FFmpeg for Unity

目次

1. 概要

これはUnityのエディタ上やアプリ上でFFmpegを使用できるアセットです。

以下のようなことが出来ます。

- 動画の再生(mp4, avi, mov etc.)
- ゲーム画面、もしくはゲーム内カメラからの録画
- 動画の変換
- Youtubeへのライブ配信(rtmp etc.) など

2. 対応環境

- Unityエディタ(Windows/Mac/Linux)
- スタンドアローン(Windows/Mac/Linux)(Mono/IL2CPP)
- Android(IL2CPP)
- iOS

3. FFmpeg本体について

このアセットでは以下のスクリプトからビルドしたFFmpeg本体(Windows/Mac/Linux)、もしくはプラグイン(Windows/Mac/Android/iOS)を使用しております。

外部ライブラリは一部のみ（「ライセンス」を参照）にしておりますので、使用したい機能がない場合は手順に従って再ビルドし、それに置き換えてください。

Windows、Mac(Library)およびLinux(Library)は1.を実行して生成したものを2.から置き換えて実行してください。

- Windows:

1. https://github.com/NON906/ffmpeg-windows-build-helpers
2. https://github.com/NON906/fftools_lib

- Mac(Library):

1. https://github.com/NON906/ffmpeg-kit
2. https://github.com/NON906/FfmpegUnityMacPlugin

- Linux(Library):

1. https://github.com/NON906/ffmpeg-build-script
2. https://github.com/NON906/fftools_lib

- Android/iOS: https://github.com/NON906/ffmpeg-kit
- Mac/Linux(Binary): https://github.com/NON906/ffmpeg-build-script

4. 使用方法

1. このアセットをインポートします。
2. 必要であれば、「FFmpeg本体について」にあるURLからFFmpegをビルドし、置き換えます。
3. 以下のように設定を変更します。
   3-1. メニューバーから'File -> Build Settings'を選択してください。
   3-2. 'Player Settings...'をクリックしてください。
   3-3. (Unity2020の場合)'Other Settings -> Api Compatibility Level'を'.NET 4.x'に変更してください。
   3-4. (Androidの場合)'Other Settings -> Scripting Backend'を'IL2CPP'に変更してください。
   3-5. (Androidの場合)'Other Settings -> Target Architectures'の'ARM64'をオンにしてください。
4. サンプルシーンを実行したい場合、以下を行います。

- `Assets/StreamingAssets`に再生（もしくは変換）したい動画を`sample.mp4`という名前で配置してください。
- PlayerVR, ListVR: 上記に加え、Package ManagerからXR Interaction Toolkitとそれに対応したPluginをインポートしてください。
- YoutubeLive: URLにある`[STREAM_KEY]`をYoutubeのライブ配信の画面から取得したストリームキーに置換してください。

5. (iOSの場合)出力データを確認したい場合、ビルド後にxcodeのUnity-iPhone -> Infoから以下の設定を行ってください。

- Application supports iTunes file sharing: YES
- Supports opening documents in place: YES

NOTE: ver1.6から、`com.arthenica.smartexception`を同梱するようになりました。
アップデート後にエラーが発生した場合は、`Assets/Plugins/Android/mainTemplate.gradle`の設定を削除してください。

NOTE: ver2.7以降、Androidでビルドしたアプリがクラッシュする問題が発生する場合があります。
もし発生した場合は以下のどちらかをお試しください。

- Unityのバージョンを2021.3.27f1以降に変更する
- Export Projectをオンにしてビルドし、そのプロジェクトをAndroid Studioでビルドする（Gradleを更新する必要あり）

5. 各種設定

メニューバーのTools→Ffmpeg for Unity→Open Setting Windowから設定を行えます。
動作の不具合などがない場合は、デフォルト設定のまま使用するのを推奨します。

- Library

ライブラリ化したffmpegを使用します

- Built In Binary

以下のパスに存在するffmpeg本体を使用します
アプリとしてビルドする場合は自動で組み込まれます
ver1.6.1以前でUse Built InをONにしたときと同じ動作になります

Windows: Assets/FfmpegUnity/Bin/Windows
Mac: Assets/FfmpegUnity/Bin/Mac
Linux: Assets/FfmpegUnity/Bin/Linux

- Installed Binary

実行環境にインストールされているffmpeg本体を使用します
実行環境にffmpeg本体を事前にインストールしており、なおかつPATHが通っている必要があります
ver1.6.1以前でUse Built InをOFFにしたときと同じ動作になります

NOTE:

- for Windows:
  WindowsでBuilt In Binaryもしくは Installed Binaryを使用したい場合は、こちらからffmpeg本体をインポートする必要があります
- for Mac:
  MacでBuilt In Binaryを使用したい場合は、こちらからffmpeg本体をインポートする必要があります
  なお、この本体はM1 Macでは動作しない可能性があるため、該当する場合は対応する本体を用意するか、Built In Binary以外を使用してください。

6. シーン詳細

- List
- ListVR

使用できる機能の一覧を表示します
後者はVR用です

- Convert
- ConvertProgress

動画形式の変換のサンプルです

- FfplayPlayer

動画プレイヤーのサンプルです

- FfplayTexture
- FfplayRenderTexture

動画をTextureに適用する方法のサンプルです

- Capture
- CaptureRenderTexture

動画録画のサンプルです
それぞれで録画元が異なります

- SendStream

動画のストリーミング配信のサンプルです

- FfplayStream

動画のストリーミング配信の受信を行うサンプルです

- YoutubeLive

YoutubeLiveへの配信のサンプルです

- BytesInput
- BytesOutput
- ConvertBytes
- FfplayBytesTexture

バイトを直接入出力する場合のサンプルです

- BatchTexture
- BatchTextureWithSound

フレーム毎に処理する場合のサンプルです

7. コンポーネント詳細

7-1. 基本

7-1-1. FfmpegCommand

設定されたFFmpegのコマンドを実行します。 動画の変換など、表示が不要な処理に使用してください。

インスペクターの設定:

Execute On Start: シーン開始時にコマンドを実行します。使用しない場合は、`StartFfmpeg()`から実行してください。

Options: 実行したいコマンドのオプションを設定します。 以下を記載した場合、対応するパスに変換されます。

```
{STREAMING_ASSETS_PATH} : Application.streamingAssetsPath
{PERSISTENT_DATA_PATH} : Application.persistentDataPath
{TEMPORARY_CACHE_PATH} : Application.temporaryCachePath
```

NOTE: Use Built Inは設定ウィンドウに引っ越しました

7-1-2. FfplayCommand (New Player)

設定したFfmpegPlayerVideoTextureとAudioSourceに、映像や音声を設定します。

インスペクターの設定:

Execute On Start: FfmpegCommandと同じです。

Options: 実行したいコマンドのオプションを設定します。
NOTE: ffplayのオプションを指定してください。他で指定するffmpegのコマンドとは異なります。

Default Path: 基準となるパスを指定します。 以下の2つは同じです。

```
Default Path: STREAMING_ASSETS_PATH
Input Path: sample.mp4
```

```
Default Path: NONE
Input Path: {STREAMING_ASSETS_PATH}/sample.mp4
```

Input Path: 動画ファイルのパスを指定します。

Video Texture: 設定したFfmpegPlayerVideoTextureの設定に応じて、以下の処理が行われます。

```
FfmpegPlayerVideoTextureのVideoTextureが空の場合: VideoTextureに新しくTexture2Dが設定されます。
スクリプトでVideoTextureをRendererなどに設定してください。

FfmpegPlayerVideoTextureのVideoTextureにRenderTextureがある場合: 設定したRenderTextureに書き込みます。
```

Audio Source Component: 設定したAudioSourceのclipに音声が設定されます。

7-1-3. FfmpegPlayerCommand (Old Player)

設定したFfmpegPlayerVideoTextureとAudioSourceに、映像や音声を設定します。

インスペクターの設定:

Execute On Start, Options: FfmpegCommandと同じです。

Input Options: 動画の指定する前のコマンドのオプションを設定します。

Default Path: 基準となるパスを指定します。

Input Path: 動画ファイルのパスを指定します。

Auto Settings: 自動で動画の大きさなどを設定します。 基本的にONで構いませんが、一度設定を取得してから、再度実行することになるため、ストリーミングなどでは上手くいかない場合があります。

Video Textures: 設定したFfmpegPlayerVideoTextureの設定に応じて、以下の処理が行われます。

> `FfmpegPlayerVideoTexture`の`VideoTexture`が空の場合: `VideoTexture`に新しく`Texture2D`が設定されます。
> スクリプトで`VideoTexture`を`Renderer`などに設定してください。
>
> `FfmpegPlayerVideoTexture`の`VideoTexture`に`RenderTexture`がある場合: 設定した`RenderTexture`に書き込みます。

Audio Sources: 設定したAudioSourceのclipに音声が設定されます。

Buffer Time: バッファを行う遅延時間（秒）です。
動作が不安定な場合を除いて、0で構いません。
負の値を指定した場合はバッファを無効にします。

7-1-4. FfmpegCaptureCommand

映像や音声を取得して、動画の保存や配信を行います。

NOTE: スマートフォンなどで処理が間に合わない場合は、映像の速度が異常に早くなることがあります。 その場合、サイズなどの設定を行って、処理を軽減させてください。

NOTE: URPやHDRPではVideo_Cameraが使用できません。
またVR機器の場合、Video_GameViewやVideo_Cameraが使用できない場合があります。
これらに該当する場合はVideo_RenderTextureを使用してください。

インスペクターの設定:

Execute On Start, Options: FfmpegCommandと同じです。

Capture Sources: キャプチャしたい内容を設定します。 Video Sizeが設定出来る場合に0以下の値を設定した場合、その値は自動で設定されます。

7-2. バイト処理

7-2-1. FfmpegBytesCommand

byte[]での入力と出力ができるFfmpegCommandの派生コンポーネントです。

`AddInputBytes(bytes, inputNo)`で入力ができます。
（`inputNo` は'Input Option'に対応したストリーム番号です）

`GetOutputBytes(outputNo)`で出力内容を取得できます。
（戻り値はbyte[]です）
（`outputNo`は'Output Option'に対応したストリーム番号です）

Input Options: 入力ストリームの数とその前に指定するオプションを設定します。

NOTE: このリストの数とストリームの数は一致します。 オプションを指定しない場合でも、（空欄で構わないので）必要なストリームの数を用意する必要があります。

Output Options: 出力ストリームの数とその前に指定するオプションを設定します。

NOTE: 'Input Options'と同様に一致させる必要があります。

7-2-2. FfmpegBytesPlayerCommand

byte[]での入力ができるFfmpegPlayerCommandの派生コンポーネントです。

詳細はFfmpegPlayerCommandとFfmpegBytesCommandを参考にしてください。

7-2-3. FfmpegBytesCaptureCommand

byte[]での出力ができるFfmpegCaptureCommandの派生コンポーネントです。

詳細はFfmpegCaptureCommandとFfmpegBytesCommandを参考にしてください。

7-3. フレーム毎バッチ処理

7-3-1. FfmpegGetTexturePerFrameCommand

フレーム毎に取得が可能なFfmpegPlayerCommandの派生コンポーネントです。

`GetNextFrame()`コルーチンを実行することで次のフレームに更新します。
（実行しない間は次のフレームに進みません）

7-3-2. FfmpegWriteFromTexturesCommand

Textureの内容を動画に書き込むためのコンポーネントです。

`WriteTexture(inputTexture)`コルーチンで指定のTextureの内容を書き込むことが出来ます。