Filter цветовые фильтры

Продолжаем тренироваться на кошках. В графических редакторах есть возможность устанавливать цветовые фильтры (в Photoshop это Photo Filter). Тем, кому приходилось проявлять фотоплёнку в прошлом веке, знаком красный фильтр - красная лампа в ванной была обязательным атрибутом домашнего фотографа.

Реализуем это на Android. За основу возьмём [предыдущий пример](http://developer.alexanderklimov.ru/android/catshop/matrix.php#prescale) и удалим из него вторую кнопку. Как и прежние операции, связанные с обработкой каждого пикселя изображения, данный процесс занимает значительное время.

Перейдём к коду.

butDemo.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

BitmapDrawable mydrawable = (BitmapDrawable) imageView.getDrawable();

Bitmap b = mydrawable.getBitmap();

b = doColorFilter(b, 1.5, 1.5, 1.5);

imageView.setImageBitmap(b);

setTitle("Фильтр: 1.5, 1.5, 1.5");

}

});

public static Bitmap doColorFilter(Bitmap src, double red, double green, double blue) {

// размеры изображения

int width = src.getWidth();

int height = src.getHeight();

// создаем финальное изображение 1098

Bitmap bmOut = Bitmap.createBitmap(width, height, src.getConfig());

// информация о цвете

int A, R, G, B;

int pixel;

// проходим через каждый пиксель

for(int x = 0; x < width; ++x) {

for(int y = 0; y < height; ++y) {

// получаем цвет пикселя

pixel = src.getPixel(x, y);

// применяем фильтр к каждому каналу R, G, B

A = Color.alpha(pixel);

R = (int)(Color.red(pixel) \* red);

G = (int)(Color.green(pixel) \* green);

B = (int)(Color.blue(pixel) \* blue);

// применяем новые значения цвета к изображению

bmOut.setPixel(x, y, Color.argb(A, R, G, B));

}

}

// возвращаем новое изображение

return bmOut;

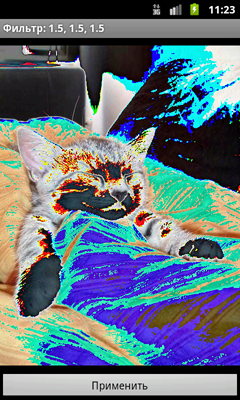
}

Попробуйте самостоятельно поиграться с различными значениями.

Это наш источник. Какая прелесть!



А вот что сделали вы с ним, садист.

Р