

# OpenCV for Unity 2.3.6

WebGL support  
iOS & Android support  
Windows10 UWP support  
Win & Mac & Linux Standalone support  
Support for preview in the **Editor**  
Work with Unity Free & Pro

## System Requirements

Build Win Standalone & Preview Editor : Windows8 or later  
Build Mac Standalone & Preview Editor : OSX 10.9 or later  
Build Linux Standalone & Preview Editor : Ubuntu16.04 or later  
Build Android : API level 21 or later  
Build iOS : iOS Version 8.0 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 4.1.0(git: [opencv,opencv-contrib](#)).
- You can image processing in **real-time** by using the **WebCamTexture** capabilities of Unity. (**real-time face detection works smoothly on 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.
- **Examples of integration with other publisher assets** are available. (e.g. [PlayMaker](#), [NatCam](#), [NatCorder](#))

[Official Site](#) | [ExampleCode](#) | [Android Demo](#) | [WebGL Demo](#) | [Tutorial & Demo Video](#) | [Forum](#)  
| [API Reference](#) | [Support Modules](#)

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

**OpenCV for Unity** uses **OpenCV** under **3-clause BSD License**; see Third-Party Notices.txt file in package for details

Example code using OpenCV for Unity is available.

- [MarkerBased AR Example](#)
- [MarkerLess AR Example](#)
- [FaceTracker Example](#)
- [FaceSwapper Example](#)
- [FaceMask Example](#)
- [RealTime FaceRecognition Example](#)
- [GoogleVRWithOpenCVForUnityExample](#)
- [Voforia with OpenCV for Unity Example](#)
- [Kinect with OpenCV for Unity Example](#)
- [AVPro with OpenCV for Unity Example](#)

- [HoloLens with OpenCV for Unity Example](#)
- [PlayMakerActions for OpenCVforUnity](#)
- [NatCam with OpenCVForUnity Example](#)
- [NatCorder with OpenCVForUnity Example](#)
- [MagicLeapWithOpenCVForUnityExample](#)

## Version changes

**2.3.6** [WebGL]Fixed “Plugins/WebGL/2018.2/opencvforunity.bc”. [Common]Added multi-dimensional Mat example to MatBasicProcessingExample. [Common]Fixed ARUtils.cs.

**2.3.5** [Common]Updated to OpenCV4.1.0. [Windows, Android]Added dynamic link library version.

**2.3.4** [Common]Added MaskRCNNExample. [WebGL]Added Unity2019.1 or later support.

**2.3.3** [Common]Updated to OpenCV4.0.0. [Common]Re-assigned namespace as was classified by module names. [Common]Updated to WebCamTextureToMatHelper.cs v1.1.0. [Common]Updated to ImageOptimizationHelper v1.1.0 [Common] Added Utils\_GetFilePathExample, FaceMarkExample and QRCodeDetectorExample.

**2.3.2** [macOS]Removed 32bit architecture(i386) from opencvforunity.bundle.

**2.3.1** [Common]Updated to OpenCV3.4.2. [Android,UWP]Fixed Utils.setDebugMode() method on the IL2CPP backend. [Common]Added DnnObjectDetectionExample and DnnObjectDetectionWebCamTextureExample.

**2.3.0** [iOS]Added a function to automatically remove the simulator architecture(i386,x86\_64) at build time. [Common] Improved OpenCVForUnityMenuItem.setPluginImportSettings() method.

**2.2.9** [Linux]Simplified the Linux platform setup procedure. [Common]Added support for Utils. setDebugMode() method on all platforms. [Common]Updated to WebCamTextureToMatHelper.cs v1.0.9. [Common]Added MatToTextureInRenderThreadExample and AlphaBlendingExample.

**2.2.8** [Common]Updated to WebCamTextureToMatHelper.cs v1.0.7. [Common]Added MatBasicProcessingExample. [Common]Fixed WebCamTextureToMatExample, WebCamTextureToMatHelperExample, ArUcoExample. [Common]Added flip flag to Utils.fastMatToTexture2D() method and Utils.fastTexture2DToMat() method. [Common]Added throwException flag to Utils.setDebugMode() method.

**2.2.7** [Common]Updated to OpenCV3.4.1. [Common]Added OpenPoseExample(The model file is not included in this asset.), KalmanFilterExample, ArUcoCameraCalibrationExample. [Common]Fixed VideoWriterExample, VideoCaptureExample, ImwriteScreenCaptureExample, CamShiftExample, TrackingExample, HandPoseEstimationExample, ArUcoCreateMarkerExample, ArUcoExample, ArUcoWebCamTextureExample. [Common] Updated to WebCamTextureToMatHelper.cs v1.0.6.

**2.2.6** [Android]Added arm64-v8a Architecture. [Common]Added ImwriteScreenCaptureExample.

**2.2.5** [Common] Updated to WebCamTextureToMatHelper.cs v1.0.4. [Common] Fixed MobileNetSSDExample and MobileNetSSDWebCamTextureExample.

**2.2.4** [Common]Updated to OpenCV3.3.1. [Common]Added ResnetSSDFaceDetectionExample, YoloObjectDetectionExample, YoloObjectDetectionWebCamTextureExample.

**2.2.3** [Common]Updated to WebCamTextureToMatHelper.cs v1.0.3. [iOS] opencv2.framework is changed from static framework to embeddd framework. (Target

minimum iOS Version must be set to 8.0 or higher.)

**2.2.2** [Common]Added TextRecognitionExample.

**2.2.1** [Common]Updated to OpenCV3.3.0. [Common]Added dnn module.(win,mac,ios,android platform) [Common]Added img\_hash, reg, text module.(all platform) [Common]Added MobileNetSSDExample, MobileNetSSDWebCamTextureExample, TensorFlowWebCamTextureExample, ThinPlateSplineShapeTransformerExample, TextDetectionExample, VideoWriterExample. [Common]WindowsStoreApp8.1 & WindowsPhone8.1 support have been deprecated.

**2.2.0** [Common]Updated WebCamTextureToMatHelper.cs v1.0.2 [Common]Improved Utils.getFilePathAsync().

**2.1.9** [WebGL]Fixed Utils.getFilePathAsync() method.

**2.1.8** [Common]Added PCAExample. [Common]Updated WebCamTextureToMatHelper.cs and OptimizationWebCamTextureToMatHelper.cs(Changed several method names.).

**2.1.7** [Common]Improved Utils.getFilePath() and Utils.getFilePathAsync(). [Common]Improved WebCamTextureAsyncDetectFaceExample.cs. [Common] Fixed the const value of Calib3d class.

**2.1.6** [Common]Fixed fastMatToTexture2D() method.

**2.1.5** [Common]Updated to OpenCV3.2.0. [Common]Added fuzzy, phase\_unwrapping, saliency, shape, tracking module. [Common]Added TrackingSample. [iOS]Added ios\_exclude\_contrib.zip for build size reduction. [Android]Added android\_exclude\_contrib.zip for build size reduction.

**2.1.4** [Common]Changed the scene name("Sample" to "Example") [Common]Fixed ArUcoTexture2DExample and ArUcoWebCamTextureExample. [Common]Added ConnectedComponentsExample. [Common]Added GreenScreenExample.

**2.1.3** [UWP]Added OpenCVForUnityUWP\_Beta3.zip.

**2.1.2** [Common]Fixed WebCamTextureToMatHelper.cs.(flipVertical and flipHorizontal flag)

**2.1.1** [Common]Fixed OpenCVForUnityMenuItem.cs.(No valid name for platform: 11 Error) [Common]Added Utils.textureToTexture2D() method. [Common]Added Mat class operators. [Common]Added PolygonFilterSample.

**2.1.0** [Common]Fixed WebCamTextureToMatHelper class. [Common]Added Utils.getVersion(). [Common]Fixed Utils.getFilePathAsync().

**2.0.9** [WebGL]Added WebGL(beta) support.(Unity5.3 or later)

**2.0.8** [Common]Improved WebCamTextureHelper class. [Common]Fixed ArUcoSample.

**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 SVMExample. [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.

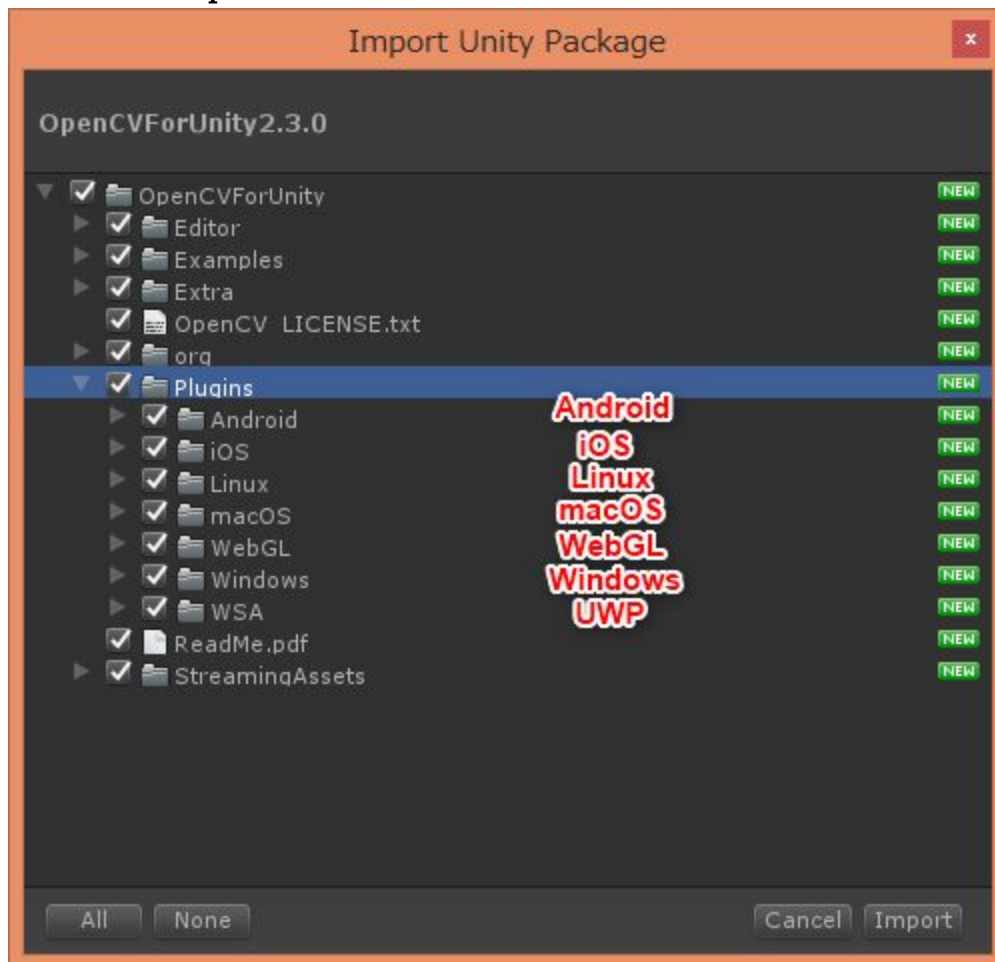
**Beta4** [Common]Add Utils. getGraphicsDeviceType(). [Common]Add SampleScene Setup Tutorial Video for Unity5.

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

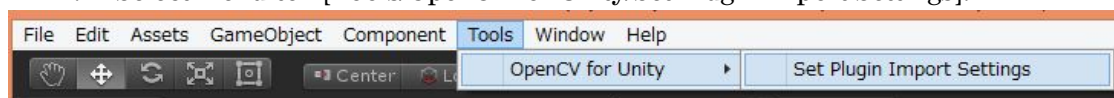
**Beta2** [iOS] Fix problem when working with Metaio(UnityAppController problem). [Common]Add [System.Serializable] to basic class. [Common] change folder name from “OpenCVForUnity/OpenCVForUnity\_Editor/” to “OpenCVForUnity/Editor/”. [iOS]Move “OpenCVForUnity/OpenCVForUnity\_Editor/opencv2.framework”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

**Quick setup procedure to run the example scenes ([Setup Tutorial Video](#))**

1. Import the OpenCVForUnity.package. You do not need to import plug-in files for platforms not supported by your project. **If there is a previous version of OpenCVForUnity in the project, please delete the OpenCVForUnity folder first and then import the new version.**



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



3. Move the "OpenCVForUnity/StreamingAssets/" folder to the "Assets/" folder.
  - Additional Setup for MobileNetSSDExample or MobileNetSSDWebCamTextureExample : Download <https://github.com/chuanqi305/MobileNet-SSD/blob/master/images/004545.jpg>. Copy 004545.jpg to "Assets/StreamingAssets/dnn/" folder. Download <https://drive.google.com/file/d/0B3gersZ2cHlxRm5PMWRoTkdHdHc/view>. Copy MobileNetSSD\_deploy.caffemodel to "Assets/StreamingAssets/dnn/" folder. Download [https://raw.githubusercontent.com/chuanqi305/MobileNet-SSD/master/MobileNetSSD\\_deploy.prototxt](https://raw.githubusercontent.com/chuanqi305/MobileNet-SSD/master/MobileNetSSD_deploy.prototxt). Copy MobileNetSSD\_deploy.prototxt to "Assets/StreamingAssets/dnn/" folder.
  - Additional Setup for OpenPoseExample : Download

[https://github.com/CMU-Perceptual-Computing-Lab/openpose/blob/master/examples/media/COCO\\_val2014\\_000000000589.jpg](https://github.com/CMU-Perceptual-Computing-Lab/openpose/blob/master/examples/media/COCO_val2014_000000000589.jpg). Copy

COCO\_val2014\_000000000589.jpg to "Assets/StreamingAssets/dnn/" folder. Download

[http://posefs1.perception.cs.cmu.edu/OpenPose/models/pose/mpi/pose\\_iter\\_160000.caffemodel](http://posefs1.perception.cs.cmu.edu/OpenPose/models/pose/mpi/pose_iter_160000.caffemodel). Copy pose\_iter\_160000.caffemodel to

"Assets/StreamingAssets/dnn/" folder. Download

[https://raw.githubusercontent.com/opencv/opencv\\_extra/master/testdata/dnn/openpose\\_pose\\_mpi\\_faster\\_4\\_stages.prototxt](https://raw.githubusercontent.com/opencv/opencv_extra/master/testdata/dnn/openpose_pose_mpi_faster_4_stages.prototxt). Copy

openpose\_pose\_mpi\_faster\_4\_stages.prototxt to "Assets/StreamingAssets/dnn/" folder.

- Additional Setup for ResnetSSDFaceDetectionExample : Download [https://raw.githubusercontent.com/opencv/opencv\\_3rdparty/b2bfc75f6aea5b1f834ff0f0b865a7c18ff1459f/res10\\_300x300\\_ssd\\_iter\\_140000.caffemodel](https://raw.githubusercontent.com/opencv/opencv_3rdparty/b2bfc75f6aea5b1f834ff0f0b865a7c18ff1459f/res10_300x300_ssd_iter_140000.caffemodel). Copy

res10\_300x300\_ssd\_iter\_140000.caffemodel to "Assets/StreamingAssets/dnn/" folder. Download

[https://raw.githubusercontent.com/opencv/opencv/master/samples/dnn/face\\_detector/deploy.prototxt](https://raw.githubusercontent.com/opencv/opencv/master/samples/dnn/face_detector/deploy.prototxt). Copy deploy.prototxt to "Assets/StreamingAssets/dnn/" folder.

- Additional Setup for MaskRCNNExample : Download <https://github.com/chuanqi305/MobileNet-SSD/blob/master/images/004545.jpg>.

Copy 004545.jpg to "Assets/StreamingAssets/dnn/" folder. Download and unzip [http://download.tensorflow.org/models/object\\_detection/mask\\_rcnn\\_inception\\_v2\\_coco\\_2018\\_01\\_28.tar.gz](http://download.tensorflow.org/models/object_detection/mask_rcnn_inception_v2_coco_2018_01_28.tar.gz).

Rename frozen\_inference\_graph.pb to mask\_rcnn\_inception\_v2\_coco\_2018\_01\_28.pb. Copy

mask\_rcnn\_inception\_v2\_coco\_2018\_01\_28.pb to "Assets/StreamingAssets/dnn/" folder. Download

[https://raw.githubusercontent.com/opencv/opencv\\_extra/master/testdata/dnn/mask\\_rcnn\\_inception\\_v2\\_coco\\_2018\\_01\\_28.pbt.txt](https://raw.githubusercontent.com/opencv/opencv_extra/master/testdata/dnn/mask_rcnn_inception_v2_coco_2018_01_28.pbt.txt). Copy

mask\_rcnn\_inception\_v2\_coco\_2018\_01\_28.pbt.txt to "Assets/StreamingAssets/dnn/" folder. Download

[https://raw.githubusercontent.com/spmallick/learnopencv/master/Mask-RCNN/mscoco\\_labels.names](https://raw.githubusercontent.com/spmallick/learnopencv/master/Mask-RCNN/mscoco_labels.names). Copy mscoco\_labels.names to "Assets/StreamingAssets/dnn/" folder.

- Additional Setup for TensorFlowWebCamTextureExample : Download and unzip <https://storage.googleapis.com/download.tensorflow.org/models/inception5h.zip>.

Copy tensorflow\_inception\_graph.pb and imagenet\_comp\_graph\_label\_strings.txt to "Assets/StreamingAssets/dnn/" folder.

- Additional Setup for YoloObjectDetectionExample or YoloObjectDetectionWebCamTextureExample : Download

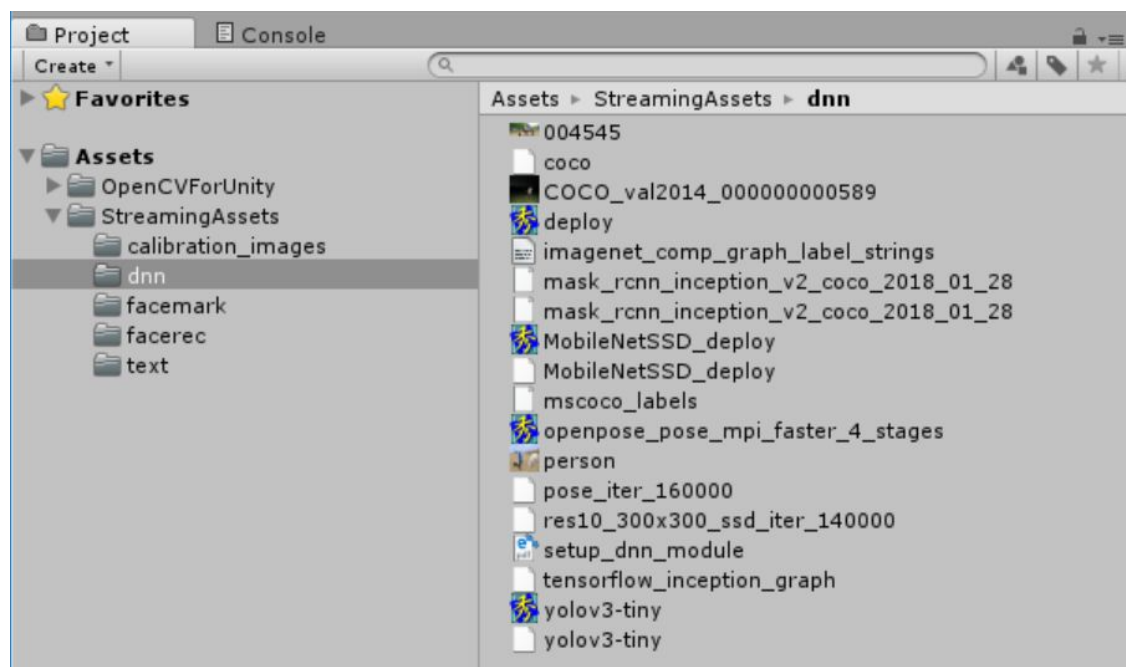
<https://github.com/pjreddie/darknet/blob/master/data/person.jpg>. Copy person.jpg to "Assets/StreamingAssets/dnn/" folder. Download

<https://raw.githubusercontent.com/pjreddie/darknet/master/cfg/yolov3-tiny.cfg>.

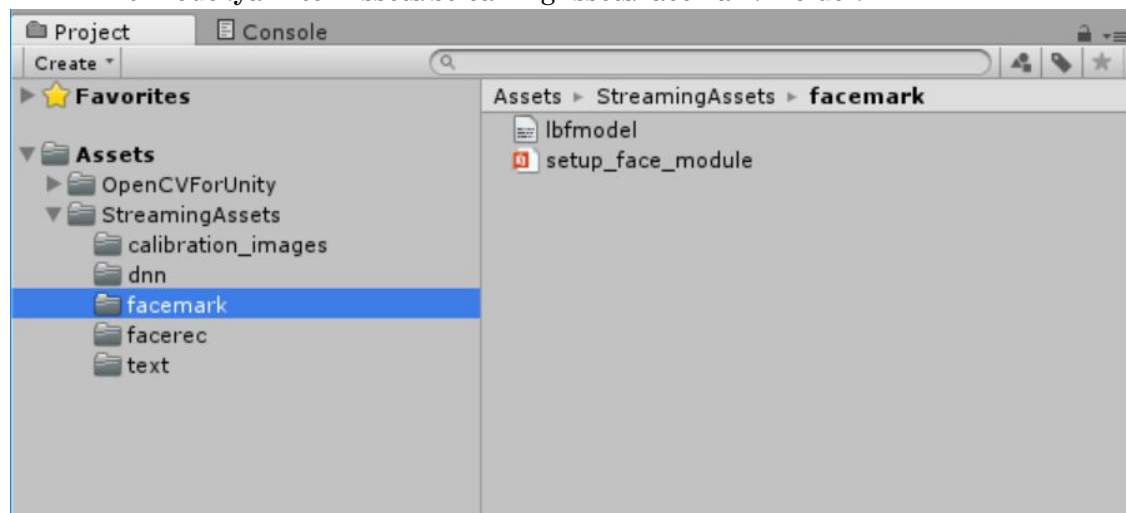
Copy yolov3-tiny.cfg to "Assets/StreamingAssets/dnn/" folder. Download <https://pjreddie.com/media/files/yolov3-tiny.weights>. Copy yolov3-tiny.weights to

"Assets/StreamingAssets/dnn/" folder. Download

<https://github.com/pjreddie/darknet/tree/master/data/coco.names>. Copy coco.names to "Assets/StreamingAssets/dnn/" folder.

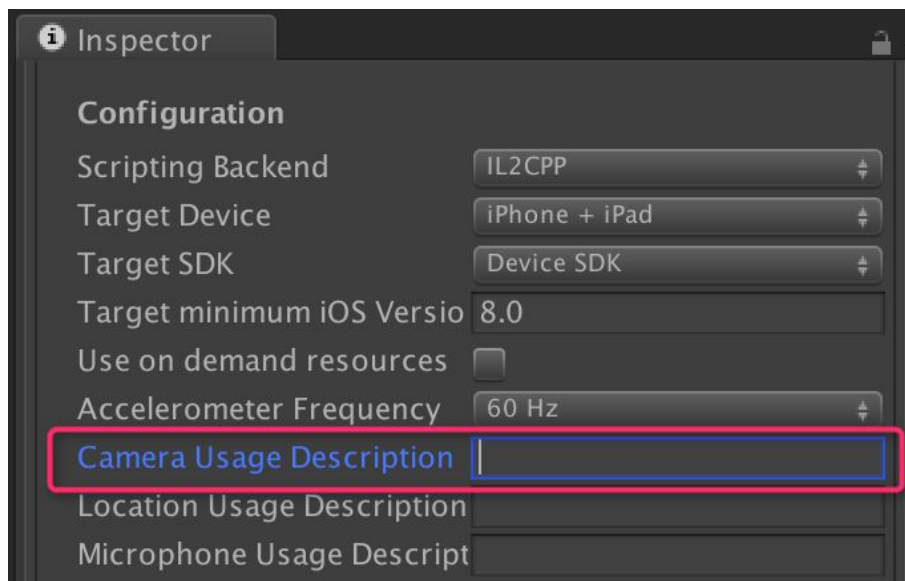


- Additional Setup for FaceMarkExample : Download <https://github.com/spmallick/GSOC2017/blob/master/data/lbfmodel.yaml>. Copy lbfmodel.yaml to "Assets/StreamingAssets/facemark/" folder.

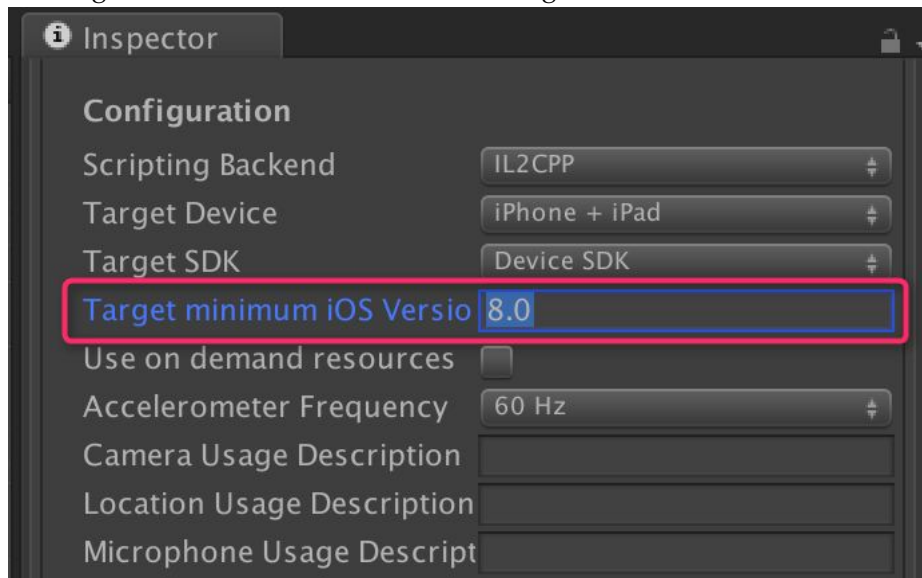


4. [iOS] Set [PlayerSettings]-[Other Settings]-[Configuration]-[Camera Usage Description].



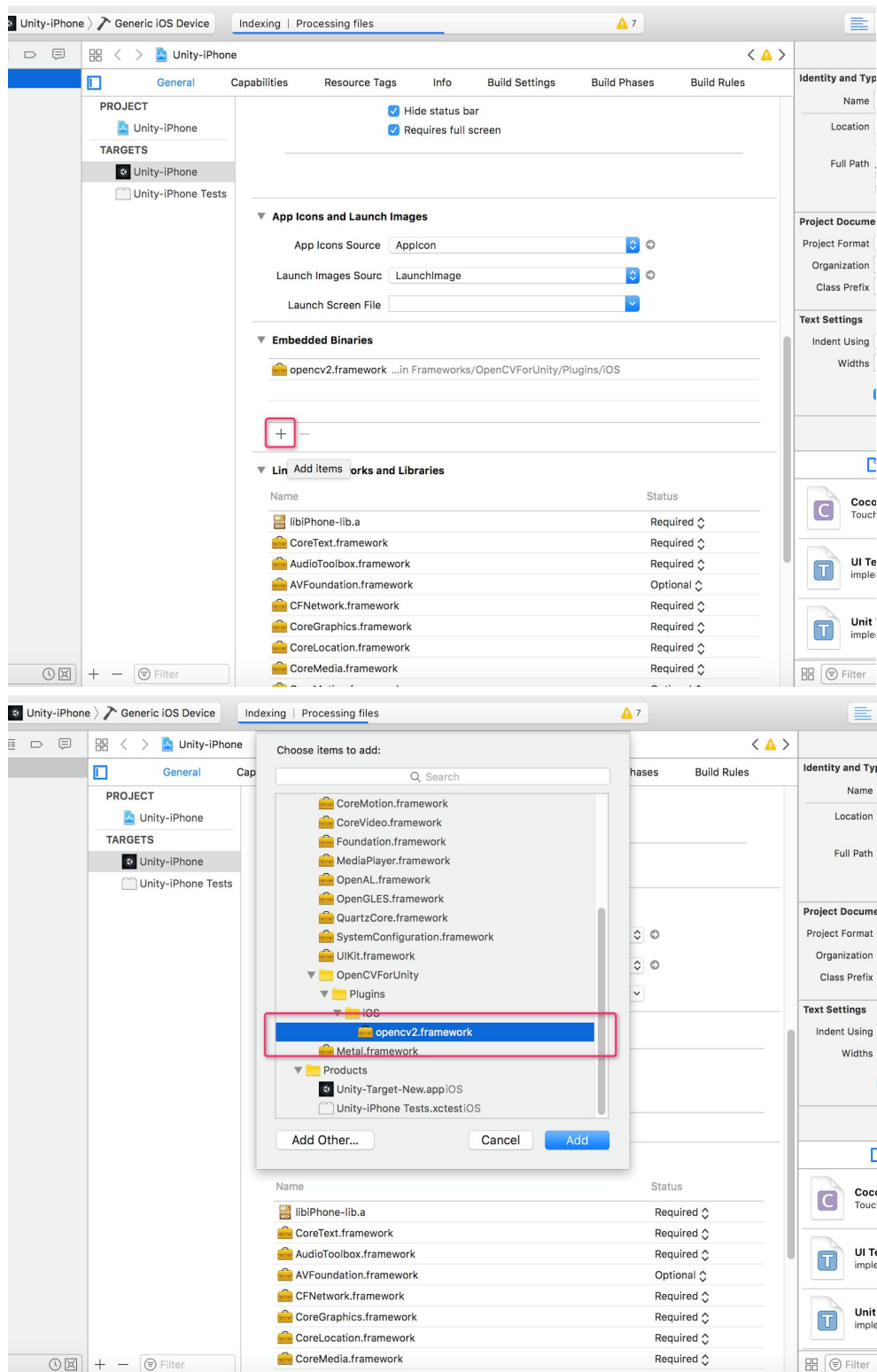


Set Target minimum iOS Version to 8.0 or higher.

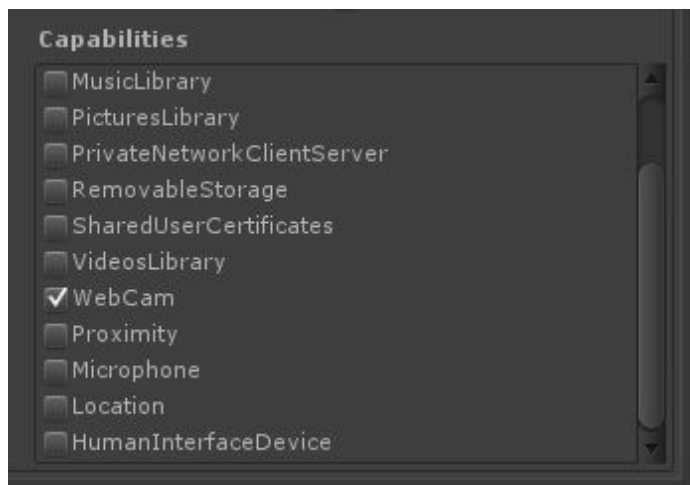


If the version of Unity is less than 2017.2, you have to set opencv2.framework to Embedded Binaries manually.



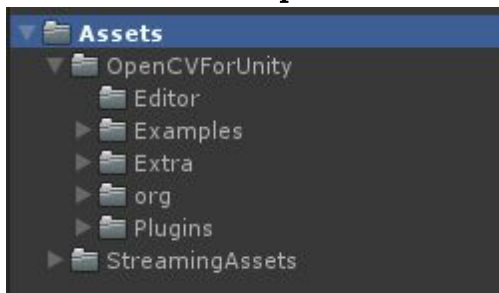


5. [Windows10 UWP] If use webCamTextue class, Please choose “WebCam” in [PlayerSettings]-[PublishingSettings]-[Capabilities].



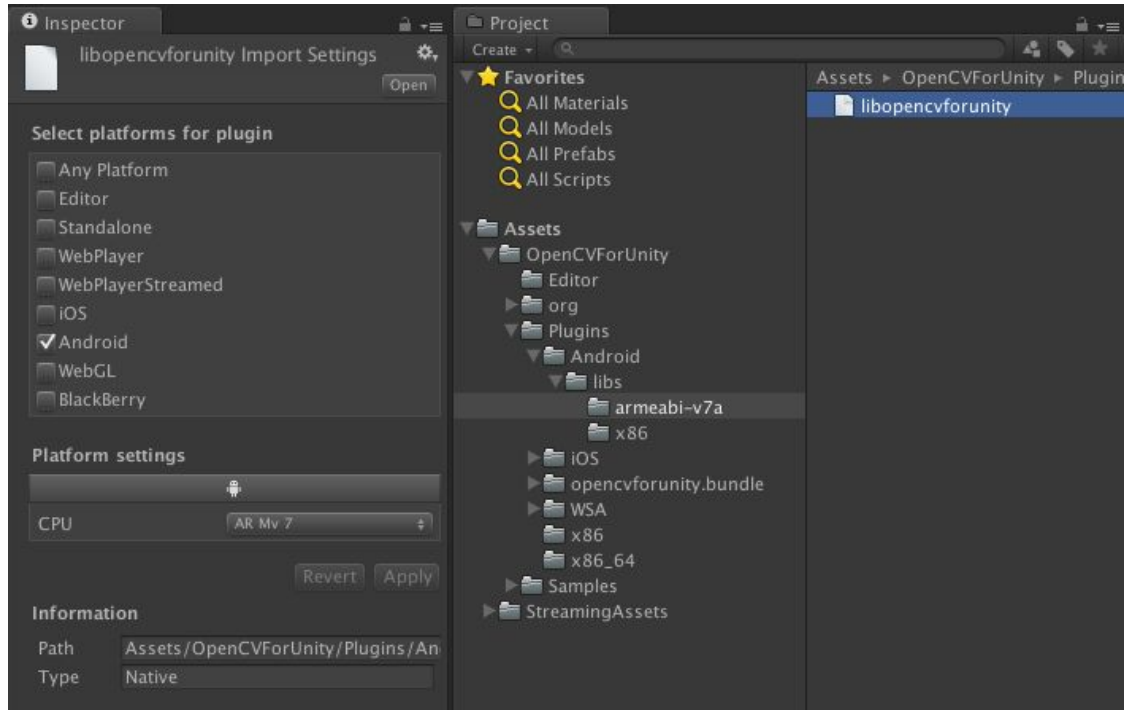
6. Add all of the “\*\*\*.unity” in the “OpenCVForUnity/Examples” folder to [Build Settings] – [Scene In Build].

**Screenshot after the setup**

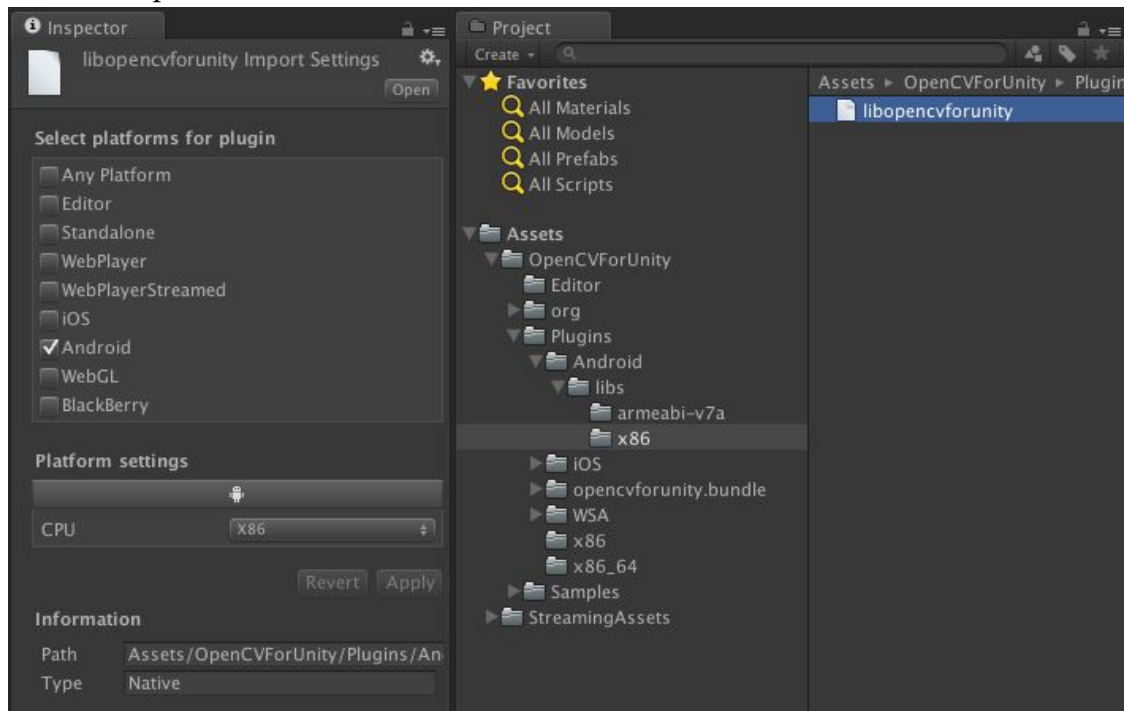


### Android Setup Procedure

- “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.

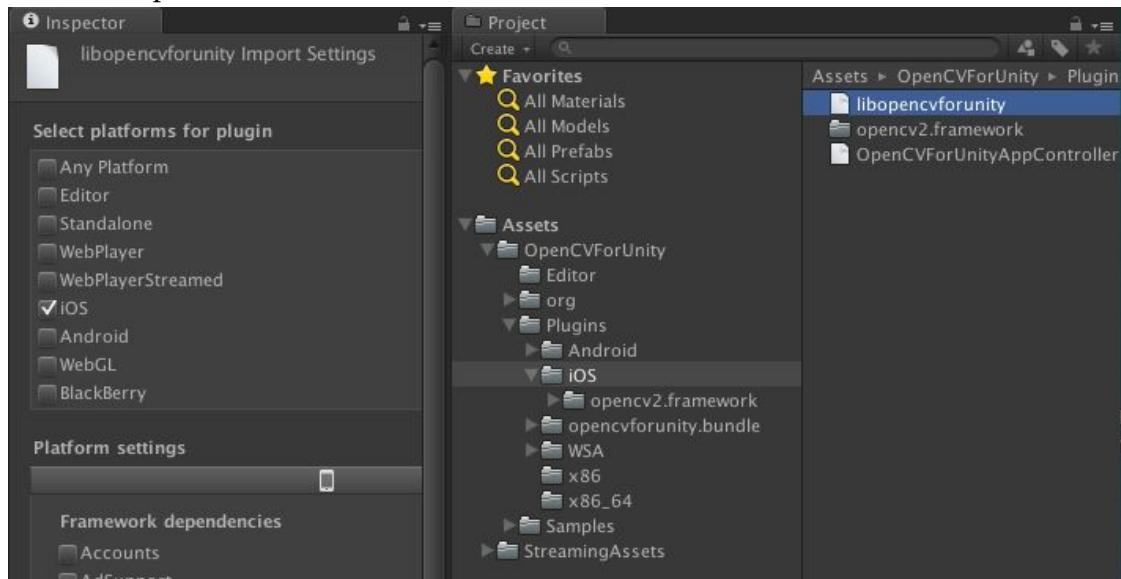


- If you do not use opencv\_contrib module, build size will be reduced by using native plugin file excluding opencv\_contrib module.

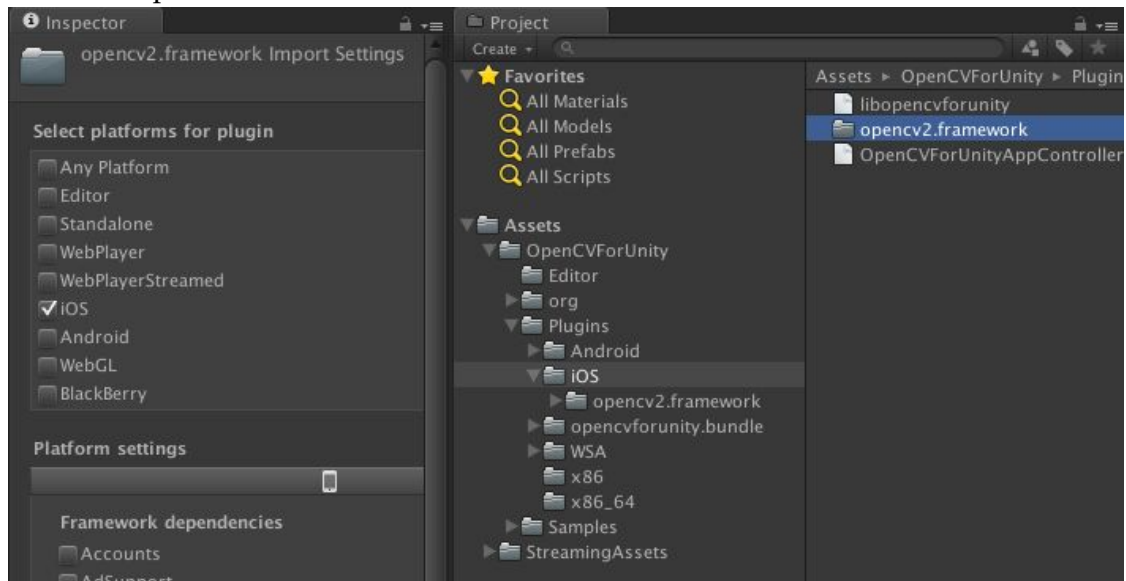
1. Replace the OpenCVForUnity/Plugins/Android/libs folder to the OpenCVForUnity/Extra/exclude\_contrib/Android/libs folder.
2. Select MenuItem[Tools/OpenCV for Unity/Set Plugin Import Settings].
3. Delete the OpenCVForUnity/Assets/OpenCVForUnity/org/opencv\_contrib folder and the OpenCVForUnity/Examples/ContribModules folder.

### iOS Setup Procedure

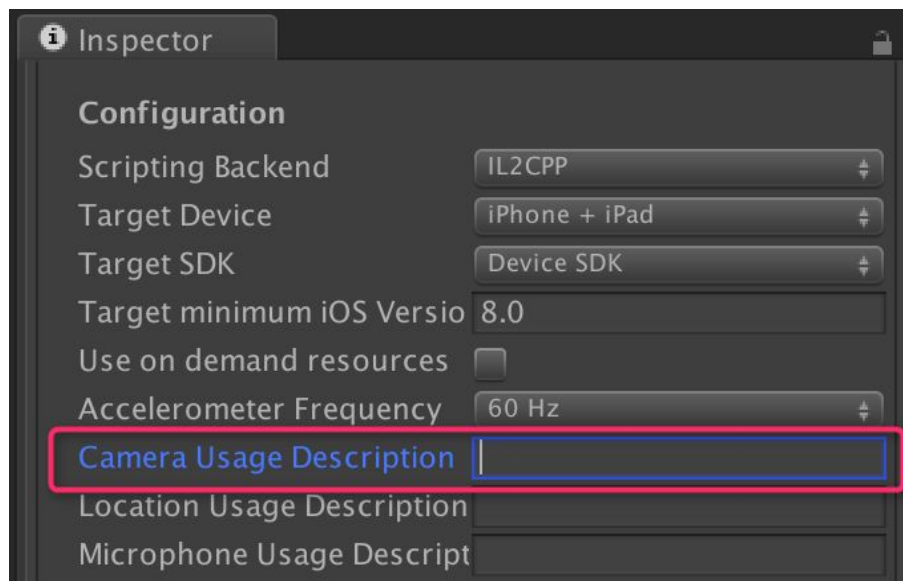
- “OpenCVForUnity/Plugins/iOS/libopencvforunity.a” – Select platform iOS in Inspector.



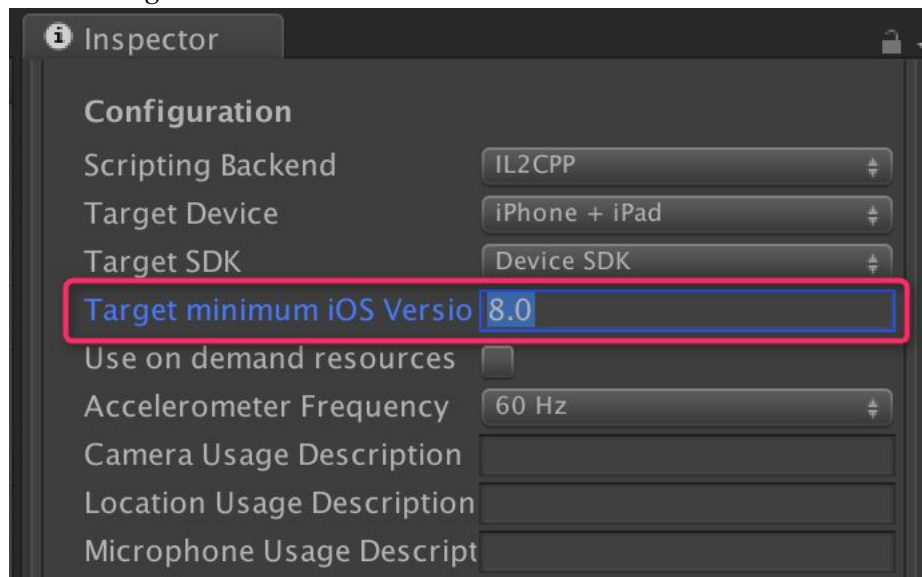
- “OpenCVForUnity/Plugins/iOS/opencv2.framework” – Select platform iOS in Inspector.



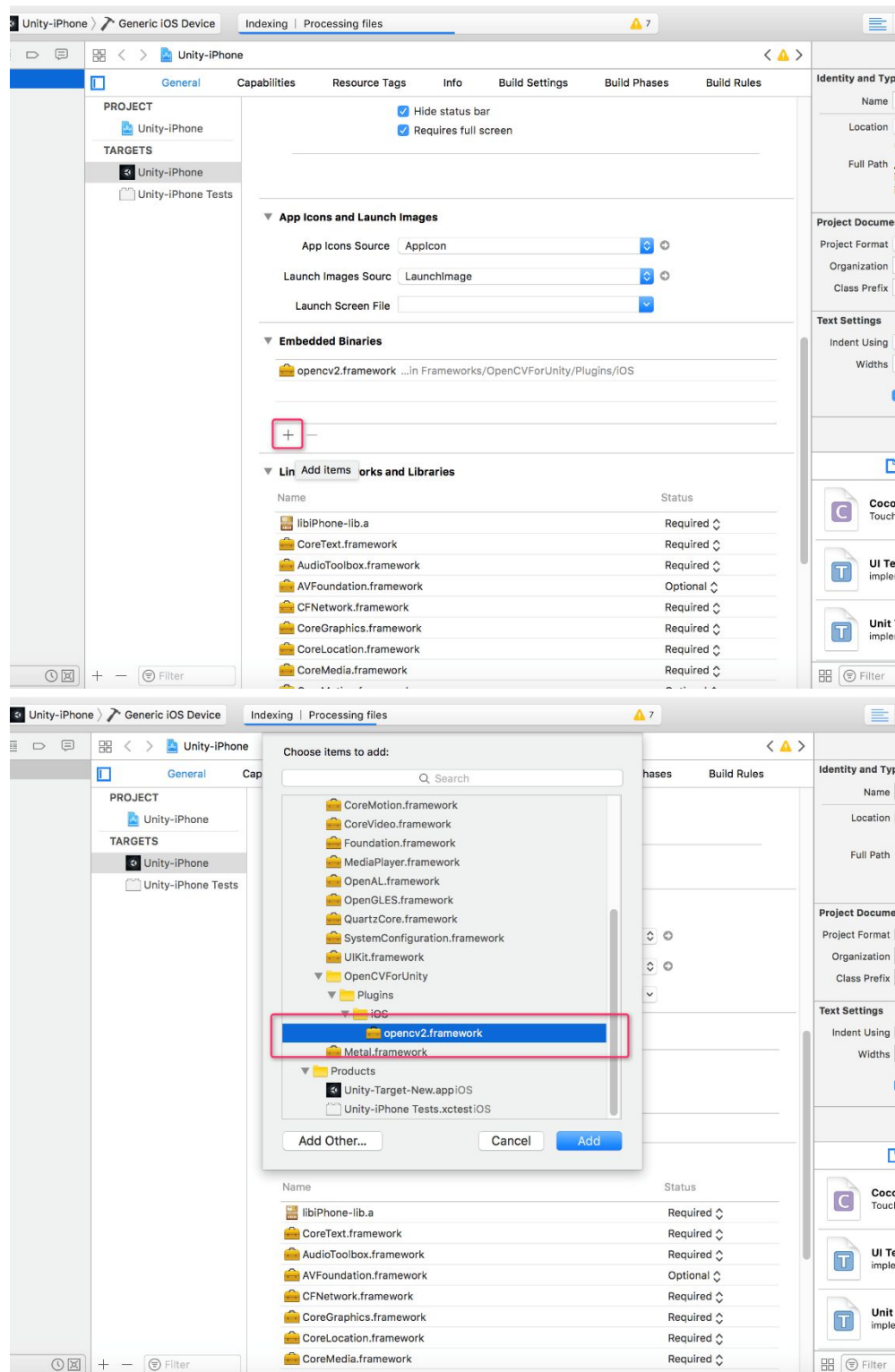
- If iOS platform, Set [PlayerSettings]-[Other Settings]-[Configuration]-[Camera Usage Description].



- Set Target minimum iOS Version to 8.0 or higher.

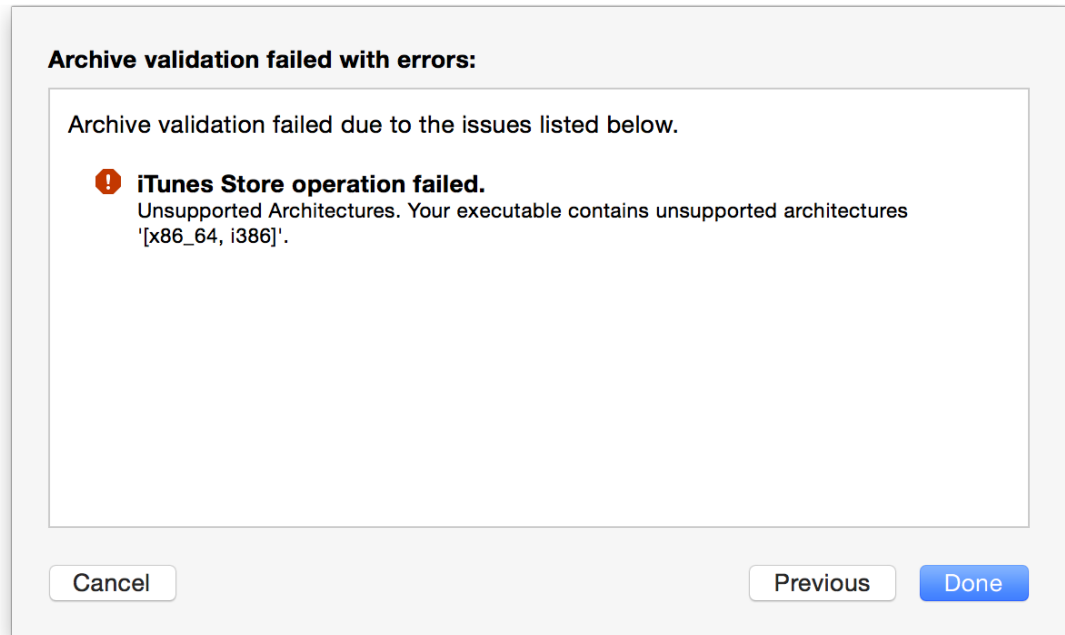


- If the version of Unity is less than 2017.2, you have to set opencv2.framework to Embedded Binaries manually.



- When exporting ipa file, you need to remove the unneeded architectures from opencv2.framework, before submitting it.





Please see Q & A No.9 for details.

- When "-ObjC" is set to "OTHER\_LDFLAGS" by other Asset, the following error may occur.

Undefined symbols for architecture armv7:

"\_OBJC\_CLASS\_\$\_ALAssetsLibrary", referenced from:

objc-class-ref in opencv2(cap\_ios\_video\_camera.o)

ld: symbol(s) not found for architecture armv7

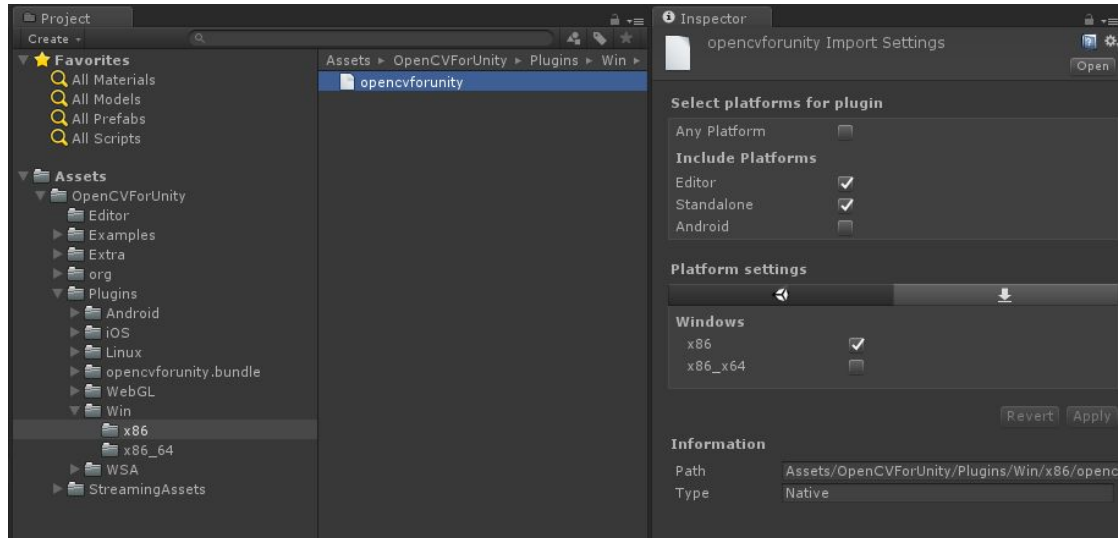
clang: error: linker command failed with exit code 1 (use -v to see invocation)

In that case, add "proj.AddFrameworkToProject (target, "AssetsLibrary.framework", false);" to "Assets/OpenCVForUnity/Editor/iOS\_BuildPostprocessor.cs".

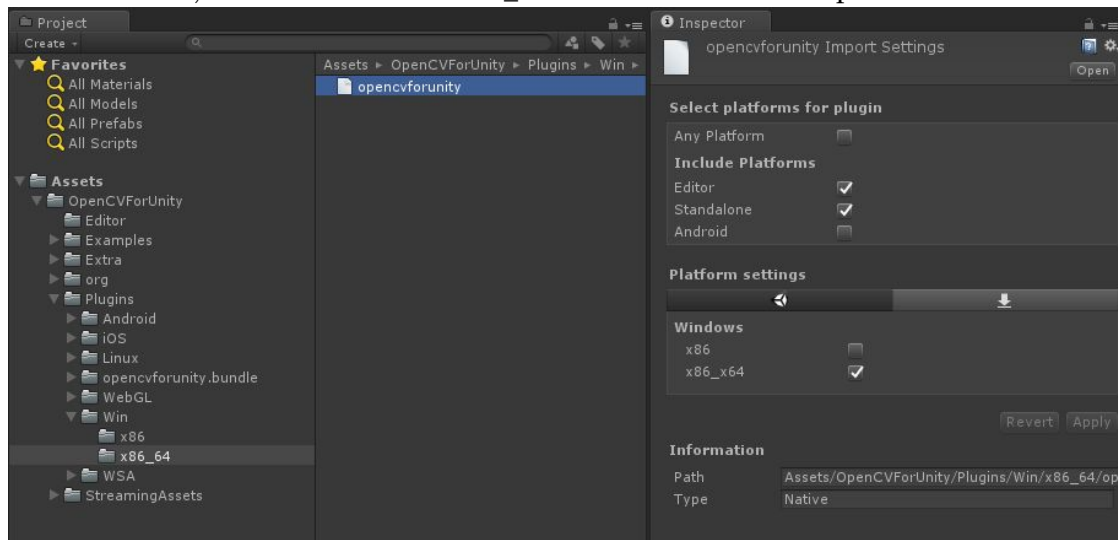
- If you do not use opencv\_contrib module, build size will be reduced by using native plugin file excluding opencv\_contrib module.
  1. Replace the OpenCVForUnity/Plugins/iOS folder to the OpenCVForUnity/Extra/exclude\_contrib/iOS folder.
  2. Select MenuItem[Tools/OpenCV for Unity/Set Plugin Import Settings].
  3. Delete the OpenCVForUnity/Assets/OpenCVForUnity/org/opencv\_contrib folder and the OpenCVForUnity/Examples/ContribModules folder.

## Windows Standalone Setup Procedure

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



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

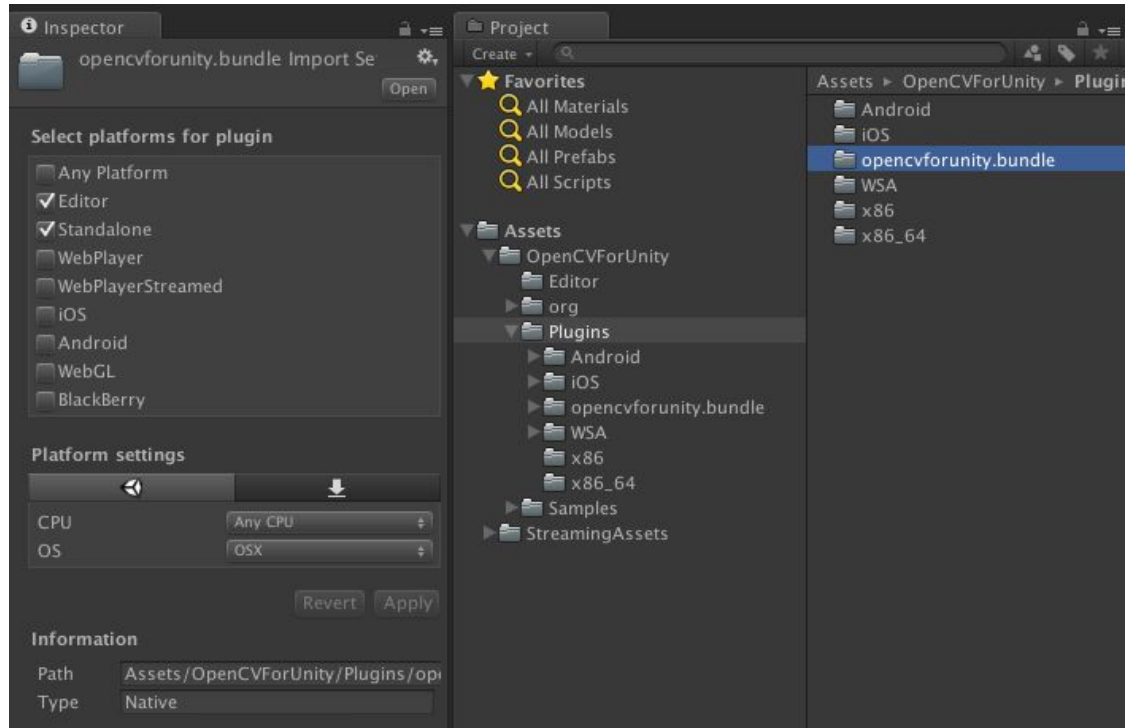


- If you want to use more video formats with the "Video Capture (string filename)" or "VideoWriter" method, setup is required.
  - 1) Download "OpenCV" for Windows Version 4.1.0" (<http://opencv.org/downloads.html>).
  - 2) Set PATH variable to "opencv\_ffmpeg4.1.0.dll" or "opencv\_ffmpeg4.1.0\_64.dll".
    - if 32bit, "path\to\opencv\build\x86\vc14\bin\".
    - if 64bit, "path\to\opencv\build\x64\vc14\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

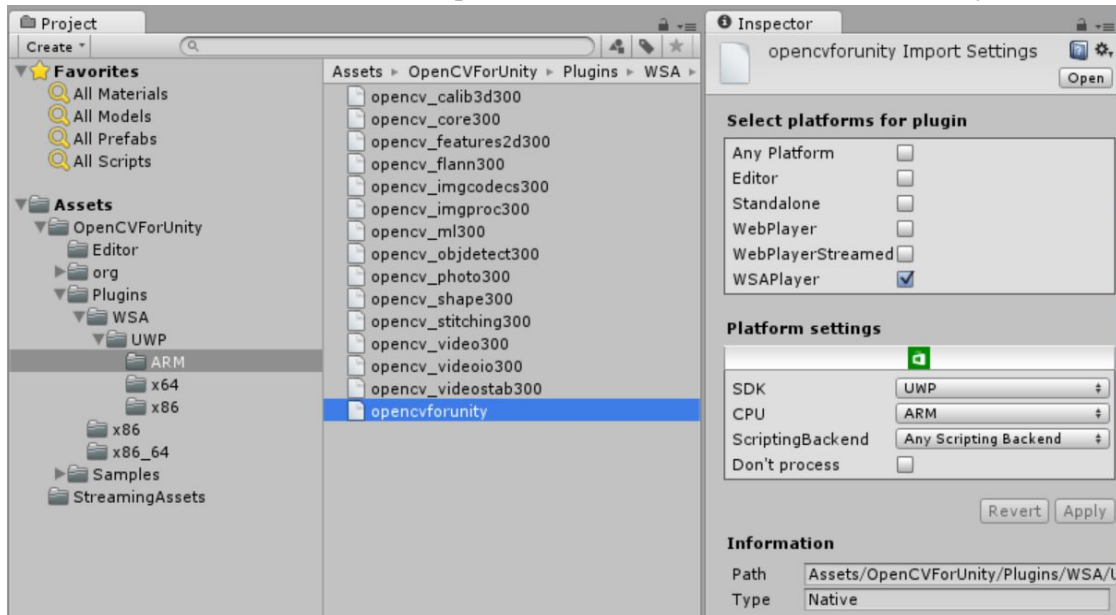
### Mac Standalone Setup Procedure

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

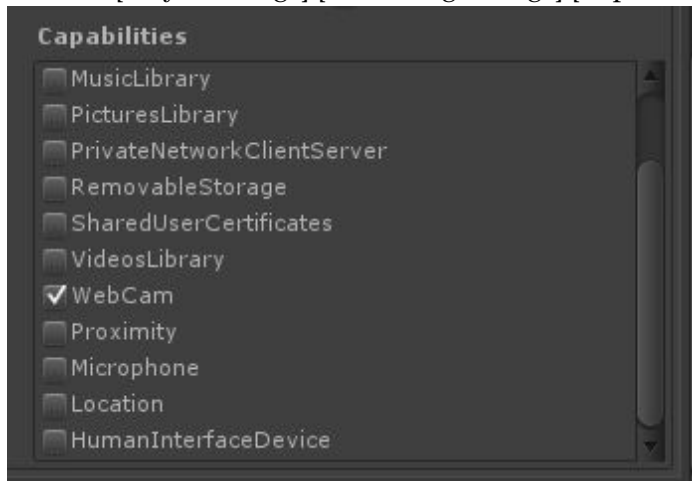


## UWP Setup Procedure

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

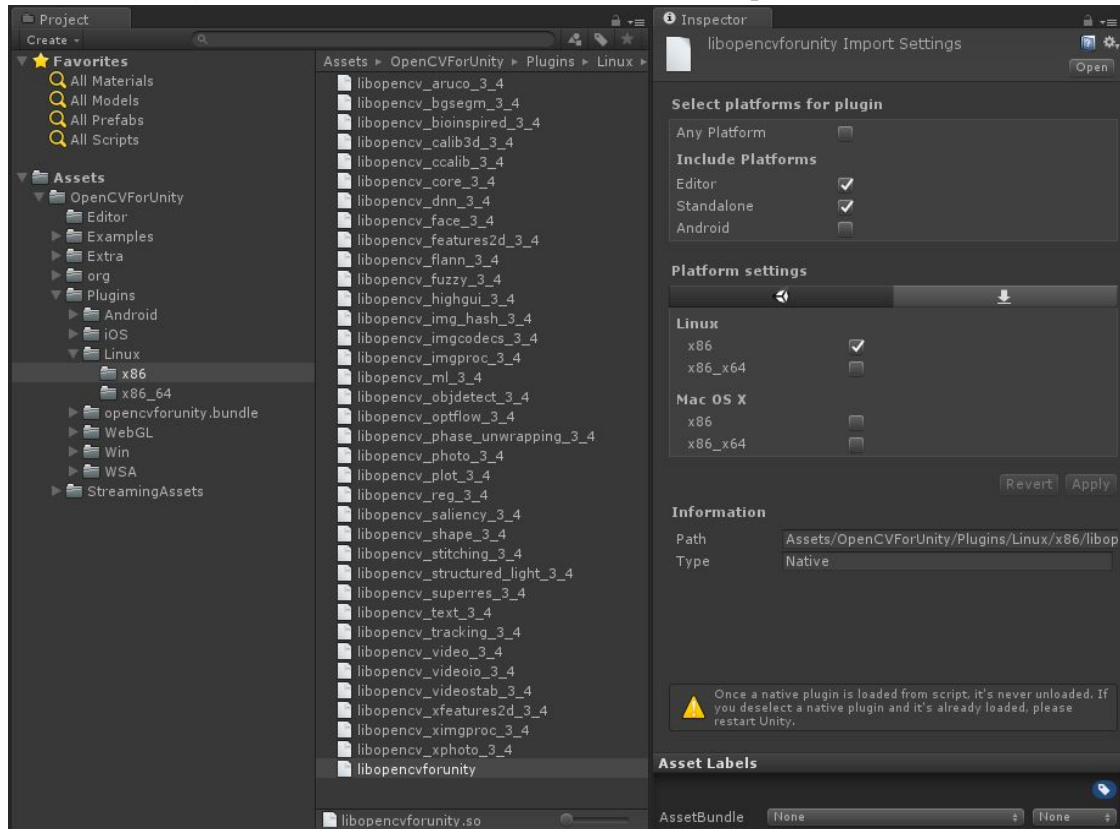


- If use webCamTextue class, Please choose “WebCam” in [PlayerSettings]-[PublishingSettings]-[Capabilities].

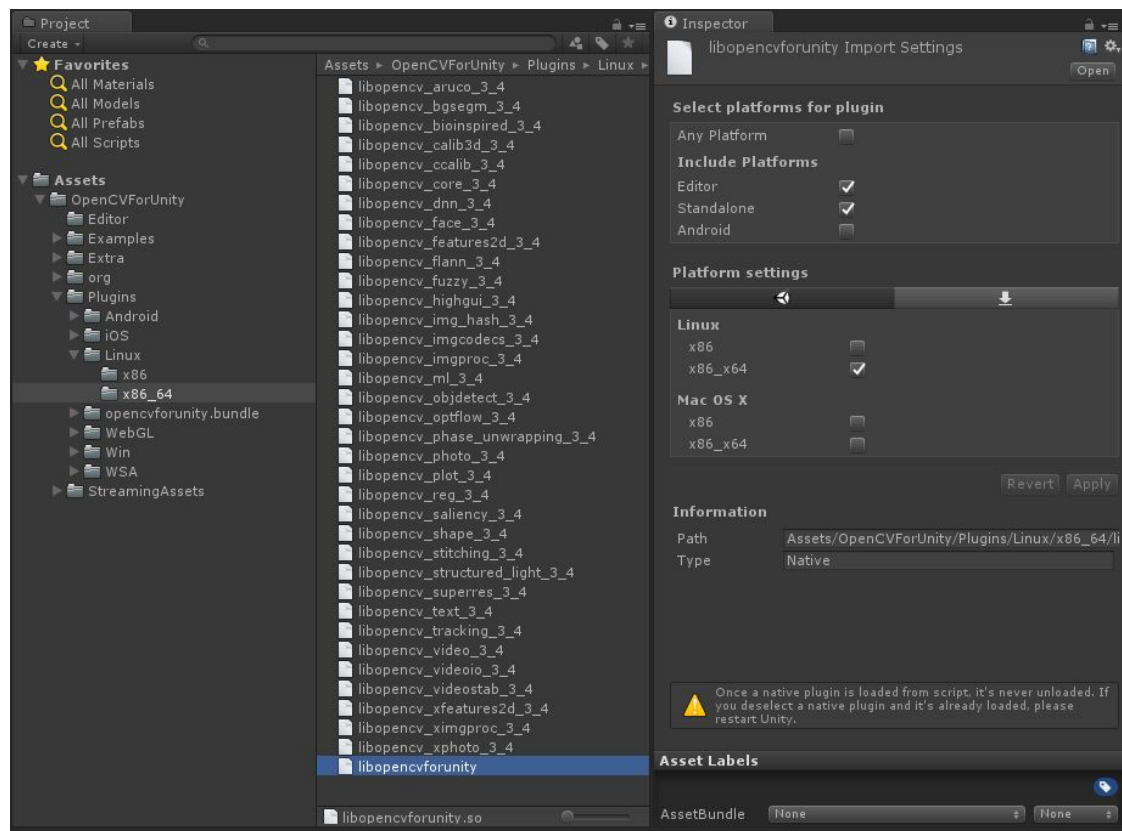


## Linux Setup Procedure

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



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

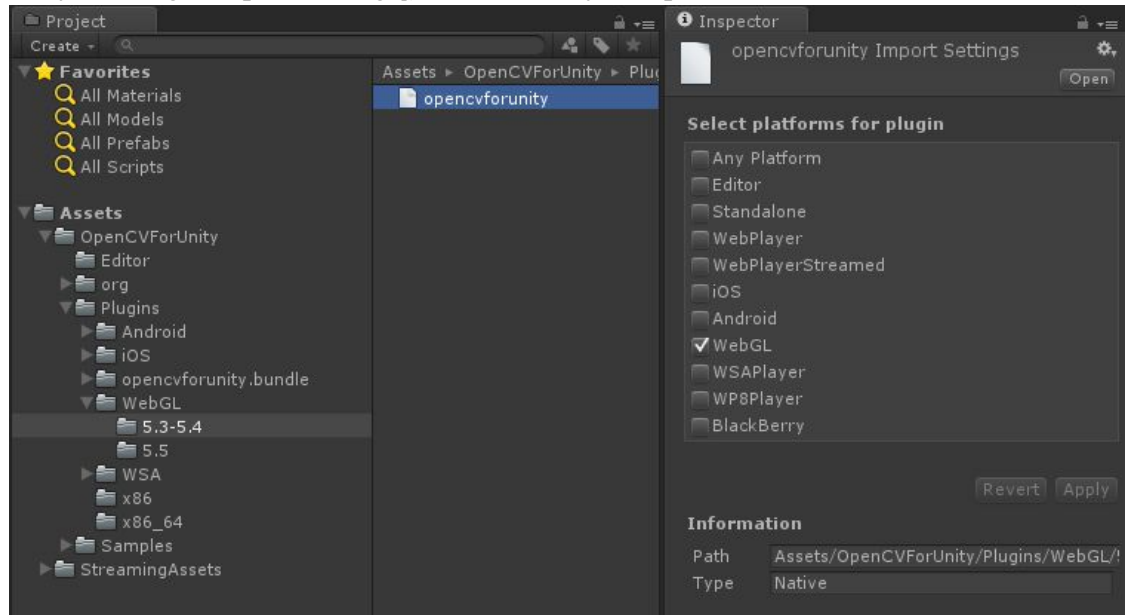


- Additional Setting is required to run on the editor.  
<http://forum.unity3d.com/threads/native-plugin-in-editor-steam-specifically.384970/>



### WebGL Setup Procedure

- “OpenCVForUnity/Plugins/WebGL/unity\_version/opencvforunity.bc” – Select platform WebGL in Inspector. By Selecting MenuItem [Tools/OpenCV for Unity/Set Plugin Import Settings], You can easily set up.



- Put the file that you want to use for `Utils.getFilePathAsync()` in the “Assets/StreamingAssets/”. In Case of WebGL platform, you need to use `Utils.getFilePathAsync()` instead of `Utils.getFilePath()`. (haarcascade\_frontalface\_alt.xml is for OpenCVForUnityExample.scene. Please copy only when necessary.)
- In the WebGL (asm.js) platform, the calculation result of Float type may be significantly different from other platforms. When using the OpenCV’s method that use the Mat class (CvType is CV\_32F) as an argument, you need to pay attention to the calculation precision.

### MagicLeap Setup Procedure

- Please see this page.  
<https://github.com/EnoxSoftware/MagicLeapWithOpenCVForUnityExample>

## How to use OpenCV Dynamic Link Library with customized build settings

- Download OpenCV 4.1.0 repository ( git: [opencv](#), [opencv-contrib](#) ).

### Android

1. Build the Android SDK with “opencv/platforms/android/build\_sdk.py”.

```
python ../opencv/platforms/android/build_sdk.py ../build ../opencv
--ndk_path=C://android-ndk --sdk_path=C://android-sdk
--extra_modules_path=../opencv_contrib/modules --use_android_buildtools
```

2. Copy the output file ( native\libs\arm64-v8a\libopencv\_java4.so ) to “OpenCVForUnity\Plugins\Android\libs\arm64-v8a\”. Copy the output files ( native\libs\arm64-v8a\libopencv\_java4.so ) to “OpenCVForUnity\Plugins\Android\libs\armeabi-v7a\”. Copy the output files ( native\libs\x86\libopencv\_java4.so ) to “OpenCVForUnity\Plugins\Android\libs\x86\”.
3. Copy “OpenCVForUnity\Extra\dll\_version\Android\libs\” to “OpenCVForUnity\Plugins\Android\libs\”.

### iOS

1. Build the iOS framework with “opencv/platforms/ios/build\_framework.py”.

```
python opencv/platforms/ios/build_framework.py --contrib opencv_contrib
--dynamic ios
```

2. Copy the output file ( opencv2.framework ) to “OpenCVForUnity\Plugins\iOS”.

### Windows

1. Build the OpenCV dynamic library.

```
OPENCV_EXTRA_MODULES_PATH:PATH=C:/Users/xxxxxx/opencv_contrib/modules
BUILD_SHARED_LIBS:BOOL=ON
```

2. Set PATH variable to “C:\path\to\opencv\x64\vc15\bin”.
3. Copy “OpenCVForUnity\Extra\dll\_version\Windows\” to “OpenCVForUnity\Plugins\Windows\”.

### macOS

1. Build the OpenCV library.

```
CMAKE_OSX_ARCHITECTURESSTRING=x86_64
OPENCV_EXTRA_MODULES_PATHPATH=/Users/xxxxxx/opencv_contrib/modules
```

2. Copy the output files ( libopencv\_\*.4.1.0.dylib ) to

- opencvforunity.bundle\Contents\MacOS\
3. Relink libopencv\_\*.4.1.0.dylib using otool and install\_name\_tool.  
<http://phenixyu.blogspot.com/2016/09/how-to-load-dynamic-library-with-unity.html>

```
//example : aruco module
otool -L libopencv_aruco.4.1.0.dylib

install_name_tool -id @loader_path/libopencv_aruco.4.1.0.dylib
libopencv_aruco.4.1.0.dylib

install_name_tool -change @rpath/libopencv_calib3d.4.1.dylib
@loader_path/libopencv_calib3d.4.1.0.dylib libopencv_aruco.4.1.0.dylib

install_name_tool -change @rpath/libopencv_features2d.4.1.dylib
@loader_path/libopencv_features2d.4.1.0.dylib libopencv_aruco.4.1.0.dylib

install_name_tool -change @rpath/libopencv_flann.4.1.dylib
@loader_path/libopencv_flann.4.1.0.dylib libopencv_aruco.4.1.0.dylib

install_name_tool -change @rpath/libopencv_highgui.4.1.dylib
@loader_path/libopencv_highgui.4.1.0.dylib libopencv_aruco.4.1.0.dylib

install_name_tool -change @rpath/libopencv_videoio.4.1.dylib
@loader_path/libopencv_videoio.4.1.0.dylib libopencv_aruco.4.1.0.dylib

install_name_tool -change @rpath/libopencv_imgcodecs.4.1.dylib
@loader_path/libopencv_imgcodecs.4.1.0.dylib libopencv_aruco.4.1.0.dylib

install_name_tool -change @rpath/libopencv_imgproc.4.1.dylib
@loader_path/libopencv_imgproc.4.1.0.dylib libopencv_aruco.4.1.0.dylib

install_name_tool -change @rpath/libopencv_core.4.1.dylib
@loader_path/libopencv_core.4.1.0.dylib libopencv_aruco.4.1.0.dylib
```

## Linux

1. Build the OpenCV library.
2. Rename output files ( libopencv\_\*.so.4.1.0 ).

```
sudo apt-get install rpl
rpl -R -e .so.4.1 "_4_1.so" libopencv_*.so.4.1.0
rename "s/".so.4.1.0"/"_4_1.so"/;" libopencv_*.so.4.1.0
```

3. Copy libopencv\_\*\_4\_0.so to “/OpenCVForUnity/Plugins/Linux/x86\_64” folder.

## UWP

1. Build the OpenCV library.

```
cd C:\Users\satoo\Desktop\opencv\platforms\winrt
setup_winrt.bat "WS" "10.0" "x64"
```

2. Copy “install\WS\10.0\ARM\ARM\vc15\bin” to “OpenCVForUnity\Plugins\WSA\UWP\ARM”.  
Copy “install\WS\10.0\x64\x64\vc15\bin” to “OpenCVForUnity\Plugins\WSA\UWP\x64”.  
Copy “install\WS\10.0\x86\x86\vc15\bin” to “OpenCVForUnity\Plugins\WSA\UWP\x86”.

## Q & A

### Q1.

Asset package size is large. Is there a way to reduce?

### A1.

Please remove plugin folders of non-output target platforms that are included in OpenCVforUnity package. You do not need to import plug-in files for platforms not supported by your project.

### Q2.

Support Web platform?

### A2.

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

A WebGL platform was added as an alternative. (Unity 5.3 or higher).

### Q3.

How do learn the details of OpenCV's method and argument?

### A3.

Please refer to OpenCV official document (<http://docs.opencv.org/4.1.0/index.html>) and OpenCV Tutorials ([http://docs.opencv.org/4.1.0/d9/df8/tutorial\\_root.html](http://docs.opencv.org/4.1.0/d9/df8/tutorial_root.html)) for the details of the argument of the method..

### Q4.

How can I convert Mat class operators defined in C++?

### A4.

Way to translation of Mat class operators defined in C++.

<https://enoxsoftware.com/opencvforunity/way-to-translation-of-mat-class-operators-defined-in-cpp/>

### Q5.

"DllNotFoundException: opencvforunity" is displayed on the console when run the example scene.

### A5.

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

### Q6.

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

### A6.

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

### Q7.

"Level 'Texture2DtoMatExample' (-1) could not be loaded because it has not been added to the build settings." is displayed on the console when run the example scene.

### A7.

Please add all of "\*\*\*\*.unity" scenes into the "Assets/OpenCVForUnity/Examples" folder to

[Build Settings] – [Scene In Build].

**Q8.**

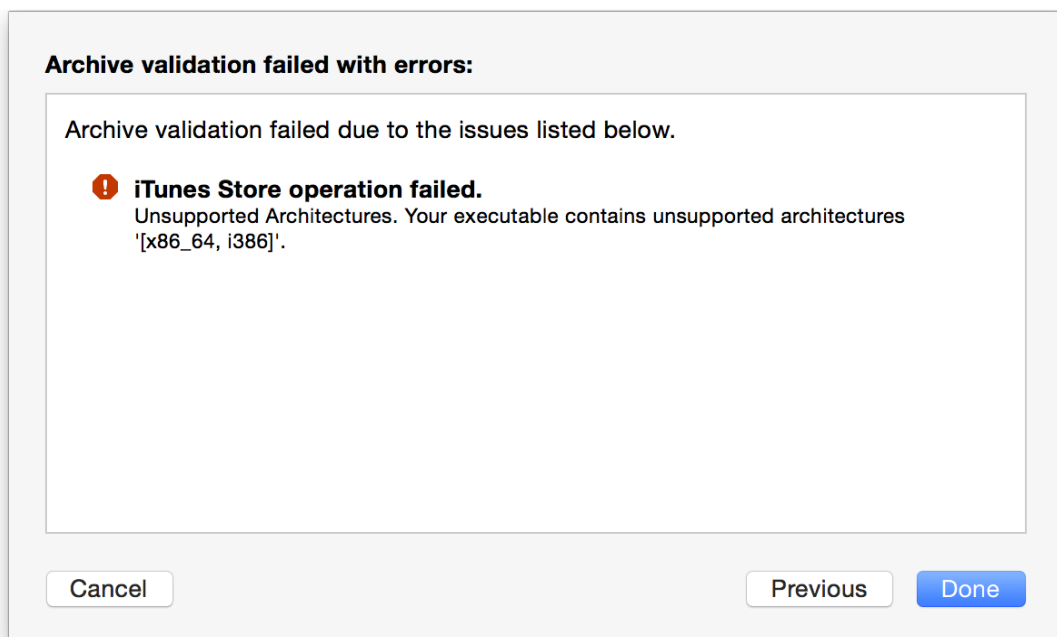
In DetectFaceExample or WebCamTextureDetectFaceExample, red rectangle is not displayed around a face.

**A8.**

You might have failed to read the “haarcascade\_frontalface\_alt.xml”. Please confirm whether there is the “StreamingAssets” folder at the right position.

**Q9.**

[iOS]Submit to App Store issues: Unsupported Architecture x86, i386“Unsupported Architecture. Your executable contains unsupported architecture '[x86\_64, i386]'.”



**A9.**

<http://ioscake.com/submit-to-app-store-issues-unsupported-architecture-x86.html>

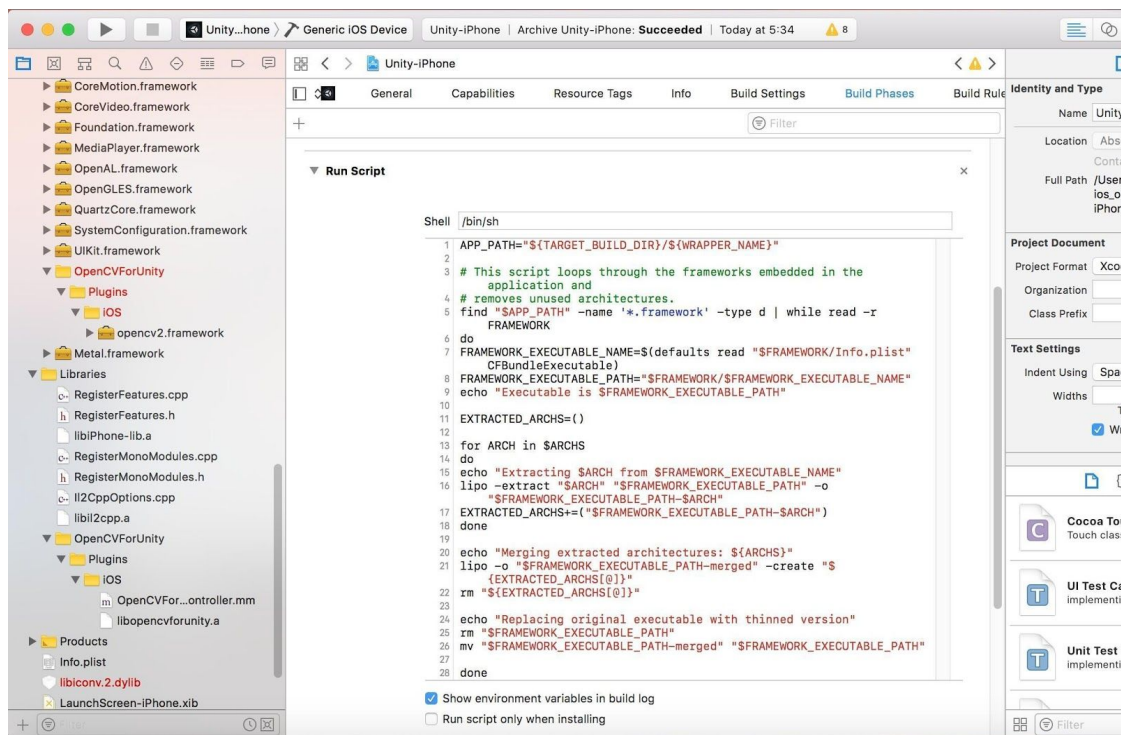
“The problem is that the Buy framework contains a build for both the simulator (x86\_64) and the actual devices (ARM).

Of course, you aren’t allowed to submit to the App Store a binary for an unsupported architecture, so the solution is to “manually” remove the unneeded architectures from the final binary, before submitting it.”

There are **two ways** to solve this error.

1. Please add the script of this page to BuildPhases->RunScript.

<http://ikennd.ac/blog/2015/02/stripping-unwanted-architectures-from-dynamic-libraries-in-xcode/>



2. Please execute the following command on terminal.

[https://stackoverflow.com/questions/42641806/check-and-remove-unsupported-architecture-x86\\_64-i386-in-ipa-archive](https://stackoverflow.com/questions/42641806/check-and-remove-unsupported-architecture-x86_64-i386-in-ipa-archive)

//remove i386 architectures.

`lipo -remove i386 opencv2.framework/opencv2 -o opencv2.framework/opencv2`

//remove x86\_64 architectures.

`lipo -remove x86_64 opencv2.framework/opencv2 -o opencv2.framework/opencv2`

//check the architectures.

`lipo -info opencv2.framework/opencv2`

**Q10**

How can I use SIFT or SURF algorithms?

**A10.**

The native library included in OpenCVForUnity is built with the `OPENCV_ENABLE_NONFREE` flag disabled. To use the SIFT and SURF algorithms, rebuild OPENCV library with `OPENCV_ENABLE_NONFREE` enabled. For more details, see the section on “How to use OpenCV Dynamic Link Library with customized build settings” in this document.

**Q11.**

How to catch native OpenCV's errors code (CvException handling)

**A11.**

<https://enoxsoftware.com/opencvforunity/how-to-catch-native-opencvs-errors-code/>