

Stack Overflow is a community of 4.7 million programmers, just like you, helping each other.

Join them; it only takes a minute:

Sign up

Join the Stack Overflow community to:

Ask programming questions

Answer and help your peers

Get recognized for your expertise

Programmatically generate video or animated GIF in Python?

I have a series of images that I want to create a video from. Ideally I could specify a frame duration for each frame but a fixed frame rate would be fine too. I'm doing this in wxPython, so I can render to a wxDC or I can save the images to files, like PNG. Is there a Python library that will allow me to create either a video (AVI, MPG, etc) or an animated GIF from these frames?

Edit: I've already tried PIL and it doesn't seem to work. Can someone correct me with this conclusion or suggest another toolkit? This link seems to backup my conclusion regarding PIL: <http://www.somethinkodd.com/oddthinking/2005/12/06/python-imaging-library-pil-and-animated-gifs/>

[python](#) [video](#) [wxpython](#) [animated-gif](#)

edited Apr 15 '09 at 19:59

asked Apr 15 '09 at 18:57



[FogleBird](#)

33.9k

12

83

112

12 Answers

As of June 2009 the originally cited blog post has a method to create animated GIFs [in the comments](#). Download the script [images2gif.py](#) (formerly [images2gif.py](#), update courtesy of [@geographika](#)).

Then, to reverse the frames in a gif, for instance:

```
#!/usr/bin/env python
```

```
from PIL import Image, ImageSequence
import sys, os
filename = sys.argv[1]
im = Image.open(filename)
original_duration = im.info['duration']
frames = [frame.copy() for frame in ImageSequence.Iterator(im)]
frames.reverse()
```

```
from images2gif import writeGif
writeGif("reverse_" + os.path.basename(filename), frames,
duration=original_duration/1000.0, dither=0)
```

edited May 24 '13 at 5:57

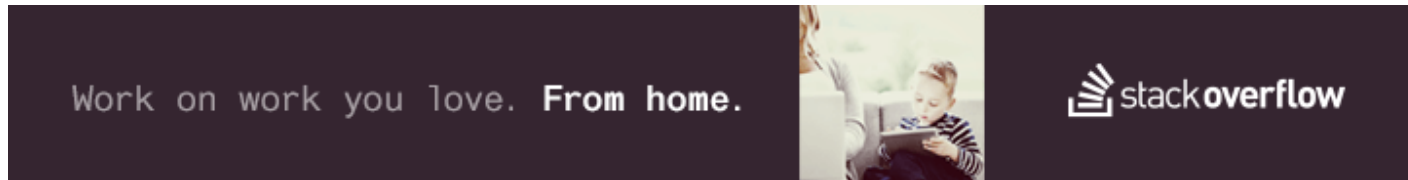
answered Mar 4 '10 at 0:13



kostmo

3,483 2 25 41

-
- 2 There is a new version of this script that makes much better quality output at visvis.googlecode.com/hg/vvmovie/images2gif.py it can be used as a standalone script separate from the package. – [geographika](#) May 9 '12 at 11:19
-
- 1 The script mentioned in this comment consistently gives a segmentation fault for me when used on Mac, even when simply run (using the `name__=='__main'` example). I'm trying the script mentioned in the answer, in hopes that it will work properly. EDIT - I can confirm that the script referenced in the answer above works correctly on my Mac. – [scubbo](#) May 22 '13 at 21:25
-
- 4 Rather than just download the script use pip e.g. `pip install visvis`, then in your script `from visvis.vvmovie.images2gif import writeGif`. – [boyfarrell](#) Jun 26 '13 at 0:56
-
- 8 I tried this with Python 2.7.3 on windows 8 and I get `UnicodeDecodeError: 'ascii' codec can't decode byte 0xc8 in position 6: ordinal not in range(128)`. From running `python images2gif.py` – [reckoner](#) Jan 16 '14 at 21:49
-
- 2 I am the author of visvis (and images2gif) and recommend against using it for this purpose. I've been working on a better solution as part of the imageio project (see my answer). – [Almar](#) Mar 11 at 15:20
-



Well, now I'm using ImageMagick. I save my frames as PNG files and then invoke ImageMagick's convert.exe from Python to create an animated GIF. The nice thing about this approach is I can specify a frame duration for each frame individually. Unfortunately this depends on ImageMagick being installed on the machine. They have a Python wrapper but it looks pretty crappy and unsupported. Still open to other suggestions.

answered Apr 15 '09 at 21:34



[FogleBird](#)

33.9k 12 83 112

12 I'm a Python guy but found ImageMagick much easier here. I just made my sequence of images and ran something like `convert -delay 20 -loop 0 *jpg animated.gif` – [Nick](#) Apr 3 '14 at 1:41

I agree, this is the best solution that I've come across. Here's a minimal example (based on the user Steve B's example code posted at stackoverflow.com/questions/10922285/): pastebin.com/JJ6ZuXdz – [andreasdr](#) Nov 4 '14 at 19:42

I used [images2gif.py](#) which was easy to use. It did seem to double the file size though..

26 110kb PNG files, I expected $26 \times 110\text{kb} = 2860\text{kb}$, but my_gif.GIF was 5.7mb

Also because the GIF was 8bit, the nice png's became a little fuzzy in the GIF

Here is the code I used:

```
__author__ = 'Robert'
from images2gif import writeGif
from PIL import Image
import os

file_names = sorted((fn for fn in os.listdir('.') if fn.endswith('.png')))
#['animationframa.png', 'animationframb.png', 'animationframc.png', ...]
```

```
images = [Image.open(fn) for fn in file_names]

print writeGif.__doc__
# writeGif(filename, images, duration=0.1, loops=0, dither=1)
#   Write an animated gif from the specified images.
#   images should be a list of numpy arrays of PIL images.
#   Numpy images of type float should have pixels between 0 and 1.
#   Numpy images of other types are expected to have values between 0 and 255.

#images.extend(reversed(images)) #infinite loop will go backwards and forwards.

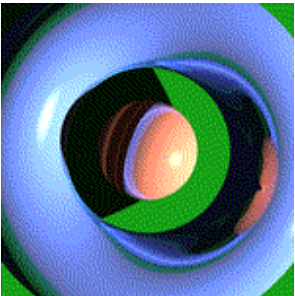
filename = "my_gif.GIF"
writeGif(filename, images, duration=0.2)
#54 frames written
#
#Process finished with exit code 0
```

Here are 3 of the 26 frames:



shrinking the images reduced the size:

```
size = (150,150)
for im in images:
    im.thumbnail(size, Image.ANTIALIAS)
```



edited Apr 29 '12 at 23:05

answered Apr 29 '12 at 22:51



robert king

6,974 2 32 64

I made a blog post about this.. robert-king.com/#post2-python-makes-gif – robert king Apr 30 '12 at 4:30

2 I get errors .. File "C:\Python27\lib\images2gif.py" , line 418, in writeGifToFile globalPalette = palettes[occur.index(max(occur))] ValueError: max() arg is an empty sequence – Harry Oct 23 '12 at 1:58

occur is probably empty. My images2gif.py file has no "globalPalette" variable. – robert king Oct 23 '12 at 2:07

how do I change that? I'm using the most recent images2gif.py script out there (bit.ly/XMMn5h) – Harry Oct 23 '12 at 3:00

4 @robertking with the code I got an error saying fp.write(globalPalette) TypeError: must be string or buffer, not list – LWZ Aug 20 '13 at 1:13

To create a video, you could use [opencv](#),

```
#Load your frames
frames = ...
#create a video writer
writer = cvCreateVideoWriter(filename, -1, fps, frame_size, is_color=1)
#and write your frames in a loop if you want
cvWriteFrame(writer, frames[i])
```

answered Jul 30 '09 at 4:29



attwad

591 5 20

I'd recommend not using images2gif from visvis because it has problems with PIL/Pillow and is not actively maintained (I should know, because I am the author).

Instead, please use imageio, which was developed to solve this problem and more, and is intended to stay.

Quick and dirty solution:

```
import imageio
images = []
for filename in filenames:
    images.append(imageio.imread(filename))
imageio.mimsave('/path/to/movie.gif', images)
```

For longer movies, use the streaming approach:

```
import imageio
with imageio.get_writer('/path/to/movie.gif', mode='I') as writer:
    for filename in filenames:
        image = imageio.imread(filename)
        writer.append_data(image)
```

answered Mar 11 at 15:19



[Almar](#)

281 2 7

1 Link to imageio -> pypi.python.org/pypi/imageio – [fedmich](#) Jun 10 at 13:07

It's not a python library, but mencoder can do that: [Encoding from multiple input image files](#). You can execute mencoder from python like this:

```
import os

os.system("mencoder ...")
```

answered Apr 15 '09 at 22:24



[willurd](#)

3,760 5 20 21

Have you tried [PyMedia](#)? I am not 100% sure but it looks like [this tutorial example](#) targets your problem.

[edited May 7 '09 at 6:59](#)

answered Apr 22 '09 at 15:08

[Jakub Šturc](#)



17.3k

18

68

95

With windows7, python2.7, opencv 3.0, the following works for me:

```
import cv2
import os

vww =
cv2.VideoWriter('mymovie.avi',cv2.VideoWriter_fourcc('X','V','I','D'),24,(640,480))
frameslist = os.listdir('.\\frames')
howmanyframes = len(frameslist)
print('Frames count: '+str(howmanyframes)) #just for debugging

for i in range(0,howmanyframes):
    print(i)
    theframe = cv2.imread('.\\frames\\'+frameslist[i])
    vww.write(theframe)
```

edited Jan 1 at 12:23



Milap

3,539 5 13 37

answered Jan 1 at 12:02



Jacek Słoma

46 3

The matplotlib user manual has a solution example using mencoder.

http://matplotlib.sourceforge.net/faq/howto_faq.html#make-a-movie

answered Aug 23 '12 at 13:38



Daniele

21 1

The task can be completed by running the two line python script from the same folder as the sequence of picture files. For png formatted files the script is -

```
from scitools.std import movie
```



```
movie('*.png',fps=1,output_file='thisismygif.gif')
```

answered Feb 2 '15 at 10:07



ArKE

41 5

1 Tried it... didn't work for me under Python 2.6. Returned: "scitools.easyviz.movie function runs the command: / convert -delay 100 g4testC_*.png g4testC.gif / Invalid Parameter - 100" – [Dan H](#) Aug 13 '15 at 19:31

Problem is not with Python for sure. Reinstall imagemagick on your system and retry. – [ArKE](#) Oct 14 '15 at 9:19

Old question, lots of good answers, but there might still be interest in another alternative...

The `numpngw` module that I recently put up on github (<https://github.com/WarrenWeckesser/numpngw>) can write animated PNG files from numpy arrays. (*Update:* `numpngw` is now on pypi: <https://pypi.python.org/pypi/numpngw>.)

For example, this script:

```
import numpy as np
import numpngw

img0 = np.zeros((64, 64, 3), dtype=np.uint8)
img0[:,32, :] = 255
img1 = np.zeros((64, 64, 3), dtype=np.uint8)
img1[32:, :32, 0] = 255
img2 = np.zeros((64, 64, 3), dtype=np.uint8)
img2[32:, 32:, 1] = 255
img3 = np.zeros((64, 64, 3), dtype=np.uint8)
img3[:,32, 2] = 255
seq = [img0, img1, img2, img3]
for img in seq:
    img[16:-16, 16:-16] = 127
    img[0, :] = 127
    img[-1, :] = 127
    img[:, 0] = 127
    img[:, -1] = 127

numpngw.write_apng('foo.png', seq, delay=250, use_palette=True)
```

creates:



You'll need a browser that supports animated PNG to see the animation. Firefox does, Safari doesn't, and Chrome has a plugin for it.

edited Nov 14 '15 at 1:32

answered Oct 6 '15 at 19:55



Warren Weckesser

35.5k 4 38 62

Like Warren said *last year*, this is an old question. Since people still seem to be viewing the page, I'd like to redirect them to a more modern solution. Like blakev said [here](#), there is a Pillow example on [github](#).

```
import ImageSequence
import Image
import gifmaker
sequence = []

im = Image.open(...)

# im is your original image
frames = [frame.copy() for frame in ImageSequence.Iterator(im)]

# write GIF animation
fp = open("out.gif", "wb")
gifmaker.makedelta(fp, frames)
fp.close()
```

answered Jun 11 at 0:46



Nate

1 1

