Фильтры для масок EmbossMaskFilter, BlurMaskFilter

3-й курс/Закрытая зона



[EmbossMaskFilter](http://developer.alexanderklimov.ru/android/catshop/maskfilters.php#embossmaskfilter)  
[BlurMaskFilter](http://developer.alexanderklimov.ru/android/catshop/maskfilters.php#blurmaskfilter)

Класс **MaskFilter** позволяет назначать контурные эффекты для объекта **Paint**. Классы, наследующие **MaskFilter**, применяют преобразования к альфа-каналу объекта **Paint** вдоль его внешней границы.

Android включает следующие фильтры для масок:

* **EmbossMaskFilter** — задает направление источника света и уровень освещённости, создавая эффект рельефа.
* **BlurMaskFilter** — задает стиль размытия и радиус выступов для краев объекта **Paint**;

Чтобы применить фильтр для маски, используйте метод **setMaskFilter()**, передавая ему в качестве параметра объект **MaskFilter**.

EmbossMaskFilter

Нарисуем круг и применим фильтр **EmbossMaskFilter** к объекту **Paint**, чтобы увидеть, как будет выглядеть круг после применения фильтра. Заодно применим этот же фильтр и к тексту

package ru.alexanderklimov.test;

// Если этот код работает, его написал Александр Климов,

// а если нет, то не знаю, кто его писал.

import android.app.Activity;

import android.content.Context;

import android.graphics.Canvas;

import android.graphics.Color;

import android.graphics.EmbossMaskFilter;

import android.graphics.Paint;

import android.os.Bundle;

import android.view.View;

public class MainActivity extends Activity {

/\*\* Called when the activity is first created. \*/

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(new Draw2D(this));

}

public class Draw2D extends View {

private Paint mPaint;

private EmbossMaskFilter mEmbossFilter;

public Draw2D(Context context) {

super(context);

mPaint = new Paint();

setLayerType(LAYER\_TYPE\_SOFTWARE, mPaint);

mPaint.setColor(Color.GREEN); // установим зеленый цвет

mPaint.setStyle(Paint.Style.FILL);

mPaint.setStrokeWidth(5);

mPaint.setTextSize(78.0f); // размер текста

mPaint.setAntiAlias(true);

// Задайте направление источника света

float[] direction = new float[] { 1, 1, 1 };

// Установите уровень освещенности

float light = 0.2f;

// Выберите степень зеркальности

float specular = 3.0f;

// Укажите, какой уровень размытия должен применяться к маске

float blur = 10.5f;

mEmbossFilter = new EmbossMaskFilter(direction, light, specular,

blur);

}

@Override

protected void onDraw(Canvas canvas) {

super.onDraw(canvas);

float width, height, cx, cy, radius;

width = getWidth();

height = getHeight();

cx = width / 2;

cy = height / 2;

if (width > height) {

radius = height / 4;

} else {

radius = width / 4;

}

// Примените маску

mPaint.setMaskFilter(mEmbossFilter);

canvas.drawText("EmbossMaskFilter", 10, 80, mPaint);

canvas.drawCircle(cx, cy, radius, mPaint);

}

}

}

Первая редакция статьи писалась ещё для Android 2.3. Когда спустя некоторое время я решил использовать этот код на новом устройстве, то с удивлением обнаружил, что текст не выводится. Для решения проблемы пришлось добавить вызов метода **setLayerType(LAYER\_TYPE\_SOFTWARE, mPaint)**. Ау, Гугл, предупреждать надо.



Эффект можно применить и к TextView.

public void onClick(View v) {

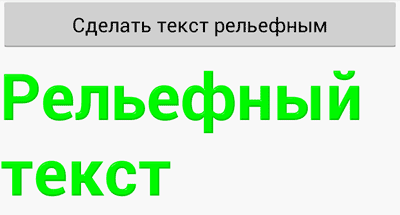
TextView tvInfo = (TextView)findViewById(R.id.textView1);

MaskFilter filter = new EmbossMaskFilter(new float[]{0.5f, 0.8f, 1f}, 0.8f, 15f, 2f);

tvInfo.getPaint().setMaskFilter(filter);

tvInfo.invalidate();

}



BlurMaskFilter

Немного изменим код, чтобы применить фильтр размытия. Удалим код для маски **EmbossMaskFilter** и добавим маску **BlurMaskFilter** с одним из четырёх возможных вариантов.

// Примените какую-нибудь маску

mPaint.setMaskFilter(new BlurMaskFilter(5, Blur.NORMAL));

mPaint.setMaskFilter(new BlurMaskFilter(3, Blur.SOLID));

mPaint.setMaskFilter(new BlurMaskFilter(5, Blur.INNER));

mPaint.setMaskFilter(new BlurMaskFilter(5, Blur.OUTER));

На рисунке представлены варианты для **Blur.NORMAL** и **Blur.OUTER**.