.  
Copyright (C) 2008-2011, Willow Garage Inc., all rights reserved.  
Third party copyrights are property of their respective owners.

Redistribution and use in source and binary forms, with or without modification,  
are permitted provided that the following conditions are met:

- \* Redistributions of source code must retain the above copyright notice,  
this list of conditions and the following disclaimer.
- \* Redistributions in binary form must reproduce the above copyright notice,  
this list of conditions and the following disclaimer in the documentation  
and/or other materials provided with the distribution.
- \* The name of the copyright holders may not be used to endorse or promote products  
derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and  
any express or implied warranties, including, but not limited to, the implied  
warranties of merchantability and fitness for a particular purpose are disclaimed.  
In no event shall the Intel Corporation or contributors be liable for any direct,  
indirect, incidental, special, exemplary, or consequential damages  
(including, but not limited to, procurement of substitute goods or services;  
loss of use, data, or profits; or business interruption) however caused  
and on any theory of liability, whether in contract, strict liability,

or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.