Fundamentals of Image Processing 2020 P. Fasouli, F. Mitterer

Project
Cropper/Annotator

Application flow

- The program receives input/output directory on start
- Check if the directories exist
- Load the first image
- Register mouse event handler
- Display the first image
- Start application loop, waiting for user input
- Reacting to user input
  - Selecting cropping area (and accept annotation)
  - Moving to next/previous image
  - Crop and store to disk
  - Quit

Pressing the mouse button down starts the cropping area selection, the mouse can be moved to the desired location. On letting the left mouse button go, the selection will end. A rectangle will help to visualize the cropping area during and after selection. The area that lies outside of the cropping area gets darkened as soon as the mouse button is lifted.

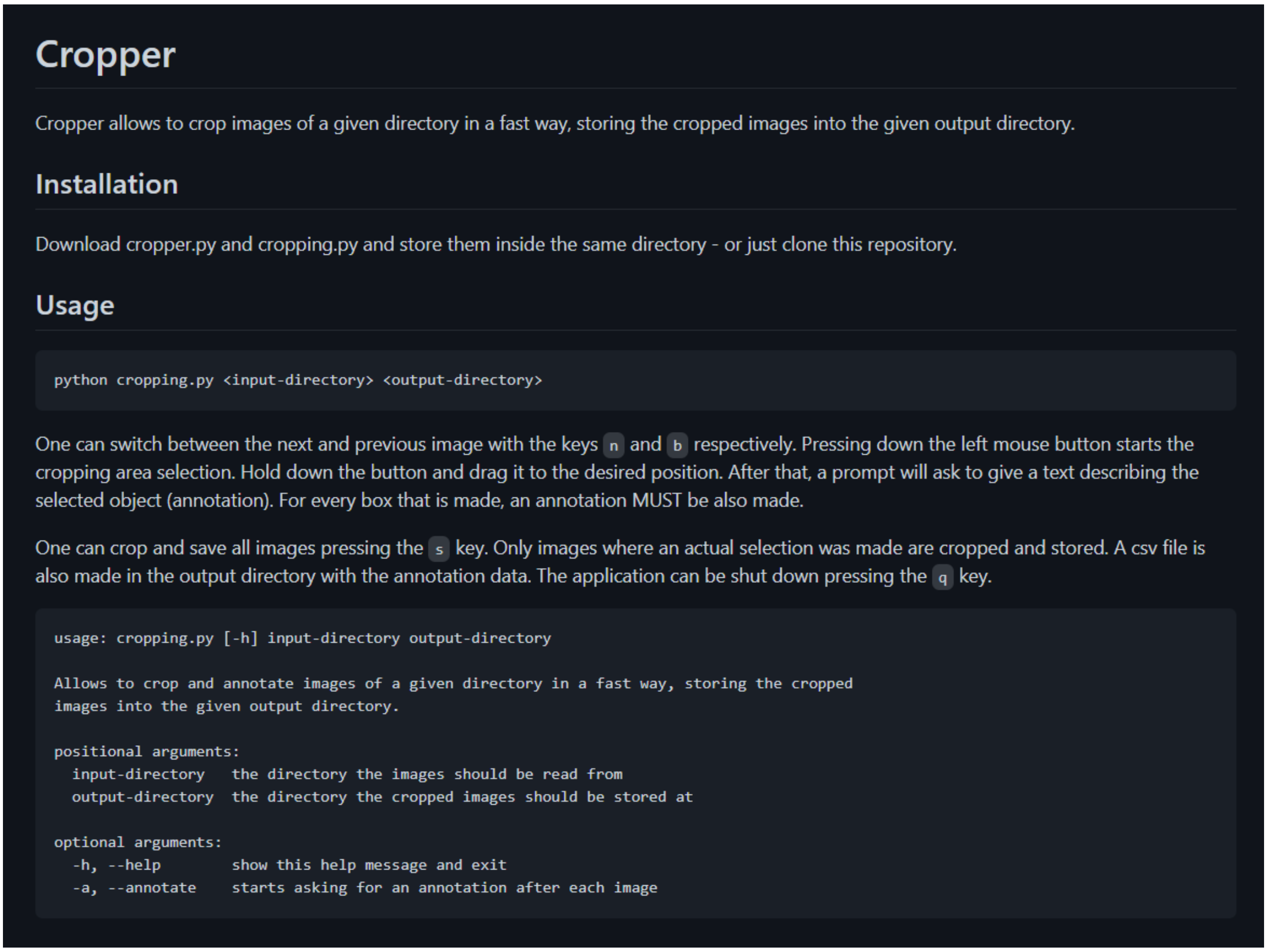
If annotation mode is enabled, an annotation for the selection has to be typed into the command line (a graphical textbox or message box would have required a GUI-library – but we preferred to only keep using OpenCV).

The user can navigate to the next or previous image by pressing the N or B key respectively or quit the application with the Q key.
Pressing the S key crops and saves all images where a selection has been made. If annotation mode is enabled, a CSV file containing information about the cropped image and associated annotation is created and written to the output directory as well.

For the usage, please see the README. The link to the GitHub repository is as follows:
https://github.com/fischly/FoIP-2020/tree/master/project
We attached a screenshot of the README file to this document as well.

Each step is commented in the code, we tried to make as many comments as we can.

README.md



Cropper

Cropper allows to crop images of a given directory in a fast way, storing the cropped images into the given output directory.

Installation

Download cropper.py and cropping.py and store them inside the same directory - or just clone this repository.

Usage

```
python cropping.py <input-directory> <output-directory>
```

One can switch between the next and previous image with the keys `n` and `b` respectively. Pressing down the left mouse button starts the cropping area selection. Hold down the button and drag it to the desired position. After that, a prompt will ask to give a text describing the selected object (annotation). For every box that is made, an annotation MUST be also made.

One can crop and save all images pressing the `s` key. Only images where an actual selection was made are cropped and stored. A csv file is also made in the output directory with the annotation data. The application can be shut down pressing the `q` key.

```
usage: cropping.py [-h] input-directory output-directory

Allows to crop and annotate images of a given directory in a fast way, storing the cropped
images into the given output directory.

positional arguments:
  input-directory   the directory the images should be read from
  output-directory  the directory the cropped images should be stored at

optional arguments:
  -h, --help        show this help message and exit
  -a, --annotate    starts asking for an annotation after each image
```