

GenGif: Python Program for Generating Animated GIFs

An animated GIF (Graphics Interchange Format) is a file format that allows for the storage of multiple images or frames in a single file, which can be played in a loop to create an animation.

Precondition for Each Program Run

Directory Structure

1. Input Directory (`jpg_dir`):

- **Path:** `./img`
- **Content:** This directory contains image files in JPEG format (`.jpg`) that will be used to create the animated GIF.

2. **Reference Directory:

- **Path:** `./refimg`
- **Content:** This directory contains the reference image that will be used to align the input images and also determines the portion of the image to be included in the GIF.

3. Output Directory:

- **Path:** `./result`
- **Content:** The generated animated GIF will be saved in this directory under the name `output.gif`.

Summary

To successfully run the program, ensure that the directories mentioned above exist and that the corresponding image files are present in their respective directories.

Functionality of the Program

The program `gengif` is designed to create an animated GIF from a series of images by aligning them and cropping to a common area. The program does not have any input parameters.

Overview

1. Imports and Logging:

- The program imports necessary libraries: `os`, `imageio`, `numpy`, `cv2` (OpenCV), and `PIL` (Pillow).
- It sets up logging to record the process and outputs messages to a log file named `gengif.log`.

2. Functions:

- `align_images(image1, image2)`: This function aligns two images using feature matching. It converts the images to grayscale, detects keypoints and descriptors using the ORB (Oriented FAST and Rotated BRIEF) algorithm, and matches these features. It then computes a homography matrix to warp the second image to align with the first.

- `crop_image(image, crop_box)`: This function crops an image to a specified rectangular area defined by `crop_box`. It converts the image to a PIL format for cropping and then converts it back to OpenCV format.
- `find_common_area(images)`: This function determines the largest common area across all aligned images. It creates a binary mask for each image, finds the intersection of these masks, and computes the bounding rectangle of the intersection area.

3. Main Execution:

- The program defines input and reference image directories (`jpg_dir` and `ref_img_dir`).
- It logs the absolute paths of these directories and lists the input files.
- It reads a reference image from the reference directory or selects a middle image from the input directory if no reference images are found.
- It iterates through the list of input images, aligns each image with the reference image, and stores the aligned images in a list.

4. Cropping and GIF Creation:

- After aligning all images, the program finds the largest common area using the `find_common_area` function.
- It crops all aligned images to this common area.
- Finally, it saves the cropped images as an animated GIF in the `./result` directory with a specified duration between frames.

Log File

- The program creates a log file named `gengif.log`, which records information about the program's execution, including the input and output file paths.

Summary

The `gengif.py` program effectively processes a set of images by aligning them, determining a common cropping area, and generating an animated GIF. It utilizes feature matching and image processing techniques to ensure that the final output is visually coherent and focused on the relevant content across all images. The use of logging helps track the process and any issues that may arise during execution.

Installation

The ZIP archive `gengif.zip` contains a fully functional working directory, structured as described above, along with sample data. Its contents can be placed in any location.

- `gengif.exe`: An executable file with no external dependencies, generated using *PyInstaller* with the command `pyinstaller --onefile 'gengif.py'`.
- `readme.md`: This file.
- `gengif.py`: Python source code.
- `gengif.ini`: Settings (only one parameter: `duration`). The duration parameter in the `gengif.ini` file specifies the time each frame is displayed in the animated GIF, measured in milliseconds. This parameter allows you to control the speed of the animation. For example, a duration of 400 means each frame will be displayed for 400 milliseconds before moving to

the next frame.

Usage

Capture Input Images

Images should be captured in the chronological order of the movement sequence, ensuring that all other factors remain as consistent as possible aside from the movement:

- Same image crop
- Same focus plane
- Same exposure
- Same color balance

The alphabetical order of the image file names should correspond to the chronological sequence.

Create GIF

Before creating the GIF, delete any existing images in the *img* and *refimg* folders.

Next, copy the captured images into the *img* folder.

Select the middle image of the sequence as the reference image and copy it into the *refimg* folder (only one image!). This reference image will also determine the crop for the GIF. The crop should be taken from the central part of the image, excluding the border area to avoid artifacts.

The reference image is utilized in the processing as follows:

1. Each image is aligned with the reference image.
2. The common rectangular area shared by all images is selected, and each image is cropped accordingly.

The aligned and cropped images are then used as frames in the animated GIF.