

# Gif Converter

by Leo Traub

## Quick Start

Note that there are two ways, you can get your gifs into the scene with this asset. Both make use of "GifData" objects I implemented as ScriptableObjects. These objects can either be created at runtime or precomputed in the editor and therefore stored in your asset folder.

These GifData objects then have a Texture2D array, that can be iterated through, a frame delayTime extracted from the gif and an endurance, of how long the gif lasts, which is basically the delay time multiplied with the number of Frames.

I deliver three Scripts to play the gifs, either on 3D Meshes, 2D Sprites (with a SpriteRenderer) or the Canvas (with a RawImage Component).

## Precomputation Scene

- Just hit play, to see what the scene does! Basically it creates some data in your asset folder (At the given save path) and makes sure the references are done right.

- Use the following semantics, when you type in the source and save path of the gif:

(These are examples)

Source Path: "Assets/ToastsGifConverter/ExampleScene/fox.gif"

Save Path: "Assets/Gifs"

Save Name: "foxy"

Of course you might need to reference other directories in your project or on your pc, but make sure, not to have too many backslashes. The save name is the name of the GifData object, that will be created and it will be used for the generated Texture2D frames.

- The bool value hasAlpha should only be true, if the gif your are deriving your data from, has and alpha channel. It affects wether the generated Texture2D frames are saved as .png (with alpha) or .jpg (without) files.

- When there is no alpha, you can enable jpg compression, to lower the quality of the frames. I highly recommend using the compression, because the amount of data in a gif can be quite high and at around 70-80% you still dont see a lot of difference on moving pictures.

## Gif loading at runtime

- For the runtime conversion I highly recommend looking in the ExampleManager script in the example scene. Try to stick to this code and especially use a similar coroutine in your script. Gif conversion is quite a heavy computation so coroutines are needed if you dont wanna really bad fps crashes.

## GifPlayer Components

- Just add one of the three GifPlayer Components to the object, you want a gif to be played on. (GifPlayerMesh for 3D Meshes, GifPlayerSprite for 2D Sprites, GifPlayerUI for RawImage)
- GifPlayer Initialization can be easily done just like in the ExampleManager script

## Notes

- Due to image conversion all gifs get mirrored along the y-axis in the process, but you can just set the y-scale of your objects to -1 or find other solutions to this issue.
- Gif conversion at runtime is not recommended, if you can avoid it
- Expect long loading times for gifs with more than 500x500 pixels and 50 frames
- Precomputation can also take a while, but these are calculation that only happen once
- Keep in mind that GifData objects can be modified, if you want the animation to play faster for example, just decrease the delayTime

Have fun with my asset! I am really looking forward to see it in your games! ;D Please give me credit somewhere in your game and maybe let me see your creations, as soon as they are ready!!

For questions and feedback contact me: [toastarve@gmail.com](mailto:toastarve@gmail.com)