|  |  |
| --- | --- |
| **Ex No: 9** | USING ANDROID ML KIT TO EXTRACT TEXT FROM IMAGE |
| **Date:** |  |

**AIM:**

To develop an application with android ML Kit to extract text from image using android studio.

**DESCRIPTION:**

The Text Extraction from Image application in Android utilizes the ML Vision framework to extract text from images efficiently. It employs the `FirebaseVisionImage` class to process images, `FirebaseVisionTextRecognizer` to initiate text recognition, and asynchronous callbacks to handle the results. The extracted text and its metadata are encapsulated within the `FirebaseVisionText` class, providing a user-friendly solution for accurate and seamless text extraction from images.

Key Components:

1. ML Vision Text Recognition: ML Vision's Text Recognition API is the core component used for detecting and extracting text from images. It employs machine learning models to recognize text and its structure within the image.
2. FirebaseVisionImage: To process images, the application utilizes the FirebaseVisionImage class to convert the image data into a format compatible with ML Vision's Text Recognition.
3. FirebaseVisionTextRecognizer: The FirebaseVisionTextRecognizer class is responsible for initializing and executing the text recognition process. It's used to analyze the image and extract the textual content.
4. Callbacks: Asynchronous processing is crucial, and the application utilizes callbacks to handle the results from the text recognition process. Callbacks like addOnSuccessListener and addOnFailureListener are used to manage successful text extraction or errors.
5. FirebaseVisionText: This class encapsulates the results of text recognition, including information about the detected text, its position in the image, and any associated metadata.

**PROGRAM:**

MainActivity.java:

package com.example.madex9;

import androidx.annotation.NonNull;

import androidx.annotation.Nullable;

import androidx.appcompat.app.AppCompatActivity;

import android.Manifest;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.graphics.Bitmap;

import android.os.Bundle;

import android.provider.MediaStore;

import android.view.View;

import android.widget.ImageView;

import android.widget.TextView;

import android.widget.Toast;

import com.google.android.gms.tasks.OnFailureListener;

import com.google.android.gms.tasks.OnSuccessListener;

import com.google.android.gms.tasks.Task;

import com.google.firebase.ml.vision.FirebaseVision;

import com.google.firebase.ml.vision.common.FirebaseVisionImage;

import com.google.firebase.ml.vision.text.FirebaseVisionText;

import com.google.firebase.ml.vision.text.FirebaseVisionTextRecognizer;

public class MainActivity extends AppCompatActivity {

ImageView imageView;

TextView textView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

imageView = findViewById(R.id.imageId);

textView = findViewById(R.id.textId);

if(checkSelfPermission(android.Manifest.permission.CAMERA)!= PackageManager.PERMISSION\_GRANTED){

requestPermissions(new String[]{Manifest.permission.CAMERA},101);

}

}

public void doProcess(View view){

Intent intent = new Intent(MediaStore.ACTION\_IMAGE\_CAPTURE);

startActivityForResult(intent, 101);

}

@Override

protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data){

super.onActivityResult(requestCode, resultCode, data);

Bundle bundle = data.getExtras();

Bitmap bitmap = (Bitmap) bundle.get("data");

imageView.setImageBitmap(bitmap);

FirebaseVisionImage firebaseVisionImage = FirebaseVisionImage.fromBitmap(bitmap);

FirebaseVision firebaseVision = FirebaseVision.getInstance();

FirebaseVisionTextRecognizer firebaseVisionTextRecognizer = firebaseVision.getOnDeviceTextRecognizer();

Task<FirebaseVisionText> task = firebaseVisionTextRecognizer.processImage(firebaseVisionImage);

task.addOnSuccessListener(new OnSuccessListener<FirebaseVisionText>() {

@Override

public void onSuccess(FirebaseVisionText firebaseVisionText) {

String s = firebaseVisionText.getText();

textView.setText(s);

}

});

task.addOnFailureListener(new OnFailureListener() {

@Override

public void onFailure(@NonNull Exception e) {

Toast.makeText(getApplicationContext(), e.getMessage(), Toast.LENGTH\_LONG).show();

}

     });

    }

}

activity\_main.xml:

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<ImageView

android:id="@+id/imageId"

android:layout\_width="411dp"

android:layout\_height="297dp"

android:contentDescription="@string/image"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.0" />

<TextView

android:id="@+id/textId"

android:layout\_width="393dp"

android:layout\_height="57dp"

android:layout\_marginStart="11dp"

android:layout\_marginTop="25dp"

android:layout\_marginEnd="11dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.493" />

<Button

android:id="@+id/buttonId"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:onClick="doProcess"

android:text="@string/click\_me"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="1.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.434" />

</androidx.constraintlayout.widget.ConstraintLayout>

build.gradle (:app):

dependencies {

implementation 'androidx.appcompat:appcompat:1.6.1'

implementation 'com.google.android.material:material:1.10.0'

implementation 'androidx.constraintlayout:constraintlayout:2.1.4'

implementation 'com.google.firebase:firebase-ml-vision:24.1.0'

implementation 'com.google.android.gms:play-services-vision:20.1.3'

testImplementation 'junit:junit:4.13.2'

androidTestImplementation 'androidx.test.ext:junit:1.1.5'

androidTestImplementation 'androidx.test.espresso:espresso-core:3.5.1'

}

AndroidManifest.xml:

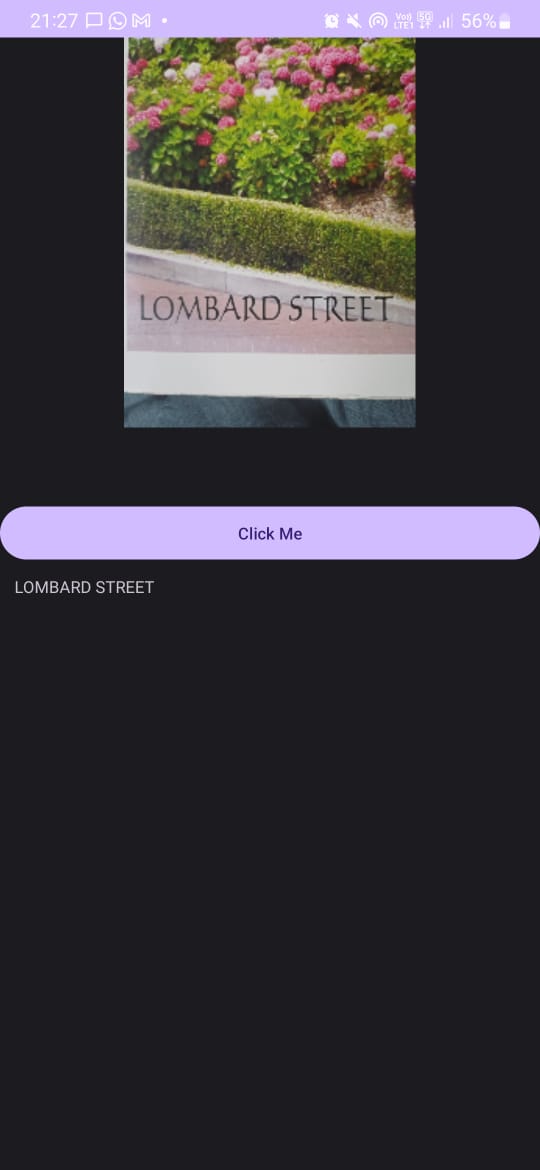
<uses-feature

android:name="android.hardware.camera"

android:required="false"/>

<uses-permission android:name="android.permission.CAMERA"/>

**OUTPUT:**



**RESULT:**

Thus, the application that extracts text from images has been created with the help of Android studio and the output has been verified.