

Video Player Extreme

Version 1.0

<https://marketplace.yoyogames.com/assets/9062/video-player-extreme-for-gms-2>

Introduction

'Video Player Extreme' for GMS 2.x is a high performance multi-threaded video decoder and renderer, which can play multiple videos simultaneously. It supports all codecs that Windows Media Player supports, including MPEG, AVI, MP4, etc... Without the need for prior conversion. Just point to the file and go.

It is highly recommended that you familiarize yourself with the demo included in the asset. 'obj_video' contains examples of every function and common, but certainly not limited to all, use cases. The limit is your imagination.

Basic Usage

Initialization

The basic initialization is a requirement for any application using Video Player Pro. This only needs to ever be done once but must be done before any video is loaded, preferably in the create event.

```
video_init(window_device());
```

Termination (Optional)

To fully terminate the extension which will release all resources you can call video_release. This is optional as the extension will call automatically when the program has ended.

```
video_release();
```

Loading a video

To play a video we must first load it. We can load a video from the projects included files or directly from any location on the computer. We are not restricted to the GameMaker 'sandbox'.

```
video_load("filename");
```

Formatting an absolute path "c:\\videos\\movie.mp4" or "c:/videos/movie.mp4" will work also.

On success, this function will return the unique instance ID for the loaded video. This instance ID can then be used for all other video related functions. It is the instance ID that distinguishes each video during simultaneous file playback.

Drawing the video

To draw the video simply call `video_render` from the draw event.

```
video_render(instance_id);
```

Function list

(Alphabetical order)

Function	Arguments	Return value	Notes
<code>video_dll_version</code>	N/A	Numeric	Version information
<code>video_render</code>	Instance ID	1 Success -1 Failure	Renders current frame
<code>video_get_current_time</code>	Instance ID	Numeric	Current playback time
<code>video_get_duration</code>	Instance ID	Numeric	Duration of video
<code>video_get_height</code>	Instance ID	Numeric	Video height
<code>video_get_loop</code>	Instance ID	0 Non looping 1 Looping	Check if video is set to automatically loop
<code>video_get_playback_ended</code>	Instance ID	0 Ended 1 Playing	Check to see if video has reached end of timeline.
<code>video_get_rate</code>	Instance ID	Numeric	Video playback rate
<code>video_get_volume</code>	Instance ID	Numeric	Returns video volume between 0.0 to 1.0
<code>video_get_muted</code>	Instance ID	0 Non muted 1 Muted	Returns if video sound is muted
<code>video_get_width</code>	Instance ID	Numeric	Video width
<code>video_init</code>	<code>window_device()</code> *Required	1 Success -1 Failure	Initializes extension
<code>video_release</code>	N/A	N/A	Releases extension from memory
<code>video_load</code>	String	1 Success -1 Failure	Filename of video
<code>video_pause</code>	Instance ID	1 Success -1 Failure	Pauses playback
<code>video_play</code>	Instance ID	1 Success -1 Failure	Resumes playback
<code>video_set_current_time</code>	Instance ID, time	1 Success -1 Failure	Sets playback location in video
<code>video_set_loop</code>	Instance ID, looped (1 or 0)	1 Success -1 Failure	Enables or disables looping
<code>video_set_rate</code>	Instance ID, rate	1 Success -1 Failure	Sets playback rate
<code>video_set_volume</code>	Instance ID, rate	1 Success -1 Failure	Set volume
<code>video_set_muted</code>	Instance ID, muted (1 or 0)	1 Success -1 Failure	Enables or disables video audio

A note on return codes

In general and unless otherwise noted, a successful call will return 1 and a failure will return -1.

The reasoning behind breaking industry standards here is because if GameMaker can't find the extension for whatever reason (for example the DLL missing), it will return 0. Which can cause unnecessarily lengthy debugging issues.

With this extension, when a function ever returns zero, when the expected return code is one or minus one, you can bet the extension hasn't been included correctly.