



链滴

# Android-表情键盘的制作

作者: [wind](#)

原文链接: <https://ld246.com/article/1449630118556>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

IM应用中必不可少的就是表情键盘，最近研究了一下，发现表情键盘还是蛮容易做的。

布局

布局文件主要有两个文件了，一个是要放在聊天对话框里的ViewPager，一个是给ViewPager添加的View。

很明显ViewPager是放在聊天布局的最底部，并且visibility="gone"，当点击表情按钮的时候使visibility="visible"。这个布局里我是把高度给写死了，因为如果不写死的话，在代码中好像布局会出现问题在网上也看到大多数的人都是给写死的，当然也可以用onGlobalLayout的监听去动态配置尺寸。

第2个布局就是ViewPager中的View，这里我使用GridView，并设置column=7。

配置ViewPager中的View

搞好布局接下来就是ViewPager中的View了。毫无疑问，一个是ViewPagerAdapter，一个是GridViewAdapter，大体思路就是：计算表情包中的表情需要几个page，分好之后对GridView实例进行加载数据，然后把一个一个gridview加载到ViewPager中去。

首先看看计算表情各种属性的代码，我放在EmotionHelper类中：

```
int pages=emojiCodes.length/ONE_PAGE_SIZE+(emojiCodes.length%ONE_PAGE_SIZE==0?0:1);
for(int i=0;i<pages;i++){
    List<String> onePageEmojis = new ArrayList<>();
    int start=i*ONE_PAGE_SIZE;
    int end=start+Math.min(ONE_PAGE_SIZE,emojiCodes.length-start);
    for(int j=start;j<end;j++){
        onePageEmojis.add(emojiCodes[j]);
    }
    emogroups.add(onePageEmojis);
}
```

emojiCodes是表情相对应的字符串，page是得到表情的页数，emogroups是多个page的数组列表这里已经完成了表情框各个数据的初始化了，接下来就是在adapter中利用字符串查找到对应的表情片，完成加载。

ImageView在TextView中的显示问题

最开始这个还想了我好一会，要在TextView中显示ImageView该怎么做？后来还上网找到了一个Spannable的东西，看英文解释就是可以把类似于ImageView的东西attach到TextView上吧。看代码如何成可以显示ImageView的SpannableString吧：

```
public static CharSequence replace(Context context, String text) {
    if (TextUtils.isEmpty(text)) {
        return text;
    }
    SpannableString spannableString = new SpannableString(text);
    Matcher matcher = pattern.matcher(text);
    while (matcher.find()) {
        String factText = matcher.group();
        String key = factText.substring(1, factText.length() - 1);
        if (contain(emojiCodes, factText)) {
            Bitmap bitmap = getDrawableBitmap(context, key);
            ImageSpan image = new ImageSpan(context, bitmap);
        }
    }
}
```

```
        int start = matcher.start();
        int end = matcher.end();
        spannableString.setSpan(image, start, end,
            Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
    }
}
return spannableString;
}
```

这里还了解了一下Pattern和matcher的知识，例如我这里的表情格式是这样的：":clap:"，对应的pattern就是

```
`pattern = Pattern.compile("\\:[a-z]*\\:");`
```

pattern.matcher就是基于text，利用这个pattern去寻找匹配字符串，faceText就是得到的表情对应字符串值，之后将对应的bitmap初始化为ImageSpan放入spannableString中去。

表情键盘做好之后，和软键盘的冲突没有很好的协调，下次有机会再好好研究吧。