

# 美妆bundle参数说明

## landmark修改

landmark修改主要为编辑美妆使用的特征点，优化效果使用。

landmark的修改主要通过两个参数控制，客户端同学需要设置这两个参数来修改和使用landmark：

1. is\_use\_fix：这个参数控制是否使用修改过得landmark点，如果设为1为使用，0为不使用
2. fix\_makeup\_data：这个参数为一个数组，需要客户端传递一个数组进去，传递的数组的长度为 150\*人脸数，也就是将所有的点位信息存储的数组中传递进来。

### 注意：

这里fix\_makeup\_data长度为150的原因是：咱们一共返回75个landmark点，每个landmark点有x和y两个数据，所以为  $75 * 2 = 150$

## 其他参数

```
is_makeup_on: 1,      //美妆开关
makeup_intensity:1.0,    //美妆程度
//下面是每个妆容单独的参数，intensity设置为0即为关闭这种妆效
makeup_intensity_lip:1.0,    //kouhong
makeup_intensity_pupil:1.0,  //meitong
makeup_intensity_eye:1.0,    //yanying
makeup_intensity_eyeLiner:1.0, //yanxian
makeup_intensity_eyelash:1.0, //jiemao
makeup_intensity_eyeBrow:1.0, //meimao
makeup_intensity_blusher: 1.0, //saihong
makeup_lip_color: [0,0,0,0]  //长度为4的数组，rgba颜色值
makeup_lip_mask: 0.0        //嘴唇优化效果开关，1.0为开 0为关
```

## 使用face\_makeup.bundle参数替换当前妆容的纹理

共7个纹理可供替换

使用fultemSetParam(itemid,str\_name,value)

str\_name 可以取以下值

tex\_brow 眉毛

tex\_eye 眼影

tex\_pupil 美瞳

tex\_eyeLash 睫毛

tex\_highlight 口红高光

tex\_eyeLiner 眼线

tex\_blusher腮红

value对应一个u8类型的数组长度为n，其中前[0-7]bytes是图片宽和高,[8-n]是图片rgba数据

或者可以使用**fuCreateTexForItem**接口，直接在接口中传入图片数据即可，建议使用**fuCreateTexForItem**，参数名称同上。

## 美妆bundle加载顺序（如果拆分使用）

---

眼影，眼线，睫毛，美瞳，眉毛，嘴唇

**注意：**这里眼影，眼线，睫毛的加载顺序不能调换，因为加载顺序就是绘制顺序，涉及到妆容的效果