

# Tinker board

IoT를 위한 최선의 엣지 단말기



MAKER SPACE  
**G·CAMP**

# Contents

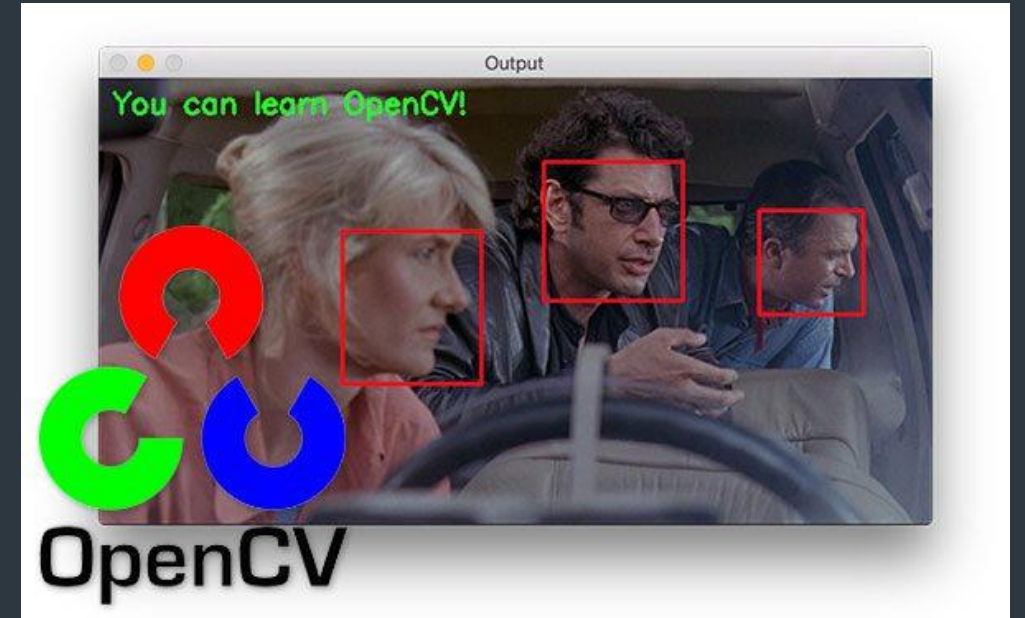
- OpenCV 소개
- OpenCV 다운로드 및 import
- OpenCV Camera View 출력
- OpenCV method 실습

\*eMMC by SKU

# OpenCV 소개

# OpenCV

- Computer Vision을 위한 오픈 소스 라이브러리
- 2차원 혹은 3차원의 이미지 데이터를 처리
- 이미지 프레임 내에서 원하는 영역 혹은 물체를 찾기 위해 사용



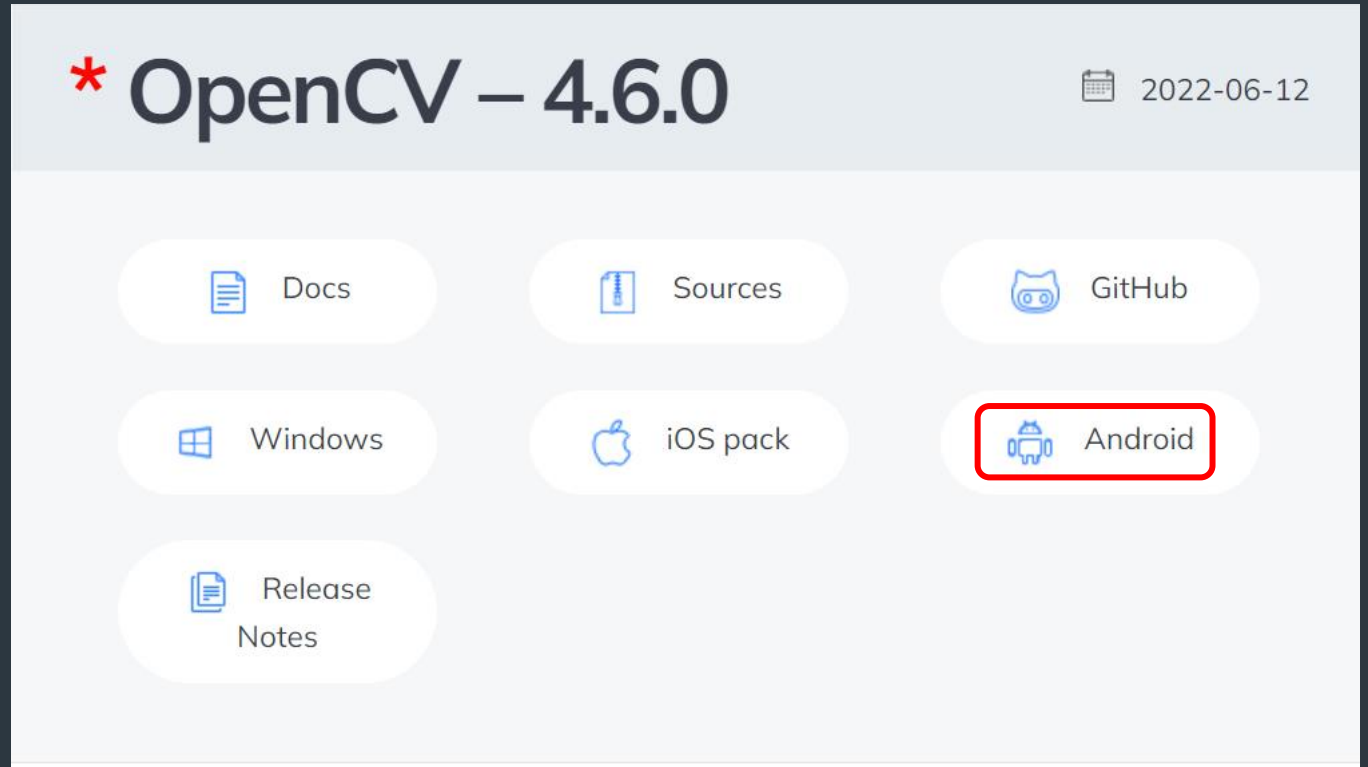
# OpenCV 다운로드 및 import

# 카메라 연결

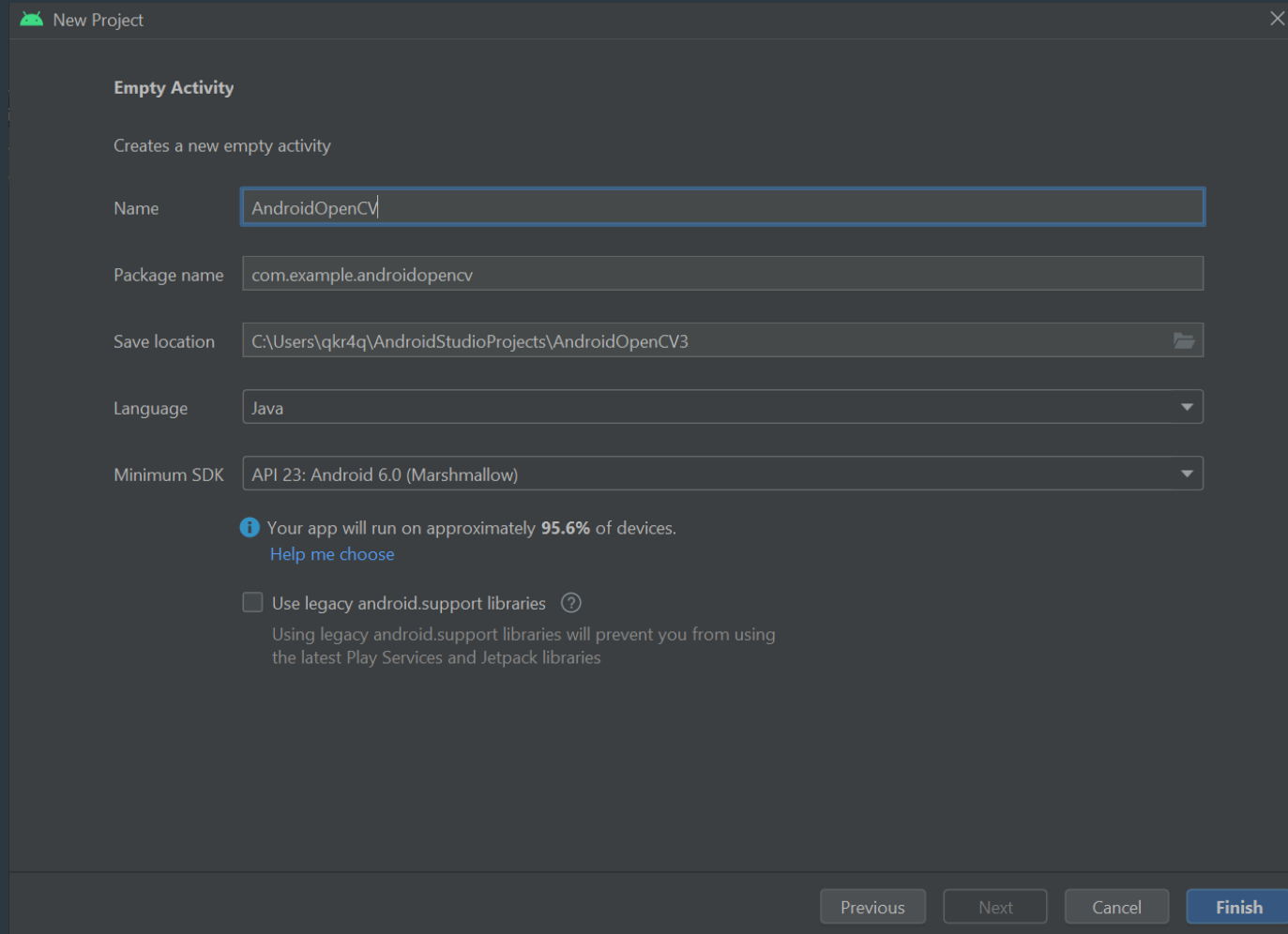


# OpenCV download

1. 링크를 클릭하여 페이지 접속 – [link](#)
2. Android 아이콘 클릭하여 다운로드
3. 다운로드 후 압축 풀기



# 안드로이드 프로젝트 생성



**New Project**

**Empty Activity**

Creates a new empty activity

Name:

Package name:

Save location:

Language:

Minimum SDK:

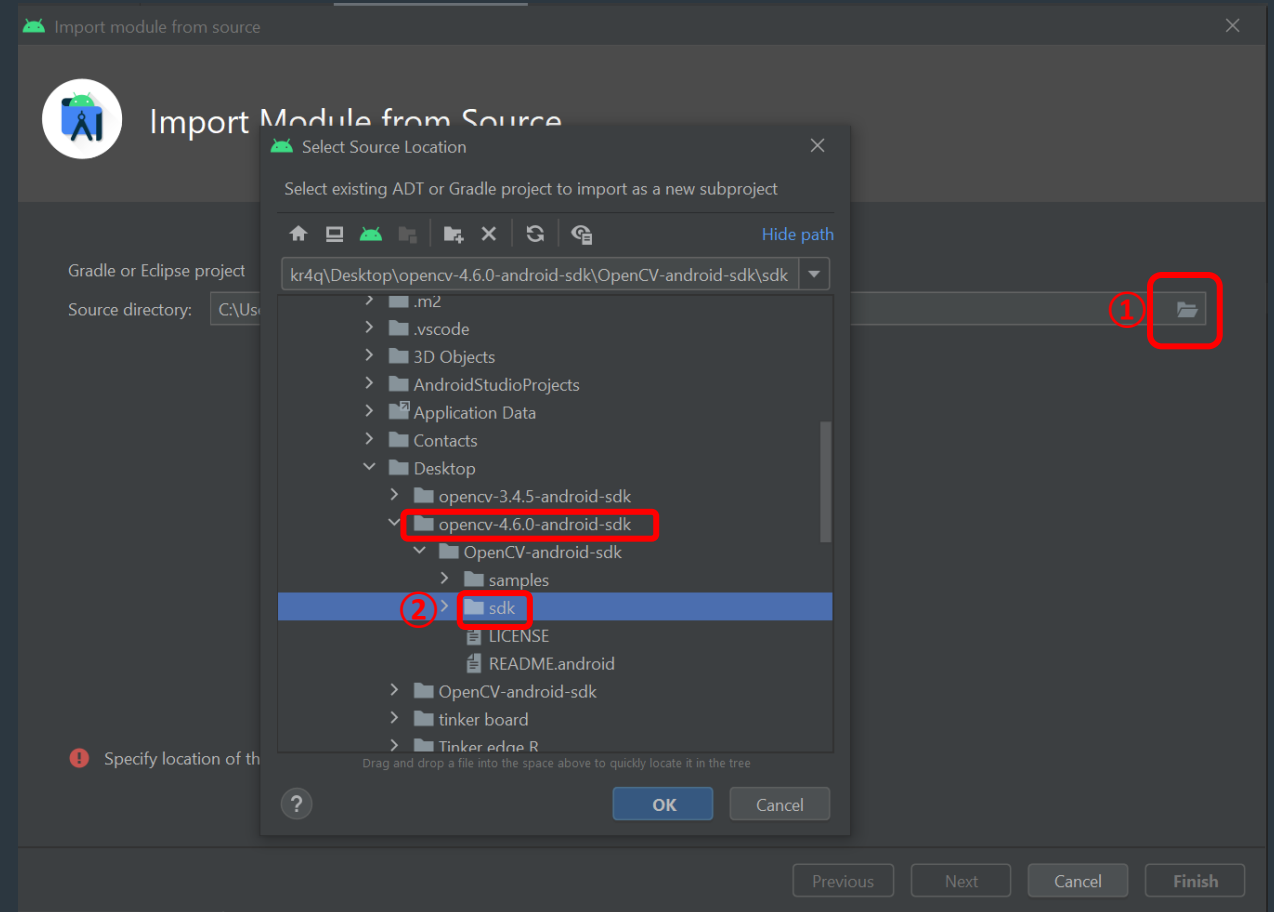
**i** Your app will run on approximately **95.6%** of devices.  
[Help me choose](#)

☐ Use legacy android.support libraries **?**  
Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries



# Import sdk

1. File → New → Import Modules
2. 디렉토리 위치 선택: OpenCV 폴더 내에 sdk 선택
3. Module name을 “:sdk”을 “OpenCV”로 수정

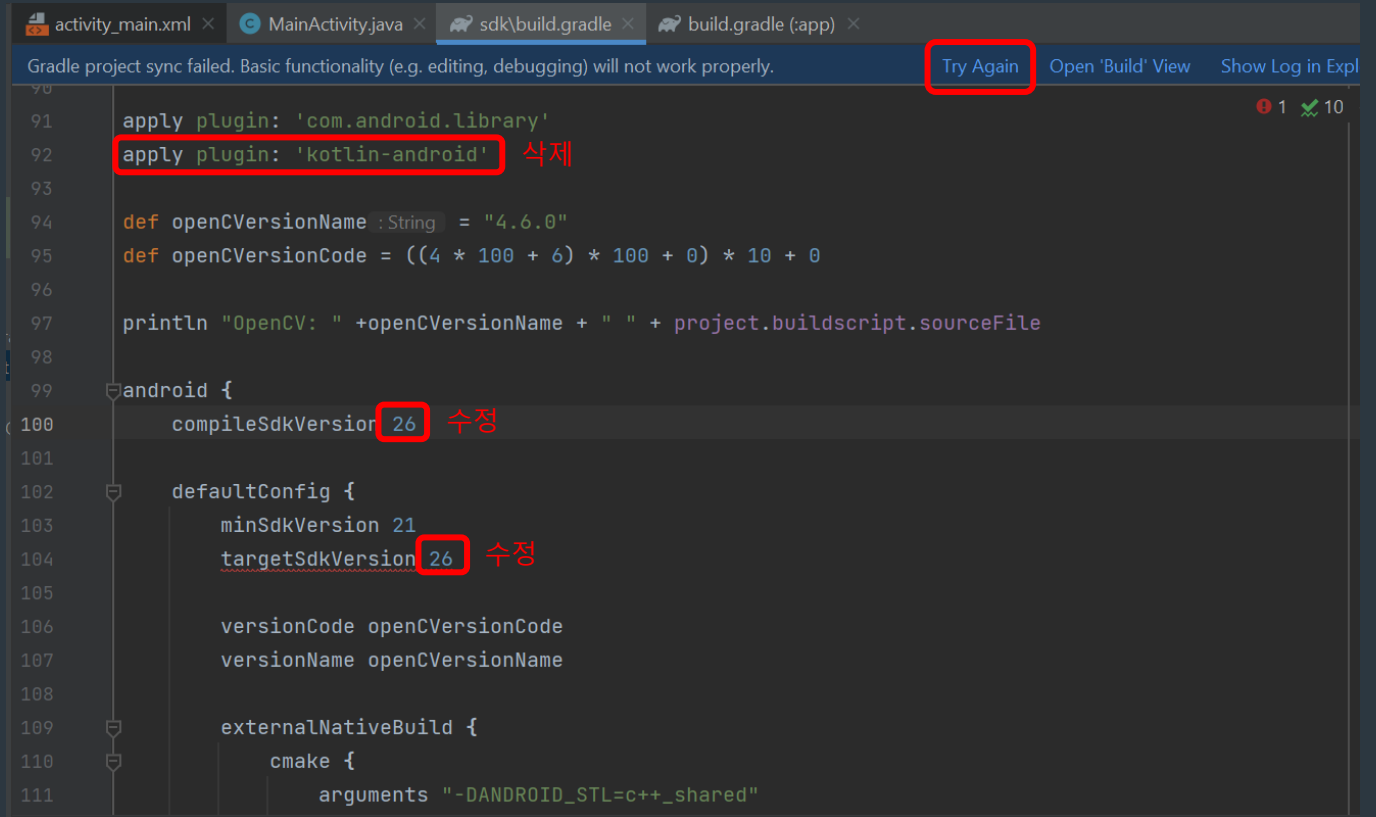


# build.gradle 파일 수정

sdk 폴더를 import 하고 나면 오류가 발생

## 해결 방법

1. 92번째 line 삭제
2. compileSdkVersion 수정
3. targetSdkVersion 수정
4. Try Again을 클릭하여 gradle sync하기

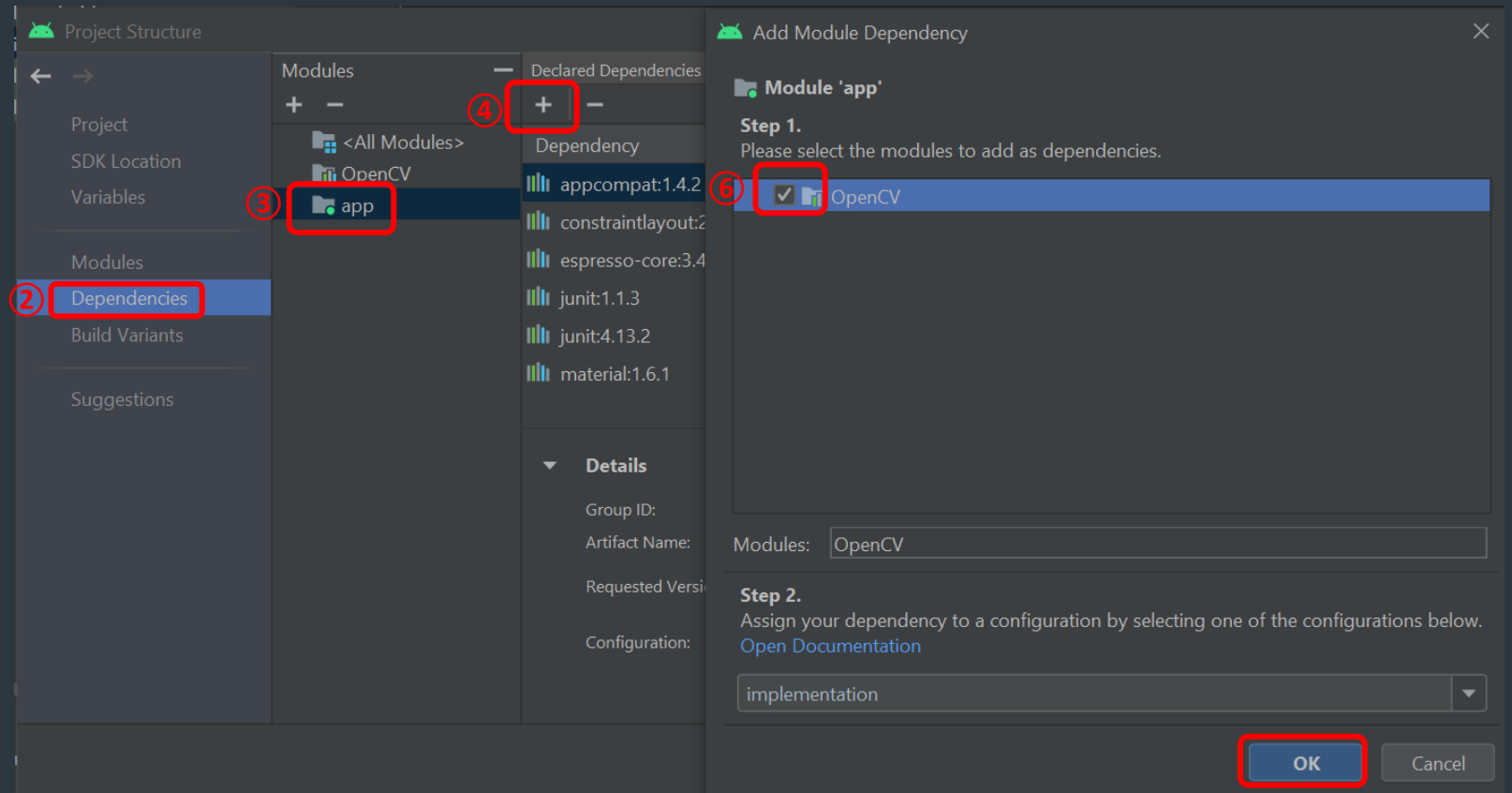


The screenshot shows the Android Studio IDE with the `build.gradle` file open. A red box highlights the `Try Again` button in the top bar. The code is as follows:

```
91 apply plugin: 'com.android.library'
92 apply plugin: 'kotlin-android' 삭제
93
94 def openCVVersionName :String = "4.6.0"
95 def openCVVersionCode = ((4 * 100 + 6) * 100 + 0) * 10 + 0
96
97 println "OpenCV: " + openCVVersionName + " " + project.buildscript.sourceFile
98
99 android {
100     compileSdkVersion 26 수정
101
102     defaultConfig {
103         minSdkVersion 21
104         targetSdkVersion 26 수정
105
106         versionCode openCVVersionCode
107         versionName openCVVersionName
108
109         externalNativeBuild {
110             cmake {
111                 arguments "-DANDROID_STL=c++_shared"
```

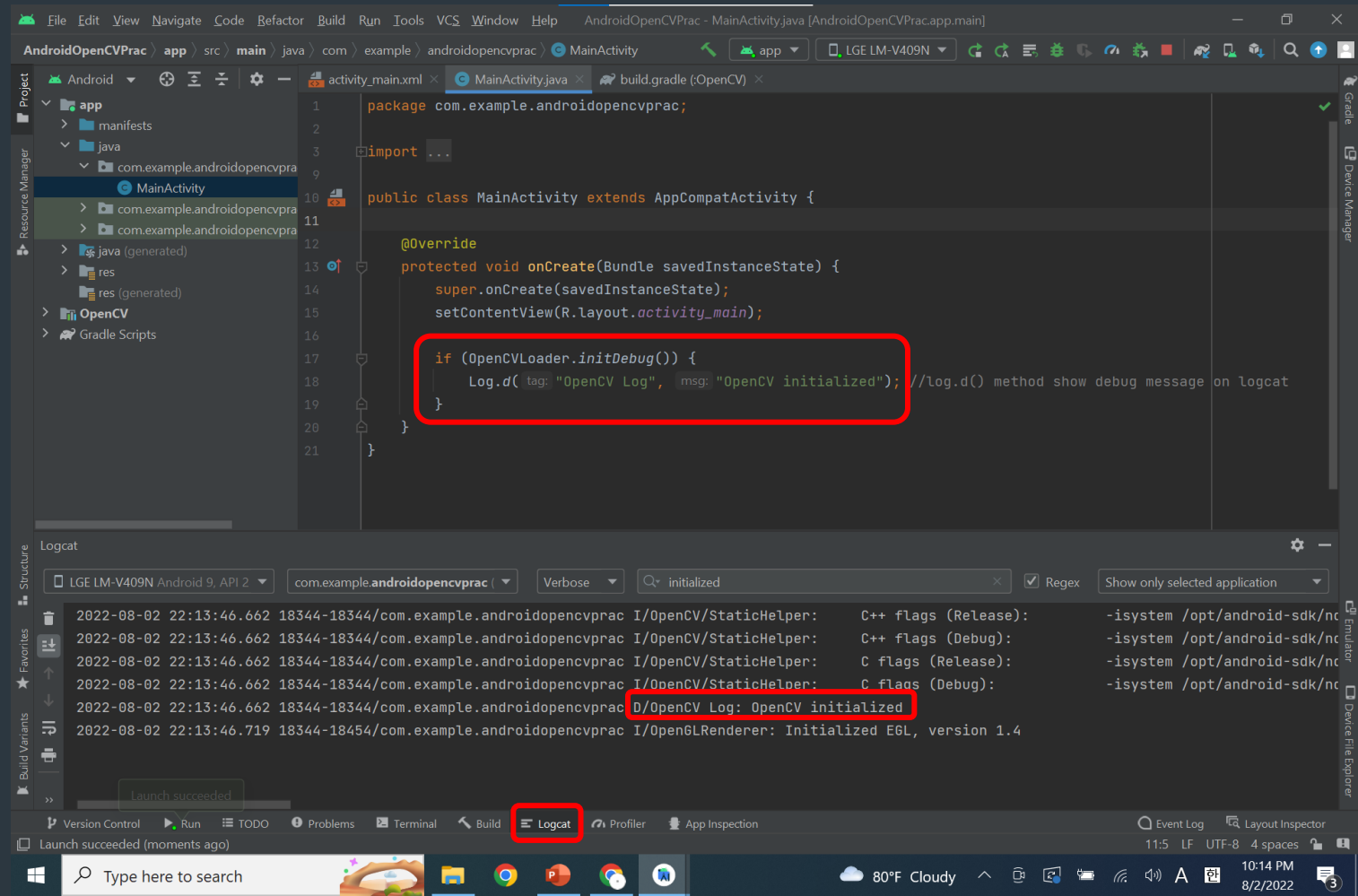
# Module Dependency 추가

1. File → Project Structure
2. 왼쪽 바에서 Dependencies 클릭
3. module에서 app 클릭
4. “+” sign 선택
5. module dependency 선택
6. OpenCV 체크하고 Ok
7. Apply 클릭 OK



# OpenCV 라이브러리 loading 확인

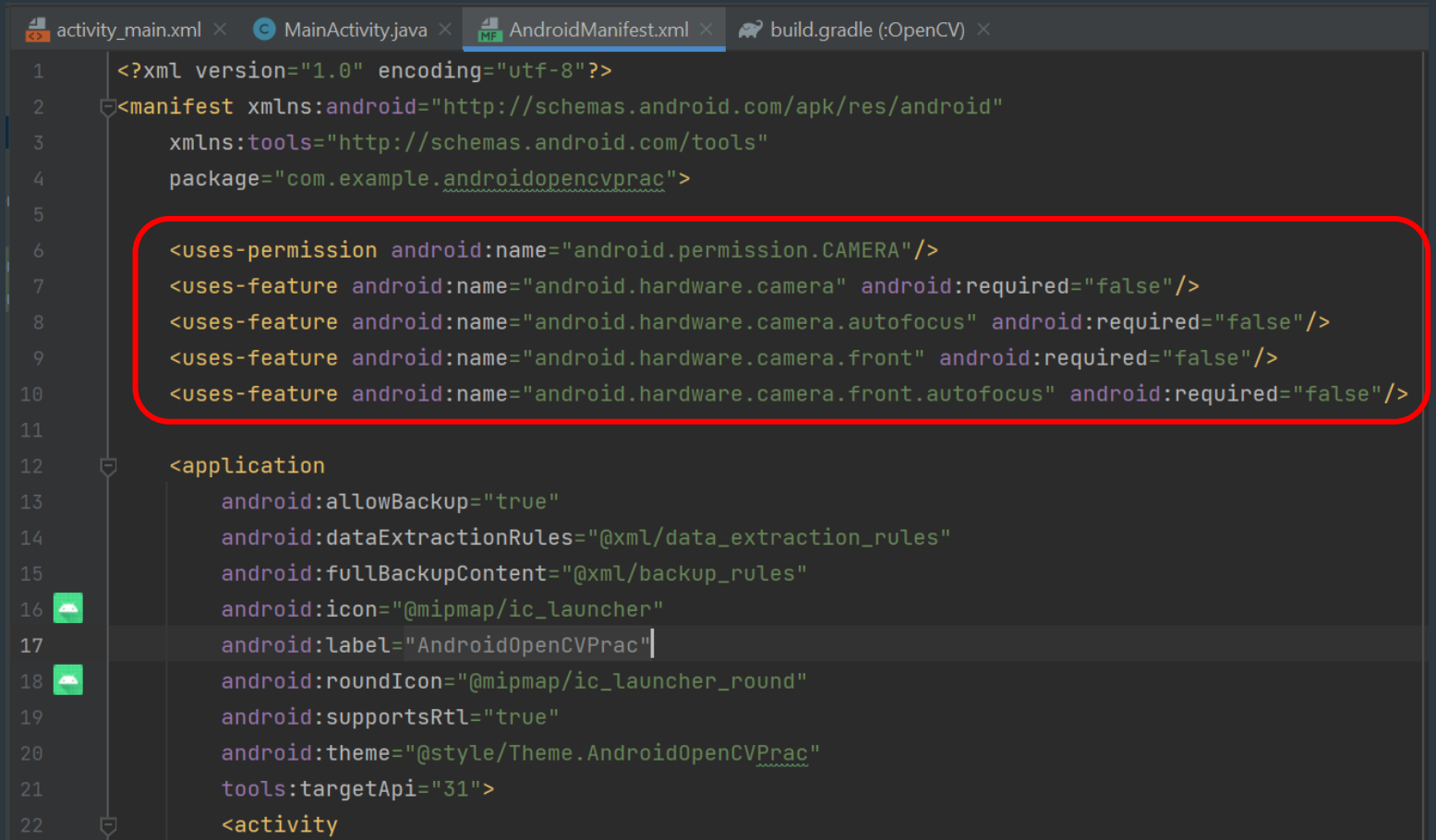
1. MainActivity.java에서 오른쪽 이미지의 if block 작성
2. 툴킷보드가 연결된 상태에서 RUN
3. LogCat에서 debug message 확인



# OpenCV Camera View 출력 코드 작성

# Manifest 파일에서 카메라 관련 permission 추가

1. AndroidManifest.xml 파일 열기
2. 오른쪽 빨간 블록 코드 추가



```
activity_main.xml x MainActivity.java x MF AndroidManifest.xml x build.gradle (:OpenCV) x
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     package="com.example.androidopencvprac">
5
6     <uses-permission android:name="android.permission.CAMERA"/>
7     <uses-feature android:name="android.hardware.camera" android:required="false"/>
8     <uses-feature android:name="android.hardware.camera.autofocus" android:required="false"/>
9     <uses-feature android:name="android.hardware.camera.front" android:required="false"/>
10    <uses-feature android:name="android.hardware.camera.front.autofocus" android:required="false"/>
11
12    <application
13        android:allowBackup="true"
14        android:dataExtractionRules="@xml/data_extraction_rules"
15        android:fullBackupContent="@xml/backup_rules"
16        android:icon="@mipmap/ic_launcher"
17        android:label="@string/app_name"
18        android:roundIcon="@mipmap/ic_launcher_round"
19        android:supportRtl="true"
20        android:theme="@style/Theme.AndroidOpenCVPrac"
21        tools:targetApi="31">
22        <activity
```

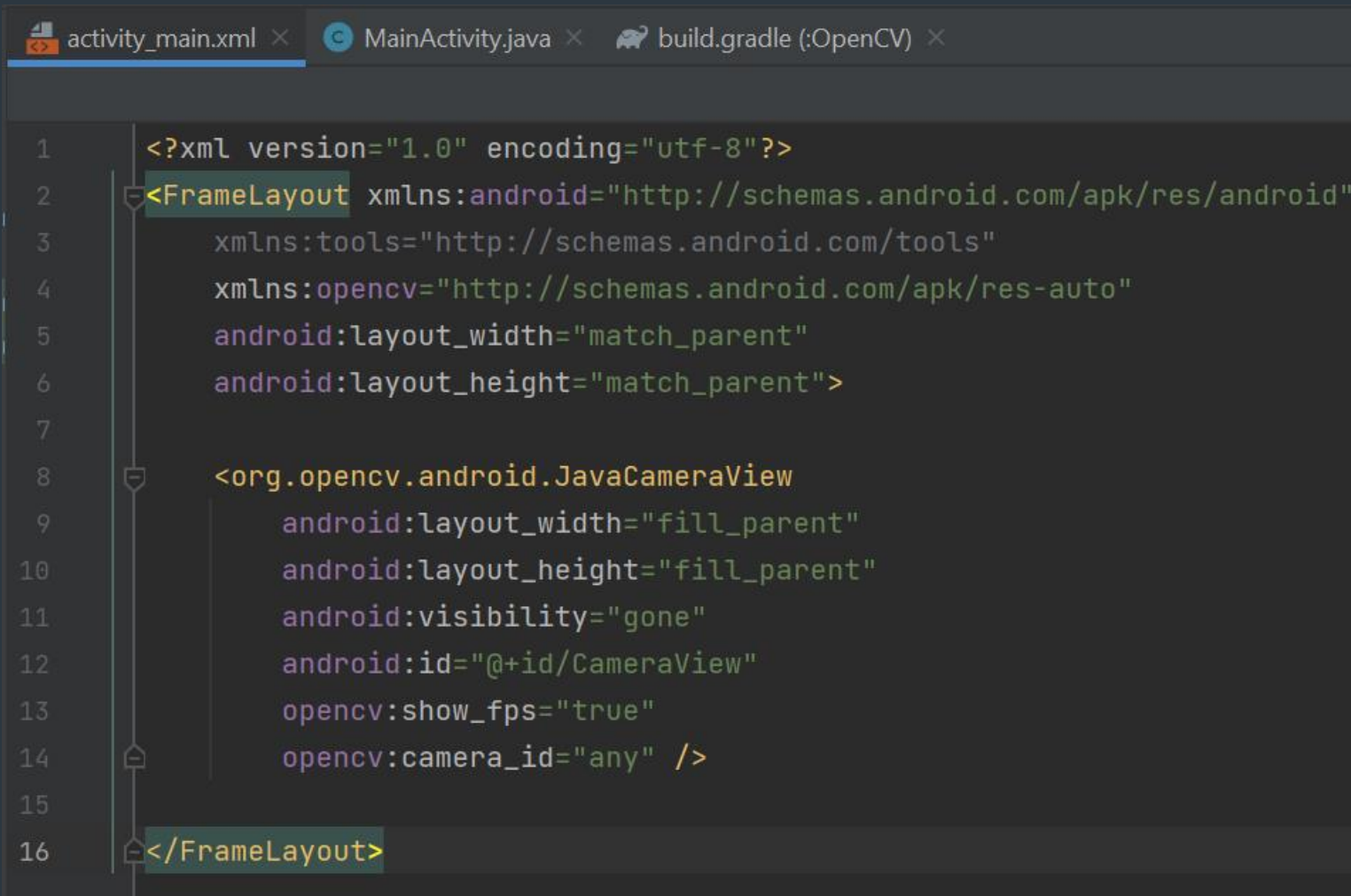
# Manifest 파일에서 카메라 관련 permission 추가

아래 코드 copy and paste

```
<uses-permission android:name="android.permission.CAMERA"/>
<uses-feature android:name="android.hardware.camera" android:required="false"/>
<uses-feature android:name="android.hardware.camera.autofocus" android:required="false"/>
<uses-feature android:name="android.hardware.camera.front" android:required="false"/>
<uses-feature android:name="android.hardware.camera.front.autofocus" android:required="false"/>
```

# Layout 파일 수정

1. activity\_main.xml 파일 열기
2. 오른쪽 이미지와 같이 코드 수정



```
1  <?xml version="1.0" encoding="utf-8"?>
2  <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      xmlns:opencv="http://schemas.android.com/apk/res-auto"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent">
7
8      <org.opencv.android.JavaCameraView
9          android:layout_width="fill_parent"
10         android:layout_height="fill_parent"
11         android:visibility="gone"
12         android:id="@+id/CameraView"
13         opencv:show_fps="true"
14         opencv:camera_id="any" />
15
16  </FrameLayout>
```



# Layout 파일 수정

아래 코드 copy and paste

constraint layout에서 Frame Layout으로 변경 ←

opencv 라이브러리에 있는 속성을 사용←

CameraView를 띄우기 위한 속성태그←

오른쪽 상단에 현재 fps 표시←  
camera\_id의 값은 any, back, front 값을 가짐←

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:opencv="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <org.opencv.android.JavaCameraView
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:visibility="gone"
        android:id="@+id/CameraView"
        opencv:show_fps="true"
        opencv:camera_id="any" />

</FrameLayout>
```

# 자바 소스코드 작성 1

1. MainActivity.java 파일 열기
2. 상속받는 class를 AppCompatActivity에서 CameraActivity로 변경
3. CameraBridgeViewBase 참조 변수 선언

```
activity_main.xml x MainActivity.java x AndroidManifest.xml x build.gradle (:OpenCV) x
1 package com.example.androidopencvprac;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Bundle;
6 import android.util.Log;
7
8 import org.opencv.android.BaseLoaderCallback;
9 import org.opencv.android.CameraActivity;
10 import org.opencv.android.CameraBridgeViewBase;
11 import org.opencv.android.OpenCVLoader;
12
13 public class MainActivity extends CameraActivity { ②
14     private CameraBridgeViewBase cameraBridgeViewBase; ③
15
16     @Override
17     protected void onCreate(Bundle savedInstanceState) {
```

# 자바 소스코드 작성 2

1. BaseLoaderCallback 인스턴스 생성
2. onManagerConnected 오버라이드
3. if 구문 작성

```
activity_main.xml x MainActivity.java x AndroidManifest.xml x build.gradle (:OpenCV) x
14
15 public class MainActivity extends CameraActivity {
16     private CameraBridgeViewBase cameraBridgeViewBase;
17
18     ① private BaseLoaderCallback baseLoaderCallback = new BaseLoaderCallback( AppContext: this) {
19         @Override
20         ② public void onManagerConnected(int status) {
21             if (status == LoaderCallbackInterface.SUCCESS) {
22                 Log.v( tag: "OpenCV Log", msg: "OpenCV initialized");
23                 cameraBridgeViewBase.enableView(); //connection이 성공이면 cameraview 활성화
24             } else {
25                 ③ super.onManagerConnected(status);
26             }
27         }
28     };
```

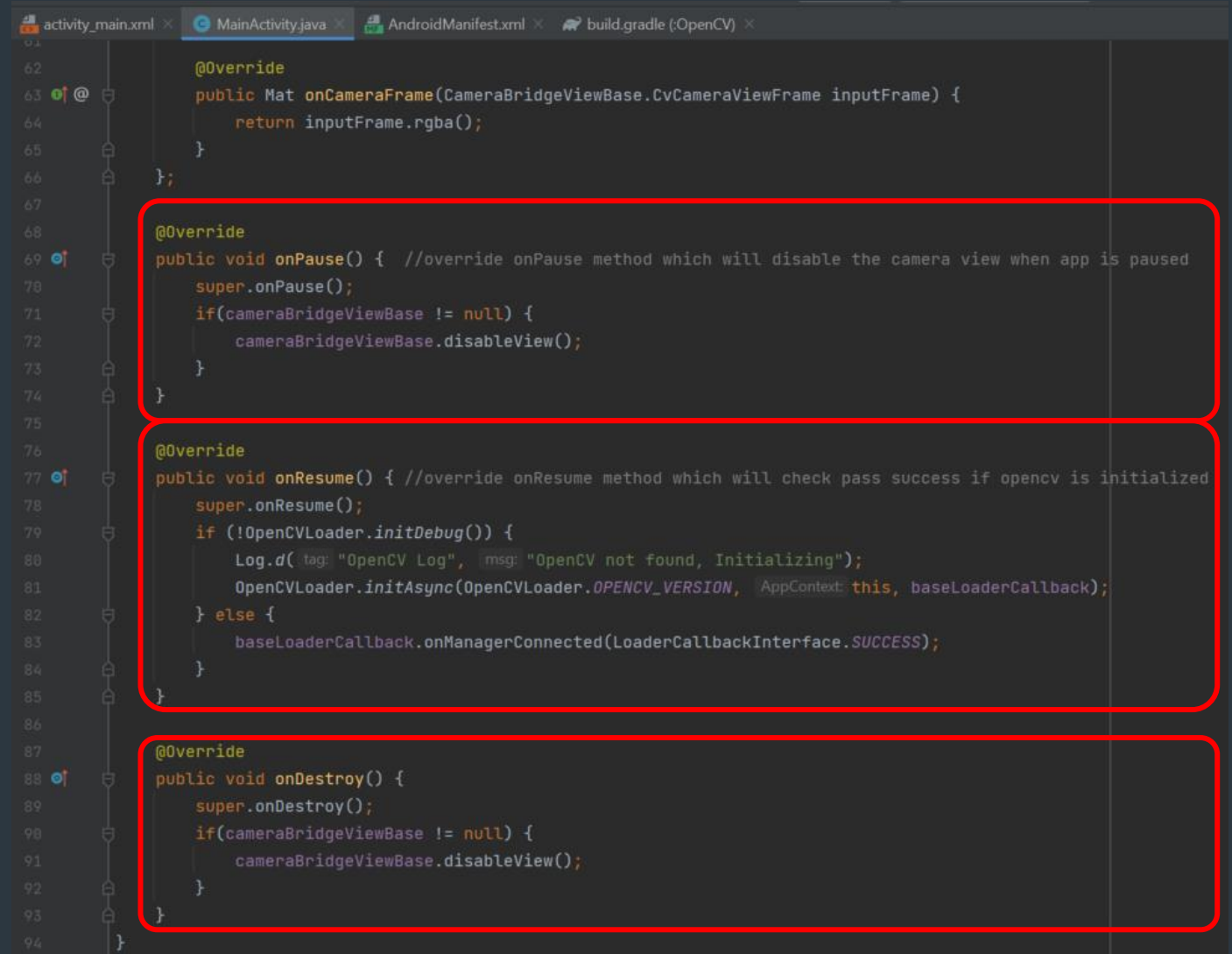
# 자바 소스코드 작성 3

1. layout에서 CameraView와 연결
2. camera view 가시화
3. cvCameraListener 인스턴스 passing
4. 모든 camera view를 get하는
5. cvCameraListner 인스턴스 생성
6. onCameraFrame의 return 값 변경

```
36      @Override
37      protected void onCreate(Bundle savedInstanceState) {
38          super.onCreate(savedInstanceState);
39          setContentView(R.layout.activity_main);
40
41          ① cameraBridgeViewBase = (CameraBridgeViewBase) findViewById(R.id.CameraView);
42          ② cameraBridgeViewBase.setVisibility(SurfaceView.VISIBLE);
43          ③ cameraBridgeViewBase.setCvCameraViewListener(cvCameraViewListener);
44      }
45
46      @Override
47      protected List<?extends CameraBridgeViewBase> getCameraViewList() { //override the method to get all camera views for opencv camera
48          ④ return Collections.singletonList(cameraBridgeViewBase);
49      }
50
51      ⑤ private CameraBridgeViewBase.CvCameraViewListener2 cvCameraViewListener = new CameraBridgeViewBase.CvCameraViewListener2() {
52          @Override
53          public void onCameraViewStarted(int width, int height) {
54
55          }
56
57          @Override
58          public void onCameraViewStopped() {
59
60          }
61
62          @Override
63          public Mat onCameraFrame(CameraBridgeViewBase.CvCameraViewFrame inputFrame) {
64              ⑥ return inputFrame.rgba();
65          }
66      };
```

# 자바 소스코드 작성 4

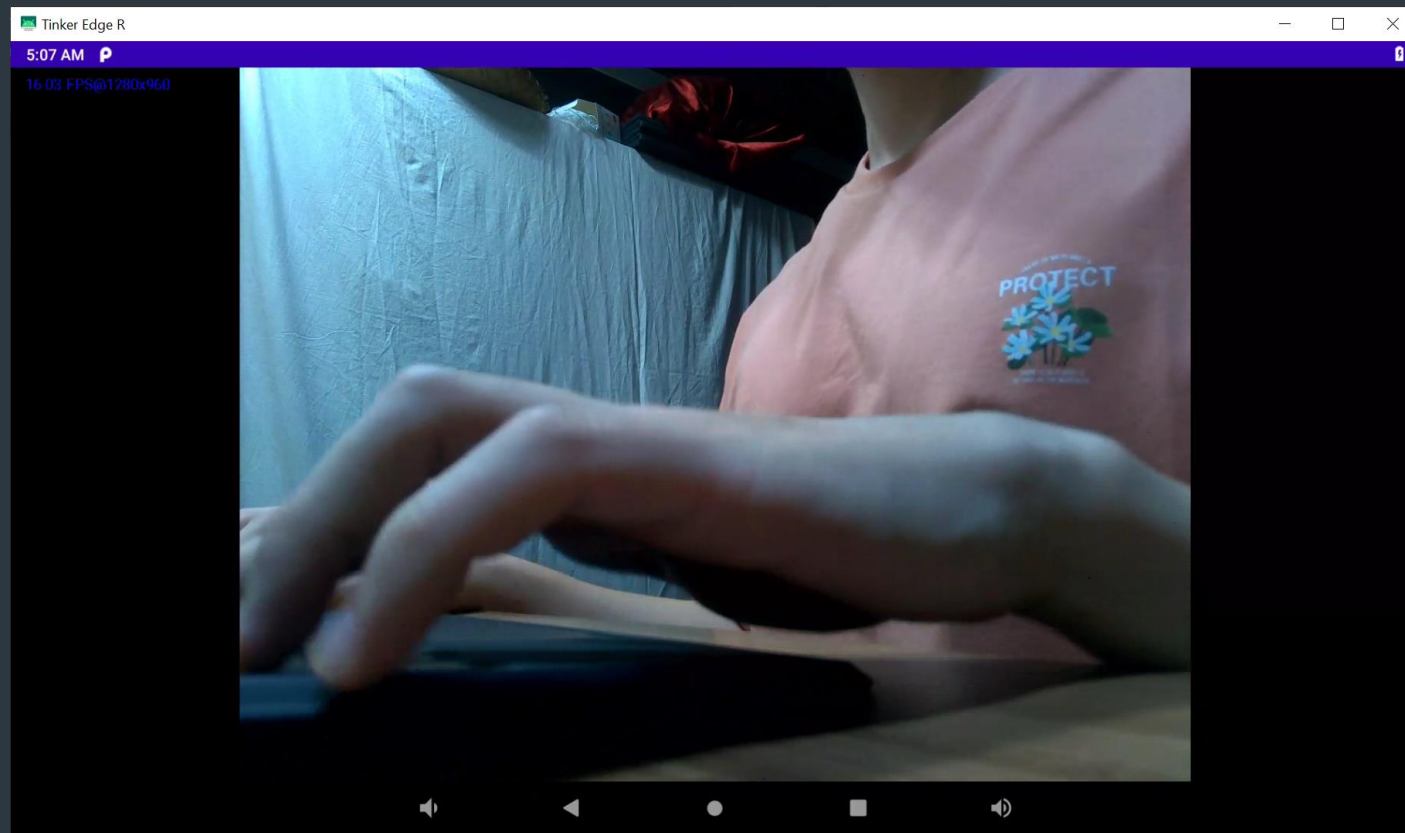
1. onPause 함수 재정의의 추가
2. onResume 함수 재정의의 추가
3. onDestroy 함수 재정의의 추가



```
activity_main.xml x MainActivity.java x AndroidManifest.xml x build.gradle (:OpenCV) x
62
63 @
64
65
66 };
67
68
69 @Override
70 public void onPause() { //override onPause method which will disable the camera view when app is paused
71     super.onPause();
72     if(cameraBridgeViewBase != null) {
73         cameraBridgeViewBase.disableView();
74     }
75 }
76
77 @Override
78 public void onResume() { //override onResume method which will check pass success if opencv is initialized
79     super.onResume();
80     if (!OpenCVLoader.initDebug()) {
81         Log.d( tag: "OpenCV Log", msg: "OpenCV not found, Initializing");
82         OpenCVLoader.initAsync(OpenCVLoader.OPENCV_VERSION, AppContext: this, baseLoaderCallback);
83     } else {
84         baseLoaderCallback.onManagerConnected(LoaderCallbackInterface.SUCCESS);
85     }
86 }
87
88 @Override
89 public void onDestroy() {
90     super.onDestroy();
91     if(cameraBridgeViewBase != null) {
92         cameraBridgeViewBase.disableView();
93     }
94 }
```

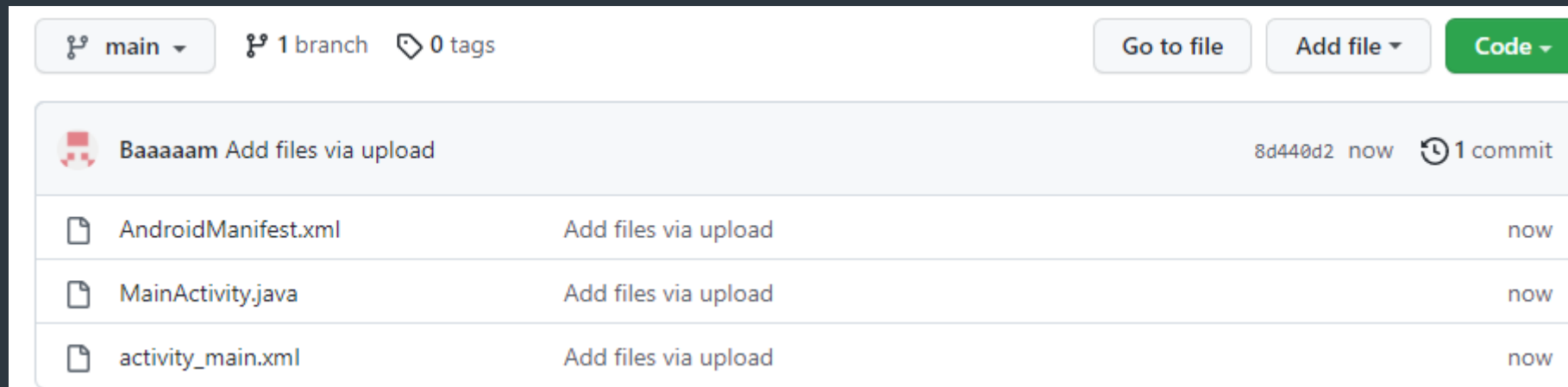
# Camera View 출력 테스트

안드로이드 스튜디오에서 실행버튼을 눌러 앱 실행 테스트



# 자바소스코드, Layout, Manifest 파일 공유

[github 링크](#)





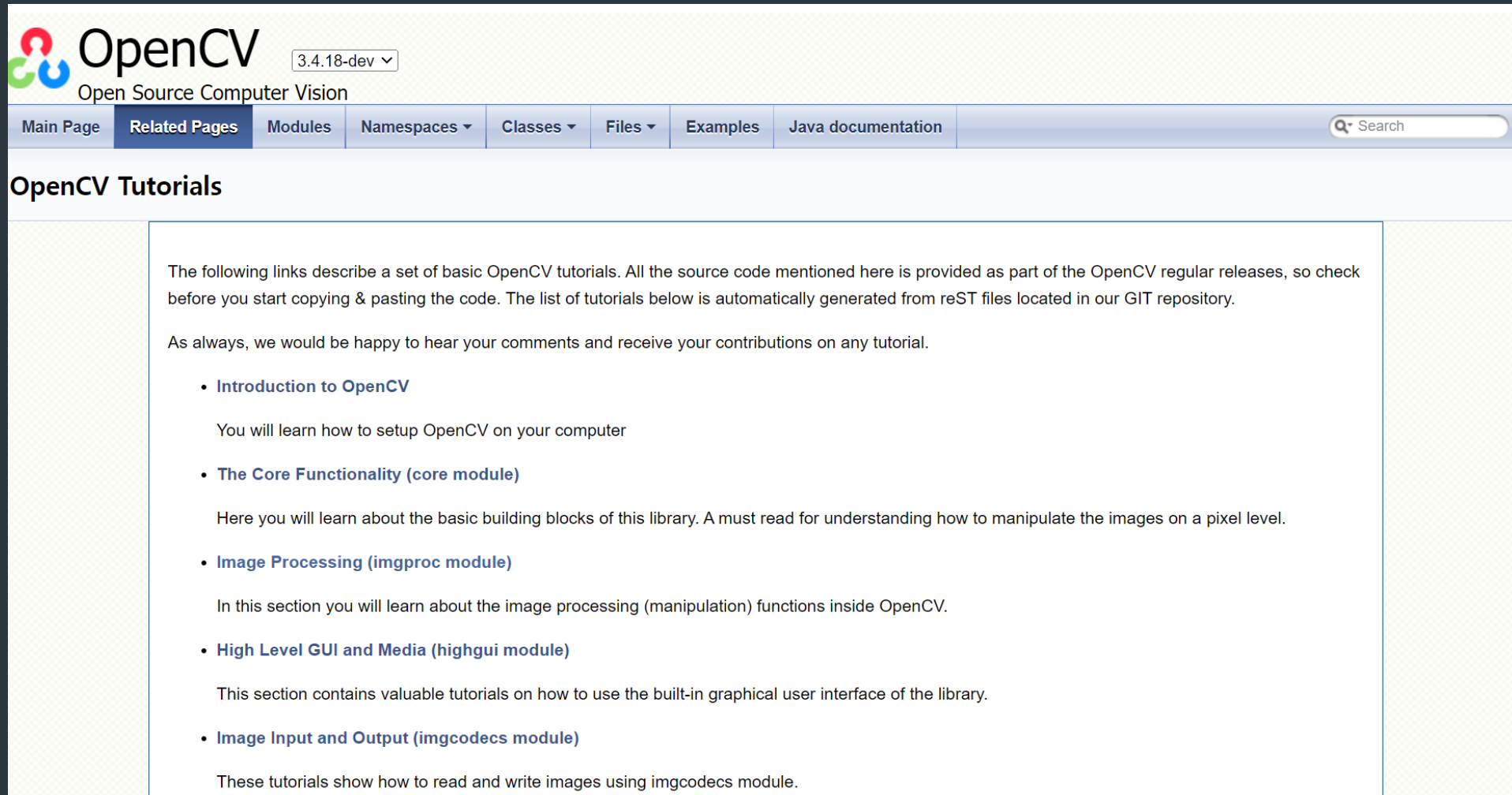
# OpenCV methods 실습





# OpenCV 공식 documentation

[https://docs.opencv.org/3.4/d9/df8/tutorial\\_root.html](https://docs.opencv.org/3.4/d9/df8/tutorial_root.html)

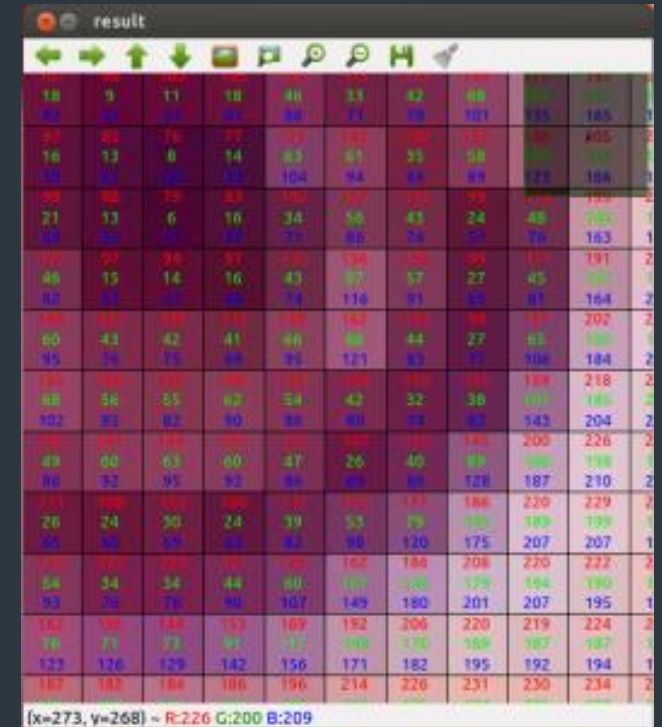


The screenshot shows the OpenCV official documentation website. At the top, there is the OpenCV logo and the text 'OpenCV Open Source Computer Vision'. A dropdown menu shows '3.4.18-dev'. Below this is a navigation bar with links: 'Main Page', 'Related Pages', 'Modules', 'Namespaces', 'Classes', 'Files', 'Examples', and 'Java documentation'. A search bar is on the right. The main content area is titled 'OpenCV Tutorials'. It contains a paragraph explaining that the following links describe a set of basic OpenCV tutorials, all of which are provided as part of the OpenCV regular releases. It also mentions that the list of tutorials is automatically generated from reST files located in the GIT repository. Below this, there is a list of tutorials with their descriptions:

- **Introduction to OpenCV**  
You will learn how to setup OpenCV on your computer
- **The Core Functionality (core module)**  
Here you will learn about the basic building blocks of this library. A must read for understanding how to manipulate the images on a pixel level.
- **Image Processing (imgproc module)**  
In this section you will learn about the image processing (manipulation) functions inside OpenCV.
- **High Level GUI and Media (highgui module)**  
This section contains valuable tutorials on how to use the built-in graphical user interface of the library.
- **Image Input and Output (imgcodecs module)**  
These tutorials show how to read and write images using imgcodecs module.

# Mat class

- OpenCV에서 가장 많이 사용되는 클래스
- 행렬 (Matrix)를 표현하기 위해 사용
- 영상은 2차원 혹은 3차원 행렬 내에 RGB value를 가지고 있음

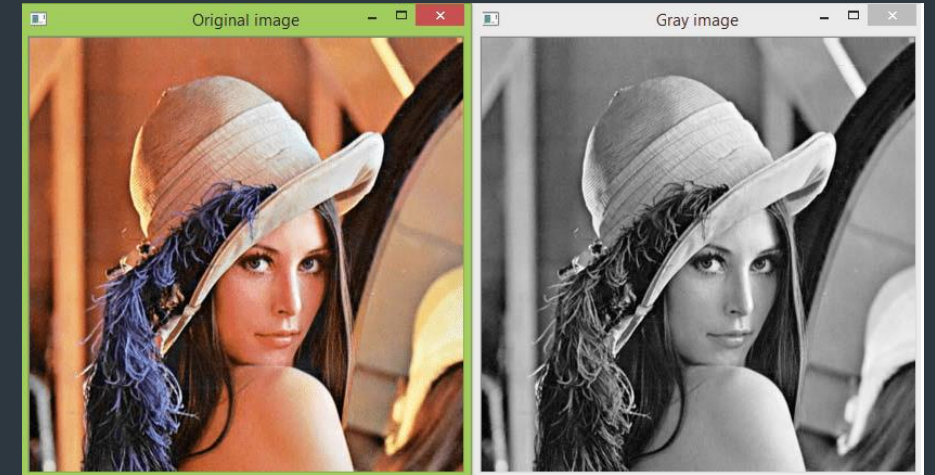


```
72
73 ①↑ @
74
75
76
77
```

```
@Override
public Mat onCameraFrame(CameraBridgeViewBase.CvCameraViewFrame inputFrame) {
    Mat frame = inputFrame.rgba();
    return frame;
}
```

# cvtColor

- Image Frame의 RGB 데이터 형식을 Gray scale이나 HSV로 변환

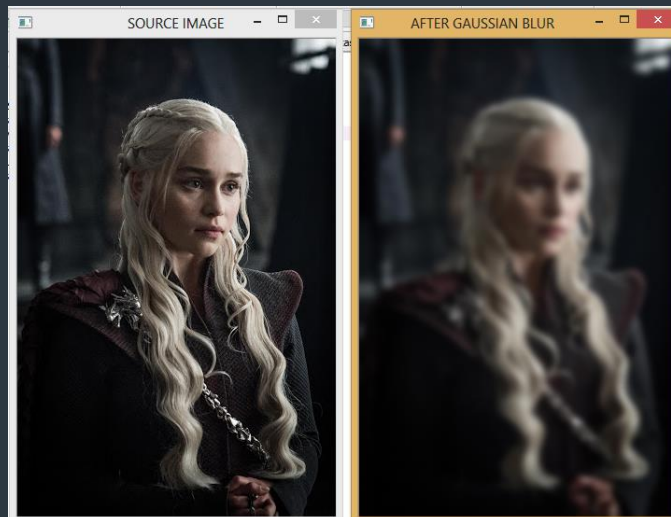


```
72  
73 @ @  
74  
75 @Override  
76 public Mat onCameraFrame(CameraBridgeViewBase.CvCameraViewFrame inputFrame) {  
77     Mat frame = inputFrame.rgba();  
    Imgproc.cvtColor(frame, frame, Imgproc.COLOR_RGB2GRAY);  
    return frame;  
}
```

# Gaussian Blur

- Image Frame을 흐릿하게 하여 noise reduction

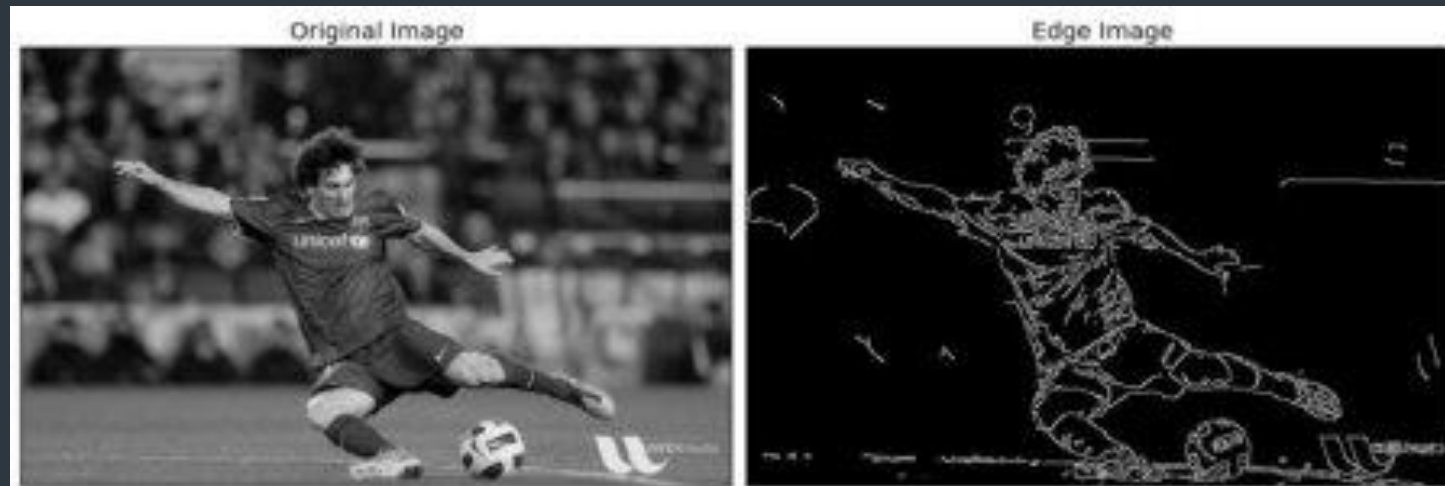
```
72      @Override
73      public Mat onCameraFrame(CameraBridgeViewBase.CvCameraViewFrame inputFrame) {
74          Mat frame = inputFrame.rgba();
75          Imgproc.cvtColor(frame, frame, Imgproc.COLOR_RGB2GRAY);
76          Imgproc.GaussianBlur(frame, frame, new Size( width: 5, height: 5), sigmaX: 0);
77          return frame;
78      }
```



# Canny Edge Detector

- image frame에서 사물의 간에 경계만을 검출하는 데 사용

```
72      @Override
73      public Mat onCameraFrame(CameraBridgeViewBase.CvCameraViewFrame inputFrame) {
74          Mat frame = inputFrame.rgba();
75          Imgproc.cvtColor(frame, frame, Imgproc.COLOR_RGB2GRAY);
76          Imgproc.Canny(frame, frame, threshold1: 100, threshold2: 200);
77          return frame;
78      }
```



# Region of Interest 출력 함수 작성

```
public void ROI( Mat img, double w, double h) {
```

```
    Mat mask = Mat.zeros(img.rows(), img.cols(), img.type());
```

→ input frame과 동일한 사이즈의 행렬을 생성하고 모든 값을 0으로 가지는 mask 생성

```
    Point[] rook_points = new Point[4];
```

```
    rook_points[0] = new Point(0, h * 1.0); //start drawing from 0 to 1 to 2 to3
```

```
    rook_points[1] = new Point(w * 0.45, h * 0.6);
```

```
    rook_points[2] = new Point(w * 0.55, h * 0.6);
```

```
    rook_points[3] = new Point(w * 1.0, h * 1.0);
```

→사다리꼴의 4개의 꼭짓점 위치를 지정

```
    MatOfPoint matPt = new MatOfPoint();
```

```
    matPt.fromArray(rook_points);
```

```
    List<MatOfPoint> ppt = new ArrayList<MatOfPoint>();
```

```
    ppt.add(matPt);
```

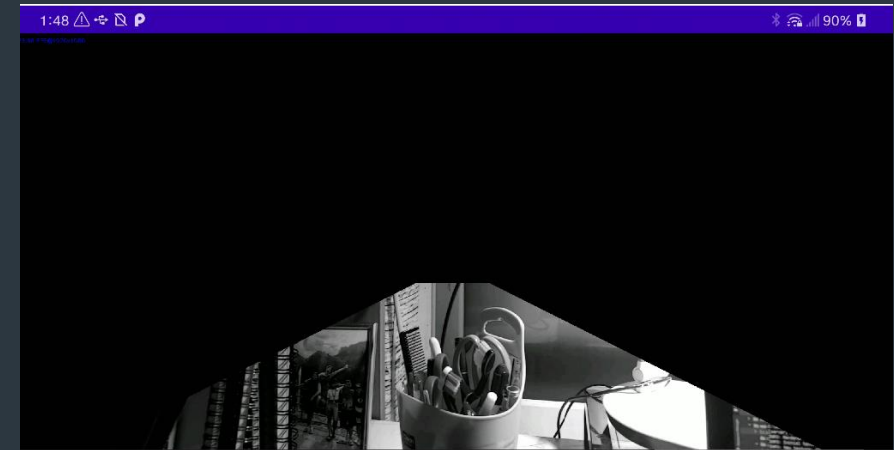
```
    Imgproc.fillPoly(mask, ppt, new Scalar( 255 ));
```

→사다리꼴의 내부의 값을 255로 채움

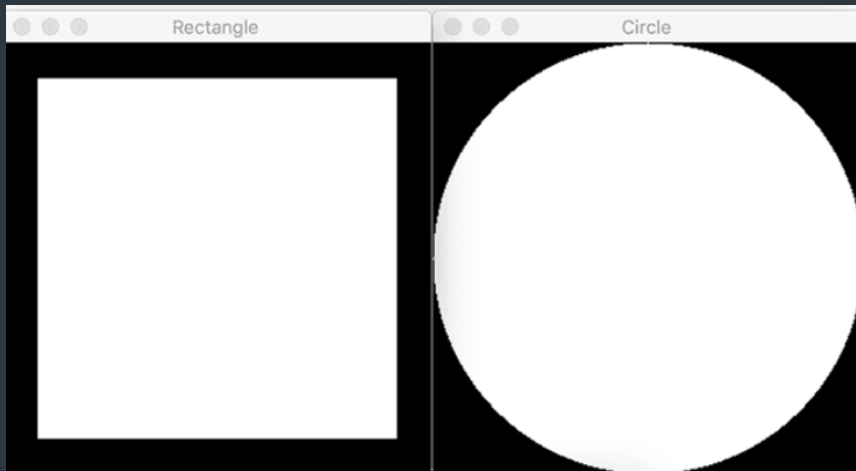
```
    Core.bitwise_and(img, mask, img);
```

```
}
```

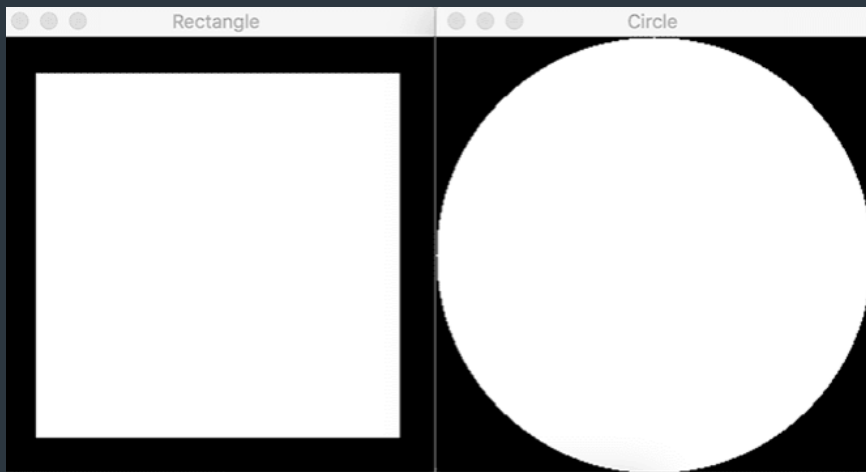
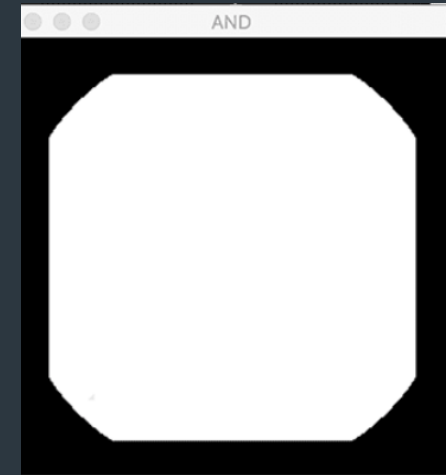
→bitwise 연산으로 mask와 input frame을 and 연산으로 합침



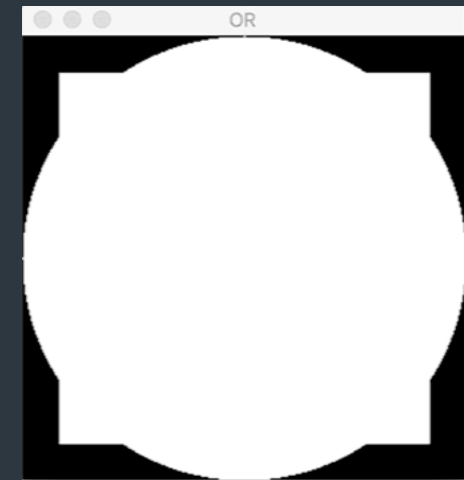
# Region of Interest 출력 함수 작성 – Bitwise Operation



`bitwise_and(img1, img2)`



`bitwise_or(img1, img2)`



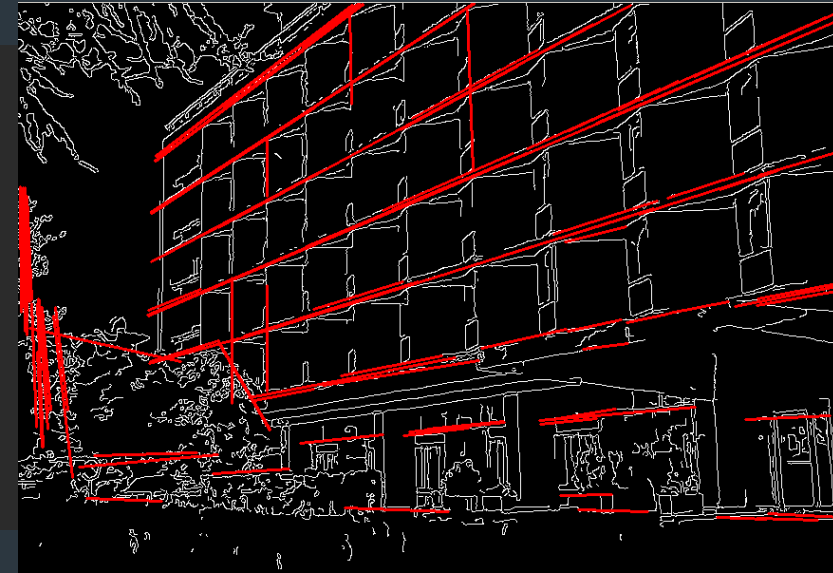


# draw line 함수 작성

```
public void draw_the_line(Mat img, Mat img2) {  
  
    Mat linesP = new Mat();  
    Imgproc.HoughLinesP(img, linesP, 6, Math.PI/180, 160, 40, 25);  
    for (int x = 0; x < linesP.rows(); x++) {  
        double[] l = linesP.get(x, 0);  
        Imgproc.line(img2, new Point(l[0], l[1]), new Point(l[2], l[3]), new Scalar(0,255,0), 10, Imgproc.LINE_AA, 0);  
    }  
}
```

→ 직선을 검출하기 위한 메소드

→ 검출한 직선을 frame 위에 표시하기 위한 for문





# Lane Detection 소스코드 공유

[github 링크](#)

```
158 lines (118 sloc) | 4.72 KB
Raw Blame
1 package com.example.androidopencvprac;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.graphics.Camera;
6 import android.os.Bundle;
7 import android.util.Log;
8 import android.view.SurfaceView;
9
10 import org.opencv.android.BaseLoaderCallback;
11 import org.opencv.android.CameraActivity;
12 import org.opencv.android.CameraBridgeViewBase;
13 import org.opencv.android.LoaderCallbackInterface;
14 import org.opencv.android.OpenCVLoader;
15 import org.opencv.*;
16 import org.opencv.core.Core;
17 import org.opencv.core.Mat;
18 import org.opencv.core.MatOfPoint;
19 import org.opencv.core.Point;
20 import org.opencv.core.Scalar;
21 import org.opencv.core.Size;
22 import org.opencv.imgproc.Imgproc;
23
24 import java.util.ArrayList;
25 import java.util.Collections;
26 import java.util.List;
```