

Including animations in L^AT_EX documents

Arthur Nishikawa

April 13, 2022

There are a few different options, but the one I had more success with is using the `animate` package. However, this will **only work with PDF viewers that support JavaScript**, like Adobe Reader or PDF-XChange (on Ubuntu is installed with Wine and runs smoothly). Funnily, as far as I know, browser readers do not support JavaScript!

A basic header looks like this:

```
\documentclass{article}

\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}

\usepackage{graphicx}
\usepackage{animate}
```

You need to have the frames you want to animate saved somewhere with the names being indexed with numbers (see Fig. 1).

If you have a your animation as a gif, but not as separate frames, you can split it using `imagemagick`'s `convert` command (imagemagick.org):

```
convert -coalesce cute_gif.gif cute_frame_%05d.pdf
```

I suggest to always embed images (in this case, frames) that are in the same format as the document that you want to generate. This makes the compilation much faster. For that reason, I normally include images already in the pdf format.

Then, you use the `\animategraphics` command to include the animation:

```
\begin{figure}
  \centering
  \animategraphics[loop,autoplay,width=.5\textwidth]{30}{img/cute_gif/cute_frame_}{00000}{00023}
  \caption{Example of animation}
  \label{gif:cute_animation}
\end{figure}
```



Figure 1: Gif split into frame files

`[loop,autoplay,width=.5\textwidth]` are self-explanatory options, `{30}` is the frame rate in fps, `img/cute_gif/cute_` is the base name, and `{00000}{00023}` are the indices of the first and last frames (what goes after the base name). `animate` guesses the extension automatically.

The final result:

Figure 2: Example of animation. You can only properly view it in a PDF viewer that support JavaScript, otherwise it will be frozen in the first frame.