Mountain Tapir Collage Maker Documentation

Release 1.1.0

tttppp

Apr 28, 2017

Contents

Contents:

CHAPTER 1

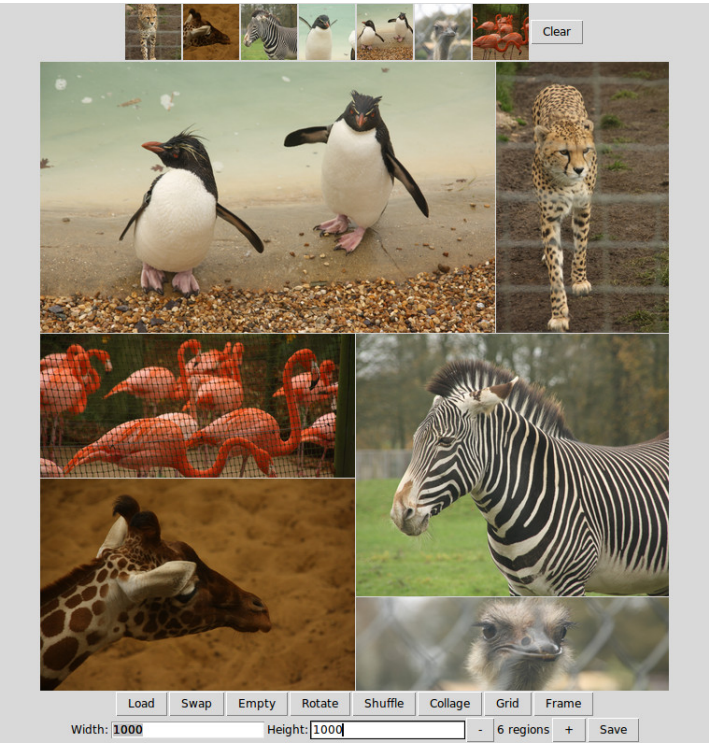
Mountain Tapir Collage Maker

Mountain Tapir Collage Maker is a tool for combining images into collages.

- Free software: GPLv3+ license
- Documentation: https://mountain_tapir.readthedocs.org.

Features

- Arrange images into a collage, and save the result as a new image.
- Graphical interface gives a preview of the finished collage.
- Multiplatform (in theory).



Credits

This package was created with Cookiecutter and the audreyr/cookiecutter-pypackage project template. The photos used in the screenshot, and the screenshot itself are by tttppp and licensed under a Creative Commons Attribution 4.0 International License (cc-by-4.0).

CHAPTER 2

Installation

As a snap:

```
$ snap install mountain-tapir
```

At the command line:

```
$ easy_install mountain_tapir
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv mountain_tapir
$ pip install mountain_tapir
```

CHAPTER 3

Usage

Launching

If you installed the snap, then you can launch with:

```
mountain-tapir
```

To launch Mountain Tapir Collage Maker from the command line:

```
python -m mountain_tapir.mountain_tapir
```

Assuming Mountain Tapir has write access then a configuration file will be stored in:

```
[user home]/.mountain_tapir/mountain_tapir.properties
```

User Interface

At the top of the UI is a section containing recently used images. Initially this just contains a button labelled "Clear", but once a few images have been loaded then thumbnails can be selected here to allow filling regions by clicking.

The middle of the UI contains a preview of what the finished collage will look like. When a region is empty then it has a random solid colour to help distinguish it from neighbouring regions. This solid colour block will not appear in the final output.

At the bottom of the UI are several buttons representing different tools and methods of changing the layout of the collage.

Tools

Load: This tool is selected when the application is first started, and causes a file selection dialog to appear when clicking on a region. The selected file (assuming it's an image file) will be loaded into that region.

Swap: When this tool is selected then the images in two regions can be swapped by clicking on one then the other.

Empty: Using this tool the current image in a region can be removed by clicking on it.

Rotate: When this tool is selected then clicks on an image will cause it to be rotated by ninety degrees. It will also cause any copies of the image to be rotated by ninety degrees too (a "copy" is created using the thumbnails at the top, using the Load button will create a separate version of the image).

Shuffle: Clicking this button will randomly rearrange the images within the regions (while keeping the layout the same).

Layout

Collage: Clicking this button converts the collage to the default "Collage" mode - i.e. a slightly random arrangement of rectangles. When in collage mode this button can be used to create a new arrangement of regions.

Grid: Clicking this button converts the collage to a grid of regions. All regions will be the same width and height.

Frame: This button generates a collage with a central picture surrounded by a border of other images. Note that if there are fewer than five regions then this uses grid mode instead.

Width and Height: These settings control the resolution in pixels of the finished collage.

-/+ regions: This controls the number of regions in the collage.

Save: This generates the finished collage, and saves it to a specified file. The default file extension is "*.jpg", but other file formats can be chosen too.

CHAPTER 4

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

Report Bugs

Report bugs at https://github.com/tttppp/mountain_tapir/issues.

If you are reporting a bug, please include:

- Your operating system name and version.
- Your installation method.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

Write Documentation

Mountain Tapir Collage Maker could always use more documentation, whether as part of the official Mountain Tapir Collage Maker docs, in docstrings, or even on the web in blog posts, articles, and such.

Submit Feedback

The best way to send feedback is to file an issue at https://github.com/tttppp/mountain_tapir/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

Get Started!

Ready to contribute? Here's how to set up *mountain_tapir* for local development.

1. Fork the *mountain_tapir* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/mountain_tapir.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv mountain_tapir
$ cd mountain_tapir/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

   Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 --max-line-length=120 mountain_tapir tests
$ python setup.py test
$ tox
```

   To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. You may want to include a test for any new functionality or fixed bug.
2. If the pull request adds functionality, consider if the docs need updating. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.7, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/tttppp/mountain_tapir/pull_requests and make sure that the tests pass for all supported Python versions.

Tips

To run a subset of tests:

```
$ python -m unittest tests.test_mountain_tapir
```

CHAPTER 5

Credits

Development Lead

- tttppp <tttppp@users.noreply.github.com>

Contributors

None yet. Why not be the first?

CHAPTER 6

History

1.1.0 (2017-04-27)

- New open image dialog containing preview thumbnails.

1.0.4 (2017-03-23)

- Add icons for algorithm buttons.
- Minor fix for shuffle for Python 3.

1.0.3 (2017-02-22)

- Maintain the selected images when changing layout and number of regions.
- Remove support for Python 2.6.
- Fix button to clear recent images.

1.0.2 (2017-02-13)

- Keep the same images loaded when changing algorithm.
- Playing around with source code to make it look prettier.

1.0.1 (2017-02-06)

- Update snap to be stable release.

- Fix a minor issue with open file dialog.

1.0.0 (2017-02-05)

- Create snapcraft file to allow easier installation.
- Slightly larger icon.
- Minor fix to documentation.

0.2.0 (2017-02-01)

- Support for rotating images by ninety degrees (also pi by two radians).
- Persistance of user specified settings in home directory file (e.g. last visited directory, algorithm, etc.).
- Add a default file extension of jpg.
- Various bug fixes.

0.1.2 (2016-10-24)

- Implement new frame algorithm for creating collages based around a central image.
- Fix bug with collage mode so that regions are more likely to have a sensible aspect ratio.
- Various fixes to make code more compatible with Python 2.6 and Python 3.

0.1.1 (2016-04-02)

- Updates to make build pass for Python 3.

0.1.0 (2016-03-25)

- First release on PyPI.

CHAPTER 7

Indices and tables

- genindex
- modindex
- search