

LAMPIRAN A *Confusion Matrix*

60	11	0	0
9	64	0	0
0	0	56	1
0	0	2	66

Confusion Matrix Dataset 50

65	6	0	0
9	64	0	0
0	0	56	1
1	0	1	66

Confusion Matrix Dataset 100

65	6	0	0
2	71	0	0
0	2	53	2
0	0	2	66

Confusion Matrix *Dataset 200*

67	4	0	0
2	71	0	0
0	0	55	2
0	0	2	66

Confusion Matrix *Dataset 250*

67	4	0	0
2	71	0	0
0	0	57	2
0	0	2	66

Confusion Matrix *Dataset 300*

LAMPIRAN B Listing Program *ClassifierQuantizedMobileNet*

```
package org.Tensorflow.lite.examples.classification.tflite;

import                                Android.app.Activity;
import                                java.io.IOException;
import
org.Tensorflow.lite.examples.classification.tflite.Classifier
r.Device;
import    org.Tensorflow.lite.support.common.TensorOperator;
import    org.Tensorflow.lite.support.common.ops.NormalizeOp;

/** This Tensorflow Lite classifier works with the quantized
MobileNet                                model.                                */
public class ClassifierQuantizedMobileNet extends Classifier
{

    /**
     * The quantized model does not require normalization, thus
set    mean    as    0.0f,    and    std    as    1.0f    to
     *    bypass    the    normalization.
     */
    private    static    final    float    IMAGE_MEAN    =    0.0f;

    private    static    final    float    IMAGE_STD    =    1.0f;

    /** Quantized MobileNet requires additional dequantization
to    the    output    probability.                                */
    private    static    final    float    PROBABILITY_MEAN    =    0.0f;

    private    static    final    float    PROBABILITY_STD    =    255.0f;
```

```

/**
 * Initializes a {@code ClassifierQuantizedMobileNet}.
 *
 *
 * @param activity
 */
public ClassifierQuantizedMobileNet(Activity activity,
Device device, int numThreads)
    throws IOException {
    super(activity, device, numThreads);
}

@Override
protected String getModelPath() {
    // you can download this file from
    // see build.gradle for where to obtain this file. It
should be auto
    // downloaded into assets.
    return "model.tflite";
}

@Override
protected String getLabelPath() {
    return "labels.txt";
}

@Override
protected TensorOperator getPreprocessNormalizeOp() {
    return new NormalizeOp(IMAGE_MEAN, IMAGE_STD);
}

@Override

```

```
protected TensorOperator getPostprocessNormalizeOp() {  
    return new NormalizeOp(PROBABILITY_MEAN,  
PROBABILITY_STD);  
}  
}
```

LAMPIRAN C Listing Program *ClassifierFloatMobileNet*

```
package org.Tensorflow.lite.examples.classification.tflite;

import                                Android.app.Activity;
import                                java.io.IOException;
import

org.Tensorflow.lite.examples.classification.tflite.Classifier
r.Device;
import    org.Tensorflow.lite.support.common.TensorOperator;
import    org.Tensorflow.lite.support.common.ops.NormalizeOp;

/** This TensorflowLite classifier works with the float
MobileNet                                model.                                */
public class ClassifierFloatMobileNet extends Classifier {

    /** Float MobileNet requires additional normalization of
the                                used                                input.                                */
    private static final float IMAGE_MEAN = 127.5f;

    private static final float IMAGE_STD = 127.5f;

    /**
     * Float model does not need dequantization in the post-
processing. Setting mean and std as 0.0f
     * and 1.0f, repectively, to bypass the normalization.
     */
    private static final float PROBABILITY_MEAN = 0.0f;

    private static final float PROBABILITY_STD = 1.0f;
```

```

/**
 * Initializes a {@code ClassifierFloatMobileNet}.
 *
 * @param activity
 */
public ClassifierFloatMobileNet(Activity activity, Device
device, int numThreads)
    throws IOException {
    super(activity, device, numThreads);
}

@Override
protected String getModelPath() {
    // you can download this file from
    // see build.gradle for where to obtain this file. It
should be auto
    // downloaded into assets.
    return "model_unquant.tflite";
}

@Override
protected String getLabelPath() {
    return "labels.txt";
}

@Override
protected TensorOperator getPreprocessNormalizeOp() {
    return new NormalizeOp(IMAGE_MEAN, IMAGE_STD);
}

```



```
@Override
protected TensorOperator getPostprocessNormalizeOp() {
    return new NormalizeOp(PROBABILITY_MEAN,
PROBABILITY_STD);
}
}
```

LAMPIRAN D Listing Program Build.Gridl Android

```
apply          plugin:          'com.Android.application'

Android                                               {
    compileSdkVersion                                28
    defaultConfig                                    {
        applicationId
        "org.Tensorflow.lite.examples.classification"
        minSdkVersion                                21
        targetSdkVersion                              28
        versionCode                                  1
        versionName                                  "1.0"
    }
    buildTypes                                       {
        release                                       {
            minifyEnabled                             false
            proguardFiles getDefaultProguardFile('proguard-
Android.txt'),                                     'proguard-rules.pro'
        }
    }
    aaptOptions                                       {
        noCompress                                    "tflite"
    }
    compileOptions                                    {
        sourceCompatibility                            =          '1.8'
        targetCompatibility                            =          '1.8'
    }
    lintOptions                                       {
        abortOnError false
    }
}
```

```

    }
}

// Download default models; if you wish to use your own models
then
// place them in the "assets" directory and comment out this
line.

apply from: 'download.gradle'

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'Androidx.appcompat:appcompat:1.0.0'
    implementation
'Androidx.coordinatorlayout:coordinatorlayout:1.0.0'
    implementation
'com.google.Android.material:material:1.0.0'

    //Build off of nightly Tensorflow Lite
    implementation('org.Tensorflow:Tensorflow-lite:0.0.0-
nightly') { changing = true }
    implementation('org.Tensorflow:Tensorflow-lite-
gpu:0.0.0-nightly') { changing = true }
    implementation('org.Tensorflow:Tensorflow-lite-
support:0.0.0-nightly') { changing = true }
    // Use local Tensorflow library
    // implementation 'org.Tensorflow:Tensorflow-lite-
local:0.0.0'

    AndroidTestImplementation
'Androidx.test.ext:junit:1.1.1'
    AndroidTestImplementation

```

```
'com.Android.support.test:rules:1.0.2'  
  AndroidTestImplementation 'com.google.truth:truth:1.0.1'  
}
```

LAMPIRAN E *Build Model Tensorflow Lite Menggunakan Web Tools Teachable machine*

