
PYTHON FOR QUICK, USEFUL MULTIMEDIA MANIPULATION

Tanim Islam, 8 JULY 2020

ONLINE PRESENTATION

Anecdotes from a Python Programmer

PYTHON FOR QUICK, USEFUL MULTIMEDIA MANIPULATION

tanim.islam@gmail.com

ONLINE PRESENTATION

WHAT THIS TALK IS ABOUT

- **I make and manipulate lots of images and videos in my day-to-day work.**



WHAT THIS TALK IS ABOUT

- **I show code and demonstrate these things I do all the time.**
- **auto-cropping images.**
- **Videos from a collection of images.**
- **Animated GIFs.**
- **YouTube + song metadata → M4A file.**



WHAT THIS TALK IS ABOUT

- **Functionality comes from three Github repos**
 - **GITHUB GIST THAT MAKES MOVIES FROM IMAGES**
 - **NPRSTUFF**
 - **COVID19_STATS**
 - **PLEXSTUFF**



AUTOCROPPING IMAGES

AUTCROPPING LOSSY (PNG, JPEG, TIFF, ETC.) IMAGES



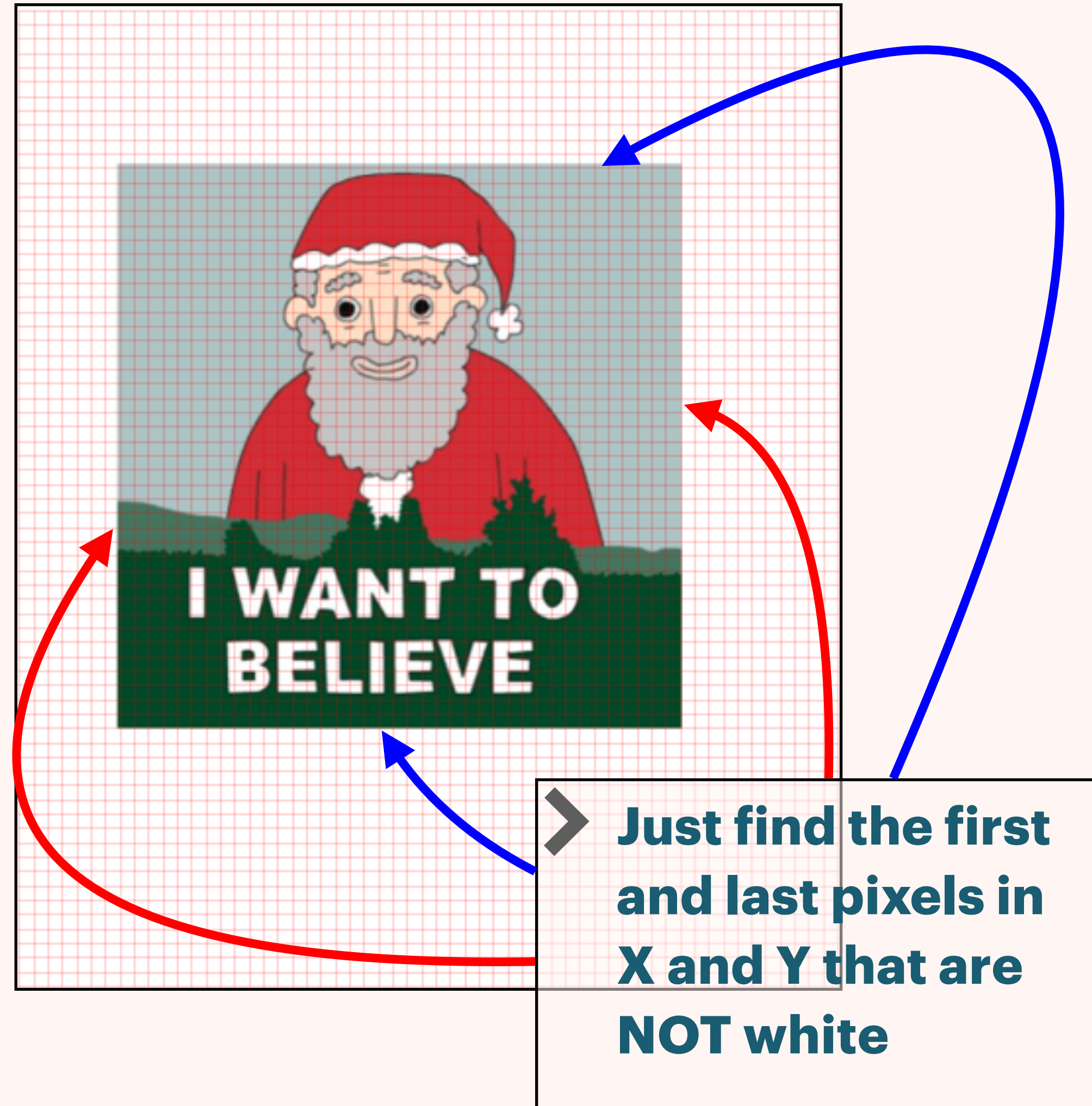
➤ First, load image file from the imread function.

➤ `imread(...)`



➤ Image is stored as a 2D `numpy.array` of **RGBA** values.

AUTCROPPING LOSSY (PNG, JPEG, TIFF, ETC.) IMAGES

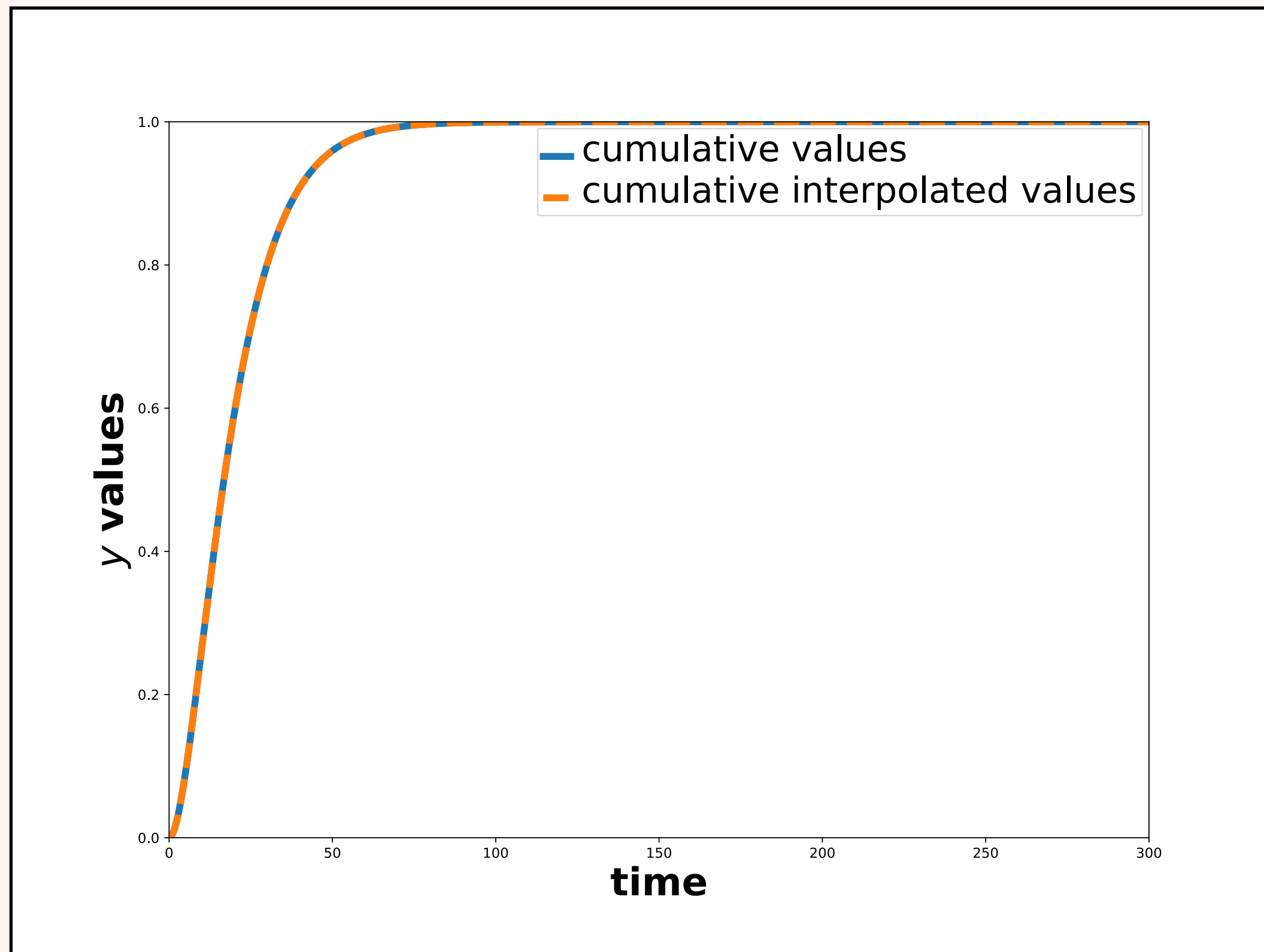


> That is the bounding box. Crop!

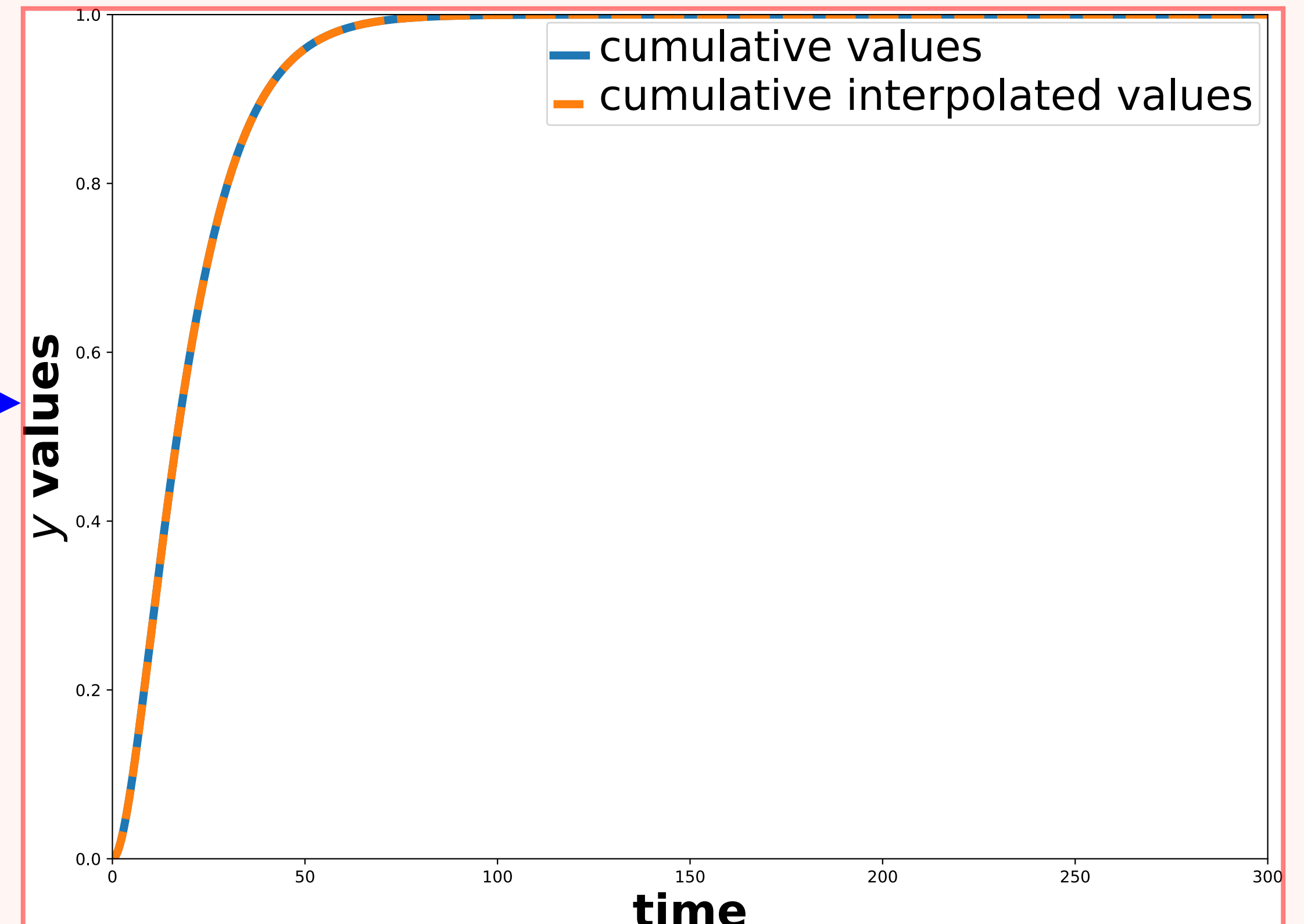
AUTOCROPPING PDF IMAGES

- **Uses PyPDF2 and Adobe Ghostscript at a lower level.**
- **Uses Ghostscript to estimate the bounding box.**
- **Crops PDF according to bounding box using PyPDF2.**

AUTOCROPPING PDF IMAGES



UNCROPPED



AUTOCROPPED

DEMONSTRATION

➤ **autoCropImage, from NPRSTUFF Github Repo, automatically crops images.**

```
usage: autoCropImage [-h] --input INPUT [--output OUTPUT] [--color COLOR]
                        [--trans] [--newwidth NEWWIDTH] [--show]
```

optional arguments:

<code>-h, --help</code>	show this help message and exit
<code>--input INPUT</code>	Name of the input file.
<code>--output OUTPUT</code>	Name of the output file. Optional.
<code>--color COLOR</code>	Name of the color over which to autocrop. Default is white.
<code>--trans</code>	If chosen, also remove the transparency wrapping around the image. Works only for non-PDF images.
<code>--newwidth NEWWIDTH</code>	New width of the image.
<code>--show</code>	If chosen, then show the final image after cropped.

CREATING MOVIES

CREATING MOVIES

- **One functionality, *MP4 movie from a sequence of images*, is created for this presentation and as a GitHub Gist.**
 - **Other functionality, *MP4 movie into animated GIF*, uses convertImage's other functionalities.**
 - **Unlike other tools, low-level functionality is handled by FFMPEG (video and audio creation SDK and tools) through subprocess.**
-

MOVIE FROM IMAGE SEQUENCE

- Python CLI follows [this website](#) on creating an MP4 movie from sequence of images.
- *Image width and height must be divisible by 2.*



DEMONSTRATE MOVIE FROM IMAGE SEQUENCE #1

```
python3.7 demo_create_movie_sequence.py \  
--prefix=img --output=mcmc_images.mp4 \  
--dirname="mcmc_animation_images" --fps=10
```

001

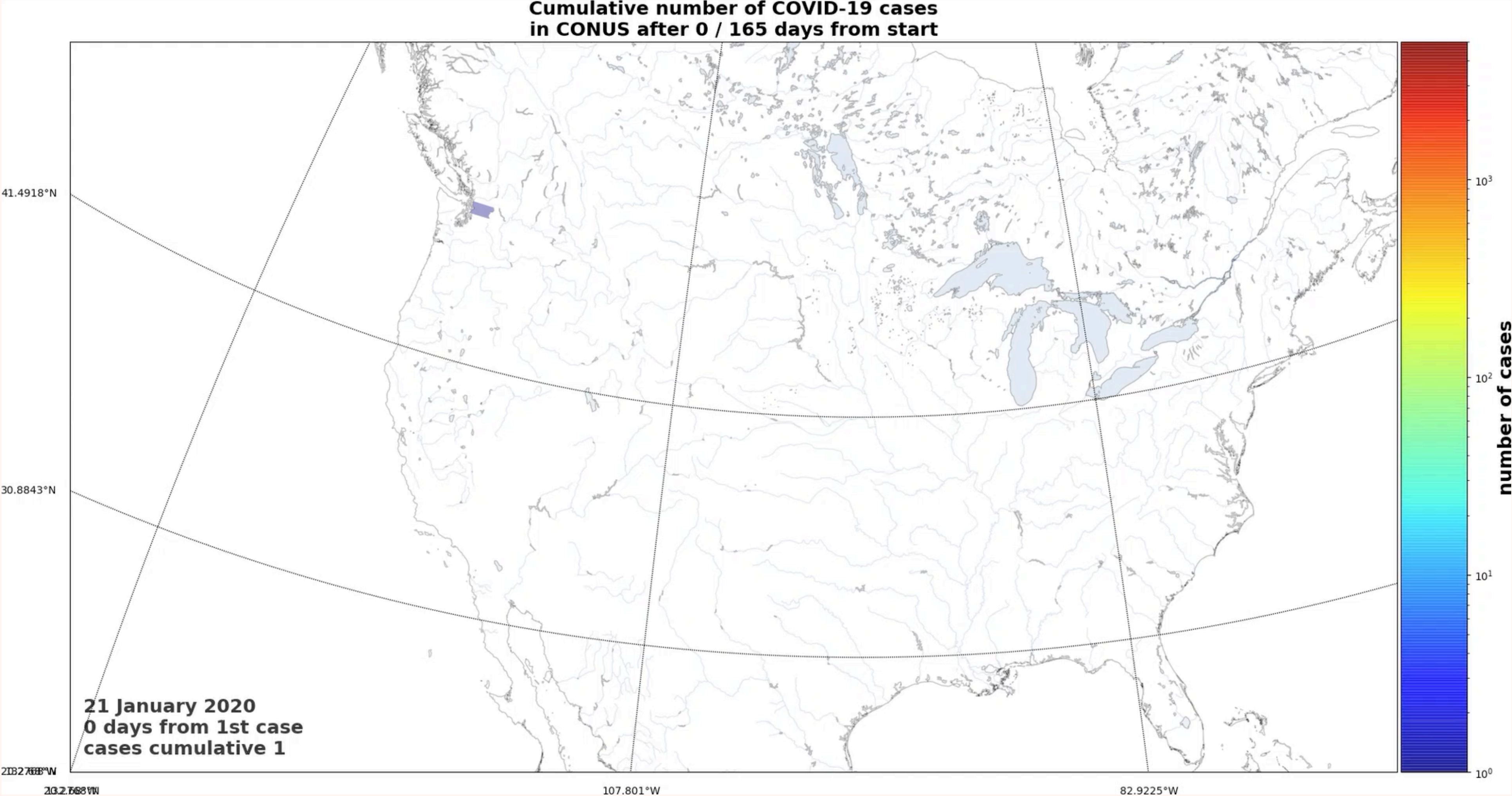


DEMONSTRATE MOVIE FROM IMAGE SEQUENCE #2

- From the [COVID-19 statistics Github repo](#).
- Cumulative COVID-19 cases in CONUS from beginning to 4 JULY 2020.

```
python3.7 demo_create_movie_sequence.py \  
--prefix="covid19_conus_cases_04072020." \  
--output="covid19_conus_cases_04072020.mp4" \  
--dirname="covid19_conus_cases_04072020_imagefiles" \  
--fps=5
```

DEMONSTRATE MOVIE FROM IMAGE SEQUENCE #2



ANIMATED GIF FROM MP4 OR YOUTUBE

- **Why use? Some services (such as [GitHub](#) or [Read the Docs](#)) do not allow movie files in documentation**
 - **They DO allow animated GIFs.**
 - **Two-step process creates a GIF from an MP4 file, taken from this website**
 - **1st pass creates a high-quality global color palette representative of movie file.**
 - **2nd pass converts to animated GIF using 1st pass color palette.**
 - **YouTube to GIF (YouTube → MP4 → GIF) uses [youtube-dl](#) to download YT clip as MP4.**
-

ANIMATED GIF FROM MP4 OR YOUTUBE

➤ Also uses NPRSTUFF's convertImage's other functionalities

```
usage: convertImage [-h] [--noverify] [--info]
                  {image,movie,youtube,square} ...

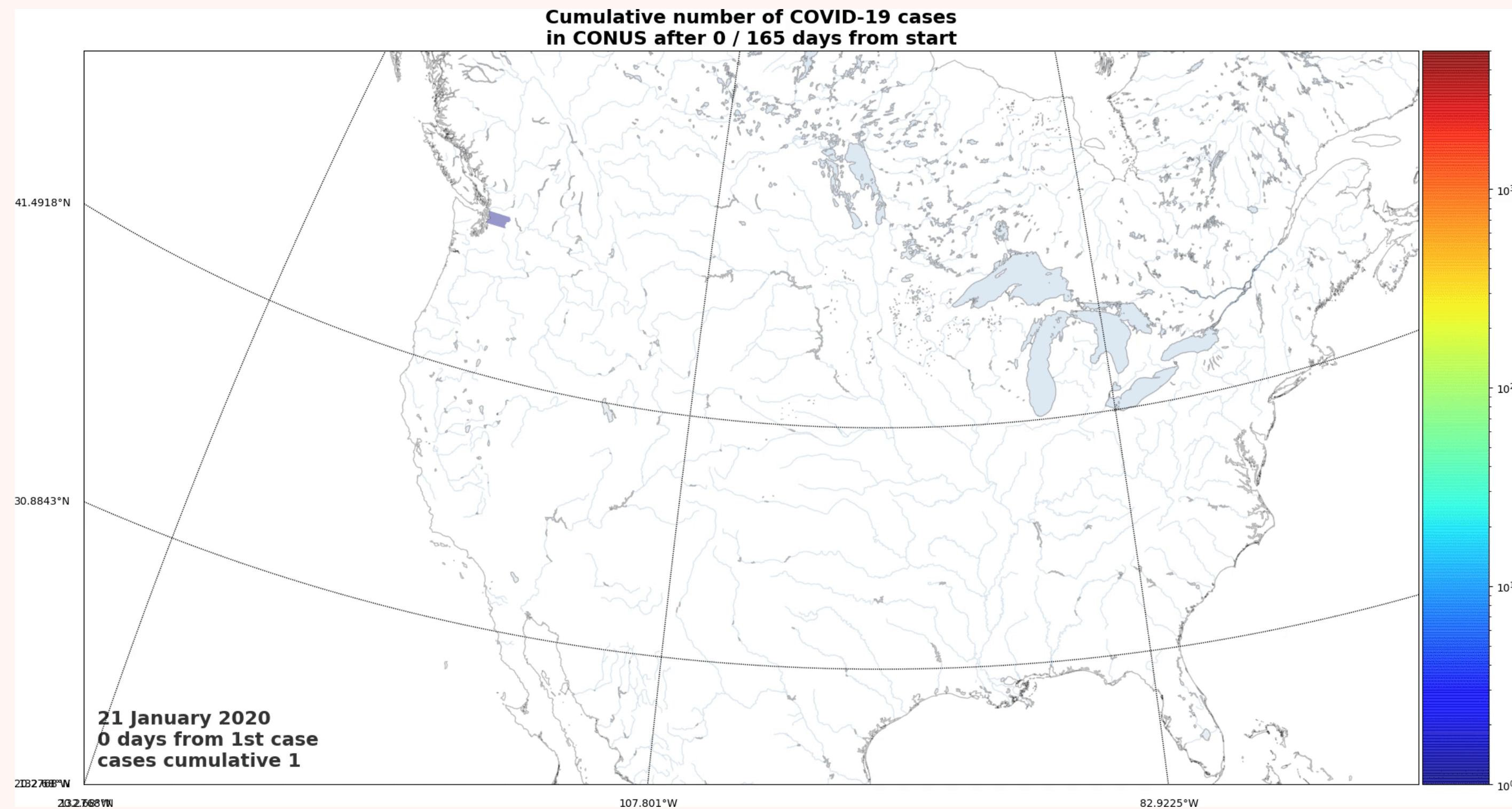
Uses CloudConvert to convert image or video files.

positional arguments:
  {image,movie,youtube,square}
    image               Choose whether to convert a video or an image
                        If chosen, convert an SVG(Z), PDF, or PNG into PNG.
    movie               If chosen, convert an MP4 into an animated GIF.
    youtube             If chosen, convert a YOUTUBE video with URL into an
                        animated GIF.
    square              If chosen, create a square MP4 file from an input MP4
                        file.

optional arguments:
  -h, --help            show this help message and exit
  --noverify            If chosen, do not verify the SSL connection.
  --info                If chosen, then print out INFO level logging.
```

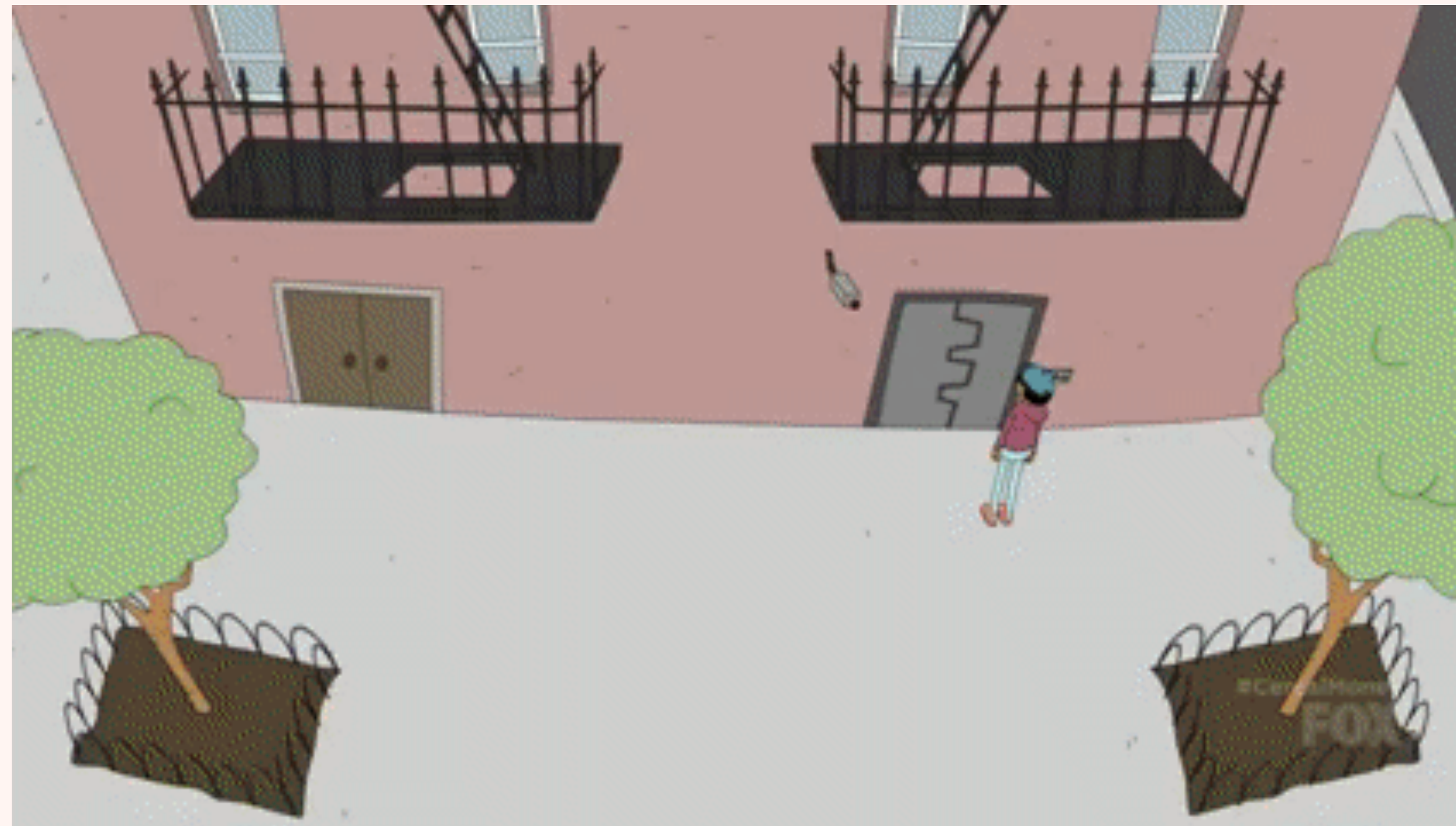
ANIMATED GIF FROM MP4

- `convertImage movie -f covid19_conus_cases_04072020.mp4`
- **input: 1.8M; output: 32M**



ANIMATED GIF FROM YOUTUBE

- A busy scene from the animated show, Lucas Bros. Moving Co.
- `convertImage youtube -u "https://www.youtube.com/watch?v=R-pmYwr8zbU" -o "lucas_bros.gif" -q high.`



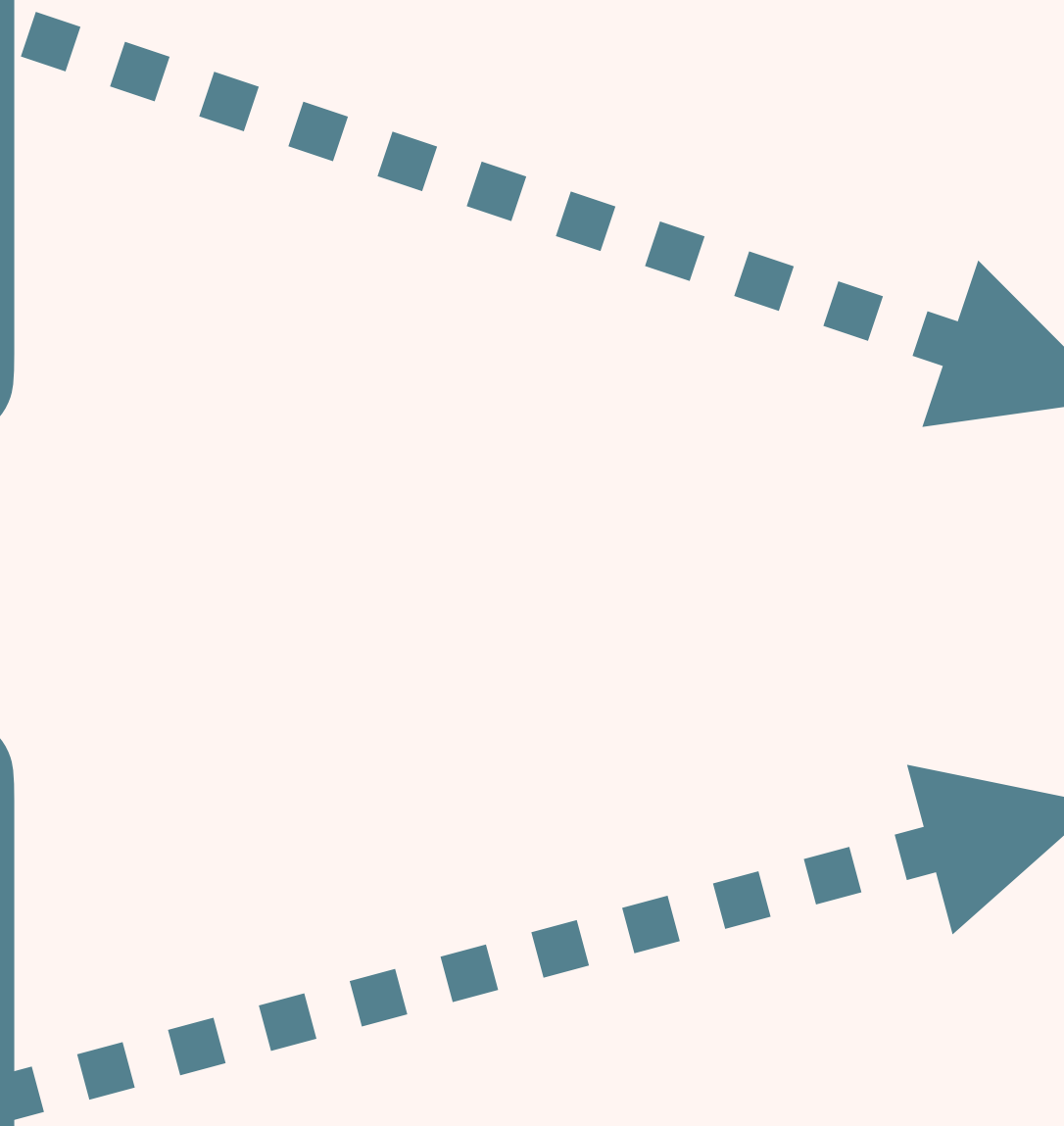
MAKING MUSIC

I USE PLEX_MUSIC_SONGS TO DOWNLOAD MUSIC

**METADATA (FROM
MUSICBRAINZ)**

**AUDIO (FROM
YOUTUBE-DL)**

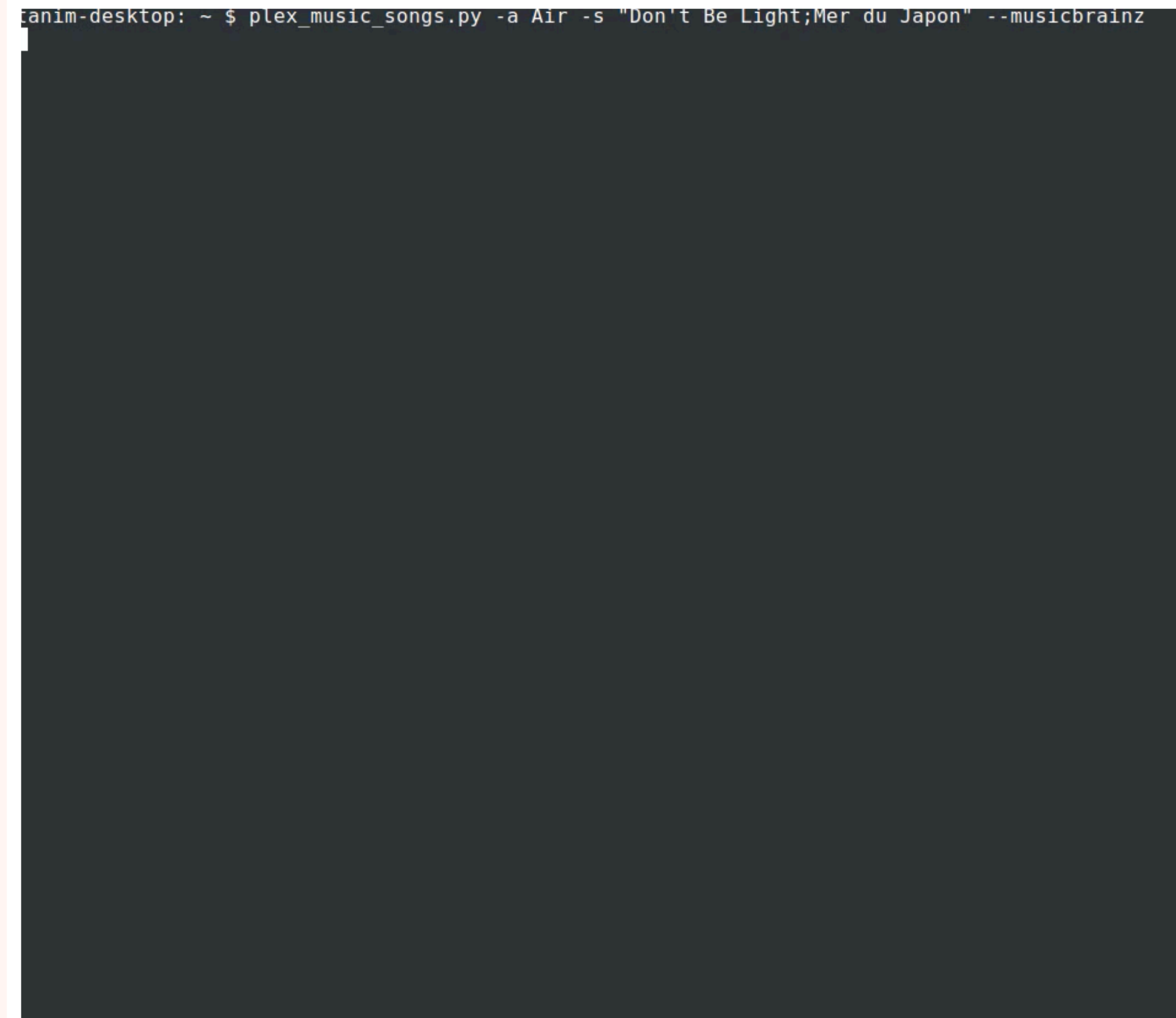
**M4A FILE (WITH ALL
METADATA)**



I USE PLEX_MUSIC_SONGS TO DOWNLOAD MUSIC

- **plex_music_songs** lives in [PLEXSTUFF](#) repository.
- Its documentation [lives here](#).

```
anim-desktop: ~ $ plex_music_songs.py -a Air -s "Don't Be Light;Mer du Japon" --musicbrainz
```



RESOURCES

RESOURCES IN THIS PRESENTATION

- **NPRSTUFF**: `python3 -m pip install --user git+https://github.com/tanimislam/nprstuff.git#egginfo=nprstuff`
 - **PLEXSTUFF**: `python3 -m pip install --user git+https://github.com/tanimislam/plexstuff.git#egginfo=plexstuff`
 - **COVID19_STATS**: `python3 -m pip install -user git+https://github.com/tanimislam/covid19_stats.git#egginfo=covid19_stats`
 - **GITHUB GIST THAT MAKES MOVIES FROM IMAGES**
-