User Generated Short Video SDK FAQs

Copyright Notice

Trademark Notice

Service Notice

This document provides an overview of the as-is details of Tencent Cloud's products and services in their entirety or part. The descriptions of certain products and services may be subject to adjustments from time to time.

The commercial contract concluded by you and Tencent Cloud will provide the specific types of Tencent Cloud products and services you purchase and the service standards. Unless otherwise agreed upon by both parties, Tencent Cloud does not make any explicit or implied commitments or warranties regarding the content of this document.

Contact Us

We are committed to providing personalized pre-sales consultation and technical after-sale support. Don't hesitate to contact us at 4009100100 or 95716 for any inquiries or concerns.

Contents

FAQs
- Supported Features FAQs
- Common Problems of License
- Uploading Common Problems
- Android FAQs
- iOS FAQs

FAQs
Supported Features FAQs

Last updated：2025-03-17 16:51:51

According to customer feedback, we have elaborately prepared answers to some common questions, which will be continuously updated subsequently, hoping to help you resolve queries about product features.

Version Related

SDK Supported System Editions?

iOS: Mobile phone system 8.0 or higher, Xcode 9 or higher version, OS X 10.10 or higher version.
Android: Mobile phone system 4.0.3 (API 15) or higher. Note that only systems with Android 4.3 (API 18) or higher can enable hardware encoding.

Whether Mini Program Is Supported?

It is not supported.

Whether X86 Is Supported?

- iOS: Support.
- Android: Not supported.

Existence of Older Versions of the SDK (For Example, 3.9)

It does not exist. Recommend you use the latest version.

Whether H5 Integration Is Supported?

It is not supported.

Whether Flutter Version Integration Is Supported?

It is not supported.

Other Related

How Does the SDK Obtain Video Information (Such As Width and Height)?

TXVideoEditer > TXVideoInfoReader > getVideoInfo

What Is the Key For SDK Decompression?

The simplified edition and basic version SDKs do not have a decompression key.

Is There a Limit On the Number of Viewers For the SDK?

There is no limit on the number of viewers.

Whether a Video License Is Required For Short Video Integration?

Not required.

Can Customers Use the BGM in Demos and Short Videos?

Not allowed. The BGM in the Demo and short video is only used for feature demonstration. If you use it in a commercial App, there will be legal risks.

Does the SDK Have a Prop Editor?

It is not supported.

Is the SDK Single-Thread or Multi-Thread Upload?

User Generated Short Video SDK (UGSV SDK) supports **multi-thread upload**.

Does the SDK Support UnionPay Mini Programs?

The User Generated Short Video SDK (UGSV SDK) only supports Android and iOS platforms and cannot be used for mini programs.

Photo and Recording Related

Whether Photo-Taking Function Is Supported?

Supported. You can call the photo-taking interface to take a photo inside the recording API (TXUGCRecord > snapshot).

Whether SDK Recording Supports Anti-Shake?

It is not supported.

Whether Video Recording and Editing Support Converting to GIF?

Not supported. You can obtain the video sample image list based on our SDK API (TXVideoEditer > TXVideoInfoReader > getSampleImages), and then generate GIFs on your own.

Is It Possible to Try the Photo-Taking Function On the Basis of the Simplified Edition?

Supports photo components that can be integrated and used in the short video SDK, which can be integrated into the short video SDK for use.

Whether Autofocus and Manual Focus Can Be Used Together?

Short video recording does not support using autofocus and manual focus together. Autofocus is the default.

Preview of Cropping and Splicing Related

Whether Video Editing Is Supported?

The short video SDK for video editing includes features such as video trimming, time-based special effects (slow motion, reverse playback, and repetition), special effect filters (dynamic light wave, dark phantom, out-of-body experience, and picture splitting), filter styles (aesthetic, rosy, blue tones, etc.), music mixing, animated stickers, static stickers, and bubble subtitles.
For detailed information, you can refer to the following documents:

- Video editing (iOS)
- Video editing (Android)

Whether Video Stitching Is Supported?

Short video SDK versions above the basic version support video stitching. For detailed information, please see:

- Video splicing (iOS)
- Video stitching (Android)

Whether Video Editing Supports Inserting Images At Any Position in the Video?

It is not supported.

Whether Background Music Functionality Is Supported?

Support. It is available to select built-in audio files or MP3s from the user's phone locally as background audio. Simultaneously support the trimming of background music and setting the volume size.

Why Is Setting Background Music Unavailable?

- The simplified edition does not support setting the background music file feature. It is advisable to view the current integration version.
- If your current SDK is upgraded based on the simplified edition, please check whether the current SDK package is still the simplified edition.
- Please check if the current License has the same package name as the simplified edition License. If they are the same, please submit a ticket and contact contacts to handle it.

Using the Video Editing Feature to Insert Music, What Sources of Music Are Available For Use?

For short videos, you need to add music in code. Currently, there is no music library provided for you to use. You can select the music path on your own. For more details, please refer to Add Background Music.

SDK'S Editing Feature, Can It Be Used in WeChat Mini Program?

Integration in WeChat mini program is not supported.

Upload and Playback Related

How to Use a Player?

The main class of the short video player is SuperPlayerView, and it can play video once created. For usage details, please refer to:.

- iOS player usage
- Android player usage

Can the Generated Video By SDK Be Directly Uploaded to a Non-Tencent Cloud Platform (For Example, WeChat Official Account)?

The generated video by the SDK can be uploaded to Tencent Cloud's VOD server. There is also corresponding source code for reference in the Demo. If you want to upload it to other platforms, please check the upload requirements of other platforms on your own.

Whether the Playback Interface of SDK Supports Setting Rounded Corners?

Not supported temporarily. Currently, the player UI requires business side to perform UI design.

Beauty-Related

Whether Enlarging Eyes and Slimming Face Is Supported?

Not supported, but additional capabilities such as intelligent beauty feature, special effect filter, animated sticker, AI matting, and makeup can be integrated through Tencent Effect SDK. For details, please refer to Tencent Effect SDK.

Is the Demo Experience Only With the Basic Beauty Filter Functionality?

In the Demo experience, only basic beauty filter functionality is supported. If you need to experience other special effects such as big eyes, additional purchase is required.

Whether Third-Party Beauty Feature Can Be Accessed and Used?

The short video SDK does not support integrating third-party beauty features.
You can additionally integrate capabilities such as smart beauty, special effects filters, animated stickers, AI matting, and makeup through the Tencent Effect SDK. For details, refer to Tencent Effect SDK.

Whether Background Wall Is Supported?

The short video SDK supports advanced beauty functions, among which the green screen functionality can be customized with a picture background. For details, see Enlarging Eyes and Slimming Face and Widgets (Android).

Supports the Use of Integrating Advanced Beauty Function in Demo Experience?

The Demo is for trial only. If you need to integrate advanced beauty features on this basis, you need to activate this feature first.

Is the Beauty Effect Material in the Short Video SDK Required to Be Your Own Design?

You don't need to design by yourself. After completing the purchase of beauty effect materials, you can select the materials you need to use from the beauty effect material library we provide. If you need to customize, you can contact business communication to submit a request.

Advanced Functions and Special Effects Related

Whether Customization of Animated Stickers Is Supported? How to Implement?

Support. Use through SDK source code. For details, see Stickers and Subtitles (iOS).

Supports Filters?

Short video SDK supports filter effect and video editing functionality. For details, see TikTok-like special effects, SDK integration (XCode).

Whether Noise Reduction Is Supported?

Currently, the short video SDK does not support noise reduction functionality.

Whether Video Transition Functionality Is Supported?

Video transition functionality is not supported yet. Currently, only image transitions are supported. For related documents, see Image Transition Effect.

Whether Applying a Filter to an Image Is Supported?

Processing of images is not supported. Only video processing is supported. For related documents, see TikTok-like special effects (iOS).

How to Integrate AI Pornography Detection?

For integrating short video SDK with AI identification, please refer to AI Identification Access Example.

How to Obtain the Playback Volume Data of a Certain Short Video?

Through following steps:

- Enter Video on Demand Console > **Data Analysis** to query. For related documents, see Data Analysis.
- Obtain through querying the video-on-demand CDN usage data interface.

Common Problems of License

Last updated: 2025-03-17 16:52:17

About License Version Issue?

The User Generated Short Video SDK (UGSV SDK) has a simplified edition and a basic version. Starting from version 4.5, a License is required. The basic version requires the License for short videos (TXUgcSDK.licence). Place the License in the project directory and change its name to the corresponding one just.
Starting from version 4.9, the usage of License has changed. You can choose whether to package the License into the project. When in use, you need to call the `setLicenseURL:` and `key:` APIs to set the URL and key of the License.

> **Note:**
> The SDK versions from 4.5 to 4.7 do not support automatic renewal of License. Automatic renewal is only supported starting from version 4.9. The SDK version 4.9 is compatible with previous Licences (the url and key cannot be null; you can pass any string), but the new License cannot be used on the SDK versions before 4.9.

Is It Possible to Postpone a Trial License Upon Expiration?

You can apply for a trial License for free (the free trial period is 14 days, renewable once, for a total of 28 days) for experience testing. Please renew as soon as possible upon expiration purchase an official License.

> **Note:**
> If you apply for test renewal during the trial period, the expiration time of the renewal will be based on the moment of applying for testing; if you apply for test renewal after the trial period ends, the expiration time of the renewal will be based on the moment of applying for test renewal.
> - When the start time of the trial application is `May 25, 2022, 11:34:55`, the expiration time after 14 days is `June 9, 2022, 00:00:00`.
> - When you renew for free once, if you apply for renewal within 14 days of the trial period, the expiration date is `2022-06-23 00:00:00`; if you apply for renewal after the 14-day trial period, the application time is `2022-07-03 22:26:20`, then the expiration date of the renewal is `2022-07-18 00:00:00`.

Can the Trial License Change the PackageName of Android and the BundleID of iOS?

The trial License supports changes. On the VOD console, select the trial License information in the upper-right corner and click **Edit** to proceed with the modification.

Can an Official License Change the PackageName of Android and the BundleID of iOS?

The official License cannot modify the Package Name and Bundle ID. Please verify whether the Package Name is occupied in the app store before adding the official License. Modification and replacement are not supported after submission.

Can a License Support Multiple Apps Simultaneously?

A License can only correspond to one PackageName and BundleID and does not support multiple Apps.

How to Confirm the Binding Relationship (Android'S PackageName and iOS'S BundleID) For a License?

When filling in, users need to confirm the Bundle ID corresponding to the iOS officially launched on the App Store and the Package Name of the Android officially launched in the app market.

How to Resolve the Issue of "License not exist" When Renewing a License?

You can log in to **VOD console** > **License Management** > SDK License and troubleshoot as follows:

1. Please confirm whether to perform License binding renewal on the **administrator** page.

2. If you are performing operations on the **non-administrator** page, contact **administrator** to help you perform License modification operations.

How to Resolve That the License Cannot Be Added?

- Confirm whether there is a boundable resource package. Enter **VOD console** > Resource Package, and confirm whether there is one of the boundable VOD traffic resource packages of 10T, 50T, 200T or 1PB under your account.
- View whether the binding page is an **administrator** page. Please select the administrator page to proceed to bind.

How to Troubleshoot License Verification Failure?

You can refer to the following points for troubleshooting:

- Confirm whether your License is within the valid period.
- Confirm whether the Package Name in the License information matches the Package Name in the project.
- Confirm whether the LicenseUrl protocol in the License is HTTPS.

> **Note:**
> If the above methods cannot solve your problem, it is recommended to **uninstall and reinstall the application** or submit a ticket to resolve it.

No bundleid, Is It Impossible to Use License On the Android Side?

The bundleid is similar to the package name on the Android side. If you do not integrate the iOS side, you can fill in anything. It can be left non-use.

How to Upgrade From a Simplified Edition of the Short Video SDK to a Basic Version License?

Purchase on-demand traffic packages of 50TB, 200TB or 1PB to obtain usage rights of Basic Version License.

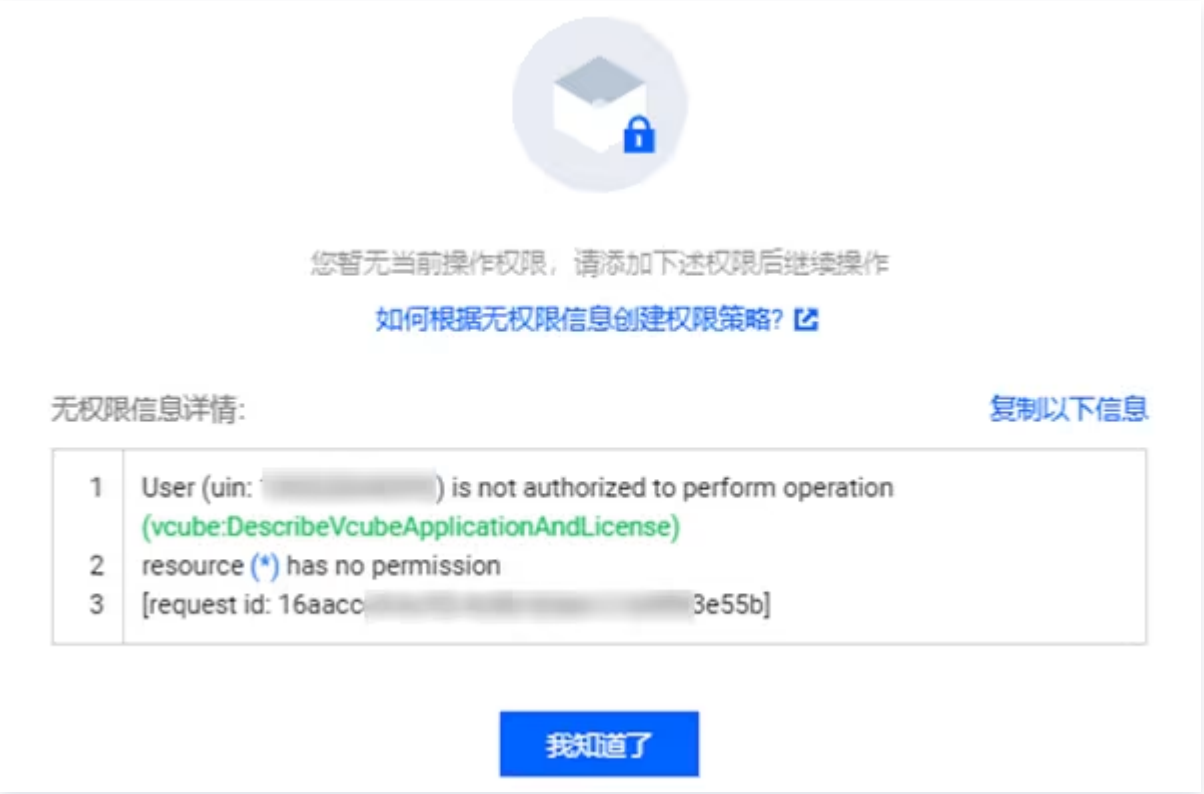
> **Note:**
> Currently, only the upgrade from the simplified edition to the basic edition of the Short Video License is supported. The upgraded License is the License specification given by the corresponding resource package.

Can the Short Video SDK License Purchased By an Individual Be Used For a Business?

The short video SDK currently only supports purchase and use in the account. There are no limitations on individual or enterprise identity verification.

Why Has My Subaccount Already Been Authorized All Privileges of Live Stream and On-Demand Video, but Still Cannot Access the Related Interface of the License Console?

Screenshot of the Issue:



Issue Analysis:

The new SDK License has been upgraded and updated this time (for details, see Old and New License Descriptions). The root account needs to reauthorize the policy independently for the sub-account before accessing the License console page.

- If you only need to provide the sub-account with the permission to query the License, authorize the QcloudVCUBEReadOnlyAccess policy.
- If you need to provide the sub-account with all License operation permissions, authorize the QcloudVCUBEFullAccess policy.

For the association guide to associate policies for users/user groups to authorize related operation permissions, please refer to Policy Authorization Management.

> **Note:**
> All feature operations on the License interface are independent of the Cloud Streaming Services (CSS) and VOD policies. That is, the original QcloudVODFullAccess and QcloudLIVEFullAccess policies no longer include relevant APIs for License. Authorization needs to be performed separately as per the above instructions.

Why Unable to Receive Relevant Message Notification of License Expiration?

The Short Video License is used for the authorization and unlocking of the management feature module of the Audio and Video Terminal SDK (Tencent Cloud RT-Cube). You can subscribe to the Audio and Video Terminal SDK in Message Subscription and configure **Message Center/Email/Short Message Service/WeChat/WeCom** and other message reception channels to receive expiry reminders for the official license. The Short Video official license will send an expiry reminder to you once when the expiration date is 30 days, 15 days, 7 days, and 1 day away from the current time respectively, prompting you to renew on time to avoid impacting normal business operations.

Why Does the End-User App Prompt "Invalid License" After I Just Renewed the License and the License Has Been Displayed As Within Its Validity Period in the Console?

Possibly because of the impact of local cache, the License failed to auto-update. Please follow these steps:

1. Ensure that the device is connected to the network.
2. Restart the application.
3. Wait for the license to update automatically.

Meanwhile, it is recommended that you renew the License in advance before the License expires to avoid affecting your business.

Uploading Common Problems

Last updated: 2026-03-20 15:01:20

What Is Client Video Upload?

Client video upload refers to App users directly uploading local videos to VOD.

Video Upload Function TXUGCPublish Not Found?

The video upload module has been made independent from the SDK and open sourced in the Demo. Users are advised to integrate short video upload themselves. The steps are as follows:

1. Download Demo.
2. Copy the uploaded jar package under the `app\libs\upload` directory to the `..\app\libs\upload` directory of your project.
3. Copy the source code directory `Demo\app\src\main\java\com\tencent\liteav\demo\videoupload` to your own project directory and modify the package name in the source code.
4. In the build.gradle under the project App directory, add the code for referencing the jar package.

```
dependencies {
compile fileTree(include: ['*.jar'], dir: 'libs/upload')
}
```

5. Configure the App's permission in `AndroidManifest.xml`.

```
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission
android:name="android.permission.READ_EXTERNAL_STORAGE"/>
```

What Should I Do If Upload Fails With the Internal Error 1000?

Please check whether the VOD service is enabled.

Short Video Upload Parameter Class Error?

Please check whether the video file URL and image address are correct, and whether the corresponding files can be found in the path.

Short Video Upload Signature Error?

Before initiating uploading, the client needs to request an upload signature from the application server. If the application server allows the client to upload, it will generate an upload signature for the client according to signature rules. The client must carry this signature to let VOD verify whether the client upload is granted.
The steps for client upload signature are as follows:

1. Obtain API key.
2. Concatenate plaintext strings.
3. Convert plaintext strings to final signature.
4. After the service is set up, developers can verify the correctness of the signature through the tools provided by VOD:

- Signature generation tool: Quickly generate a signature based on parameters and a key.
- Signature verification tool: Parse the signature to obtain the various parameters used when generating the signature.

For more details, see client upload signature.

Is There a Limit to the Duration and Sizes of Maximum Allowed Upload During Video Upload?

The User Generated Short Video SDK (UGSV SDK) has no limit on the duration and sizes of uploaded videos.

Is It Possible to Upload an Image?

The function of uploading pictures separately is not currently supported, but a cover image can be attached when uploading a video. For related notes, please refer to Video Upload.

Android FAQs

Last updated: 2025-03-17 16:52:55

Functionality Related

Can the Resolution Generated By Short Video Recording Be Customized? What Are the Customizable Outputs?

Short video recording can customize parameters including fps (number of frames per second), GOP (how many seconds to encode a Key I-frame), size, video bitrate (amount of audio and video data generated by the encoder per second), maximum/minimum recording duration, and recording resolution. Four resolutions are provided in a constant manner for you to choose from: 360 × 640, 540 × 960, 720 × 1280, 1080 × 1920.
Why is the recording in a constant method rather than a user-defined size? Reasons are as follows:

- The above four types are mainstream recording resolutions.
- Compatibility issues with Android phones, unsupported non-mainstream resolutions, may cause some screen glitches, green screens, and mosaics.

By calling the startCameraCustomPreview API of the TXUGCRecord class, pass in the custom recording parameters, as follows:

```
// Customize configuration
TXRecordCommon.TXUGCCustomConfig customConfig = new
TXRecordCommon.TXUGCCustomConfig();
customConfig.videoResolution =  TXRecordCommon.VIDEO_RESOLUTION_540_960;
customConfig.minDuration = mMinDuration;  // Minimum duration
customConfig.maxDuration = mMaxDuration;  // Maximum duration
customConfig.videoBitrate = mBiteRate;    // Video bitrate
customConfig.videoGop = mGop;             // GOP size
customConfig.videoFps = mFps;             // FPS
customConfig.isFront = mFront;            // Whether it is a front
camera
mTXCameraRecord.startCameraCustomPreview(customConfig, mVideoView);
```

Why Does the onRecordComplete Callback Not Receive After the Android Short Video Recording Ends?

Before starting to record a short video, please first configure the callback listener by calling the setVideoRecordListener() interface of the TXUGCRecord class.

Ending the recording requires calling the stopRecord() interface of the TXUGCRecord class.

```
// Before recording
mTXCameraRecord = TXUGCRecord.getInstance(this.getApplicationContext());
mTXCameraRecord.setVideoRecordListener(this);

...

// End recording
mTXCameraRecord.stopRecord();
```

How to Continue Recording From the Last Content When Exiting Short Video Recording and Starting the Second Recording?

After the onRecordComplete callback, the Demo calls mTXCameraRecord.getPartsManager().deleteAllParts() to clear shard files because stopRecord has already completed the synthesis of shard files.
If the recording exits and continues the last recording, there is no need to delete the shards. Do not call mTXCameraRecord.getPartsManager().deleteAllParts().

```
@Override
public void onRecordComplete(TXRecordCommon.TXRecordResult result) {
    TXCLog.i(TAG, "onRecordComplete, result retCode = " + result.retCode
+ ", descMsg = " + result.descMsg + ", videoPath + " + result.videoPath
+ ", coverPath = " + result.coverPath);
    if (mTXRecordResult.retCode < 0) {

Toast.makeText(TCVideoRecordActivity.this.getApplicationContext(),
"Recording failed. Reason: " + mTXRecordResult.descMsg,
Toast.LENGTH_SHORT).show();
    } else {
        mDuration = mTXCameraRecord.getPartsManager().getDuration();
//Total duration of recording
        if (mTXCameraRecord != null) {
            mTXCameraRecord.getPartsManager().deleteAllParts();  //
Delete the recorded shard files multiple times
        }
        startPreview(); //Enter the preview interface
```

```
    }
```

Why Does Setting the Background Music For Short Video Recording Not Take Effect?

Set background music must be set before starting recording (TXUGCRecord's startRecord) interface for the changes to take effect. Refer to the following example for code calling sequence:

```
TXRecordCommon.TXUGCSimpleConfig simpleConfig = new
TXRecordCommon.TXUGCSimpleConfig();
simpleConfig.videoQuality = TXRecordCommon.VIDEO_QUALITY_MEDIUM;
simpleConfig.minDuration = mMinDuration;
simpleConfig.maxDuration = mMaxDuration;
// 1. First, enable preview
mTXCameraRecord.startCameraSimplePreview(simpleConfig, mVideoView);
// 2. Then set the path of background music and play background music
mBGMDuration = mTXCameraRecord.setBGM(mBGMPath);
mTXCameraRecord.playBGMFromTime(0, mBGMDuration);
// 3. Start shooting (customVideoPath: video path after recording,
customPartFolder: folder for recorded videos, customCoverPath: cover
path of the recorded video)
int result = mTXCameraRecord.startRecord(customVideoPath,
customPartFolder, customCoverPath);
```

Whether There Is a Photo-Taking Function in Recording?

The User Generated Short Video SDK (UGSV SDK) has a photo-taking function. Call the snapshot API of the TXUGCRecord class, and the TXRecordCommon.ITXSnapshotListener callback will return the photo taken asynchronously. Sample code is as follows:

```
private void snapshot() {
    if (mTXCameraRecord != null) {
        mTXCameraRecord.snapshot(new
TXRecordCommon.ITXSnapshotListener() {
            @Override
            public void onSnapshot(Bitmap bmp) {
                // Image taken by photo
                saveBitmap(bmp);
            }
```

```
            });
        }
    }
```

What Is the Multiple of the Adjustable-Speed Shoot Speed?

Adjustable-speed shooting does not support custom speed.

Definition	Corresponding Constant in TXRecordCommon	Multiple
Ultra-slow	RECORD_SPEED_SLOWEST	0.5x
Slow	RECORD_SPEED_SLOW	0.8x
Standard	RECORD_SPEED_NORMAL	1x
Fast	RECORD_SPEED_FAST	1.25x
Ultra-fast	RECORD_SPEED_FASTEST	2x

Adjustable-speed shoot is achieved by calling setRecordSpeed(record) of TXUGCRecord to set different recording speeds.

```
mTXCameraRecord.setRecordSpeed(TXRecordCommon.RECORD_SPEED_FAST);
```

What Are the Format Requirements For Importing Videos? Whether Importing Videos With a Resolution Higher Than 720P (Such As 2K, 4K) Is Supported? Are There Specific Size Limitations For Imported Files?

Currently, on the Android side, only MP4 is supported for importing videos. There is no limitation on resolution. Importing files has no size limit.

- There is no limitation on the resolution of imported videos. Regardless of the size of the original video, the maximum resolution after pre-processing is 720P.
- To quickly import videos, preprocessing (some features are limited, such as reverse playback, single-frame preview, etc.) can be skipped after SDK 4.7. For videos with a resolution higher than 720P, it is recommended to add preprocessing, because the preview decodes each frame. Some mobile phones have poor performance, resulting in too long a time to decode a frame and render it on the interface, causing stuttering.

Currently What Formats of Background Music Does Short Video Editing Support?

Currently only support MP3 and M4A types.

What Are the Customizable Outputs in Current Short Video Editing?

Short video editing allows customization of video bitrate (SDK 4.5 or higher), audio bitrate (SDK 4.7 or higher), and resolution. Several resolutions are provided as constants for you to choose from: 360 x 640, 480 x 640, 540 x 960, 720 x 1280, 1080 × 1920.

Resolution	Corresponding Constant in TXVideoEditConstants
360x640	VIDEO_COMPRESSED_360P
480x640	VIDEO_COMPRESSED_480P
540x960	VIDEO_COMPRESSED_540P
720x1280	VIDEO_COMPRESSED_720P
1080x1920	VIDEO_COMPRESSED_1080P

```
// Set the output video bitrate
mTXVideoEditer.setVideoBitrate(3600);
// Set output resolution
mTXVideoEditer.generateVideo(TXVideoEditConstants.VIDEO_COMPRESSED_720P,
mVideoOutputPath);
```

Can the Audio in the Recorded Video Through the Short Video Recording Functionality Be Replaced?

Currently, short video recording does not support simultaneous recording of BGM and voice. If you want to add a custom BGM without retaining the original recorded audio, the operating method is as follows:

1. **Enter edit mode**: First, import the recorded video into video editing software.
2. **Replace original audio**: In the editing software, find the audio editing feature. Set the original sound volume to 0 to achieve the effect of hiding the original audio. This way, the original recorded sound will not be heard when playing the video.
3. **Replace background music**: Next, you can select a new BGM file from a local file or other sources and add it to the video. Ensure the new BGM coordinates with the video content and adjust its volume to meet your needs.
4. **Export the edited video**: After completing audio replacement and other editing work, save and export your video.

Through these steps, you can easily add and adjust a custom background music to a video without retaining the original audio.
The specific implementation code is as follows:

```
// Set the volume of video original sound (set to 0 to remove the
recorded BGM)
mTXVideoEditer.setVideoVolume(0.0f);
// Set the local background music path
String bgmPath = getBGMPath();
mTXVideoEditer.setBGM(bgmPath);
// Set the background music volume, ranging from 0.0f to 1.0f
mTXVideoEditer.setBGMVolume(1.0f);
```

How to Switch the Video Preview Between the Same Activity Window and Fullscreen Mode?

Dynamically modify the size of the parent layout of the video preview View passed into the SDK. The SDK will dynamically adjust the size of the video according to the size of the parent layout and the aspect ratio of the video. Call sequence of SDK API:
First perform stopPlay, then modify the width and height of the FrameLayout passed into the SDK, call initWithPreview(parm), pass in the layout of the new FrameLayout hosting the playback component, and startPlay again.

```
// Stop playback
mTXVideoEditer.stopPlay();
if (isFullScreen) {
    // If it is fullscreen mode, then switch to windowed mode below
    FrameLayout.LayoutParams params = new
FrameLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, 1500);
    mVideoPlayerLayout.setLayoutParams(params);
    initPlayerLayout(false);
    isFullScreen = false;
} else {
    // If it is windowed mode, then switch to fullscreen mode below
    FrameLayout.LayoutParams params = new
FrameLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT,
ViewGroup.LayoutParams.MATCH_PARENT);
    mVideoPlayerLayout.setLayoutParams(params);
    initPlayerLayout(false);
```

```
    isFullScreen = true;
}
// Start playback
mTXVideoEditer.startPlayFromTime(startTime, endTime);

// Reset the preview View
private void initPlayerLayout(boolean isFullScreen) {
    TXVideoEditConstants.TXPreviewParam param = new
TXVideoEditConstants.TXPreviewParam();
    param.videoView = mVideoPlayerLayout;
    if (isFullScreen) {
        param.renderMode =
TXVideoEditConstants.PREVIEW_RENDER_MODE_FILL_SCREEN;
    } else {
        param.renderMode =
TXVideoEditConstants.PREVIEW_RENDER_MODE_FILL_EDGE;
    }
    mTXVideoEditer.initWithPreview(param);
}
```

During Short Video Editing, Tencent Cloud'S Short Video Demo Handles Functions Such As "Edit" and "Filter" On One Page. However, Our Company'S Product Divides the "Edit" Function and the "Filter" Into Two Pages

You can first perform cropping (setCutTimeFrom) + pre-processing (processVideo) simultaneously, resulting in a cropped and pre-processed video. Then proceed with various editing operations, set the cropping to the entire duration (setCutTimeFrom), and finally call generateVideo to generate the video, preventing compression twice from reducing video quality.

> **Note**
> The video has been cropped in pre-processing. Before the final generation of the preprocessed video, be sure to set the cropping duration to the entire video duration; otherwise, cropping will be performed again.

```
// Cropping page
mTXVideoEditer = new TXVideoEditer(mContext);
```

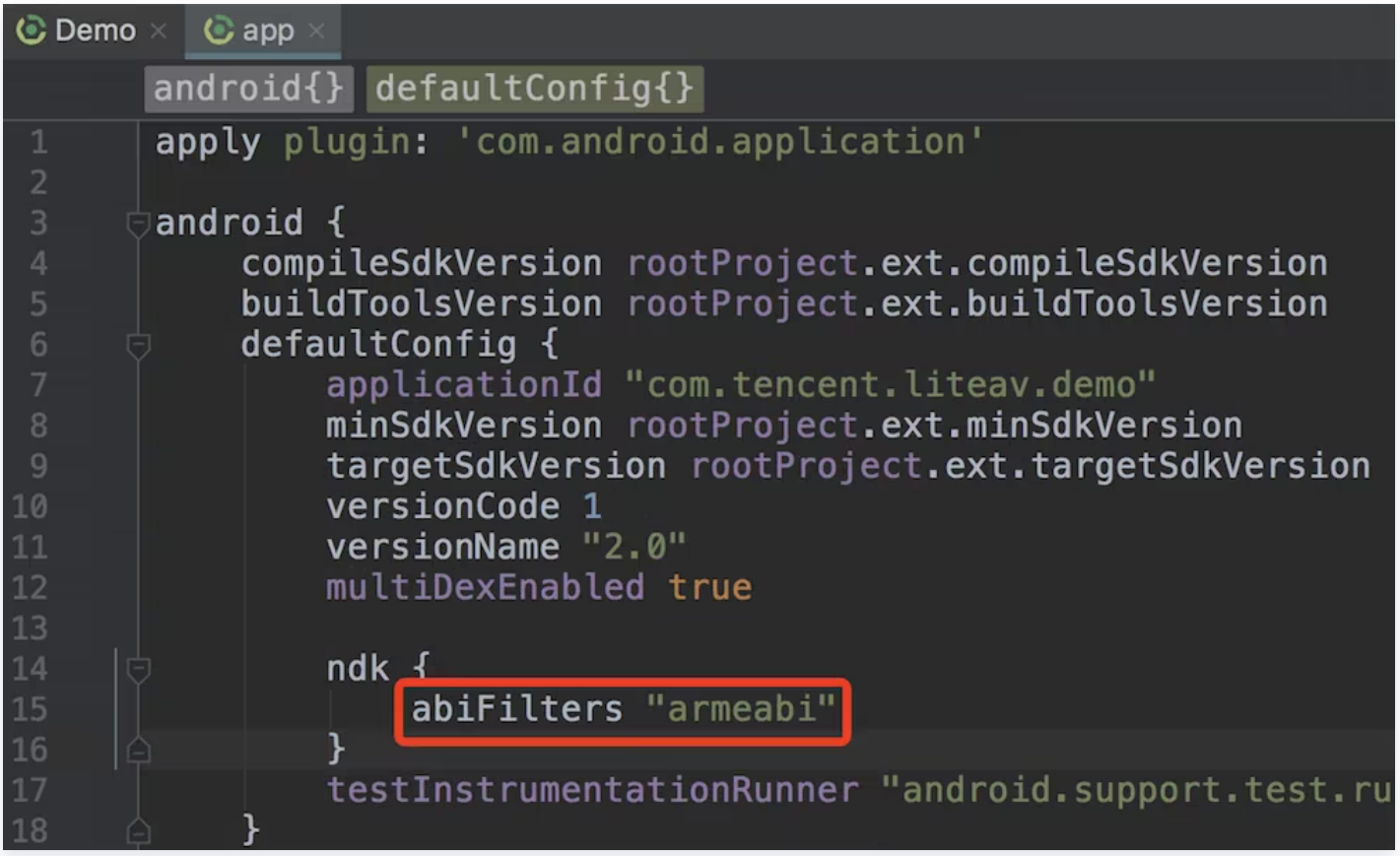
```
mTXVideoEditer.setCutFromTime(mTCVideoEditView.getSegmentFrom(),
mTCVideoEditView.getSegmentTo());
mTXVideoEditer.processVideo();

// Set the cropping to the entire duration (setCutTimeFrom)
mTXVideoEditer.setCutFromTime(0, mVideoDuration);
// Navigate to the Special Effects Page and perform generation
mTXVideoEditer.generateVideo(TXVideoEditConstants.VIDEO_COMPRESSED_720P,
mVideoOutputPath);
```

Compilation Related

What to Do When Encountering Exceptions in Integration?

```
java.lang.UnsatisfiedLinkError: No implementation found for void com.tencent.liteav.basic.log.TXCLog.nativeLogInit() (tried
Java_com_tencent_liteav_basic_log_TXCLog_nativeLogInit__)
    at com.tencent.liteav.basic.log.TXCLog.nativeLogInit(Native Method)
    at com.tencent.liteav.basic.log.TXCLog.init(TXCLog.java:31)
    at com.tencent.liteav.basic.log.TXCLog.setConsoleEnabled(TXCLog.java:52)
    at com.tencent.rtmp.TXLiveBase.setConsoleEnabled(TXLiveBase.java:73)
```

You can use armeabi and armeabi-v7a architectures.



As shown above, please assign abiFilters to "armeabi" in the `app` 's build.gradle.

What to Do When Integrating Two or More LiteAV System SDKs Simultaneously and Conflicts Occur?

If you have integrated two or more SDKs of the LiteAV system in your project, there will be a symbol conflict (symbol duplicate) problem, since the SDKs of the LiteAV system all use the same basic module.
To avoid the symbol conflict issue, the correct approach is not to integrate two SDKs simultaneously, but to integrate the full-featured SDK:

Platform	ZIP Package	Github	64-Bit Support	Installation Package Increment	Slim Installation Package
iOS	DOWNLOAD	Github	Supported	4.08M (arm64)	DOC
Android	DOWNLOAD	Github	Supported	• jar: 1.5M • so(armeabi): 6.5M • so(armv7): 6.1M • so(arm64): 7.3M	DOC

SDK Upgrade, Short Video Features Cannot Be Used?

1. If you are using Android Studio, after replacing the new aar, please modify the aar reference in the build.gradle of `app`, and check if it is consistent with the **Filename** of the aar placed under the /libs directory of your project. Then clean and rebuild your project again.
2. Confirm the SDK version. License support is required for the UGSV SDK after version 4.5. Please first apply for a License. There are two kinds of SDK licenses: **simplified edition** and **basic version**:
   - For differences, please see SDK features and corresponding License versions.
   - For detailed pricing, please see pricing documentation.
   - If you need to use advanced functions such as beauty effect, please see Tencent Effect SDK.

How to Set Suspension and Progress Bar For Short Videos On Android?

Short video playback is implemented based on the short video player. Therefore, the progress bar functionality requires you **to develop on your own**. For the feature implementation instructions, please refer to Player SDK – Progress Display.

iOS FAQs

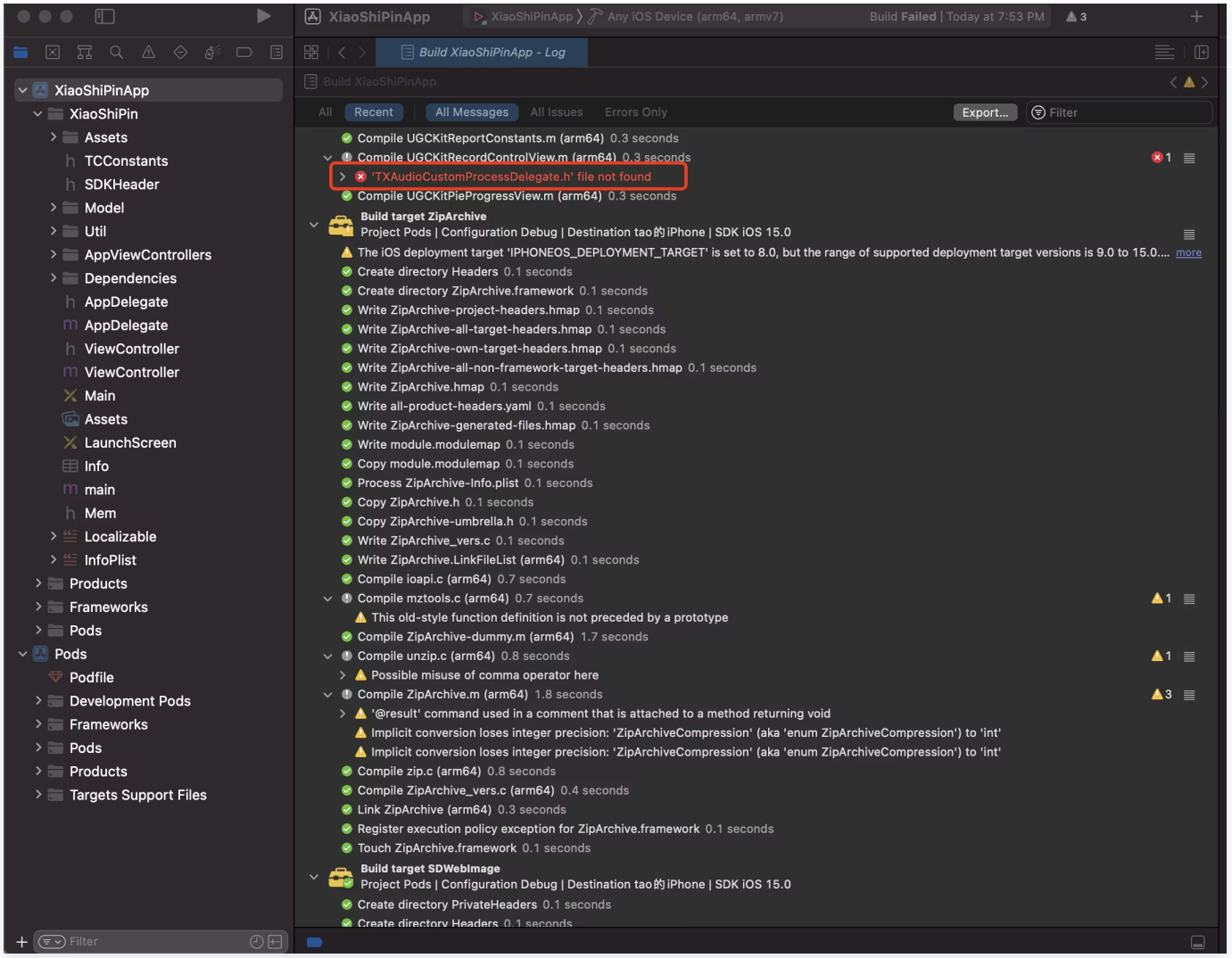
Last updated: 2025-03-17 16:53:49

Problem With TXUGCPublish.H?

Starting from version 4.5, classes related to `TXUGCPublish` have been moved from the SDK to the Demo layer. Developers who need to use them can directly drag the entire `VideoUpload` directory into their own projects during use.

Error When Running Demo Directly in Xcode?

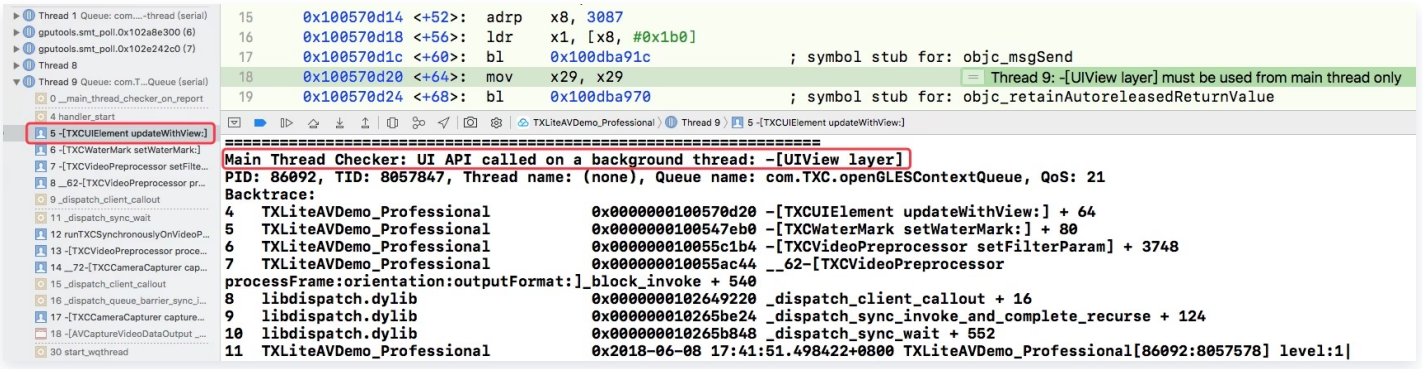
- Error information:



- Solution: Execute pod install.

Connect to Xcode For Debugging, Error in Short Video Recording?

Connect to Xcode for debugging, error in short video recording, error information: `Main Thread Checker: UI API called on a background thread`



- Reason: Some APIs (generally UI-related) need to be called in the main thread. If they are called in a non-main thread and `Main Thread Checker` is selected, an error will be reported.
- Solution: `Product` > `Scheme` > `Edit Scheme` > `Run` > `Diagnostics`, uncheck `Main Thread Checker`.

> **Note:**
> This issue has been fixed in version 4.9.

Getting a Header File Not Found Error When Using SDK?

Just run pod install.

What Should I Do If a crash Occurs or an Error Is Reported Saying That Class Methods Could Not Be Found When I Run the Project?

The SDK uses some methods of categories. To load category methods, you need to add `-ObjC` in project configuration: `Build Settings` > `Linking` > `Other Linker Flags`.

What Should I Do If the Background Music Set For Short Video Recording Is Unavailable?

1. Confirm whether there is a file under the transmitted BGM path and whether it can play normally.
2. Confirm the call sequence of the API: `startCameraSimple:preview:` > `setBGM:` > `startRecord`.

> **Note:**
> Many API calls have timing requirements; otherwise, they will be invalid. Generally, there will be notes on this.

> For example, for short video recording, APIs such as `setVideoResolution:`, `setVideoBitrate:`, `setAspectRatio:`, etc., need to be set before `startRecord` to be valid.

Recording Settings: Can'T BGM Loop Playback?

Currently, the logic does not support loop playback.

Recording Settings: BGM, No Complete Callback At `endTime`?

If the set `endTime` is less than the total duration of the music file, versions prior to 4.6 will trigger the complete callback only after the BGM has finished playing. Starting from version 4.7, the complete callback will be triggered at `endTime`.

Why Is It Slow to Turn On the Camera For the First Time When Recording?

When the mobile camera of an apple phone is turned on for the first time (cold startup), the duration is relatively long. The same is true when the camera is turned on through the system interface.
Because the operation of turning on the camera is not suitable to be placed in a child thread. After testing, the operation duration of turning on the camera in a child thread will be longer. And when the camera is continuously opened/closed in the main thread, the response delay of the child thread will be higher, resulting in a poor experience.

How to Achieve Returning to Continue Recording?

When the first recording is completed, do not call `stopRecord` and `stopCameraPreview` (after calling these, you cannot continue recording, but can only re-record). You can call `pauseRecord`, then use `TXUGCPartsManager.getVideoPathList` to obtain the already recorded video clips, and use `TXVideoJoiner.joinVideo` to synthesize the final video (for versions before 4.5). You can also directly call `TXUGCPartsManager.joinAllParts` to synthesize the final video. This method has a faster synthesis speed (supported in versions after 4.5). In this way, when you return to continue recording, all the recording statuses are retained, and you can continue recording.

Why Is the Complete Callback Not Received Upon Completion of Short Video Recording?

- Determine whether `stopRecord` has been called. Only after `stopRecord` has been called will a complete callback be received.
- Determine whether all function calls are in the main thread.

Can'T Record Audio When Playing Video With Another Player During Recording and Returning to Continue Recording?

The AudioSession in iOS is shared by all audio and video applications. When another player is used for playback, the AudioSession will be occupied. If the AudioSession does not give way or does not give way in time when playback ends, it will cause the AudioSession of the recording module to become invalid. The SDK provides two APIs: `-(void) pauseAudioSession` and `-(void) resumeAudioSession`. Call `pauseAudioSession` before going to another player preview, and call `resumeAudioSession` before returning to continue recording.

Why Is the Recorded Video Unclear?

If the bitrate and resolution do not match, the recorded video will be unclear. You can improve image quality by appropriately increasing the bitrate and enabling B-frames.

Video Generation Failure When Going Into the Background and Then Back to the Foreground During Video Editing?

Video generation defaults to hard coding (high coding efficiency and good image quality). The hard encoder will stop working when the application enters background, leading to video generation failure. The SDK provides two APIs: `pauseGenerate` and `resumeGenerate`. When the App enters background, you can call `pauseGenerate` to pause video generation. When the App returns to foreground, you can call `resumeGenerate` to continue video generation.

> **Note:**
> Call `resumeGenerate`. The SDK will restart the hardware encoder. There is a certain probability that the restart will fail, or the encoding of the first few frames of data will fail after restart. At this point, an error event will be thrown inside the SDK in `TXVideoGenerateListener`. After receiving the error event, it is necessary to regenerate the video.

File Upload Failed?

File upload status code:

```
typedef NS_ENUM(NSInteger, TXPublishResultCode)
{
    PUBLISH_RESULT_OK                           = 0,        //发布成功
    PUBLISH_RESULT_UPLOAD_REQUEST_FAILED        = 1001,     //step1: "文件上传请求"发送失败
    PUBLISH_RESULT_UPLOAD_RESPONSE_ERROR        = 1002,     //step1: "文件上传请求"收到错误响应
    PUBLISH_RESULT_UPLOAD_VIDEO_FAILED          = 1003,     //step2: "视频文件"上传失败
    PUBLISH_RESULT_UPLOAD_COVER_FAILED          = 1004,     //step2: "封面文件"上传失败
    PUBLISH_RESULT_PUBLISH_REQUEST_FAILED       = 1005,     //step3: "短视频发布请求"发送失败
    PUBLISH_RESULT_PUBLISH_RESPONSE_ERROR       = 1006,     //step3: "短视频发布请求"收到错误响应
};
```

1. Confirm whether the uploaded file is in the local sandbox. If uploading a file to the media library, it needs to be copied to the local sandbox first.
2. Return error code 1002: Signature issue, timestamp expired, video-on-demand (VOD) service issue (not enabled or service suspended).
3. Return error code 1003: Parameter issue, unsupported upload file format.

Whether There Is a Photo-Taking Function in Short Video Recording?

The User Generated Short Video SDK (UGSV SDK) can realize the photo-taking function. After starting the preview, just call the `snapshot` API of the `TXUGCRecord` class to get the image.

Error "Use of undeclared identifier 'TXVideoInfo'" Keeps Reporting During Integration?

This error occurs because the compiler does not detect the TXVideoInfo class. Suggest checking whether the SDK (framework) is correct. You can re-import the project according to Documentation steps.

What to Do If an Error "-1, Failed to enable encoder" Occurs When Calling Video Synthesis?

1. Please confirm whether the issue occurs consistently. It is recommended to change models for testing.
2. Download the latest version of Demo and reproduce the issue. If the issue is reproducible, please provide complete log information and submit a work order for resolution.