Image Tagging

Sloth

Setup

In order to tag images, we use a piece of software called Sloth. To install it from the robosub debian repository, run the following command:

```
sudo aptitude install sloth
```

Now that you have sloth, you need to clone the vision_dev repository. To do that, run this command wherever you want to clone it:

Using an SSH key (Do this if you have completed the Getting Started setup):

```
git clone git@github.com:PalouseRobosub/vision_dev.git
```

Without an SSH key:

```
git clone https://github.com/PalouseRobosub/vision_dev.git
```

Using Sloth

To use sloth and start tagging images, you can run the following command.

```
sloth -c /path/to/vision_dev/sloth/robosub_config.py
/path/to/annotation/file
```

-c Is a flag to give the path to a configuration file. This file is provided in the vision_dev/sloth directory. The path you provide should point to this file.

The last argument is a path to an annotation file. This is most likely named something like `labels.json`. You will need to provide this path in order to tag images.

Should you find this tedious, there is a script in the sloth directory of the vision_dev repository which performs some of this for you. It can be used as follows:

```
./robosub_sloth.sh /path/to/annotation/file
```

This removes the need to add the -c flag repeatedly. This script can also be symlinked to without issues.

Sloth alias

You can set up an alias in your bashrc. To do so run (replace "~/vision_dev" with path to your vision_dev repo):

```
echo "alias sloth='~/vision_dev/sloth/./robosub_sloth.sh'" >> ~/.bashrc
source ~/.bashrc
```

Sloth Keybindings

The full list of keybindings used in sloth can be found in the robosub_config.py file near the bottom. A shorthand list is provided below

Default keybindings

- **Space**   Mark image as labeled/confirmed and go to next
- **Backspace**   Next image/frame
- **PgDown**   Previous image/frame
- **PgUp**   Previous image/frame
- **Tab**   Select next annotation
- **Shift+Tab**   Select previous annotation
- **Ctrl+f**   Fit current image/frame into window
- **Del**   Delete selected annotations
- **Esc**   Exit insert mode
- **Shift+l**   Mark current image as labeled
- **Shift+c**   Mark all annotations in image as confirmed

Robosub Specific

- **F5**   Toggle the visibility of label names on annotation boxes
- **Ctrl+Shift+Del**   Delete all annotations from the current image and mark it as unlabeled
- **c**   Copy all annotations from the previous image to this one
- Mark image as labeled/confirmed and copy annotations to next image.
- **n**   (Equivalent to **Space** then **c**)

Annotations

- **s**   Start Gate Post
- **p**   Path Marker
- **t**   Torpedo Target
- **w**   Roulette Wheel
- **1**   Die 1
- **2**   Die 2
- **3**   Die 3

- **4** Die 4
- **5** Die 5
- **6** Die 6

Sloth Mouse Controls

While creating annotations, the following are useful mouse controls.

- **Right Click and Drag**

  Resize an annotation. Resizing is based upon the quadrant of the annotation clicked on.

- **Ctrl+Left Click**

  Select multiple annotations at once.

Data Management

For Linux Users

Getting, validating, and returning labeling data is handled through the `rslabel` utility program. It currently only supports python 2.x versions. To install it, run

```
sudo pip install rslabel
```

To update rslabel for new features run

```
sudo pip install --upgrade rslabel
```

There are a number of commands to be used with `rslabel`, including `show`, `get`, `return`, `upload`, and `collect`.

Command	Description
`rslabel show`	Provides information about the number of datasets labeled, number of images properly validated, and counts any labeling sessions in progress.
`rslabel get`	Grabs an image dataset for labeling and places it in your current directory. The `--validation` flag can be supplied to get labeled data to validate. The `--clarification` flag can be supplied to get labeled data to clarify. However, validation and clarification are done only by few assigned people in charge of object detection
`rslabel return [JSON]`	Returns a dataset to the server. If the data has not been completely labeled or validated, it will be returned for someone else to complete in the future.

Command	Description
`rslabel upload [ROSBAG]`	Takes a ROS bag file and break the images out into datasets for labeling. It will then upload the files to the server for labeling.
`rslabel collect`	Collects all of the labeled and validated images into a single dataset for use with object detection training.
`rslabel mark`	A tool which is used for validating data, highlights the box which is there.
`rslabel stats`	This command will show scoreboard table: how many images labeled, how many labels added, how many images validated and how many labels validated.

For Windows Users

People who want to label on Windows you can use this tool to check, download and upload labels. To use this just extract the folder in zip anywhere and run `WindowsSftp.exe`, remember that `.dll` file must be in the same folder as `.exe`

robosub-windows-tools.zip

From:
https://robosub.eecs.wsu.edu/wiki/ - **Palouse RoboSub Technical Documentation**

Permanent link:
https://robosub.eecs.wsu.edu/wiki/cs/vision/image_tagging/start?rev=1526679831

Last update: **2018/05/18 14:43**