package com.datumdroid.android.ocr.simple;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.InputStream;

import java.io.OutputStream;

import java.nio.charset.Charset;

import java.util.zip.GZIPInputStream;

import android.app.Activity;

import android.content.Intent;

import android.content.res.AssetManager;

import android.graphics.Bitmap;

import android.graphics.BitmapFactory;

import android.graphics.Matrix;

import android.media.ExifInterface;

import android.net.Uri;

import android.os.Bundle;

import android.os.Environment;

import android.provider.MediaStore;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import com.googlecode.tesseract.android.TessBaseAPI;

public class SimpleAndroidOCRActivity extends Activity {

public static final String PACKAGE\_NAME = "com.datumdroid.android.ocr.simple";

public static final String DATA\_PATH = Environment

.getExternalStorageDirectory().toString() + "/SimpleAndroidOCR/";

// You should have the trained data file in assets folder

// You can get them at:

// http://code.google.com/p/tesseract-ocr/downloads/list

public static final String lang = "knn";

private static final String TAG = "SimpleAndroidOCR.java";

protected Button \_button;

// protected ImageView \_image;

protected EditText \_field;

protected String \_path;

protected boolean \_taken;

protected static final String PHOTO\_TAKEN = "photo\_taken";

@Override

public void onCreate(Bundle savedInstanceState) {

String[] paths = new String[] { DATA\_PATH, DATA\_PATH + "tessdata/" };

for (String path : paths) {

File dir = new File(path);

if (!dir.exists()) {

if (!dir.mkdirs()) {

Log.v(TAG, "ERROR: Creation of directory " + path + " on sdcard failed");

return;

} else {

Log.v(TAG, "Created directory " + path + " on sdcard");

}

}

}

// lang.traineddata file with the app (in assets folder)

// You can get them at:

// http://code.google.com/p/tesseract-ocr/downloads/list

// This area needs work and optimization

if (!(new File(DATA\_PATH + "tessdata/" + lang + ".traineddata")).exists()) {

try {

AssetManager assetManager = getAssets();

InputStream in = assetManager.open("tessdata/" + lang + ".traineddata");

//GZIPInputStream gin = new GZIPInputStream(in);

OutputStream out = new FileOutputStream(DATA\_PATH

+ "tessdata/" + lang + ".traineddata");

// Transfer bytes from in to out

byte[] buf = new byte[1024];

int len;

//while ((lenf = gin.read(buff)) > 0) {

while ((len = in.read(buf)) > 0) {

out.write(buf, 0, len);

}

in.close();

//gin.close();

out.close();

Log.v(TAG, "Copied " + lang + " traineddata");

} catch (IOException e) {

Log.e(TAG, "Was unable to copy " + lang + " traineddata " + e.toString());

}

}

super.onCreate(savedInstanceState);

setContentView(R.layout.main);

// \_image = (ImageView) findViewById(R.id.image);

\_field = (EditText) findViewById(R.id.field);

\_button = (Button) findViewById(R.id.button);

\_button.setOnClickListener(new ButtonClickHandler());

\_path = DATA\_PATH + "/ocr.jpg";

}

public class ButtonClickHandler implements View.OnClickListener {

public void onClick(View view) {

Log.v(TAG, "Starting Camera app");

startCameraActivity();

}

}

// Simple android photo capture:

// http://labs.makemachine.net/2010/03/simple-android-photo-capture/

protected void startCameraActivity() {

File file = new File(\_path);

Uri outputFileUri = Uri.fromFile(file);

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

intent.putExtra(MediaStore.EXTRA\_OUTPUT, outputFileUri);

startActivityForResult(intent, 0);

}

@Override

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

Log.i(TAG, "resultCode: " + resultCode);

if (resultCode == -1) {

onPhotoTaken();

} else {

Log.v(TAG, "User cancelled");

}

}

@Override

protected void onSaveInstanceState(Bundle outState) {

outState.putBoolean(SimpleAndroidOCRActivity.PHOTO\_TAKEN, \_taken);

}

@Override

protected void onRestoreInstanceState(Bundle savedInstanceState) {

Log.i(TAG, "onRestoreInstanceState()");

if (savedInstanceState.getBoolean(SimpleAndroidOCRActivity.PHOTO\_TAKEN)) {

onPhotoTaken();

}

}

protected void onPhotoTaken() {

\_taken = true;

BitmapFactory.Options options = new BitmapFactory.Options();

options.inSampleSize = 4;

Bitmap bitmap = BitmapFactory.decodeFile(\_path, options);

try {

ExifInterface exif = new ExifInterface(\_path);

int exifOrientation = exif.getAttributeInt(

ExifInterface.TAG\_ORIENTATION,

ExifInterface.ORIENTATION\_NORMAL);

Log.v(TAG, "Orient: " + exifOrientation);

int rotate = 0;

switch (exifOrientation) {

case ExifInterface.ORIENTATION\_ROTATE\_90:

rotate = 90;

break;

case ExifInterface.ORIENTATION\_ROTATE\_180:

rotate = 180;

break;

case ExifInterface.ORIENTATION\_ROTATE\_270:

rotate = 270;

break;

}

Log.v(TAG, "Rotation: " + rotate);

if (rotate != 0) {

// Getting width & height of the given image.

int w = bitmap.getWidth();

int h = bitmap.getHeight();

// Setting pre rotate

Matrix mtx = new Matrix();

mtx.preRotate(rotate);

// Rotating Bitmap

bitmap = Bitmap.createBitmap(bitmap, 0, 0, w, h, mtx, false);

}

// Convert to ARGB\_8888, required by tess

bitmap = bitmap.copy(Bitmap.Config.ARGB\_8888, true);

} catch (IOException e) {

Log.e(TAG, "Couldn't correct orientation: " + e.toString());

}

// \_image.setImageBitmap( bitmap );

Log.v(TAG, "Before baseApi");

TessBaseAPI baseApi = new TessBaseAPI();

baseApi.setDebug(true);

baseApi.init(DATA\_PATH, lang);

baseApi.setImage(bitmap);

String recognizedText = baseApi.getUTF8Text();

baseApi.end();

// You now have the text in recognizedText var, you can do anything with it.

// We will display a stripped out trimmed alpha-numeric version of it (if lang is eng)

// so that garbage doesn't make it to the display.

Log.v(TAG, "OCRED TEXT: " + recognizedText);

if ( lang.equalsIgnoreCase("knn") ) {

//recognizedText = recognizedText.replaceAll("[^a-zA-Z0-9]+", " ");

}

recognizedText = recognizedText.trim();

//byte[] mar= recognizedText.getBytes(Charset.forName("UTF-8"));

//char m = '\u092E';

//char d = '\u0926';

//char n = '\u0928';

/\*

String myText=new String("\u092E" + "\u0926" + "\u0928");

if(recognizedText.equals(myText))

{

recognizedText="mdn";

}

else

{

recognizedText="no";

}

\*/

String engText=new String();

for(int i=0;i<recognizedText.length();i++)

{

switch(recognizedText.charAt(i))

{

case '\u0902':

engText=engText+"n";

break;

case '\u0904':

engText=engText+"a";

break;

case '\u0905':

engText=engText+"a";

break;

case '\u0906':

engText=engText+"aa";

break;

case '\u0907':

engText=engText+"i";

break;

case '\u0908':

engText=engText+"ii";

break;

case '\u0909':

engText=engText+"u";

break;

case '\u090A':

engText=engText+"uu";

break;

case '\u090B':

engText=engText+"r";

break;

case '\u090C':

engText=engText+"l";

break;

case '\u090D':

engText=engText+"e";

break;

case '\u090E':

engText=engText+"e";

break;

case '\u090F':

engText=engText+"e";

break;

case '\u0910':

engText=engText+"ai";

break;

case '\u0911':

engText=engText+"o";

break;

case '\u0912':

engText=engText+"o";

break;

case '\u0913':

engText=engText+"o";

break;

case '\u0914':

engText=engText+"au";

break;

case '\u0915':

engText=engText+"ka";

break;

case '\u0916':

engText=engText+"kha";

break;

case '\u0917':

engText=engText+"ga";

break;

case '\u0918':

engText=engText+"gha";

break;

case '\u0919':

engText=engText+"nga";

break;

case '\u091A':

engText=engText+"cha";

break;

case '\u091B':

engText=engText+"chha";

break;

case '\u091C':

engText=engText+"ja";

break;

case '\u091D':

engText=engText+"jha";

break;

case '\u091E':

engText=engText+"nya";

break;

case '\u091F':

engText=engText+"tta";

break;

case '\u0920':

engText=engText+"ttha";

break;

case '\u0921':

engText=engText+"da";

break;

case '\u0922':

engText=engText+"dha";

break;

case '\u0923':

engText=engText+"na";

break;

case '\u0924':

engText=engText+"ta";

break;

case '\u0925':

engText=engText+"tha";

break;

case '\u0926':

engText=engText+"da";

break;

case '\u0927':

engText=engText+"dha";

break;

case '\u0928':

engText=engText+"na";

break;

case '\u0929':

engText=engText+"nna";

break;

case '\u092A':

engText=engText+"pa";

break;

case '\u092B':

engText=engText+"fa";

break;

case '\u092C':

engText=engText+"ba";

break;

case '\u092D':

engText=engText+"bha";

break;

case '\u092E':

engText=engText+"ma";

break;

case '\u092F':

engText=engText+"ya";

break;

case '\u0930':

engText=engText+"ra";

break;

case '\u0931':

engText=engText+"rra";

break;

case '\u0932':

engText=engText+"la";

break;

case '\u0933':

engText=engText+"la";

break;

case '\u0934':

engText=engText+"lla";

break;

case '\u0935':

engText=engText+"va";

break;

case '\u0936':

engText=engText+"sha";

break;

case '\u0937':

engText=engText+"sha";

break;

case '\u0938':

engText=engText+"sa";

break;

case '\u0939':

engText=engText+"ha";

break;

case '\u093A':

engText=engText+"oe";

break;

case '\u093B':

engText=engText+"ooe";

break;

case '\u093C':

engText=engText+"";

break;

case '\u093D':

engText=engText+"";

break;

case '\u093E':

engText=engText+"a";

break;

case '\u093F':

engText=engText+"i";

break;

case '\u0940':

engText=engText+"ee";

break;

case '\u0941':

engText=engText+"u";

break;

case '\u0942':

engText=engText+"oo";

break;

case '\u0943':

engText=engText+"r";

break;

case '\u0944':

engText=engText+"rr";

break;

case '\u0945':

engText=engText+"a";

break;

case '\u0946':

engText=engText+"e";

break;

case '\u0947':

engText=engText+"e";

break;

case '\u0948':

engText=engText+"ai";

break;

case '\u0949':

engText=engText+"o";

break;

case '\u094A':

engText=engText+"o";

break;

case '\u094B':

engText=engText+"o";

break;

case '\u094C':

engText=engText+"au";

break;

case '\u094D':

engText=engText+"";

break;

case '\u094E':

engText=engText+"e";

break;

case '\u094F':

engText=engText+"aw";

break;

case '\u0966':

engText=engText+"0";

break;

case '\u0967':

engText=engText+"1";

break;

case '\u0968':

engText=engText+"2";

break;

case '\u0969':

engText=engText+"3";

break;

case '\u096A':

engText=engText+"4";

break;

case '\u096B':

engText=engText+"5";

break;

case '\u096C':

engText=engText+"6";

break;

case '\u096D':

engText=engText+"7";

break;

case '\u096E':

engText=engText+"8";

break;

case '\u096F':

engText=engText+"9";

break;

default:

engText=engText+"|";

}

}

recognizedText=engText;

if ( recognizedText.length() != 0 ) {

\_field.setText(\_field.getText().toString().length() == 0 ? recognizedText : \_field.getText() + " " + recognizedText);

\_field.setSelection(\_field.getText().toString().length());

}

// Cycle done.

}

// www.Gaut.am was here

// Thanks for reading!

}