EXIF извлекаем информацию из фотографий

3-й курс/Закрытая зона

При съёмках современными цифровыми фотоаппаратами в изображение записываются специальные метаданные, описывающий условия съёмки, размеры изображения, авторство, модель устройства и т. п. Не являются исключением и камеры, встроенные в телефоны. Метаданные записываются и считываются в соответствии со специальным стандартом EXIF. В Android есть специальный класс **android.media.ExifInterface** для работы с данным стандартом, позволяющий легко и быстро извлечь необходимую информацию.

Добавим на форму текстовую метку, в которой будем выводить полученные данные. Далее сам код:

package ru.alexanderklimov.testapp;

import ...

public class TestAppActivity extends Activity {

private TextView mInfoTextView;

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

mInfoTextView = (TextView) findViewById(R.id.textView);

// установите свой путь к фотографии

String filename = Environment.getExternalStorageDirectory() + "/20130207\_140816.jpg";

try {

ExifInterface exif = new ExifInterface(filename);

showExif(exif);

} catch (IOException e) {

e.printStackTrace();

Toast.makeText(getApplicationContext(), "Ошибка!", Toast.LENGTH\_LONG).show();

}

}

private void showExif(ExifInterface exif) {

String attribute = "Информация об Exif:\n";

attribute += getTagString(ExifInterface.TAG\_DATETIME, exif);

attribute += getTagString(ExifInterface.TAG\_FLASH, exif);

attribute += getTagString(ExifInterface.TAG\_GPS\_LATITUDE, exif);

attribute += getTagString(ExifInterface.TAG\_GPS\_LATITUDE\_REF, exif);

attribute += getTagString(ExifInterface.TAG\_GPS\_LONGITUDE, exif);

attribute += getTagString(ExifInterface.TAG\_GPS\_LONGITUDE\_REF, exif);

attribute += getTagString(ExifInterface.TAG\_IMAGE\_LENGTH, exif);

attribute += getTagString(ExifInterface.TAG\_IMAGE\_WIDTH, exif);

attribute += getTagString(ExifInterface.TAG\_MAKE, exif);

attribute += getTagString(ExifInterface.TAG\_MODEL, exif);

attribute += getTagString(ExifInterface.TAG\_ORIENTATION, exif);

attribute += getTagString(ExifInterface.TAG\_WHITE\_BALANCE, exif);

attribute += getTagString(ExifInterface.TAG\_ISO, exif);

mInfoTextView.setText(attribute);

}

private String getTagString(String tag, ExifInterface exif) {

return (tag + " : " + exif.getAttribute(tag) + "\n");

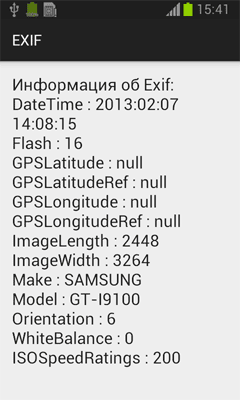
}

}

В данном случае у меня на телефоне стоит запрет на получение информации о GPS-координатах, поэтому часть данных недоступна. Подобную функциональность я добавил и в [файловый менеджер](http://developer.alexanderklimov.ru/android/fileexplorer.php).

Новые свойства в API 11

В Android 3.0 (API 11) у класса **ExifInterface** появились новые свойства **TAG\_APERTURE**, **TAG\_EXPOSURE\_TIME** и **TAG\_ISO**.



Устанавливаем свои атрибуты

Можно не только считывать данные из фотографий, но и записывать свои значения при помощи методов **setAttribute()** и **saveAttributes()**. Это может понадобиться в целях безопасности или наоборот, если в телефоне нет GPS, а вас есть другой источник данных о местоположении и вы хотите добавить информацию в фотографию. Следует учитывать, что если добавляемые данные будут некорректны, то они будут проигнорированы и не запишутся в файл.

Запишем фальшивые данные в фотографию:

public void updateGeoTag(){

// данные от балды

final String DUMMY\_GPS\_LATITUDE = "23/1,21/1,299295/32748";

final String DUMMY\_GPS\_LATITUDE\_REF = "N";

final String DUMMY\_GPS\_LONGITUDE = "104/1,3/1,207045/4096";

final String DUMMY\_GPS\_LONGITUDE\_REF = "E";

exif.setAttribute(ExifInterface.TAG\_GPS\_LATITUDE, DUMMY\_GPS\_LATITUDE);

exif.setAttribute(ExifInterface.TAG\_GPS\_LATITUDE\_REF, DUMMY\_GPS\_LATITUDE\_REF);

exif.setAttribute(ExifInterface.TAG\_GPS\_LONGITUDE, DUMMY\_GPS\_LONGITUDE);

exif.setAttribute(ExifInterface.TAG\_GPS\_LONGITUDE\_REF, DUMMY\_GPS\_LONGITUDE\_REF);

try {

exifInterface.saveAttributes();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

Конвертируем данные GPS в градусы

Данные о местоположении съёмки записываются в формате GPS, что не совсем удобно для дальнейшего использования. Поэтому нужно самостоятельно конвертировать значения в более удобный вид. Создадим новый класс GeoDegree:

import android.media.ExifInterface;

public class GeoDegree {

private boolean valid = false;

Float Latitude, Longitude;

GeoDegree(ExifInterface exif) {

String attrLATITUDE = exif.getAttribute(ExifInterface.TAG\_GPS\_LATITUDE);

String attrLATITUDE\_REF = exif

.getAttribute(ExifInterface.TAG\_GPS\_LATITUDE\_REF);

String attrLONGITUDE = exif

.getAttribute(ExifInterface.TAG\_GPS\_LONGITUDE);

String attrLONGITUDE\_REF = exif

.getAttribute(ExifInterface.TAG\_GPS\_LONGITUDE\_REF);

if ((attrLATITUDE != null) && (attrLATITUDE\_REF != null)

&& (attrLONGITUDE != null) && (attrLONGITUDE\_REF != null)) {

valid = true;

if (attrLATITUDE\_REF.equals("N")) {

Latitude = convertToDegree(attrLATITUDE);

} else {

Latitude = 0 - convertToDegree(attrLATITUDE);

}

if (attrLONGITUDE\_REF.equals("E")) {

Longitude = convertToDegree(attrLONGITUDE);

} else {

Longitude = 0 - convertToDegree(attrLONGITUDE);

}

}

};

private Float convertToDegree(String stringDMS) {

Float result = null;

String[] DMS = stringDMS.split(",", 3);

String[] stringD = DMS[0].split("/", 2);

Double D0 = Double.valueOf(stringD[0]);

Double D1 = Double.valueOf(stringD[1]);

Double FloatD = D0 / D1;

String[] stringM = DMS[1].split("/", 2);

Double M0 = Double.valueOf(stringM[0]);

Double M1 = Double.valueOf(stringM[1]);

Double FloatM = M0 / M1;

String[] stringS = DMS[2].split("/", 2);

Double S0 = Double.valueOf(stringS[0]);

Double S1 = Double.valueOf(stringS[1]);

Double FloatS = S0 / S1;

result = new Float(FloatD + (FloatM / 60) + (FloatS / 3600));

return result;

};

public boolean isValid() {

return valid;

}

@Override

public String toString() {

// TODO Auto-generated method stub

return (String.valueOf(Latitude) + ", " + String.valueOf(Longitude));

}

public int getLatitudeE6() {

return (int) (Latitude \* 1000000);

}

public int getLongitudeE6() {

return (int) (Longitude \* 1000000);

}

}

Источник: [Convert Exif GPS info to Degree format](http://android-er.blogspot.ru/2010/01/convert-exif-gps-info-to-degree-format.html)