首页\_dog\_info



RelativeLayout

占位控件

View

Textview

Textview

View

View

Textview

Textview

TV

TV

LinearLayout

ImageView

TV

Textview

ImageView

1. ImageView圆形显示效果
2. 采用自定义imageView，
3. 借鉴文章
4. 代码：

**package** com.example.jack.dogsmaganerapp.user\_difined\_view;  
  
  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.BitmapShader;  
**import** android.graphics.Canvas;  
**import** android.graphics.Matrix;  
**import** android.graphics.Paint;  
**import** android.graphics.Shader;  
**import** android.graphics.drawable.BitmapDrawable;  
**import** android.support.annotation.Nullable;  
**import** android.support.v7.widget.AppCompatImageView;  
**import** android.util.AttributeSet;  
  
**public class** CircleImageView **extends** AppCompatImageView{  
  
*// 控件大小长宽* **private float width**;  
 **private float height**;  
*// 圆形半径* **private float radius**;  
*// 画笔* **private** Paint **paint**;  
*// 控制图片旋转、缩放，裁剪的矩阵* **private** Matrix **matrix**;  
  
*// imageview的三个构造函数* **public** CircleImageView(Context context)  
 {  
 **this**(context, **null**);  
 }  
  
 **public** CircleImageView(Context context, @Nullable AttributeSet attrs)  
 {  
 **this**(context, attrs, 0);  
 }  
  
 **public** CircleImageView(Context context, @Nullable AttributeSet attrs, **int** defStyleAttr)  
 {  
 **super**(context, attrs, defStyleAttr);  
 **paint** = **new** Paint();*//初始化画笔* **paint**.setAntiAlias(**true**); *//设置抗锯齿* **matrix** = **new** Matrix(); *//初始化缩放矩阵* }  
  
 */\*\* \* 测量控件的宽高，并获取其内切圆(因控件默认是矩形，所以这里称为内切圆）的半径 \*/* @Override **protected void** onMeasure(**int** widthMeasureSpec, **int** heightMeasureSpec)  
 {  
 **super**.onMeasure(widthMeasureSpec, heightMeasureSpec);  
 **width** = getMeasuredWidth();  
 **height** = getMeasuredHeight();  
*// 半径为控件长宽中断的一边的一半* **radius** = Math.*min*(**width**, **height**) / 2;  
 }  
  
*// 绘制* @Override **protected void** onDraw(Canvas canvas)  
 {  
 **paint**.setShader(initBitmapShader());*//将着色器设置给画笔  
// 参数：圆心x、圆心y，半径、画笔* canvas.drawCircle(**width** / 2, **height** / 2, **radius**, **paint**);*//使用画笔在画布上画圆* }  
  
 */\*\* \* 获取ImageView中资源图片的Bitmap  
 \* ，利用Bitmap初始化图片着色器,通过缩放矩阵将原资源图片缩放到铺满整个绘制区域  
 \* ，避免边界填充 \*/* **private** BitmapShader initBitmapShader()  
 {  
*// bitmap代表图片信息，英语直译为位图* Bitmap bitmap = ((BitmapDrawable) getDrawable()).getBitmap();  
*// shader直译著色器* BitmapShader bitmapShader = **new** BitmapShader(bitmap, Shader.TileMode.***CLAMP***, Shader.TileMode.***CLAMP***);  
*// 求图片缩放比例，单精度类型小数。  
// 控件的长宽/图片的长宽分别获得长宽的缩放比例，有限满足大的，否则长边将无法再控件中显示* **float** scale = Math.*max*(**width** / bitmap.getWidth(), **height** / bitmap.getHeight());  
 **matrix**.setScale(scale, scale);*//将图片宽高等比例缩放，避免拉伸  
// 把设置好缩放比例的图片矩阵告诉著色器* bitmapShader.setLocalMatrix(**matrix**);  
 **return** bitmapShader;  
 }  
  
  
  
}

1. 分界线颜色渐变效果
2. 设置控件边界阴影效果有很多种方式，这里选取了一种比较简单的方式
3. 借鉴文章

<https://blog.csdn.net/yezhouyong/article/details/51329221>

代码：

*<?***xml version="1.0" encoding="utf-8"***?>*<**shape xmlns:android="http://schemas.android.com/apk/res/android" android:shape="rectangle"**>  
 <**corners  
 android:radius="1.5dp"  
 android:bottomRightRadius="0dp"**/>  
 <**stroke  
 android:width="1dp"  
 android:color="#E3E3E1"** />  
  
 <**solid android:color="@color/blanc"** />  
  
</**shape**>