

音视频通话 SDK

虚拟背景



腾讯云

【 版权声明 】

©2013–2024 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【 商标声明 】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或 95716。

文档目录

虚拟背景

Flutter

iOS

Android

uni-app (客户端)

uniapp (小程序)

微信小程序

虚拟背景 Flutter

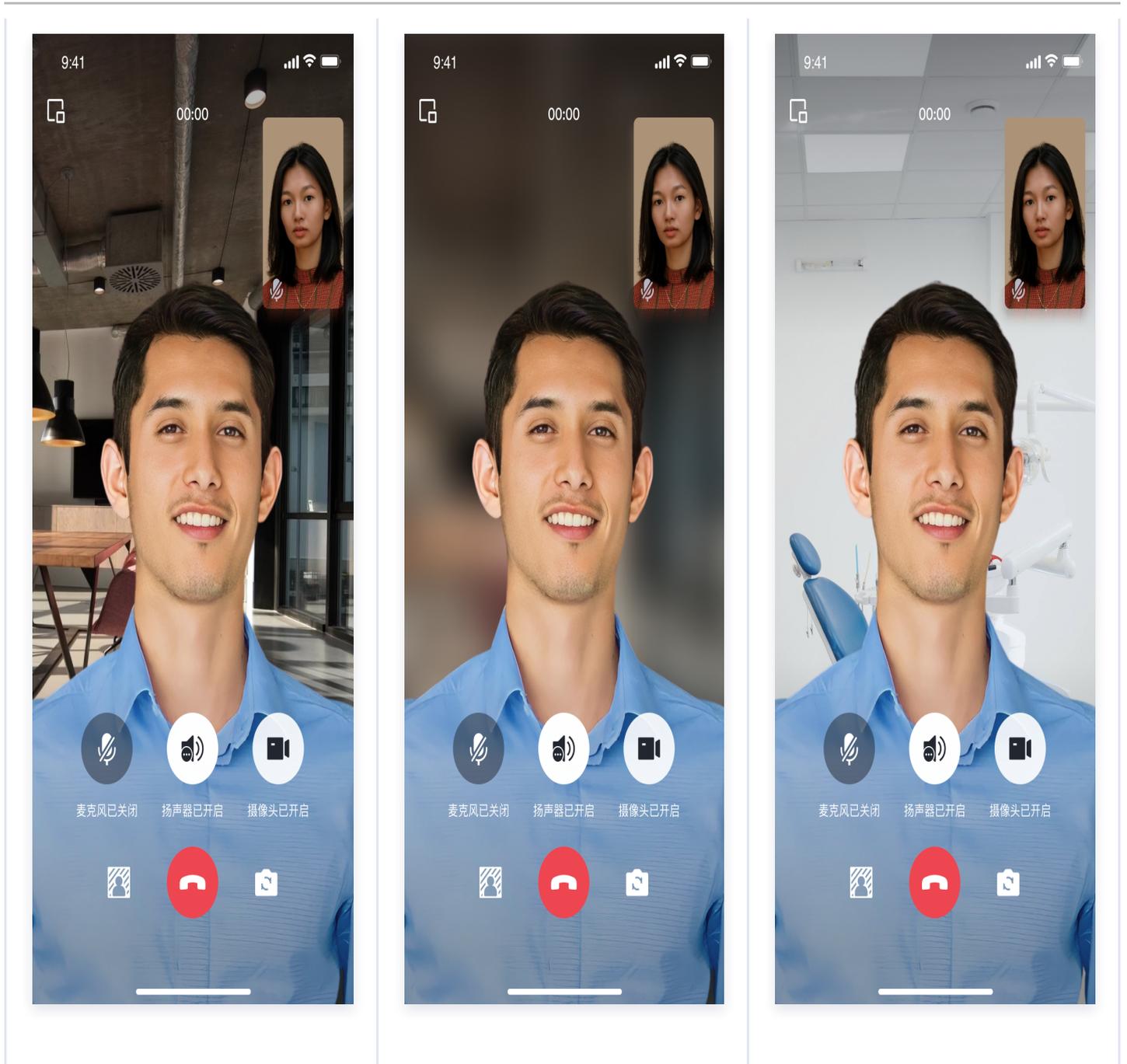
最近更新时间：2024-04-26 17:54:41

TUICallKit 推出了虚拟背景新特性，让用户在视频通话时，可设置模糊背景或图片背景，隐藏真实通话环境，保护隐私，让通话更有趣。接下来，本文将详细介绍在 TUICallKit 组件中如何使用这一特性。

集成效果

TUICallKit 组件接入虚拟背景功能后的显示效果如下：

原始摄像头	模糊背景效果	图片背景效果



准备条件

1. 在使用腾讯云提供的虚拟背景功能前，您需要前往控制台，为应用开通音视频服务，购买 **群组通话版（140万分钟）** 套餐。具体步骤请参见 [开通服务](#)。
2. 指定 LiteAVSDK_Professional SDK 版本。
虚拟背景在 tencent_calls_uikit: 2.3.2 (LiteAVSDK_Professional 11.7.0.12001) 开始支持，不同的 LiteAVSDK_Professional SDK 使用不同的模型文件。

Android

在 `build.gradle` 文件，指定 `TXLiteAVSDK_Professional` 版本，例如指定为 `11.7.12001`，可根据需求和版本迭代修改。

```
api "com.tencent.liteav:LiteAVSDK_Professional:11.7.0.12001"
```

iOS

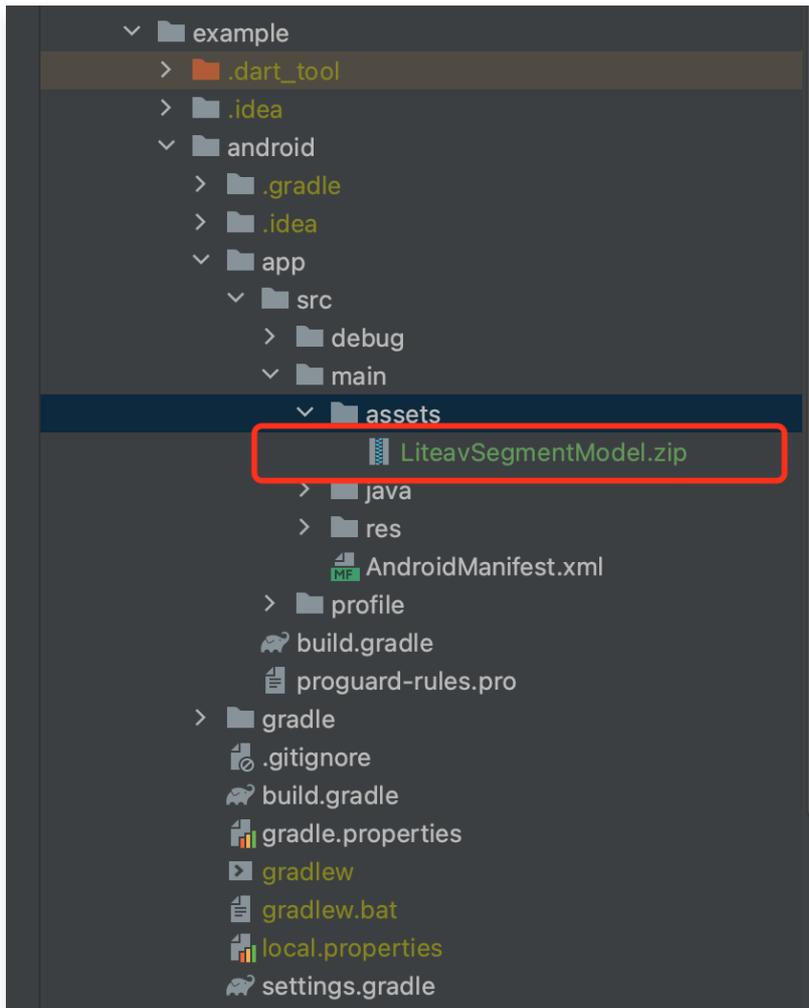
修改您的 `Podfile` 文件中的依赖，指定 `TXLiteAVSDK_Professional` 版本，例如指定为 `11.7.12001`，可根据需求和版本迭代修改。

```
pod 'TXLiteAVSDK_Professional', '11.7.12001'
```

3. 根据 [模型文件与SDK的匹配](#) 情况下载模型文件，并添加到 Android Studio 和 Xcode 工程中。

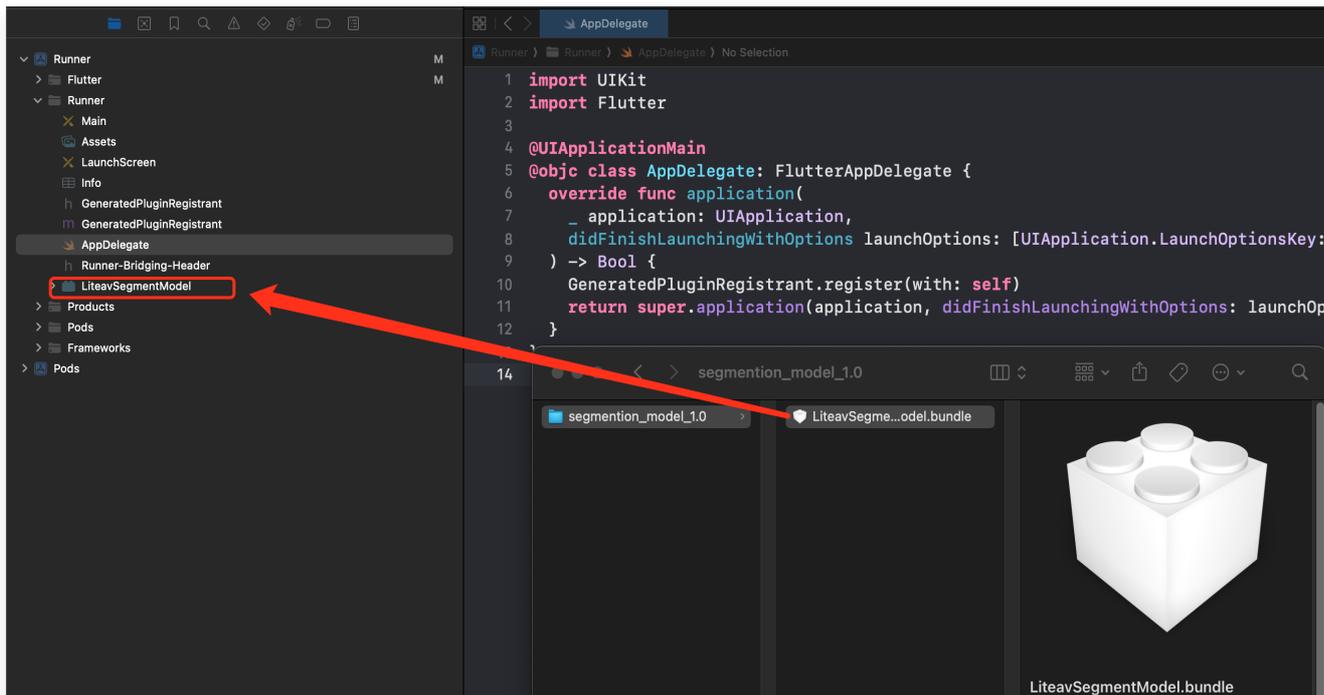
Android

解压后将 `LiteavSegmentModel.zip` 文件拷贝到 Android 工程中的 `assets` 目录下。



iOS

解压后将 `LiteavSegmentModel.bundle` 文件拖拽到您的 Xcode 工程中。



开启模糊背景

TUICallKit 的 UI 方案支持设置模糊背景。通过调用以下接口，您可以在 UI 上显示模糊背景的功能按钮，点击按钮可直接启用模糊背景功能。

```
TUICallKit.instance.enableVirtualBackground(true);
```

设置图片背景（可选）

图片背景需要用户自行实现，将图片文件添加到 Flutter 工程中（需要在 pubspec.yaml 文件中添加资源），并调用接口设置背景图片（目前只支持本地路径的图片，网络图片暂时不支持）。

```
TUICallEngine.instance.setVirtualBackground("***.png", (code, message) { });
```

常见问题

1. 开启模糊背景无反应或有延迟

- 确保已购买音视频通话 **群组通话版（140万分钟）** 套餐，详见 [开通服务](#)。
- 确保已下载模型文件至本地。

如果没有添加模型文件到本地路径中，在开启模糊背景功能的时候，SDK 内部才会去下载模型文件，正常网络情况下，下载耗时1~3s，网络越差耗时越久。

- 检查模型文件与 SDK 是否匹配。

2. 模型文件与SDK的匹配

TUICallKit 是基于 IM SDK 和 TRTC SDK 实现的音视频通话场景，虚拟背景是 TRTC SDK 提供的特色功能。要注意的是，虚拟背景模型文件与 TRTC SDK 的版本需要匹配，否则可能会导致模糊背景功能无法正常使用。下表列出了模型文件与 TRTC SDK 的对应关系：

SDK版本	虚拟背景模型文件下载地址
LiteAVSDK_Professional: 11.7.0.12001	下载

iOS

最近更新时间：2024-05-13 10:16:02

TUICallKit 推出了虚拟背景新特性，让用户在视频通话时，可设置模糊背景或图片背景，隐藏真实通话环境，保护隐私，让通话更有趣。接下来，本文将详细介绍在 TUICallKit 组件中如何使用这一特性。

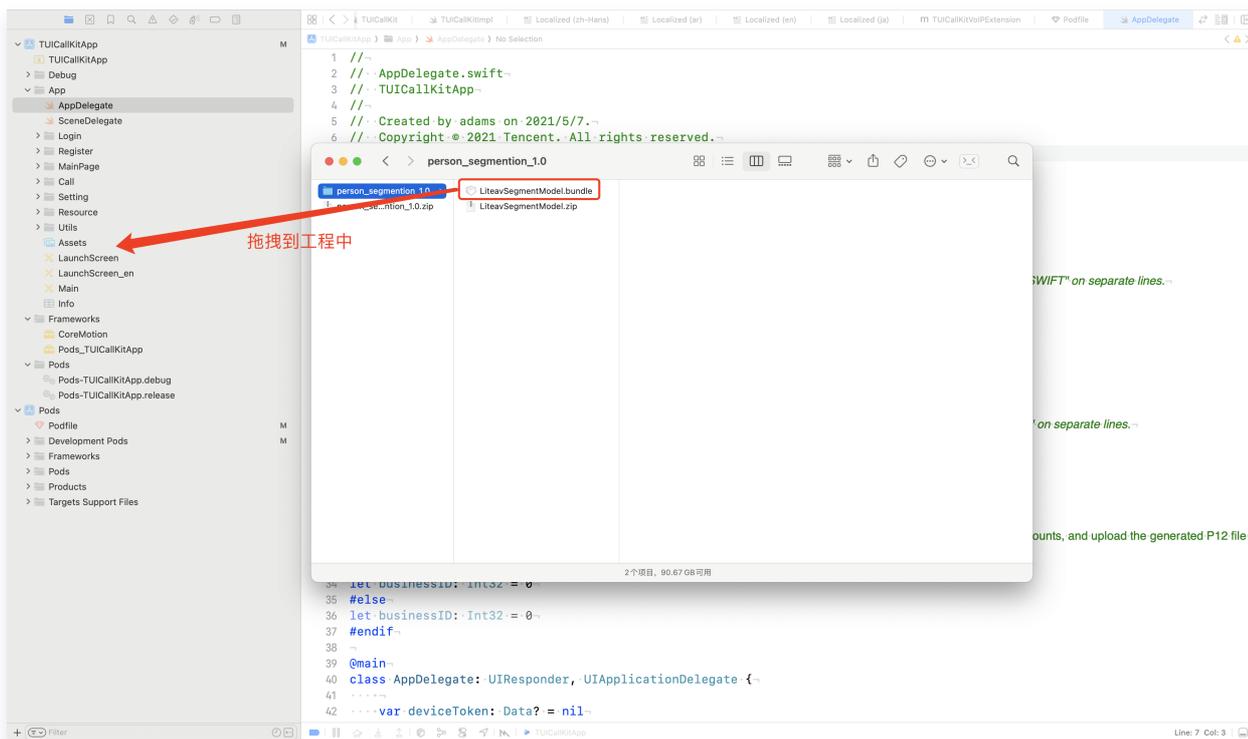
集成效果

TUICallKit 组件接入虚拟背景功能后的显示效果如下：



准备条件

1. 在使用腾讯云提供的虚拟背景功能前，您需要前往控制台，为应用开通音视频服务，并购买 **群组通话版（140万分钟）套餐**，具体步骤请参见 [开通服务](#)。
2. 下载 **虚拟背景模型** 文件，解压后将 **LiteavSegmentModel.bundle** 文件拖拽到您的工程中，并确保它在 **MainBundle** 中。



3. 修改您的 Podfile 文件中的依赖，替换为 **Professional** 版本，并指定 **TXLiteAVSDK_Professional** 版本为 **11.8.15669**。

```
pod 'TUICallKit_Swift/Professional'
pod 'TXLiteAVSDK_Professional', '11.8.15669'
```

⚠ 注意：

TRTC SDK 的版本与模型文件间有对应关系，请确保版本号与模型相匹配，详见下方：[模型文件与SDK的匹配](#)。

开启模糊背景

TUICallKit 的 UI 方案支持设置模糊背景。通过调用以下接口，您可以在 UI 上显示模糊背景的功能按钮，点击按钮可直接启用模糊背景功能。

```
import TUICallKit

TUICallKit.createInstance().enableVirtualBackground(enable: true)
```

设置图片背景（可选）

图片背景需要用户自行实现，将图片保存在本地后，调用以下接口进行设置（目前只支持本地路径的图片，网络图片暂时不支持）。

```
import TUICallEngine

TUICallEngine.createInstance().setVirtualBackground("imagePath") { code, message
in
}
```

常见问题

1. 开启模糊背景无反应或有延迟

- 确保已购买音视频通话 **群组通话版（140万分钟）** 套餐，详见 [开通服务](#)。
- 确保已下载模型文件至本地。

如果没有添加模型文件到本地路径中，在开启模糊背景功能的时候，SDK 内部才会去下载模型文件，正常网络情况下，下载耗时1~3s，网络越差耗时越久。

- 检查模型文件与 SDK 是否匹配。

2. 模型文件与SDK的匹配

TUICallKit 是基于 IM SDK 和 TRTC SDK 实现的音视频通话场景，虚拟背景是 TRTC SDK 提供的特色功能。要注意的是，虚拟背景模型文件与 TRTC SDK 的版本需要匹配，否则可能会导致模糊背景功能无法正常使用。下表列出了模型文件与 TRTC SDK 的对应关系：

SDK 版本	虚拟背景模型文件下载地址
pod 'TXLiteAVSDK_Professional', '11.8.15669'	version_2.0 下载
pod 'TXLiteAVSDK_Professional', '11.7.12001'	version_1.0 下载

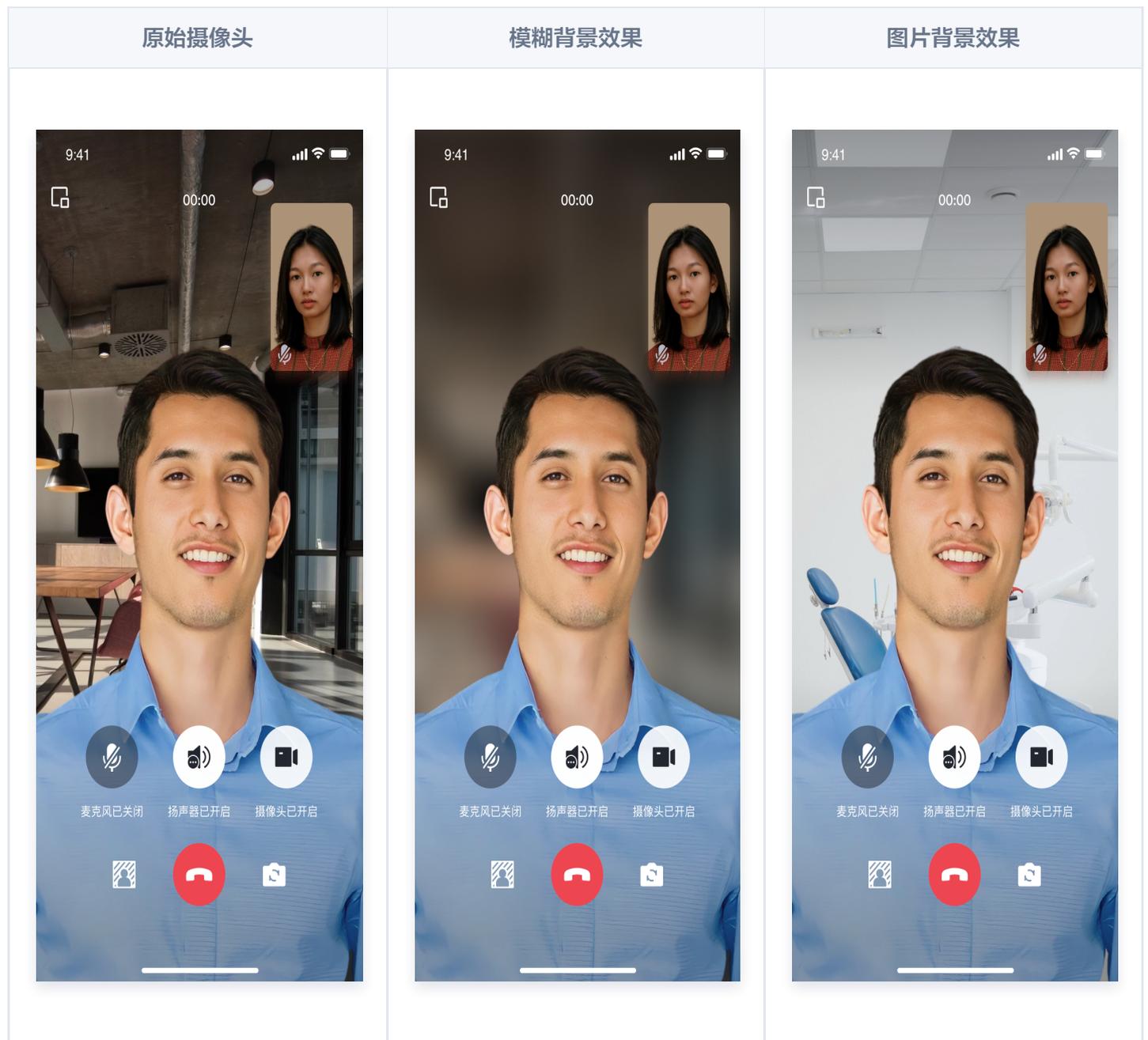
Android

最近更新时间：2024-05-13 11:25:42

TUICallKit 推出了虚拟背景新特性，让用户在视频通话时，可设置模糊背景或图片背景，隐藏真实通话环境，保护隐私，让通话更有趣。接下来，本文将详细介绍在 TUICallKit 组件中如何使用这一特性。

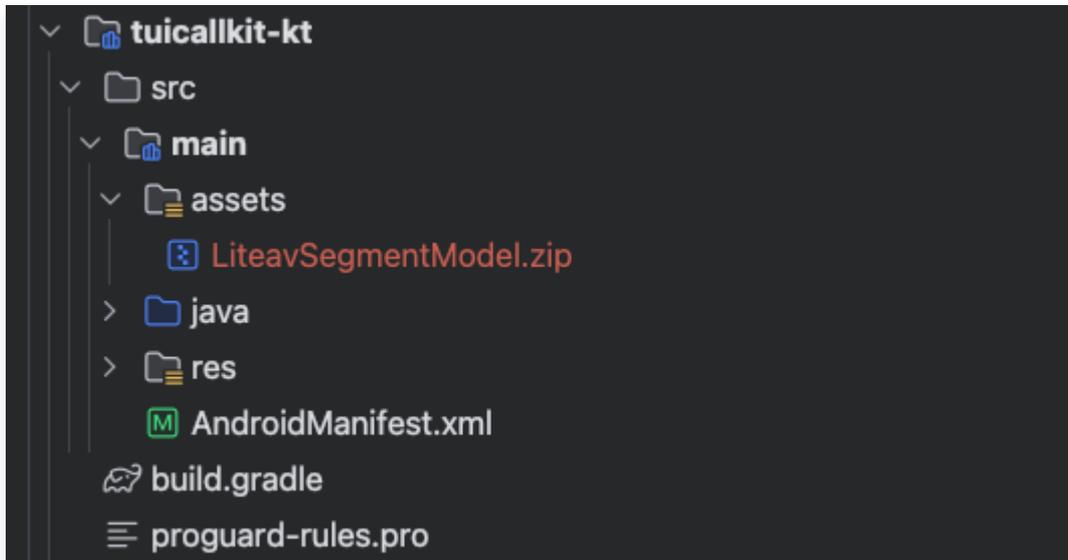
集成效果

TUICallKit 组件接入虚拟背景功能后的显示效果如下：



准备条件

1. 在使用腾讯云提供的虚拟背景功能前，您需要前往控制台，为应用开通音视频服务，购买 **群组通话版（140万分钟）** 套餐。具体步骤请参见 [开通服务](#)。
2. 下载 [虚拟背景模型](#) 文件，解压后，拷贝 `LiteavSegmentModel.zip` 文件到项目的 `assets` 目录下。



3. 在工程的 `tuicallkit-kt` 目录下，找到 `build.gradle` 文件，替换文件中的 TRTC SDK 版本为 Professional 版本。

```
api "com.tencent.liteav:LiteAVSDK_Professional:11.8.0.14176"
```

⚠ 注意：

TRTC SDK 的版本与模型文件间有对应关系，请确保版本号与模型相匹配，详见下方：[模型文件与 SDK 的匹配](#)。

开启模糊背景

TUICallKit 的 UI 方案支持设置模糊背景。通过调用以下接口，您可以在 UI 上显示模糊背景的功能按钮，点击按钮可直接启用模糊背景功能。

```
TUICallKit.createInstance(getApplicationContext()).enableVirtualBackground(true);
```

设置图片背景（可选）

图片背景需要用户自行实现，将图片保存在本地后，调用以下接口进行设置（目前只支持本地路径的图片，网络图片暂时不支持）。

```
TUICallEngine.createInstance(context).setVirtualBackground("imagePath", null)
```

常见问题

1. 开启模糊背景无反应或有延迟

- 确保已购买音视频通话 **群组通话版（140万分钟）** 套餐，详见 [开通服务](#)。
- 确保已下载模型文件至本地。

如果没有添加模型文件到本地路径中，在开启模糊背景功能的时候，SDK 内部才会去下载模型文件，正常网络情况下，下载耗时1~3s，网络越差耗时越久。

- 检查模型文件与 SDK 是否匹配。

2. 模型文件与SDK的匹配

TUICallKit 是基于 IM SDK 和 TRTC SDK 实现的音视频通话场景，虚拟背景是 TRTC SDK 提供的特色功能。虚拟背景模型文件与 TRTC SDK 的版本有匹配关系。若不匹配，可能导致开启模糊背景无效果。请查看下表了解模型文件与 TRTC SDK 的对应关系：

SDK 版本	虚拟背景模型文件下载地址
com.tencent.liteav:LiteAVSDK_Professional:11.7.0.12001	segmentation_1.0
com.tencent.liteav:LiteAVSDK_Professional:11.8.0.14176	segmentation_2.0

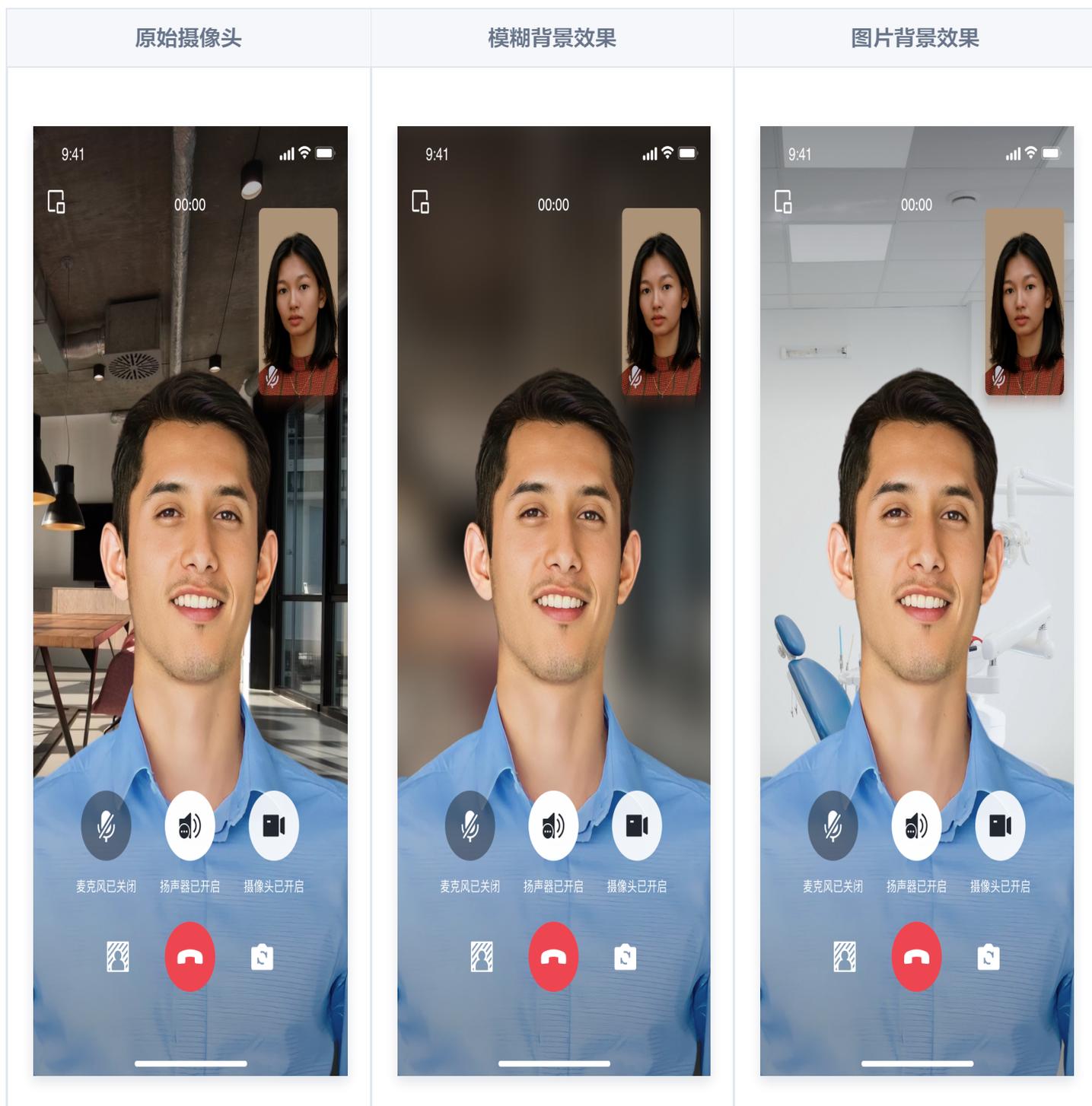
uni-app (客户端)

最近更新时间：2024-04-28 14:32:51

TUICallKit 推出了虚拟背景新特性，让用户在视频通话时，可设置模糊背景或图片背景，隐藏真实通话环境，保护隐私，让通话更有趣。接下来，本文将详细介绍在 TUICallKit 组件中如何使用这一特性。

集成效果

TUICallKit 组件接入虚拟背景功能后的显示效果如下：



准备条件

- 在使用腾讯云提供的虚拟背景功能前，您需要前往控制台，为应用开通音视频服务，购买 **群组通话版（140万分钟）** 套餐。具体步骤请参见 [开通服务](#)。
- HbuilderX 版本要求：HbuilderX 版本 \geq 3.94。

开启模糊背景

TUICallKit 的 UI 方案支持设置模糊背景。通过调用以下接口，您可以在 UI 上显示模糊背景的功能按钮，点击按钮可直接启用模糊背景功能。

ⓘ 说明：

首次使用该功能，可能会存在延迟或无反应，这是因为在开启模糊背景功能的时候，SDK 内部才会去下载模型文件，正常网络情况下，下载耗时1~3s，网络越差耗时越久。

```
const TUICallKit = uni.requireNativePlugin('TencentCloud-TUICallKit');
const enable = true;
TUICallKit.enableVirtualBackground(enable);
```

uniapp（小程序）

最近更新时间：2024-05-13 10:16:01

TUICallKit 推出了虚拟背景新特性，让用户在视频通话时，可设置模糊背景或图片背景，隐藏真实通话环境，保护隐私，让通话更有趣。接下来，本文将详细介绍在 TUICallKit 组件中如何使用这一特性。

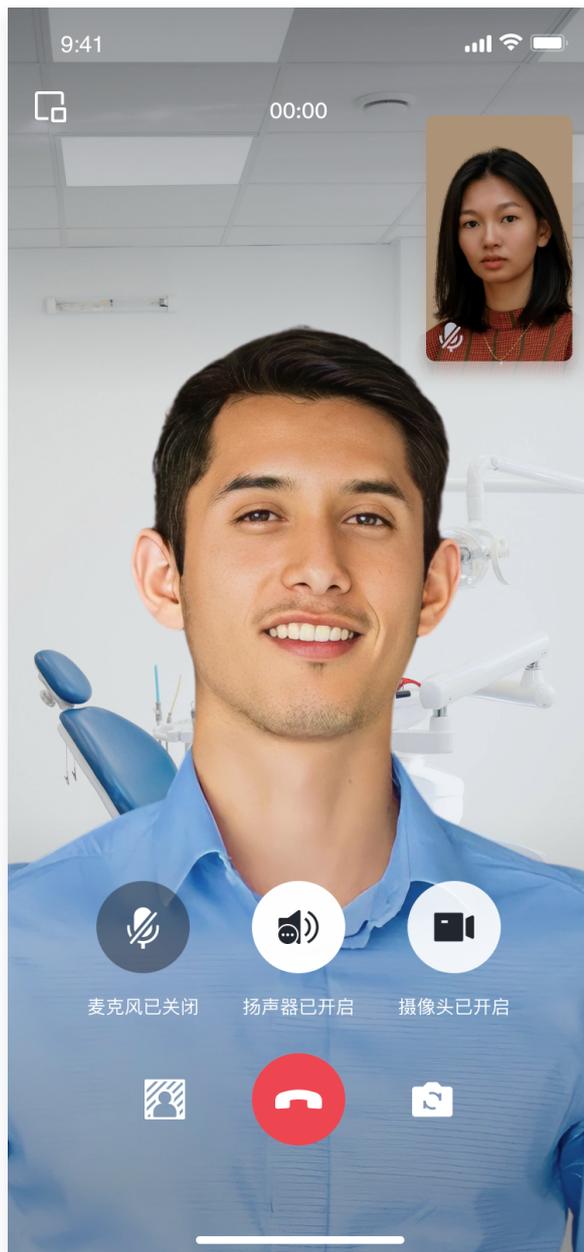
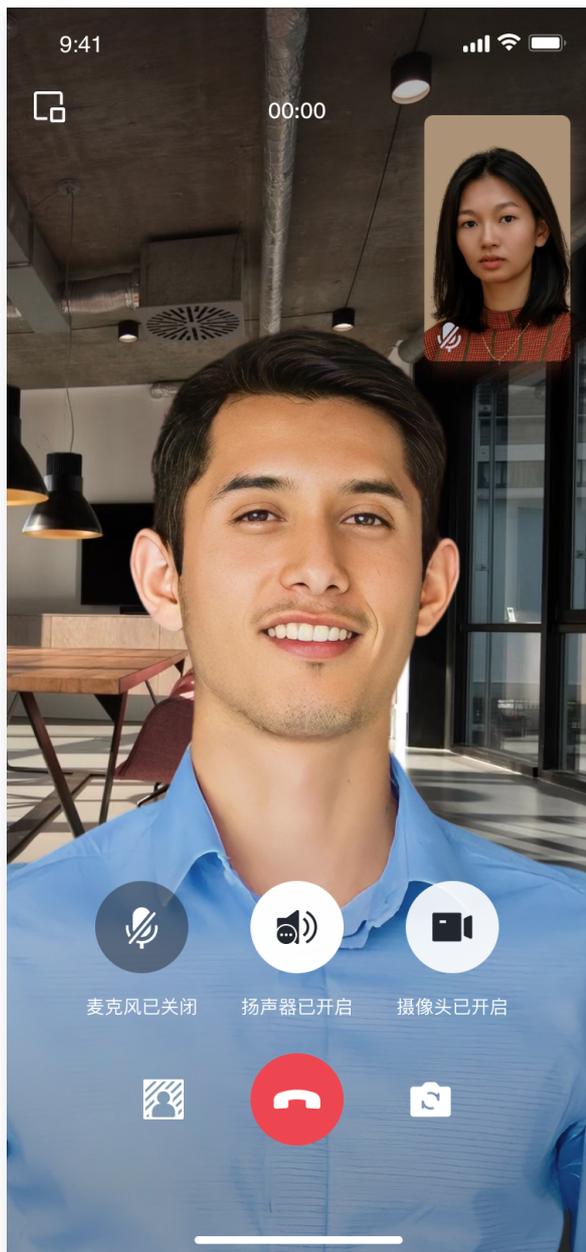
集成效果

TUICallKit 组件接入虚拟背景功能后的显示效果如下：

⚠ 注意：

小程序暂时不支持模糊背景效果。

原始摄像头	图片背景效果



准备条件

TUICallKit 虚拟背景能力基于 [腾讯特效 SDK](#)，请首先申请试用或购买视立方 WEBAR License，详见 [腾讯云视立方控制台](#)。

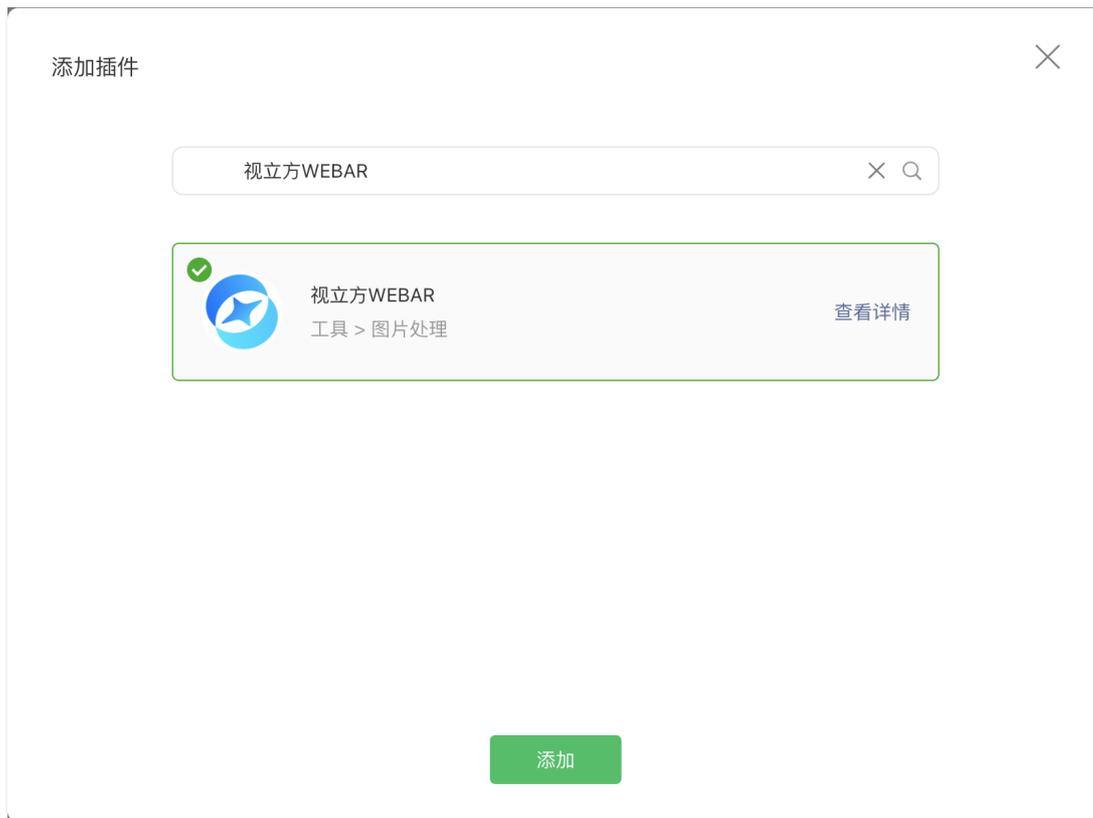
开始使用

警告：

在集成虚拟背景能力之前，请阅读 [TRTC TUICallKit 快速接入\(uniapp-小程序\)](#)，推荐使用分包方式集成 TUICallKit，并完成调试预览相关操作。

步骤一：添加插件

首先要在小程序管理后台的 **设置 > 第三方设置 > 插件管理** 中添加插件。开发者可登录小程序管理后台，搜索 **视立方 WEBAR** 申请添加插件。



步骤二：引用插件

使用组件前需在 uniapp 工程的 `manifest.json` 中声明要使用的插件：

```
{
  "mp-weixin": {
    "plugins": {
      "webarPlugin": {
        "version": "latest",
        "provider": "wx04445cff065100ed"
      }
    }
  }
}
```

步骤三：下载虚拟背景相关依赖

```
npm install tencentcloud-webar-wx crypto-js
```

步骤四：引入美颜特效 SDK

1. 在 TUICallKit/debug 目录下新增 webar-auth.js，实例代码如下：

```
import crypto from "crypto-js";
/** ----- 鉴权配置 ----- */

/**
 * 腾讯云账号 APPID
 *
 * 进入[腾讯云账号中心](https://console.cloud.tencent.com/developer) 即可查看 APPID
 */
const APP_ID = "";
/**
 * Web LicenseKey
 *
 * 登录音视频终端 SDK 控制台的[Web License 管理]
 (https://console.cloud.tencent.com/vcube/web)，创建项目即可获得 LicenseKey
 */
const LICENSE_KEY = "";
/**
 * 计算签名用的密钥 Token
 *
 * 注意：此处仅用于 DEMO 调试，正式环境中请将 Token 保管在服务端，签名方法迁移到服
务端实现，通过接口提供，前端调用拉取签名，参考
 * [签名方法]
 (https://cloud.tencent.com/document/product/616/71370#.E7.AD.BE.E5.90.8D.E6.9
6.B9.E6.B3.95)
 */
const TOKEN = "";
const sha256 = (str) => {
  return crypto.SHA256(str).toString();
};

// 此方法仅测试时使用，为防止 token 泄露，发布时请使用服务端加密，详见[腾讯云-腾讯特
效 SDK 官网文档](https://cloud.tencent.com/document/product/616/71364)
function authFunc() {
  const timestamp = Math.round(new Date().getTime() / 1000);
  const signature = sha256(
    timestamp + TOKEN + APP_ID + timestamp
  ).toUpperCase();
  return { signature, timestamp };
}
export { LICENSE_KEY, APP_ID, authFunc };
```

2. 修改

TUICallKit/src/Components/components/common/Pusher/weChatPusher/weChatPusher.vue 文件

❗ 说明:

imagePath 表示开启虚拟背景显示的图片背景，可通过修改图片路径来修改显示的背景图片，目前仅支持网络图片(不支持本地图片)

```
<template>
<div :class="localClass">
  <div class="stream-box">
    <WebarPusher
      class="stream"
      :custom-effect="true"
      :licenseKey="licenseKey"
      :appId="appid"
      :authFunc="authFunc"
      :background="setBlurBackground ? imagePath : ''"
      :url="pusher.url"
      :mode="pusher.mode"
      :autopush="true"
      :enable-camera="pusher.enableCamera"
      :enable-mic="true"
      :muted="!pusher.enableMic"
      :enable-agc="true"
      :enable-ans="true"
      :enable-ear-monitor="pusher.enableEarMonitor"
      auto-focus="false"
      :zoom="pusher.enableZoom"
      :min-bitrate="pusher.minBitrate"
      :max-bitrate="pusher.maxBitrate"
      :video-width="pusher.videoWidth"
      :video-height="pusher.videoHeight"
      :beauty="pusher.beautyLevel"
      :whiteness="pusher.whitenessLevel"
      :orientation="pusher.videoOrientation"
      :aspect="pusher.videoAspect"
      :device-position="pusher.frontCamera"
      :remote-mirror="pusher.enableRemoteMirror"
      :local-mirror="pusher.localMirror"
      :background-mute="pusher.enableBackgroundMute"
      :audio-quality="pusher.audioQuality"
      :audio-volume-type="pusher.audioVolumeType"
      :audio-reverb-type="pusher.audioReverbType"
```

```

        :waiting-img="pusher.waitingImage"
        :beauty-style="pusher.beautyStyle"
        :filter="pusher.filter"
        @statechange="pusherStateChangeHandler"
        @netstatus="pusherNetStatus"
        @error="pusherErrorHandler"
        @audiovolumenotify="pusherAudioVolumeNotify"
    ></WebarPusher>
</div>
</div>
</template>

<script lang="ts">
export default {
  options: {
    virtualHost: true,
  },
};
</script>

<script lang="ts" setup>
import {
  TUICallKitServer,
  StoreName,
  NAME,
  TUIStore,
} from "../../../../../TUICallService/index";
import { onMounted, ref, onUnmounted } from "../../../../../adapter-vue";
import {
  LICENSE_KEY,
  APP_ID,
  authFunc as auth,
} from "../../../../../debug/webar-auth";
import WebarPusher from "tencentcloud-webar-wx/WebArPusher/WebArPusher.vue";
const pusher = ref(TUIStore.getData(StoreName.CALL, NAME.PUSHER));
const callStatus = ref(TUIStore.getData(StoreName.CALL, NAME.CALL_STATUS));
const licenseKey = ref(LICENSE_KEY);
const appid = ref(APP_ID);
const authFunc = ref(auth);
const setBlurBackground = ref(false);
const imagePath =
  "https://webar-static.tencent-cloud.com/assets/back-new/class.png";

const props = defineProps({

```

```
localClass: {
  type: String,
  required: true,
},
});
const emit = defineEmits(["toggleViewSize"]);

const handlePusherChange = (value) => {
  pusher.value = Object.assign({}, pusher.value, value);
};

const handleCallStatusChange = (value) => {
  callStatus.value = value;
};

const handleBlurBackgroundChange = (value) => {
  setBlurBackground.value = value;
};

const watchOptions = {
  [NAME.PUSHER]: handlePusherChange,
  [NAME.CALL_STATUS]: handleCallStatusChange,
  [NAME.ENABLE_VIRTUAL_BACKGROUND]: handleBlurBackgroundChange,
};

onMounted(() => {
  TUIStore.watch(StoreName.CALL, watchOptions, {
    notifyRangeWhenWatch: NAME.MYSELF,
  });
});

onUnmounted(() => {
  TUIStore.unwatch(StoreName.CALL, watchOptions);
});

function pusherStateChangeHandler(e: any) {
  TUICallKitServer._tuiCallEngine._pusherStateChangeHandler(e);
}

function pusherNetStatus(e: any) {
  TUICallKitServer._tuiCallEngine._pusherNetStatus(e);
}

function pusherErrorHandler(e: any) {
  TUICallKitServer._tuiCallEngine._pusherNetStatus(e);
}
```

```
}  
  
function pusherAudioVolumeNotify(e: any) {  
  TUICallKitServer._tuiCallEngine._pusherAudioVolumeNotify(e);  
}  
</script>  
  
<style lang="scss" src="./style/index.scss" scoped>  
</style>
```

步骤五：开启模糊背景

通过调用以下接口，您可以在 UI 上显示虚拟背景的功能按钮。

ⓘ 说明：WeChat ≥ v3.2.5 支持。

```
TUICallKitServer.enableVirtualBackground(true)
```

微信小程序

最近更新时间：2024-05-13 10:16:01

TUICallKit 推出了虚拟背景新特性，让用户在视频通话时，可设置模糊背景或图片背景，隐藏真实通话环境，保护隐私，让通话更有趣。接下来，本文将详细介绍在 TUICallKit 组件中如何使用这一特性。

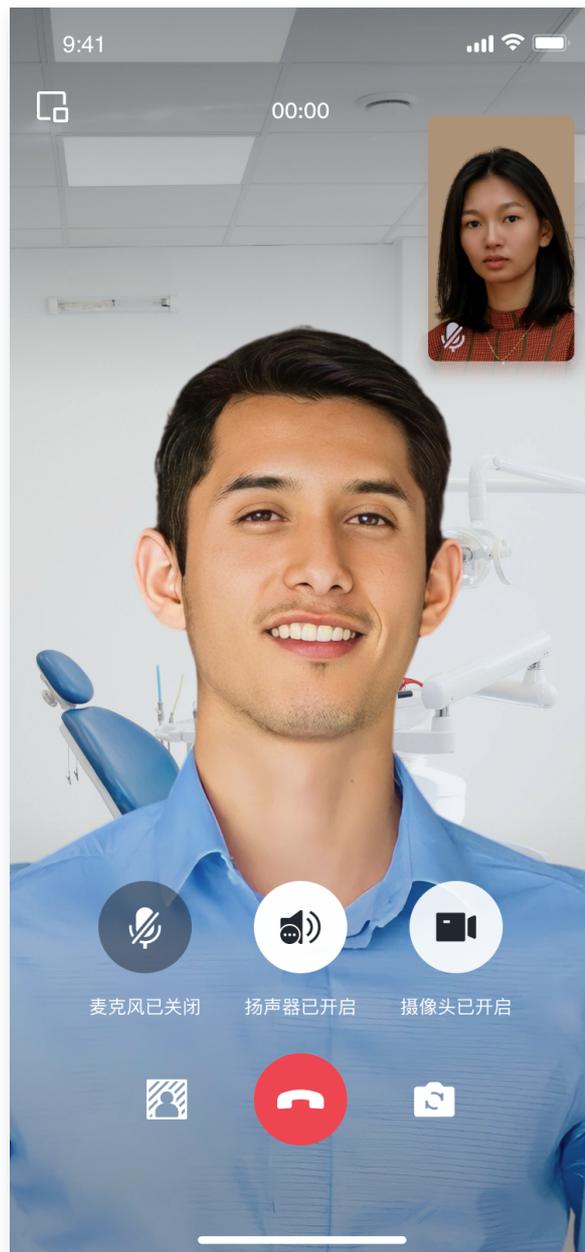
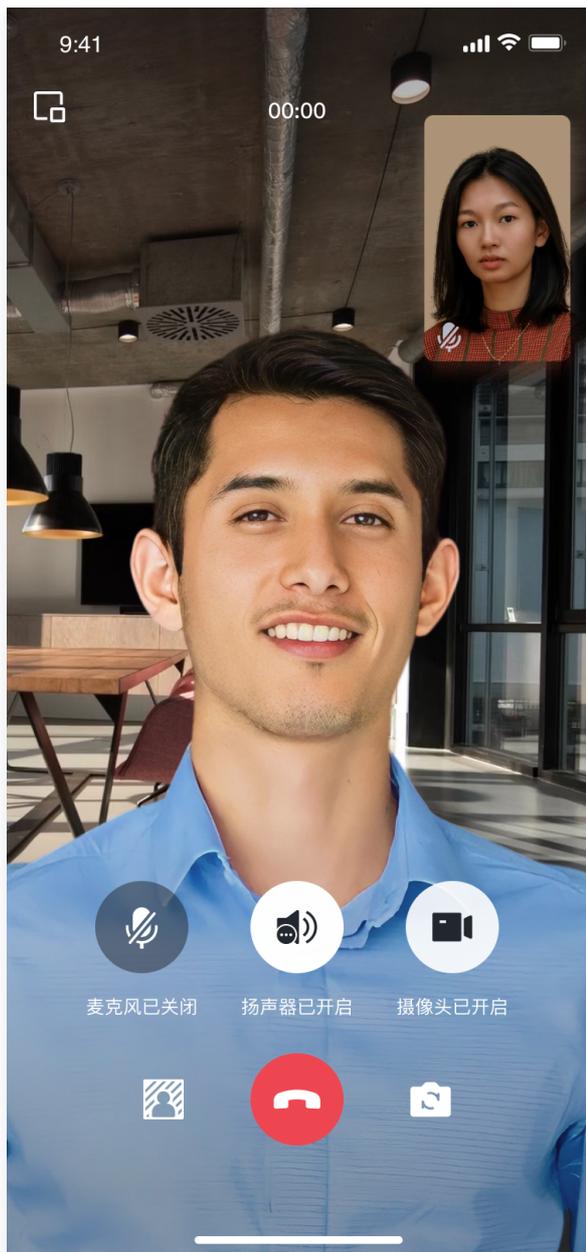
集成效果

TUICallKit 组件接入虚拟背景功能后的显示效果如下：

⚠ 注意：

小程序暂时**不支持**模糊背景效果。

原始摄像头	图片背景效果



准备条件

TUICallKit 虚拟背景能力基于 [腾讯特效 SDK](#)，请首先申请试用或购买视立方 WEBAR License，详见 [腾讯云视立方控制台](#)。

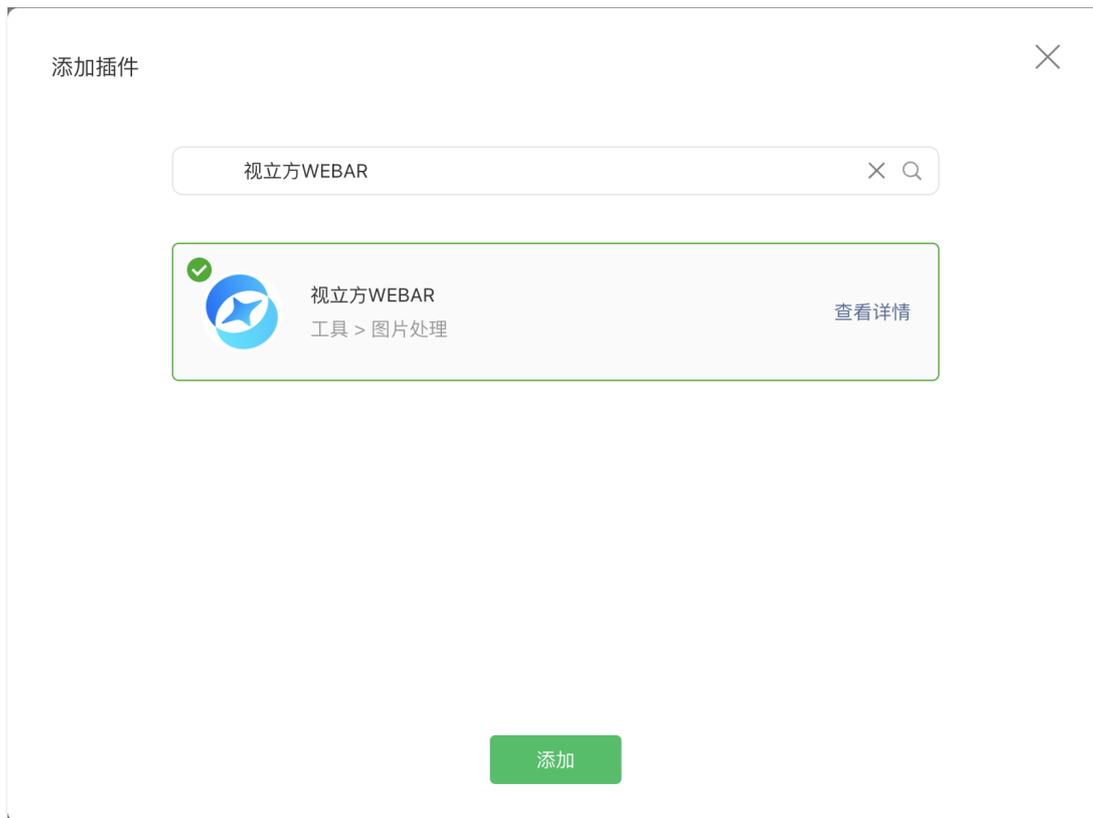
开始使用

警告：

在集成虚拟背景能力之前，请阅读 [TRTC TUICallKit 快速接入\(原生小程序\)](#)，推荐使用分包方式集成 TUICallKit，并完成调试预览相关操作。

步骤一：添加插件

首先要在小程序管理后台的[设置 > 第三方设置 > 插件管理](#)中添加插件。开发者可登录小程序管理后台，搜索 [视立方 WEBAR](#) 申请添加插件。



步骤二：引用插件

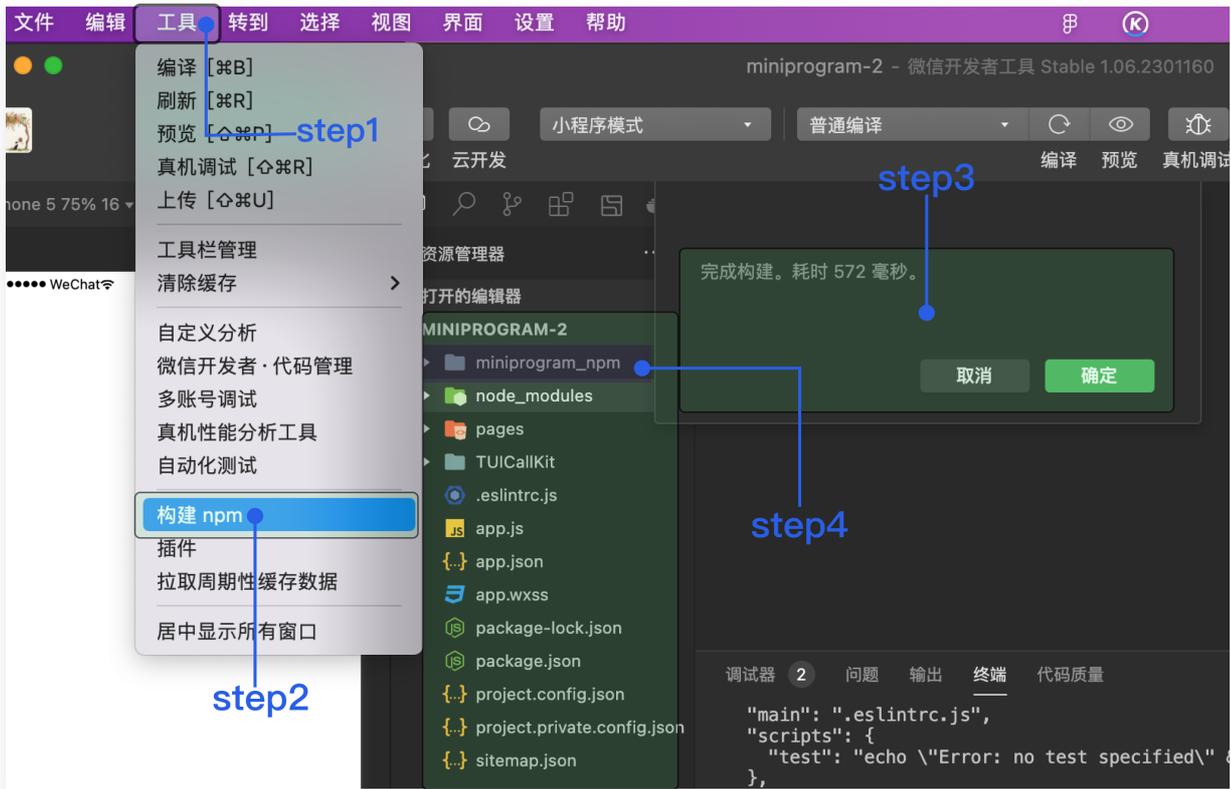
1. 使用组件前需在小程序工程的app.json中声明要使用的插件：

```
"plugins": {  
  "webarPlugin": {  
    "version": "latest",  
    "provider": "wx04445cff065100ed"  
  }  
}
```

2. 在小程序目录安装 npm 包: tencentcloud-webar-wx

```
npm install tencentcloud-webar-wx crypto-js
```

3. 构建 npm，打开微信开发者工具单击工具 > 构建 npm，新增 miniprogram_npm 目录。目录如下：



步骤三：引入美颜特效 SDK

1. 在 TUICallKit/debug 目录下新增 webar-auth.js，实例代码如下：

```
import crypto from "crypto-js";
/** ----- 鉴权配置 ----- */

/**
 * 腾讯云账号 APPID
 *
 * 进入[腾讯云账号中心](https://console.cloud.tencent.com/developer) 即可查看 APPID
 */
const APP_ID = "";
/**
 * Web LicenseKey
 *
 * 登录音视频终端 SDK 控制台的[Web License 管理]
 (https://console.cloud.tencent.com/vcube/web)，创建项目即可获得 LicenseKey
 */
const LICENSE_KEY = "";
/**
 * 计算签名用的密钥 Token
 *
 * 注意：此处仅用于 DEMO 调试，正式环境中请将 Token 保管在服务端，签名方法迁移到服
务端实现，通过接口提供，前端调用拉取签名，参考

```

```
* [签名方法]
(https://cloud.tencent.com/document/product/616/71370#.E7.AD.BE.E5.90.8D.E6.9
6.B9.E6.B3.95)
*/
const TOKEN = "";
const sha256 = (str) => {
  return crypto.SHA256(str).toString();
};

// 此方法仅测试时使用，为防止 token 泄露，发布时请使用服务端加密，详见[腾讯云-腾讯特
效 SDK 官网文档](https://cloud.tencent.com/document/product/616/71364)
function authFunc() {
  const timestamp = Math.round(new Date().getTime() / 1000);
  const signature = sha256(
    timestamp + TOKEN + APP_ID + timestamp
  ).toUpperCase();
  return { signature, timestamp };
}
export { LICENSE_KEY, APP_ID, authFunc };
```

2. 修改 `TUICallKit/component/SingleCall/SingleCall.json` 文件，引入 `webar-live-pusher` 组件。

```
{
  "usingComponents": {
    "webar-live-pusher": "tencentcloud-webar-wx/webar-live-pusher"
  }
}
```

3. 修改 `TUICallKit/component/SingleCall/SingleCall.wxml` 文件，将 `live-pusher` 替换为 `webar-live-pusher`。

⚠ 注意：

组件 `webar-live-pusher` 的使用方法和微信原生标签 `live-pusher` 的方法一致，更多属性可参考微信小程序标签 `live-pusher` 的文档说明，美颜特效相关参数请参见 [组件属性说明](#)。

```
<webar-live-pusher
  custom-effect
  licenseKey="{{licenseKey}}"
  appId="{{appId}}"
  authFunc="{{authFunc}}"
  background="{{isVirtualBackground ? imagePath : ''}}"
  // 其他 props 同 live-pusher，此处省略
```

```
/>
```

4. 修改 TUICallKit/component/SingleCall/SingleCall.js 文件

❗ 说明:

backgroundUrl 表示开启虚拟背景显示的图片背景，可通过修改图片路径来修改显示的背景图片，目前仅支持网络图片(不支持本地图片)

```
import { APP_ID, LICENSE_KEY, authFunc } from '../debug/webar-auth';

Component({
  data: {
    // 其他的 data 不变，此处忽略
    appId: APP_ID,
    licenseKey: LICENSE_KEY,
    authFunc,
    imagePath: 'https://webar-static.tencent-cloud.com/assets/back-new/class.png'
  },
});
```

步骤四：开启模糊背景

通过调用以下接口，您可以在 UI 上显示虚拟背景的功能按钮。

❗ 说明：3.0.0 > WeChat ≥ v2.1.5 支持。

```
TUICallKitServer.enableVirtualBackground(true)
```