

新版美狐SDK集成文档（Android）

山东小狐狸网络科技有限公司



目 录

一、产品简介

- 1.功能说明
- 2.平台支持
- 3.官方链接

二、集成指引

Android集成指引

- 1.概述
- 2.接入说明
- 3.详细步骤
- 4.具体直播平台接入
 - 1.腾讯云直播
 - 2.阿里云

三、MHSDK美颜相关接口部分

Android相关接口文档说明

美颜相关接口文档说明

四、常见问题

Android常见问题

- 1.认证失败原因
- 2.调用美型、贴纸、特效、哈哈镜没有效果
- 3.MHBeautyManager的特别声明
- 4.使用MHUI的注意事项
- 5.卡顿问题
- 6.闪退问题
- 7.贴纸无法下载
- 8.AndroidX(腾讯云)环境集成问题
- 9.导入新的sdk出现乱码
- 10.弹出的UI有阴影效果
- 11.Android10出现网络资源失败请检查网络
- 12.美狐可以记录设置的各种参数吗
- 13.(七牛云)替换sdk画面上下颠倒
- 14.贴纸可以自定义吗
- 15.您这边可以录制视频吗
- 16.MHSDK必须本地依赖吗
- 17.找不到某个方法
- 18.环信不显示自己的头像框
- 19.如何取消特效
- 20.水印没有找到
- 21.声网视频通话大小屏切换问题

一、产品简介

1.功能说明

2.平台支持

3.官方链接

1.功能说明

****1.功能说明 ****

MHSDK是一个为市面上主流的直播，短视频SDK提供自定义视频渲染的一个美颜类库。类库分为基础版本和高级版本。基础版本含有美颜(美白、磨皮、红润、亮度)，美型(大眼、瘦脸、嘴型、鼻型、下巴、额头)、贴纸（基础贴纸、基础面具）、滤镜(美颜滤镜)功能，此外还包括抖动特效、哈哈镜、水印等功能，高级版本除了以上功能外，增加了更多美型（眉毛、眼距、眼角、削脸、长鼻、开眼角）功能，并且提供了一键美颜以及更多贴纸样式和美颜效果，满足用户的不同需求。

2.平台支持

2.平台支持

MHSDK提供对OpenGL纹理等视频数据的渲染功能，兼容腾讯云、金山云、即构科技等平台的直播SDK、短视频SDK等。

3.官方链接

美狐官方链接地址：<http://www.facegl.com/>

二、集成指引

[Android集成指引](#)

Android集成指引

- 1.概述
- 2.接入说明
- 3.详细步骤
- 4.具体直播平台接入

1.概述

1.概述

美狐SDK整理流程比较简单，但是细节点比较大，请仔细阅读本文档，以防出现无法预料的错误。

建议集成前先查看您的项目使用的直播SDK的说明文档，理清直播SDK的整体使用流程，尤其是基础美颜功能。

2.接入说明

2.接入说明

MHSDK支持多个平台的直播SDK，但是集成方式不尽相同，所以会针对不同平台进行详细说明，但整体思路都是一样的，比较简单清晰。

MHSDK集成和使用总体分为:初始化MHSDK、初始化MhDataManager、渲染 简单几步.

3.详细步骤

1.申请产品密钥

MHSDK 首先需要向商务申请精简版、基础或者高级版的产品密钥，然后将密钥集成到项目中方可使用。

2.类库导入

将MHUI的文件夹拷贝到Android工程根目录，

MHSDK目前支持、armv7，armv8版本，defaultConfig配置节点中添加（根据自身项目需求来设置）：

ndk {

```
abiFilters "armeabi-v7a", "arm64-v8a"
```

}

引入mhui模块

```
implementation project(':mhui')
```

3.初始化SDK

```
MHSDK.init(this, "您的AppID", "您的授权密钥");//建议在Application子类的onCreate方法中初始化SDK
```

4.添加美狐UI组件

在布局文件中添加美狐的UI组件

```
<com.meihu.beauty.views.MeiHuBeautyControl  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"/>
```

5.在集成了美狐UI组件的Activity中初始化

```
//初始化  
MhDataManager.getInstance().create(this.getApplicationContext());
```

6.在直播SDK的渲染方法 调用美狐SDK的渲染方法 返回新生成的纹理ID

3.详细步骤

```
int textureId = MhDataManager.getInstance().render(texture, width, height);
return textureId;
//修改MhDataManager#render
public int render(int texture, int width, int height) {
    if (mMhManager != null) {
        try {
            texture = mMhManager.renderxxx(...);
        } catch (Exception e) {
            e.printStackTrace();
        }
    } else {
        Log.e(TAG, "render: error");
    }
    return texture;
}

//          阿里云：render18(...)
//          慕色短视频，七牛短视频：render3(...)
//          网易，render4(...)
//          声网：render16(...)
//          七牛直播，融云：render5(...)
//          即构：render6(...)
//          金山：render7(...)
//          声网视频通话(前置摄像头)：render10(...)
//          声网视频通话(后置摄像头)：render11(...)
//          腾讯直播：render12(...)
//          三体：render14(...)
```

7.如何您使用了MhDataManager结束时销毁防止内存泄漏

```
MhDataManager.getInstance().release();
```

4.具体直播平台接入

1.腾讯云直播

2.阿里云

1.腾讯云直播

1.初始化SDK

```
MHSDK.init(this, "您的AppID", "您的授权密钥");//建议在Application子类的onCreate方法中初始化SDK
```

2.添加美狐UI组件

在布局文件中添加美狐的UI组件

```
<com.meihu.beauty.views.MeiHuBeautyControl  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"/>
```

3.在集成了美狐UI组件的Activity中初始化

```
//初始化  
MhDataManager.getInstance().create(this.getApplicationContext());
```

4.在腾讯云直播的自定义渲染方法中 调用美狐SDK的渲染方法

实现TXLivePusher.VideoCustomProcessListener接口

调用 livePusher.setVideoProcessListener(this); 设置回调。

```
@Override  
public int onTextureCustomProcess(int texture, int width, int height) {  
  
    int newTexture = MhDataManager.getInstance().render(texture,width,height);  
  
    return newTexture;  
}
```

2.阿里云

1.初始化SDK

```
MHSDK.init(this, "您的AppID", "您的授权密钥");//建议在Application子类的onCreate方法中初始化SDK
```

2.添加美狐UI组件

在布局文件中添加美狐的UI组件

```
<com.meihu.beauty.views.MeiHuBeautyControl  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"/>
```

3.在集成了美狐UI组件的Activity中初始化

```
//初始化  
MhDataManager.getInstance().create(this.getApplicationContext());
```

4.在阿里的渲染方法中 调用美狐SDK的渲染方法

```
int newTexture = MhDataManager.getInstance().render(texture,width,height);  
  
return newTexture;
```

三、MHSDK美颜相关接口部分

[Android相关接口文档说明](#)

Android相关接口文档说明

[美颜相关接口文档说明](#)

美颜相关接口文档说明

如有需要，调用MHBeautyManager里面的相关接口设置美颜参数，参数列表如下：

分类	功能	对应函数	参数范围
美颜	美白	setSkinWhiting(int progress)	progress: [0,9]
美颜	磨皮	setSkinSmooth(int progress)	progress: [0,9]
美颜	红润	setSkinTenderness(int progress)	progress: [0,9]
美颜	亮度	setBrightness(int progress)	progress: [0,100]
美型	大眼	setBigEye(int progress)	progress: [0,100]
美型	瘦脸	setFaceLift(int progress)	progress: [0,100]
美型	嘴型	setMouseLift(int progress)	progress: [0,100]
美型	瘦鼻	setNoseLift(int progress)	progress: [0,100]
美型	下巴	setChinLift(int progress)	progress: [0,100]
美型	额头	setForeheadLift(int progress)	progress: [0,100]
美型	眉毛	setEyeBrow(int progress)	progress: [0,100]
美型	眼角	setEyeCorner(int progress)	progress: [0,100]
美型	眼距	setEyeLength(int progress)	progress: [0,100]
美型	开眼角	setEyeAlat(int progress)	progress: [0,100]
美型	削脸	setFaceShave(int progress)	progress: [0,100]
美型	长鼻	setLengthenNoseLift(int progress)	progress: [0,100]

贴纸	贴纸	setSticker(final String stickerName, final int action, final boolean show,final boolean isAction,final String key)	stickerName:动作贴纸名称（使用前需要先获取动作贴纸列表 然后下载动作贴纸），action 动作id（0 //无 1 // 抬头 2 //张嘴 3 //眨眼），show 是否显示，isAction 是否为动作贴纸，key贴纸分类标识 MHConfigConstants.TIE_ZHI_1//基础贴纸_2//高级贴纸_3//基础面具_4//高级面具
滤镜	滤镜	setFilter(final int filterId);	filterId：滤镜资源id
特效		setSpeciallyEffect(String speciallyName)	speciallyName 特效名称
哈哈镜		setDistortionEffect(final int distortionName, final boolean isTeXiao)	distortionName 哈哈镜名称 isTeXiao：是否为特效里的哈哈镜
水印		setWatermark(Bitmap bitmap, WaterAlignEnum waterAlignEnum)	Bitmap 水印切图， waterAlignEnum 水印显示位置（屏幕四角）
动作贴纸	2.4.3以后版本生效	setSticker(final String stickerName, final int action, final boolean show,final	stickerName:动作贴纸名称（使用前需要先获取动作贴纸列表 然后下载动作贴纸），action 动作id（0 //无 1 // 抬头 2 //张嘴 3 //眨眼），show 是否显示，isAction

		boolean isAction,final String key)	是否为动作贴纸， key贴纸分类标识 MHConfigConst ants.TIE_ZHI_1// 基础贴纸_2//高级 贴纸_3//基础面具 _4//高级面具
美妆	2.4.5以后版本生效	setMakeup(int makeupId,boole an enable)	makeupId 美妆 类型 id (MHSDK.MA KEUP_NONE // 无 MHSDK.MAKEU P_LIPSTICK //唇 彩 MHSDK.MAKEU P_EYELASH //睫 毛 MHSDK.MAKEU P_BLUSH //腮 红)，enable 是 否开启

说明：大眼、瘦脸、嘴型、鼻型、下巴、额头是基础和高级版本都可以用，其余美型只能在高级版本才能使用。
混淆

```
-keep class com.meihu.**{*;*}
```

四、常见问题

[Android常见问题](#)

Android常见问题

- 1.认证失败原因
- 2.调用美型、贴纸、特效、哈哈镜没有效果
- 3.MHBeautyManager的特别声明
- 4.使用MHUI的注意事项
- 5.卡顿问题
- 6.闪退问题
- 7.贴纸无法下载
- 8.AndroidX(腾讯云)环境集成问题
- 9.导入新的sdk出现乱码
- 10.弹出的UI有阴影效果
- 11.Android10出现网络资源失败请检查网络
- 12.美狐可以记录设置的各种参数吗
- 13.(七牛云)替换sdk画面上下颠倒
- 14.贴纸可以自定义吗
- 15.您这边可以录制视频吗
- 16.MHSDK必须本地依赖吗
- 17.找不到某个方法
- 18.环信不显示自己的头像框
- 19.如何取消特效
- 20.水印没有找到
- 21.声网视频通话大小屏切换问题

1.认证失败原因

认证失败原因

确定网络正常；

确定初始化鉴权密钥是否正确，确保没有空格，包名和鉴权密钥需要一一对应。

2.调用美型、贴纸、特效、哈哈镜没有效果

调用美型、贴纸、特效、哈哈镜没有效果

检查权限是否获取；

检查网络是否正常（需访问外网）；

检查鉴权key是否正确；

检查创建BeautyView时是否获取到鉴权，查看 BeautyViewHolderFactory.getBeautyViewHolder (...) 中 String ver = MHSDK.getInstance().getVer()的值，ver=-1表示授权失败，如果key和网络均正常，但ver值始终不对，尝试修改初始化方式为

```
MHSDK.getInstance().clearVerNote().init(context, "appkey");
```

确认MHBeautyManager是否初始化。

3.MHBeautyManager的特别声明

请确保只有一个MHBeautyManager的实例化对象，否则可能造成不可预料的问题，并且正确及时销毁。

4.使用MHUI的注意事项

如使用MHUI，请不要使用aar包内的相关类（已废弃，且新版已删除），商务会提供最新ui库，并且修改类中import导入和layout中布局文件的自定义view的路径，确保自定义view控件能被正确识别

5.卡顿问题

如果感觉有点卡，检查分辨率，码率等是否过高，分辨率 建议设为540P；

如果发生异常卡顿问题（类似幻灯片效果），请检查推流 版本是否兼容，尝试替换低版本直播推流sdk。（ 1.0.8 之前的老版本美狐已知腾讯直播sdk6.7 、 6.8版本这边测试不兼容，表现为严重卡顿或闪退，新版本未现 ）

6.闪退问题

1.MHBeautyManager销毁之前只可创建一次，不可以同时new两个!

使用腾讯直播sdk时，使用setVideoQuality(int quality, boolean adjustBitrate, boolean adjustResolution)方法来设置视频编码质量，有可能会出现闪退问题。

2.第二次进入直播页面发生闪退:

请检查MHBeautyManager是否及时销毁，onDestroy方法可能会延迟调用，导致MHBeautyManager不能及时销毁，建议在onStop方法中调用停止预览画面，例如腾讯SDK。

```
@Override

protected void onStop() {

    if (isFinishing()) {

        mLivePusher.stopCameraPreview(true);

        mLivePusher.stopPusher();

        .....

    }

}
```

7.贴纸无法下载


AndroidManifest.xml文件的application节点添加

android:usesCleartextTraffic="true"属性。

8.AndroidX(腾讯云)环境集成问题

Could not determine artifacts for androidx.recyclerview:recyclerview:1.0.0: No cached version available for offline mode

解决问题的截图

- 
1. Go to preferences
 2. Build,Execution,Deployment
 3. Gradle
 4. Global gradle settings
 5. Uncheck offline work
 6. Then Rebuild your project/Try sync

9.导入新的sdk出现乱码

导入新的sdk出现乱码。解决方式重新倒入一次

10.弹出的UI有阴影效果

弹出的UI有阴影效果。原因布局冲突,查看一下工程UI使用的地方，解决冲突

11.Android10出现网络资源失败请检查网络

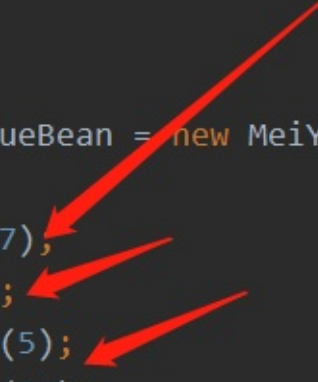
安卓10出现获取网络资源失败请检查网络，
原因安卓10自身权限问题将targetSdkVersion回退回28

12.美狐可以记录设置的各种参数吗

1.美狐可以记录设置的各种参数吗？

可以，可以在MhDataManager里的create方法里进行设置美颜参数

```
public MhDataManager create(Context context) {  
  
    mContext = context;  
    mInputValue = new MeiYanValueBean();  
    mUseValue = null;  
    mMeiYanValue = null;  
    mOneKeyValue = null;  
    try {  
        mMhManager = new MHBeautyManager(context, b: true);  
    } catch (Exception e) {  
        mMhManager = null;  
        e.printStackTrace();  
    }  
  
    MeiYanValueBean meiYanValueBean = new MeiYanValueBean();  
  
    meiYanValueBean.setMeiBai(7);  
    meiYanValueBean.setMoPi(7);  
    meiYanValueBean.setHongRun(5);  
    meiYanValueBean.setLiangDu(57);  
  
    meiYanValueBean.setDaYan(30);  
    meiYanValueBean.setMeiMao(0);  
    meiYanValueBean.setYanJu(50);  
    meiYanValueBean.setYanJiao(50);  
    meiYanValueBean.setShouLian(30);  
    meiYanValueBean.setZuiXing(50);  
}
```



使用方法

```
MhDataManager.getInstance().create(this.getApplicationContext());
```

13.(七牛云)替换sdk画面上下颠倒

(七牛云) 替换sdk画面上下颠倒。解决方案将下方图片的render5修改为render15.

```
/**
 * 视频帧的回调，回调纹理数据，可以送入第三方美颜的处理引擎
 *
 * @param texId 视频帧纹理
 * @param width 纹理宽度
 * @param height 纹理高度
 * @param transformMatrix 纹理变换矩阵
 * @return 处理后的纹理
 */
@Override
public int onDrawFrame(int texId, int width, int height, float[] transformMatrix) {
    int newTexId = mFBO.drawFrame(texId, width, height);

    if (mhBeautyManager == null) {
        return 0;
    }
    newTexId = mhBeautyManager.render5(newTexId, width, height);
    return newTexId;
}
};
```

14.贴纸可以自定义吗

1.贴纸可以自定义吗。

答案可以。需要将自定义的贴纸发送给我们来处理，目前支持2d贴纸。如果只是自己使用需要定制时间长，公开使用时间短，需收费。

15.您这边可以录制视频吗

可以集成腾讯短视频，七牛短视频，Movieous短视频的sdk。文档中有详细介绍。短视频sdk的收费需要客户自己去咨询，我们不提供方案。

16.MHSDK必须本地依赖吗

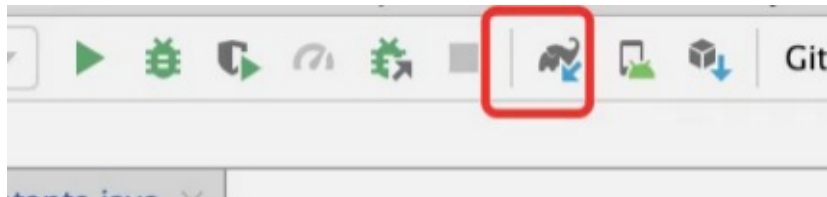
MHSDK必须本地依赖吗？

是的必须本地依赖，目前还没有远程，我们计划做成远程的。

17.找不到某个方法

找不到某个方法。

重新构建，clean。



18.环信不显示自己的头像框

环信不显示自己的头像框

需要升级一下环信的jniLibs.zip包

19.如何取消特效

如何取消特效

每个特效或者美颜的第一个按钮就是取消，这个在UI层代码里面有。

20.水印没有找到

水印没有找到。

原因水印位置超出范围，您可以通过设置setWatermarkRect来手动设置水印位置

21.声网视频通话大小屏切换问题

1.设定自定义采集

```
private void setupLocalVideo(FrameLayout frameLayout, boolean top) {

    if(mTextureCamera == null){
        mTextureCamera = new TextureCamera(this, 640, 480);
        mTextureCamera.setOnCaptureListener(new TextureCamera.OnCaptureListener(
    ) {
        @Override
        public int onTextureBufferAvailable(int textureId, byte[] buffer, i
nt width, int height) {

            int newTexture = textureId;

            try {
                if (mhBeautyManager == null) {
                    mhBeautyManager = new MHBeautyManager(getApplicationCon
text(), true);

                    mhBeautyManager.setBeautyDataModel(BeautyDataModel.getI
nstance());

                    if (beautyViewHolder != null) {
                        beautyViewHolder.setMhBeautyManager(mhBeautyManager)
;
                    }
                }else{

                    if (mCameraFacing == Camera.CameraInfo.CAMERA_FACING_F
RONT){

                        newTexture = mhBeautyManager.render10(textureId, wi
dth, height);

                    }else{
                        newTexture = mhBeautyManager.render11(textureId, wi
dth, height);
                    }
                }
            } catch (Exception e) {
                e.printStackTrace();
            }

            return newTexture;

        }

        @Override
        public void onCapturerStarted() {
            Log.d(TAG, "onCapturerStarted() called");
        }
    });
}
```

```

    }

    @Override
    public void onCapturerStopped() {
        Log.d(TAG, "onCapturerStopped() called");
    }

    @Override
    public void onCameraSwitched(int facing, int orientation) {
        Log.d(TAG, "onCameraSwitched() called with: facing = [" + facing + "], orientation = [" + orientation + "]);

        mCameraFacing = facing;
    }
});
}

mLocalView = new AgoraSurfaceView(VideoLineActivity.this);
mLocalView.init(mTextureCamera.getEglContext());
mLocalView.setBufferType(MediaIO.BufferType.TEXTURE);
mLocalView.setPixelFormat(MediaIO.PixelFormat.TEXTURE_2D);
mLocalView.setZOrderOnTop(top);
mLocalView.setZOrderMediaOverlay(top);
frameLayout.addView(mLocalView);

mRtcEngine.setLocalVideoRenderer(mLocalView);

mRtcEngine.setVideoSource(mTextureCamera);

mTextureCamera.onResume();
}

```

2.切换大小屏的时候销毁资源

```

mTextureCamera.onPause();
mTextureCamera.release();
mTextureCamera = null;
mhBeautyManager.destroy();
mhBeautyManager = null;

```

3. 设置本地渲染

大屏

```
setupLocalVideo(video_chat_big, false);
```

小屏

```
setupLocalVideo(video_chat_small, true);
```