LAMPIRAN

Lampiran - 1 Listing Program

Train_cacao.py

```
import os
import tensorflow as tf
from tensorflow.keras.preprocessing.image
import ImageDataGenerator
from tensorflow.keras.applications import
MobileNetV2
from tensorflow.keras.layers import Dense,
GlobalAveragePooling2D, Dropout
from tensorflow.keras.models import Model
from tensorflow.keras.callbacks import
ModelCheckpoint, EarlyStopping,
ReduceLROnPlateau
# --- PARAMETER ---
IMAGE_SIZE = (224, 224)
BATCH_SIZE = 32
EPOCHS = 10
DATASET_DIR =
r"D:\Skripsiku\cacao\dataset"
MODEL_SAVE_PATH = 'model_cacao.h5'
TFLITE_SAVE_PATH =
'model_cacao.tflite'
LABELS_PATH = 'labels.txt'
# --- DATA PREPARATION ---
train_datagen = ImageDataGenerator(
    rescale=1./255,
    validation_split=0.2,
    rotation_range=20,
    width_shift_range=0.1,
    height_shift_range=0.1,
    horizontal_flip=True,
    zoom_range=0.2,
    shear_range=0.2
)
train_generator =
train_datagen.flow_from_directory(
    DATASET_DIR,
    target_size=IMAGE_SIZE,
    batch_size=BATCH_SIZE,
    class_mode='categorical',
    subset='training',
    shuffle=True
)
val_generator =
train_datagen.flow_from_directory(
    DATASET_DIR,
    target_size=IMAGE_SIZE,
    batch_size=BATCH_SIZE,
    class_mode='categorical',
    subset='validation'
)
num_classes = train_generator.num_classes
# --- SIMPAN LABELS ---
class_names =
list(train_generator.class_indices.keys())
with open(LABELS_PATH, 'w') as f:
    for label in class_names:
        f.write(f"{label}\n")
print(f"✅ Labels disimpan di:
{LABELS_PATH}")
# --- MODEL ---
base_model =
MobileNetV2(weights='imagenet',
include_top=False, input_shape=(224, 224,
3))
base_model.trainable = True  # Fine-tune
model
# Freeze sebagian besar layer, hanya buka
20 terakhir
for layer in base_model.layers[:-20]:
    layer.trainable = False
x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dropout(0.3)(x)  # Tambahkan dropout
x = Dense(128, activation='relu')(x)
x = Dropout(0.3)(x)
predictions = Dense(num_classes,
activation='softmax')(x)
model = Model(inputs=base_model.input,
outputs=predictions)
model.compile(optimizer=tf.keras.optimizer
s.Adam(learning_rate=1e-4),
              loss='categorical_crossentropy',
              metrics=['accuracy'])
# --- CALLBACKS ---
checkpoint =
ModelCheckpoint(MODEL_SAVE_PATH,
monitor='val_accuracy',
save_best_only=True, verbose=1)
early_stop =
EarlyStopping(monitor='val_loss',
patience=5, restore_best_weights=True)
```

```
reduce_lr =
ReduceLROnPlateau(monitor='val_loss',
factor=0.3, patience=3, verbose=1)
# --- TRAINING ---
model.fit(train_generator,
        validation_data=val_generator,
        epochs=EPOCHS,
        callbacks=[checkpoint, early_stop,
reduce_lr])
# --- LOAD BEST MODEL ---
model =
tf.keras.models.load_model(MODEL_SAV
E_PATH)
# --- CONVERT TO TFLITE ---
converter =
tf.lite.TFLiteConverter.from_keras_model(
model)
tflite_model = converter.convert()

with open(TFLITE_SAVE_PATH, 'wb') as
f:
   f.write(tflite_model)
print(f"✅ Model TFLite disimpan di:
{TFLITE_SAVE_PATH}")
```

BerandaActivity.kt

```
import android.content.Intent
import android.os.Bundle
import android.os.Handler
import android.os.Looper
import android.view.Menu
import android.view.MenuItem
import android.widget.Toast
import
androidx.appcompat.app.AppCompatActivit
y
import androidx.appcompat.widget.Toolbar
import
androidx.core.content.ContextCompat
import
com.google.android.material.floatingactionb
utton.ExtendedFloatingActionButton
// Mengabaikan peringatan deprecated pada
fungsi tertentu
@Suppress("DEPRECATION")
class BerandaActivity :
AppCompatActivity() {
   // Deklarasi tombol
ExtendedFloatingActionButton
   private lateinit var mulaiscan:
ExtendedFloatingActionButton
   // Variabel untuk mendeteksi apakah
tombol back ditekan dua kali
   private var klikduakali = false
   override fun onCreate(savedInstanceState:
Bundle?) {
      super.onCreate(savedInstanceState)
      // Menetapkan layout yang digunakan
oleh activity ini
 setContentView(R.layout.activity_beranda)
      // Inisialisasi tombol dari layout
      mulaiscan =
findViewById(R.id.btnscan)
      // Saat tombol diklik, pindah ke
MainActivity
      mulaiscan.setOnClickListener {
         startActivity(Intent(this,
MainActivity::class.java))
      }
      // Mengubah warna status bar menjadi
coklat gelap
      window.statusBarColor =
ContextCompat.getColor(this,
R.color.warna1)
      // Menyiapkan toolbar dari layout dan
menetapkan judul
      val toolbar: Toolbar =
findViewById(R.id.toolbar)
      toolbar.title = "Deteksi Kakao"
      setSupportActionBar(toolbar) //
Menetapkan toolbar sebagai ActionBar
   }
   // Menampilkan menu di toolbar (inflasi
dari file menu XML)
   override fun
onCreateOptionsMenu(menu: Menu) :
Boolean {
menuInflater.inflate(R.menu.menu_atas,
menu) // Menambahkan item menu
      return true
   }
   // Menangani klik pada item menu
   override fun onOptionsItemSelected(item:
MenuItem): Boolean {
      return when (item.itemId) {
         // Jika menu "Tentang Aplikasi"
```

```
diklik
            R.id.action_tentang_aplikasi -> {
                // Buka activity
TentangAplikasiActivity
                startActivity(Intent(this,
TentangAplikasiActivity::class.java))
                true
            }
            // Jika bukan item yang dikenali,
gunakan handler default
            else ->
super.onOptionsItemSelected(item)
        }
    }
    // Override fungsi saat tombol kembali
(back) ditekan
    @Deprecated("This method has been
deprecated in favor of using the\n      {@link
OnBackPressedDispatcher} via {@link
#getOnBackPressedDispatcher()}.\n      The
OnBackPressedDispatcher controls how
back button events are dispatched\n      to
one or more {@link
OnBackPressedCallback} objects.")
    override fun onBackPressed() {
        // Jika sebelumnya tombol sudah
ditekan sekali
        if (klikduakali) {
            // Keluar dari aplikasi
            super.onBackPressed()
            finish()
            return
        }
        // Jika baru pertama kali ditekan
        klikduakali = true
        // Tampilkan pesan peringatan
        Toast.makeText(this, "Tekan sekali lagi
untuk keluar",
Toast.LENGTH_SHORT).show()
        // Setelah 2 detik, reset kembali
klikduakali ke false
Handler(Looper.getMainLooper()).postDela
yed({ klikduakali = false }, 2000)
    }
}
```

MainActivity.kt

```
import android.Manifest
import android.content.Intent
import android.content.pm.PackageManager
import android.graphics.*
import android.hardware.camera2.*
import android.os.*
import android.view.*
import android.widget.ImageView
import android.widget.Toast
import
androidx.appcompat.app.AppCompatActivit
y
import androidx.appcompat.widget.Toolbar
import androidx.core.app.ActivityCompat
import
androidx.core.content.ContextCompat
import
com.kelas_b.a221280027.ml.CacaoModelW
ithMetadata
import
org.tensorflow.lite.support.common.FileUtil
import
org.tensorflow.lite.support.image.ImageProc
essor
import
org.tensorflow.lite.support.image.TensorIma
ge
import
org.tensorflow.lite.support.image.ops.Resize
Op
@Suppress("DEPRECATION") //
Mengabaikan warning deprecated
class MainActivity : AppCompatActivity() {
    // Daftar label hasil klasifikasi dari model
    private lateinit var labels: List<String>
    // Objek untuk menggambar teks dan
kotak di hasil deteksi
    private val paint = Paint().apply {
        textAlign = Paint.Align.LEFT
        isAntiAlias = true
    }
    private lateinit var imageProcessor:
ImageProcessor // Untuk memproses gambar
sebelum dikirim ke model
    private lateinit var imageView:
ImageView // Tampilan hasil gambar deteksi
    private lateinit var cameraDevice:
```

```
CameraDevice // Kamera yang sedang digunakan
    private lateinit var handler: Handler // Handler thread untuk proses kamera
    private lateinit var cameraManager: CameraManager // Pengelola kamera
    private lateinit var textureView: TextureView // Tampilan preview kamera
    private lateinit var model: CacaoModelWithMetadata // Model TFLite untuk deteksi kakao
    // Menyimpan label terakhir dan jumlah deteksi berulang
    private var lastDetectedLabel: String = ""
    private var detectionCount: Int = 0
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main) // Atur layout
        // Load daftar label dari file labels.txt di assets
        labels = FileUtil.loadLabels(this, "labels.txt") // pastikan "Bukan Kakao" tidak ada di sini
        // Atur image processor untuk resize gambar ke 300x300
        imageProcessor = ImageProcessor.Builder()
            .add(ResizeOp(300, 300, ResizeOp.ResizeMethod.BILINEAR))
            .build()
        // Inisialisasi model TFLite
        model = CacaoModelWithMetadata.newInstance(this)
        // Siapkan thread khusus untuk kamera
        val handlerThread = HandlerThread("videoThread").apply { start() }
        handler = Handler(handlerThread.looper)
        // Inisialisasi manajer kamera
        cameraManager = getSystemService(CAMERA_SERVICE) as CameraManager
        // Ambil tampilan dan view dari layout
        imageView = findViewById(R.id.imageView)
        textureView = findViewById(R.id.textureView)
        // Set listener untuk preview kamera
        textureView.surfaceTextureListener = surfaceTextureListener
        // Cek dan minta permission kamera
        getPermission()
        // Atur warna status bar dan toolbar
        window.statusBarColor = ContextCompat.getColor(this, R.color.warna1)
        val toolbar: Toolbar = findViewById(R.id.toolbar)
        toolbar.title = "Deteksi Kakao"
        setSupportActionBar(toolbar)
supportActionBar?.setDisplayHomeAsUpEnabled(true)
supportActionBar?.setDisplayShowHomeEnabled(true)
    }
    // Buat menu atas di toolbar
    override fun onCreateOptionsMenu(menu: Menu): Boolean {
menuInflater.inflate(R.menu.menu_atas, menu)
        return true
    }
    // Tangani klik menu toolbar
    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        return when (item.itemId) {
            R.id.action_tentang_aplikasi -> {
                startActivity(Intent(this, TentangAplikasiActivity::class.java))
                true
            }
            android.R.id.home -> {
                onBackPressed()
                true
            }
            else -> super.onOptionsItemSelected(item)
        }
    }
    // Listener untuk TextureView (preview kamera)
```

```
    private val surfaceTextureListener =
object :
TextureView.SurfaceTextureListener {
        override fun
onSurfaceTextureAvailable(surface:
SurfaceTexture, width: Int, height: Int) {
            if
(ContextCompat.checkSelfPermission(this@
MainActivity,
Manifest.permission.CAMERA)
                ==
PackageManager.PERMISSION_GRANTED
            ) {
                openCamera() // Buka kamera saat
textute tersedia
            }
        }
        override fun
onSurfaceTextureSizeChanged(surface:
SurfaceTexture, width: Int, height: Int) {}
        override fun
onSurfaceTextureDestroyed(surface:
SurfaceTexture): Boolean = false
        // Dipanggil setiap frame baru dari
kamera
        override fun
onSurfaceTextureUpdated(surface:
SurfaceTexture) {
            val bmp = textureView.bitmap ?:
return
            // Hitung rata-rata kecerahan
            val avgBrightness =
bmp.getAverageBrightness()
            if (avgBrightness > 230) return //
Hindari deteksi jika terlalu terang
            val image =
imageProcessor.process(TensorImage.fromB
itmap(bmp))
            val outputs = model.process(image)
// Jalankan model
            val probability =
outputs.probabilityAsCategoryList
            val detected =
probability.maxByOrNull { it.score }
            val allowedLabels = listOf("Basah",
"Kering", "Setengah Kering")
            val threshold = 0.90f // Hanya label
dengan confidence > 90%
            val isValid = detected != null &&
                detected.score > threshold &&
                detected.label in allowedLabels
            if (isValid) {
                // Jika label yang sama terdeteksi
beberapa kali
                if (detected!!.label ==
lastDetectedLabel) {
                    detectionCount++
                } else {
                    detectionCount = 0
                }
                lastDetectedLabel = detected.label
                if (detectionCount >= 3) {
                    val mutable =
bmp.copy(Bitmap.Config.ARGB_8888, true)
                    val canvas = Canvas(mutable)
                    val h = mutable.height
                    val w = mutable.width
                    // Gambar kotak hijau dan label
deteksi
                    paint.textSize = h / 20f
                    paint.strokeWidth = h / 85f
                    paint.color = Color.GREEN
                    paint.style =
Paint.Style.STROKE
                    canvas.drawRect(RectF(0f, 0f,
w.toFloat(), h.toFloat()), paint)
                    paint.style = Paint.Style.FILL
                    canvas.drawText(detected.label,
(w - paint.measureText(detected.label)) / 2f,
100f, paint)
                    val scoreText = "Confidence:
%.2f%%".format(detected.score * 100)
                    paint.textSize = h / 35f
                    canvas.drawText(scoreText, (w
- paint.measureText(scoreText)) / 2f, 160f,
paint)
imageView.setImageBitmap(mutable)
                    return
                }
            } else {
                // Jika tidak valid atau bukan
kakao
                detectionCount = 0
                lastDetectedLabel = ""
                // tampilkan info "Bukan Kakao"
                val mutable =
```

```
bmp.copy(Bitmap.Config.ARGB_8888, true)
            val canvas = Canvas(mutable)
            val h = mutable.height
            val w = mutable.width
            paint.textSize = h / 20f
            paint.color = Color.RED
            canvas.drawText("Bukan Kakao",
(w - paint.measureText("Bukan Kakao")) /
2f, 100f, paint)
          imageView.setImageBitmap(mutable)
            return
          }
          // Default: tampilkan frame tanpa
anotasi
          imageView.setImageBitmap(bmp)
        }
    }
    // Ekstensi fungsi Bitmap untuk
menghitung rata-rata kecerahan
    fun Bitmap.getAverageBrightness(): Int {
        var total = 0L
        val pixels = IntArray(width * height)
        getPixels(pixels, 0, width, 0, 0, width,
height)
        for (pixel in pixels) {
            val r = Color.red(pixel)
            val g = Color.green(pixel)
            val b = Color.blue(pixel)
            total += (r + g + b) / 3
        }
        return (total / pixels.size).toInt()
    }
    // Tutup model saat activity dihancurkan
    override fun onDestroy() {
        super.onDestroy()
        model.close()
    }
    // Buka kamera dan mulai preview
    private fun openCamera() {
        val cameraId =
cameraManager.cameraIdList[0] // Ambil
kamera pertama (biasanya belakang)
        if
(ActivityCompat.checkSelfPermission(this,
Manifest.permission.CAMERA) !=
PackageManager.PERMISSION_GRANTED
) return
        cameraManager.openCamera(cameraId,
object : CameraDevice.StateCallback() {
            override fun onOpened(device:
CameraDevice) {
                cameraDevice = device
                val surface =
Surface(textureView.surfaceTexture)
                val requestBuilder =
cameraDevice.createCaptureRequest(Camer
aDevice.TEMPLATE_PREVIEW)
                requestBuilder.addTarget(surface)
                // Buat session kamera dan mulai
stream
cameraDevice.createCaptureSession(listOf(s
urface), object :
CameraCaptureSession.StateCallback() {
                    override fun
onConfigured(session:
CameraCaptureSession) {
session.setRepeatingRequest(requestBuilder.
build(), null, handler)
                    }
                    override fun
onConfigureFailed(session:
CameraCaptureSession) {}
                }, handler)
            }
            override fun onDisconnected(device:
CameraDevice) {}
            override fun onError(device:
CameraDevice, error: Int) {}
        }, handler)
    }
    // Minta permission kamera
    private fun getPermission() {
        if
(ContextCompat.checkSelfPermission(this,
Manifest.permission.CAMERA)
            !=
PackageManager.PERMISSION_GRANTED
        ) {
requestPermissions(arrayOf(Manifest.permi
ssion.CAMERA), 101)
        }
    }
    // Tangani hasil permission
    override fun
onRequestPermissionsResult(requestCode:
Int, permissions: Array<out String>,
```

```
grantResults: IntArray) {
super.onRequestPermissionsResult(requestC
ode, permissions, grantResults)
      if (requestCode == 101 &&
grantResults.isNotEmpty() &&
grantResults[0] ==
PackageManager.PERMISSION_GRANTED
) {
         if (textureView.isAvailable)
openCamera()
         else
textureView.surfaceTextureListener =
surfaceTextureListener
      } else {
         Toast.makeText(this, "Kamera
diperlukan untuk aplikasi ini",
Toast.LENGTH_LONG).show()
         finish()
      }
   }
}
```

TentangAplikasiActivity.kt

```
import android.os.Bundle
import android.view.MenuItem
import
androidx.appcompat.app.AppCompatActivit
y
import androidx.appcompat.widget.Toolbar
import
androidx.core.content.ContextCompat
// TentangAplikasiActivity adalah Activity
yang menampilkan informasi tentang
aplikasi
class TentangAplikasiActivity :
AppCompatActivity() {
   // Fungsi yang dipanggil saat activity
pertama kali dibuat
   override fun onCreate(savedInstanceState:
Bundle?) {
      super.onCreate(savedInstanceState)
      // Menetapkan tampilan/layout yang
digunakan activity ini
setContentView(R.layout.activity_tentang_a
plikasi)
      // Mengubah warna status bar menjadi
warna coklat gelap
      window.statusBarColor =
ContextCompat.getColor(this,
R.color.warna1)
      // Menghubungkan Toolbar dari layout
      val toolbar: Toolbar =
findViewById(R.id.toolbar)
      // Menetapkan judul toolbar
      toolbar.title = "Tentang Aplikasi"
      // Menetapkan toolbar sebagai
ActionBar
      setSupportActionBar(toolbar)
      // Menampilkan tombol "back" (panah
kiri) di ActionBar
supportActionBar?.setDisplayHomeAsUpEn
abled(true) // tampilkan tombol kembali
supportActionBar?.setDisplayShowHomeEn
abled(true) // aktifkan fungsi tombol kembali
   }
   // Menangani aksi ketika item menu
diklik, termasuk tombol back
   override fun onOptionsItemSelected(item:
MenuItem): Boolean {
      return when (item.itemId) {
         // Jika tombol back (home) di toolbar
ditekan
         android.R.id.home -> {
            onBackPressed() // kembali ke
halaman sebelumnya
            true
         }
         // Jika bukan item yang dikenali,
gunakan handler default
         else ->
super.onOptionsItemSelected(item)
      }
   }
}
```

activity_beranda.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.Constrain
tLayout
xmlns:android="http://schemas.android.com
/apk/res/android"
xmlns:app="http://schemas.android.com/apk
/res-auto"
xmlns:tools="http://schemas.android.com/to
ols"
   android:id="@+id/main"
```

```
   android:layout_width="match_parent"
   android:layout_height="match_parent"
 android:background="@drawable/bgkakao"
   tools:context=".BerandaActivity">
   <View
      android:id="@+id/darkOverlay"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:background="#7C000000"
      android:layout_gravity="center"
      android:elevation="1dp" />
   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:orientation="vertical"
      android:elevation="2dp">
      <androidx.appcompat.widget.Toolbar
         android:id="@+id/toolbar"
        android:layout_width="match_parent"
       android:layout_height="?actionBarSize"
        android:background="@color/warna1"
android:theme="@style/Theme._221280027
" />
      <TextView
        android:layout_width="match_parent"
android:layout_height="wrap_content"
         android:gravity="center"
         android:text="SELAMAT
DATANG"
         android:textColor="@color/white"
         android:textStyle="bold"
         android:fontFamily="sans-serif-
light"
         android:textSize="30sp"
         android:padding="50dp"/>
      <TextView
android:layout_width="match_parent"
       android:layout_height="wrap_content"
         android:gravity="center"
         android:textSize="20sp"
         android:paddingTop="20dp"
         android:paddingBottom="20dp"
         android:textColor="@color/white"
         android:textStyle="bold"
         android:fontFamily="sans-serif-
light"
         android:text="Aplikasi Pendeteksi
Tingkat Kekeringan\nPada Biji Kakao
Berbasis Android"/>
      <LinearLayout
        android:layout_width="match_parent"
       android:layout_height="match_parent"
         android:gravity="center"
         android:orientation="vertical">
<com.google.android.material.floatingaction
button.ExtendedFloatingActionButton
            android:id="@+id/btnscan"
        android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="17dp"
            android:layout_gravity="center"
app:backgroundTint="@color/warna3"
            app:iconTint="@color/white"
            app:icon="@drawable/scan"
            android:textColor="@color/white"
            android:text="Mulai Deteksi
Kakao"/>
      </LinearLayout>
   </LinearLayout>
</androidx.constraintlayout.widget.Constrai
ntLayout>
```

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.Constrain
tLayout
xmlns:android="http://schemas.android.com
/apk/res/android"
xmlns:app="http://schemas.android.com/apk
/res-auto"
xmlns:tools="http://schemas.android.com/to
ols"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">
   <RelativeLayout
      android:layout_width="match_parent"
     android:layout_height="match_parent">
      <androidx.appcompat.widget.Toolbar
         android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="?actionBarSize"
android:background="@color/warna1"
android:theme="@style/Theme._221280027
" />
```

```
        <TextureView
            android:id="@+id/textureView"
android:layout_below="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="match_parent" />
        <ImageView
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_below="@+id/toolbar"
            android:id="@+id/imageView"/>
    </RelativeLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_tentang_aplikasi.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/main"
android:background="@drawable/bgkakao"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
tools:context=".TentangAplikasiActivity">
    <View
        android:id="@+id/darkOverlay"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#7C000000"
        android:layout_gravity="center"
        android:elevation="1dp" />
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:elevation="2dp">
        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="?actionBarSize"
android:background="@color/warna1"
android:theme="@style/Theme._221280027" />
<com.google.android.material.textview.MaterialTextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
        android:layout_marginBottom="6dp"
        android:layout_marginTop="17dp"
        android:textColor="@color/white"
        android:gravity="center"
        android:textStyle="bold"
        android:text="Aplikasi Pendeteksi Tingkat Kekeringan\nPada Biji Kakao Berbasis Android"
        android:textSize="20sp"
        android:layout_gravity="center"/>
<com.google.android.material.textview.MaterialTextView
        android:id="@+id/versi"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginBottom="25dp"
        android:textColor="@color/white"
        android:text="Versi 1.0"
        android:textSize="15sp"
        android:layout_gravity="center"/>
      <View
        android:id="@+id/coloview"
        android:background="#C1C1C1"
android:layout_width="match_parent"
        android:layout_height="0.7dp"/>
<com.google.android.material.textview.MaterialTextView
android:id="@+id/deskripsi_aplikasi"
android:layout_width="match_parent"
android:layout_height="wrap_content"
        android:layout_marginStart="24dp"
        android:layout_marginEnd="24dp"
        android:layout_marginTop="60dp"
android:layout_marginBottom="16dp"
        android:maxWidth="600dp"
        android:text="Aplikasi ini digunakan untuk mendeteksi tingkat kekeringan pada biji kakao berdasarkan analisis warna menggunakan TensorFlow dan algoritma Convolutional Neural Network (CNN)."
        android:textColor="@color/white"
        android:textSize="13sp"
        android:gravity="center"
        android:layout_gravity="center"/>
```

```
      <LinearLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
         android:gravity="bottom"
         android:paddingBottom="22dp"
         android:orientation="vertical">
         <ImageView
            android:id="@+id/logo_aplikasi"
            android:layout_width="120dp"
            android:layout_height="120dp"
            android:layout_gravity="center"
android:layout_marginBottom="16dp"
            android:src="@drawable/icon"
            android:contentDescription="Logo
Aplikasi Kakao" />
<com.google.android.material.textview.Mat
erialTextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
            android:textColor="@color/white"
            android:text="Create by :"
android:layout_marginBottom="15dp"
android:layout_marginTop="25dp"
            android:textSize="20sp"
            android:layout_gravity="center"/>
<com.google.android.material.textview.Mat
erialTextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
            android:textColor="@color/white"
            android:text="Febby Surya Ayu
Nengsih"
            android:textSize="20sp"
            android:textStyle="bold"
            android:layout_gravity="center"/>
<com.google.android.material.textview.Mat
erialTextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
            android:text="221280027"
            android:textColor="@color/white"
            android:textSize="20sp"
            android:layout_gravity="center"/>
      </LinearLayout>
   </LinearLayout>
</androidx.constraintlayout.widget.Constrai
ntLayout>
```

menu_atas.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu
xmlns:app="http://schemas.android.com/apk
/res-auto"
xmlns:android="http://schemas.android.com
/apk/res/android">
   <item
android:id="@+id/action_tentang_aplikasi"
android:title="@string/tentang_aplikasi"
android:icon="@drawable/ic_baseline_info"
      app:showAsAction="always"/>
</menu>
```

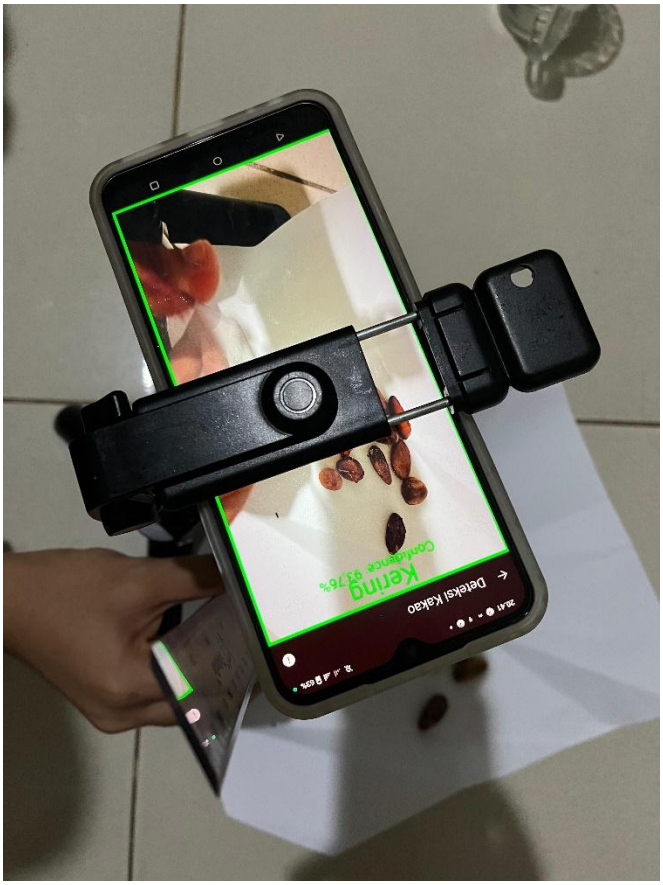
colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
   <color
name="black">#FF000000</color>
   <color
name="white">#FFFFFFFF</color>
   <color name="warna1">#240C00</color>
   <color
name="warna2">#FF8C58</color>
   <color
name="warna3">#A96200</color>
</resources>
```

strings.xml

```
<resources>
   <string
name="app_name">221280027</string>
   <string name="tentang_aplikasi">Tentang
Aplikasi</string>
</resources>
```

Lampiran - 2 Dokumentasi Teknik Pengujian Aplikasi



KARTU MONITORING BIMBINGAN

MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

PROPOSAL

Mahasiswa : Febby Surya Ayu Nengsih	**Pembimbing I : Ir. Untung Suwardoyo, S.Kom.,MT.,IPP**
NIM : 221280027	**Pembimbing II : Hj. A. Irmayani P, ST.,MT**
Judul Skripsi : Rancang Bangun Aplikasi Pendeteksi Tingkat Kekeringan pada Biji Kakao Berbasis Android	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 Perbaiki tabel kelahiran		Konsultasi 1 Kata ary (ilmiah).	19/11/24 A.Y
Konsultasi 2 Rumus Matlab, Tgrau, bab 2. Flowchart.		Konsultasi 2 D.F. (penulisannya perbaiki).	A.Y.
Konsultasi 3 Perbaiki Flowchart, Tambahkan tabel Use Case, Tambahkan Dataset biji kakao di Bab 2		Konsultasi 3 Acc u/ propose	25/12/24 A.Y
Konsultasi 4		Konsultasi 4	
Konsultasi 5		Konsultasi 5	

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton A4 berwarna Hijau muda dan dicetak timbal balik

KARTU MONITORING BIMBINGAN

MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

SKRIPSI

Mahasiswa : FEBBY SURYA AYU NENGSIH	**Pembimbing I : Ir. Untung Suwardoyo, S.Kom., MT.,IPP**
NIM 221280027	**Pembimbing II : Hj. A. Irmayani P, ST.,MT**
Judul Skripsi : RANCANG BANGUN APLIKASI PENDETEKSI TINGKAT KEKERINGAN PADA BIJI KAKAO BERBASIS ANDROID	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 Pabaca abstrak Pendeteksi biji kakao.		Konsultasi 1 - Ha 6 Ika - ABSTRAK	28/2/2025 A-1
Konsultasi 2 Pgpn bab 1		Konsultasi 2 Progaki kerangka	14/6/25 A-1
Konsultasi 3 Pgpn penulisan bab 1 dan Class diagram		Konsultasi 3 progaki abstrak	A1
Konsultasi 4		Konsultasi 4 Kata asing (Italic)	1/
Konsultasi 5		Konsultasi 5 ACC Ujian Hasil	1/

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton A4 berwarna Hijau muda dan dicetak timbal balik