**How to crop picture programmatically**

1. **Introduction**

This tutorial will help you know how to crop a picture in android SDCard.

1. **When should you apply this?**

You should apply this when you want to adjust your images in square shape and point to different locations on the photo.

1. **Advantages**

Freely edit your picture.

1. **Disadvantages**

Android phone must have SDCard, and difficult to code.

1. **Requirement**

Eclipse and emulator or real phone.

1. **Steps and Screen shots**

* First, you need to get image from SDCard. I will crop image in sdcard assuming I already know image name and its location.
* In this code below, I will get image in mnt/sdcard/Nish. mnt/sdcard is the sdcard location in DDMS and Nish is my own folder that I created. Then I get an image named temp.jpg

**private** Intent cameraIntent;

**private** Uri mImageCaptureUri;

**private** **static** **final** **int** *PICK\_FROM\_CAMERA* = 9, *CROP\_FROM\_CAMERA* = 10;

**public** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

File dir = **new** File(Environment.*getExternalStorageDirectory*()

.getAbsolutePath() + "/Nish");

dir.mkdirs();

File file = **new** File(dir, "temp.jpg");

mImageCaptureUri = Uri.*fromFile*(file);

cameraIntent.putExtra(android.provider.MediaStore.*EXTRA\_OUTPUT*,

mImageCaptureUri);

cameraIntent.putExtra("return-data", **true**);

startActivityForResult(cameraIntent, *PICK\_FROM\_CAMERA*);

}

* The last line in the above code will call a method named onActivityResult() which I use to call crop method.

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

**if** (cameraIntent != **null**) {

**if** (requestCode == *PICK\_FROM\_CAMERA*) {

doCrop();

}

**if** (requestCode == *CROP\_FROM\_CAMERA*) {

Bundle extras = data.getExtras();

**if** (extras != **null**) {

Bitmap photo = extras.getParcelable("data");

File dir = **new** File(Environment .*getExternalStorageDirectory*().getAbsolutePath()

+ "/Nish");

dir.mkdirs();

File file = **new** File(dir, "temp.jpg");

**try** {

FileOutputStream out = **new** FileOutputStream(file);

photo.compress(Bitmap.CompressFormat.*JPEG*, 100, out);

out.flush();

out.close();

} **catch** (Exception e) {

}

//do something with image after cropping.

}

}

}

* Finally, create method doCrop()

**private** **void** doCrop() {

Utility.*displayErrorConnection*(**this**);

Intent intent = **new** Intent("com.android.camera.action.CROP");

intent.setType("image/\*");

List<ResolveInfo> list = getPackageManager().queryIntentActivities(intent, 0);

**int** size = list.size();

**if** (size == 0) {

Toast.*makeText*(**this**, "Can not find image crop app",

Toast.*LENGTH\_SHORT*).show();

**return**;

} **else** {

Log.*d*("CAMERA", "size > 0");

intent.setData(mImageCaptureUri);

intent.putExtra("outputX", 200);

intent.putExtra("outputY", 200);

intent.putExtra("aspectX", 1);

intent.putExtra("aspectY", 1);

intent.putExtra("scale", **true**);

intent.putExtra("return-data", **true**);

**if** (size == 1) {

// Intent i = new Intent(intent);

ResolveInfo res = list.get(0);

intent.setComponent(**new** ComponentName(

res.activityInfo.packageName, res.activityInfo.name));

startActivityForResult(intent, *CROP\_FROM\_CAMERA*);

}

}

}

* After this method finish, which means when you press SAVE, onActivityResult() will be called again with CROP\_FROM\_CAMERA request code.
* Additional classes:

**public** **class** CropOption {

**public** CharSequence title;

**public** Drawable icon;

**public** Intent appIntent;

}

**public** **class** CropOptionAdapter **extends** ArrayAdapter<CropOption> {

**private** ArrayList<CropOption> mOptions;

**private** LayoutInflater mInflater;

**public** CropOptionAdapter(Context context, ArrayList<CropOption> options) {

**super**(context, R.layout.*crop\_selector*, options);

mOptions = options;

mInflater = LayoutInflater.*from*(context);

}

@Override

**public** View getView(**int** position, View convertView, ViewGroup group) {

**if** (convertView == **null**)

convertView = mInflater.inflate(R.layout.*crop\_selector*, **null**);

CropOption item = mOptions.get(position);

**if** (item != **null**) {

((ImageView) convertView.findViewById(R.id.*iv\_icon*))

.setImageDrawable(item.icon);

((TextView) convertView.findViewById(R.id.*tv\_name*))

.setText(item.title);

**return** convertView;

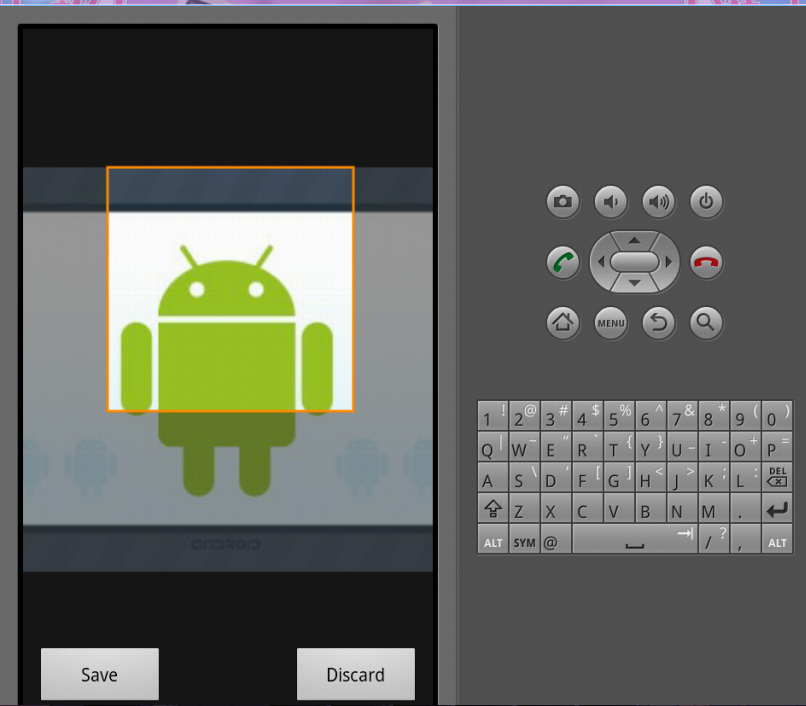
}

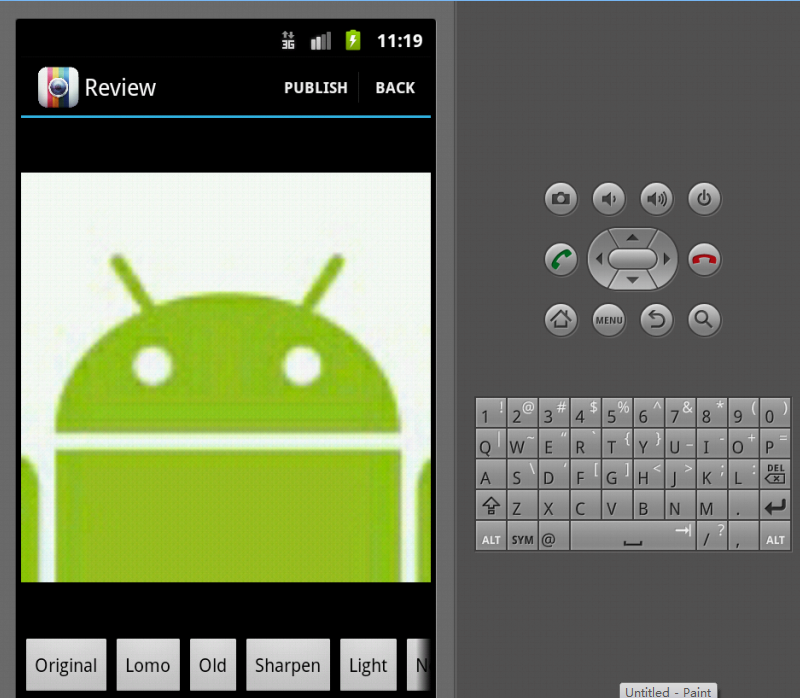
**return** **null**;

}

}

* Here is an image before and after cropping





* Remember to add this in your manifest files:

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

<uses-permission android:name=*"android.permission.WRITE\_EXTERNAL\_STORAGE"* />

1. **Reference**

<http://stackoverflow.com/questions/2085003/how-to-select-and-crop-an-image-in-android>

<https://github.com/lorensiuswlt/AndroidImageCrop>

Contact me if you have any question: bachxvu@gmail.com