Python-Relion Documentation

Release 0.1.0

Diamond Light Source - Scientific Software

Sep 11, 2020

CONTENTS:

1 Usage **3**

2 API **5**
2.1 Project object 5
2.2 Stage-specific information 5

3 Credits **9**

Index **11**

Documentation: https://python-relion.readthedocs.io.

This package provides a python interface to the information contained in a Relion project folder. It does not run Relion itself.

Currently it caters for specific fields from the Motion Correction, CTF Find, 2D Classification and 3D Classification stages of the Relion pipeline, but this could readily be expanded to more stages and fields.

CHAPTER

ONE

USAGE

To access a Relion project folder you first need to create a `relion.Project` object (c.f. *API* for more information):

```
import relion
proj = relion.Project("/path/to/relion/project/directory")
proj = relion.Project(pathlib.Path("/project/directory"))  # path objects are␣
↪supported
```

The directory structure inside a Relion directory is built up of stages and jobs. Each stage folder will contain one or more job folders. The job folder(s) contain files related to the stage, including the `*.star` files from which values can be read:

```
project_root
│
├── MotionCorr
│   └── job002
│       └── corrected_micrographs.star
│       └── ...
├── CTFFind
│   └── job003
│       └── micrographs_ctf.star
│       └── ...
├── Class2D
│   ├── job008
│   │   └── run_it025_data.star
│   │   └── run_it025_model.star
│   │   └── ...
│   └── job013
│       └── run_it_025_data.star
│       └── run_it_025_model.star
│       └── ...
└── Class3D
    └── job016
        └── run_it_025_data.star
        └── run_it_025_model.star
        └── ...
```

The desired EM values are extracted from *.star files. For example, a snippet from `MotionCorr/job002/corrected_micrographs.star` is shown below:

```
...
loop_
_rlnCtfPowerSpectrum #1
_rlnMicrographName #2
_rlnMicrographMetadata #3
```

(continues on next page)

(continued from previous page)

```
_rlnOpticsGroup #4
_rlnAccumMotionTotal #5
_rlnAccumMotionEarly #6
_rlnAccumMotionLate #7
MotionCorr/job002/Movies/20170629_00021_frameImage_PS.mrc MotionCorr/job002/Movies/
↪20170629_00021_frameImage.mrc MotionCorr/job002/Movies/20170629_00021_frameImage.
↪star            1    16.420495     2.506308    13.914187
MotionCorr/job002/Movies/20170629_00022_frameImage_PS.mrc MotionCorr/job002/Movies/
↪20170629_00022_frameImage.mrc MotionCorr/job002/Movies/20170629_00022_frameImage.
↪star            1    19.551677     2.478968    17.072709
MotionCorr/job002/Movies/20170629_00023_frameImage_PS.mrc MotionCorr/job002/Movies/
↪20170629_00023_frameImage.mrc MotionCorr/job002/Movies/20170629_00023_frameImage.
↪star            1    17.547827     1.941103    15.606724
MotionCorr/job002/Movies/20170629_00024_frameImage_PS.mrc MotionCorr/job002/Movies/
↪20170629_00024_frameImage.mrc MotionCorr/job002/Movies/20170629_00024_frameImage.
↪star            1    18.100817     1.722567    16.378250
...
```

To access the `_rlnAccumMotionTotal` column in this file you can use:

```
>>> [micrograph.total_motion for micrograph in proj.motioncorrection["job002"]]
['16.420495', '19.551677', '17.547827', '18.100817', ...]
```

Stages are dictionary-like objects, so can discover the list of all known jobs by:

```
>>> list(proj.class2D)
['job008', 'job013']
```

and use the other standard dictionary accessors (`.values()`, `.keys()`, `.items()`), too. You can also convert the stages into normal dictionaries:

```
>>> dict(p.ctffind)
{'job003': [CTFMicrograph(...), ...]}
```

For a list of supported stages and a list of supported values per stage please have a look at the *API* page.

CHAPTER

TWO

API

2.1 Project object

class relion.**Project**(*path*)

Reads information from a Relion project directory and makes it available in a structured, object-oriented, and pythonic fashion.

property class2D

access the 2D classification stage of the project. Returns a dictionary-like object with job names as keys, and lists of Class2DParticleClass namedtuples as values.

property class3D

access the 3D classification stage of the project. Returns a dictionary-like object with job names as keys, and lists of Class3DParticleClass namedtuples as values.

property ctffind

access the CTFFind stage of the project. Returns a dictionary-like object with job names as keys, and lists of CTFMicrograph namedtuples as values.

property motioncorrection

access the motion correction stage of the project. Returns a dictionary-like object with job names as keys, and lists of MCMicrograph namedtuples as values.

The individual stage accessors `.ctffind`, `.class2D`, etc. return a dictionary-like object that allows you to access individual Relion jobs within that particular stage. The dictionary key names are the relion job names (usually `jobXXX`), the dictionary value is a list of stage-specific named tuples, listed below.

2.2 Stage-specific information

class relion._parser.ctffind.**CTFMicrograph**(*micrograph_name*, *astigmatism*, *defocus_u*, *defocus_v*, *defocus_angle*, *max_resolution*, *fig_of_merit*)

Contrast Transfer Function stage.

property astigmatism

Estimated astigmatism. Units angstrom (A).

property defocus_angle

Estimated angle of astigmatism.

property defocus_u

Averaged with Defocus V to give estimated defocus. Units angstrom (A).

property defocus_v
Averaged with Defocus U to give estimated defocus. Units angstrom (A).

property fig_of_merit
Figure of merit/CC/correlation value. Confidence of the defocus estimation.

property max_resolution
Maximum resolution that the software can detect. Units angstrom (A).

property micrograph_name
Micrograph name. Useful for reference.

class relion._parser.motioncorrection.**MCMicrograph**(*micrograph_name*, *total_motion*, *early_motion*, *late_motion*)
Motion Correction stage.

property early_motion
Early motion.

property late_motion
Late motion.

property micrograph_name
Micrograph name. Useful for reference.

property total_motion
Total motion. The amount the sample moved during exposure. Units angstrom (A).

class relion._parser.class2D.**Class2DParticleClass**(*particle_sum*, *reference_image*, *class_distribution*, *accuracy_rotations*, *accuracy_translations_angst*, *estimated_resolution*, *overall_fourier_completeness*)
2D Classification stage.

property accuracy_rotations
Accuracy rotations.

property accuracy_translations_angst
Accuracy translations angst.

property class_distribution
Class Distribution. Proportional to the number of particles per class.

property estimated_resolution
Estimated resolution.

property overall_fourier_completeness
Overall Fourier completeness.

property particle_sum
Sum of all particles in the class. Gives a tuple with the class number first, then the particle sum.

property reference_image
Reference image.

class relion._parser.class3D.**Class3DParticleClass**(*particle_sum*, *reference_image*, *class_distribution*, *accuracy_rotations*, *accuracy_translations_angst*, *estimated_resolution*, *overall_fourier_completeness*)

3D Classification stage.

property accuracy_rotations
Accuracy rotations.

property accuracy_translations_angst
Accuracy translations angst.

property class_distribution
Class Distribution. Proportional to the number of particles per class.

property estimated_resolution
Estimated resolution.

property overall_fourier_completeness
Overall Fourier completeness.

property particle_sum
Sum of all particles in the class. Gives a tuple with the class number first, then the particle sum.

property reference_image
Reference image.

CHAPTER

THREE

CREDITS

- Anna Horstmann
- Markus Gerstel

INDEX

A

accuracy_rotations() (*relion._parser.class2D.Class2DParticleClass property*), 6
accuracy_rotations() (*relion._parser.class3D.Class3DParticleClass property*), 7
accuracy_translations_angst() (*relion._parser.class2D.Class2DParticleClass property*), 6
accuracy_translations_angst() (*relion._parser.class3D.Class3DParticleClass property*), 7
astigmatism() (*relion._parser.ctffind.CTFMicrograph property*), 5

C

class2D() (*relion.Project property*), 5
Class2DParticleClass (*class in relion._parser.class2D*), 6
class3D() (*relion.Project property*), 5
Class3DParticleClass (*class in relion._parser.class3D*), 6
class_distribution() (*relion._parser.class2D.Class2DParticleClass property*), 6
class_distribution() (*relion._parser.class3D.Class3DParticleClass property*), 7
ctffind() (*relion.Project property*), 5
CTFMicrograph (*class in relion._parser.ctffind*), 5

D

defocus_angle() (*relion._parser.ctffind.CTFMicrograph property*), 5
defocus_u() (*relion._parser.ctffind.CTFMicrograph property*), 5
defocus_v() (*relion._parser.ctffind.CTFMicrograph property*), 5

E

early_motion() (*relion._parser.motioncorrection.MCMicrograph property*), 6
estimated_resolution() (*relion._parser.class2D.Class2DParticleClass property*), 6
estimated_resolution() (*relion._parser.class3D.Class3DParticleClass property*), 7

F

fig_of_merit() (*relion._parser.ctffind.CTFMicrograph property*), 6

L

late_motion() (*relion._parser.motioncorrection.MCMicrograph property*), 6

M

max_resolution() (*relion._parser.ctffind.CTFMicrograph property*), 6
MCMicrograph (*class in relion._parser.motioncorrection*), 6
micrograph_name() (*relion._parser.ctffind.CTFMicrograph property*), 6
micrograph_name() (*relion._parser.motioncorrection.MCMicrograph property*), 6
motioncorrection() (*relion.Project property*), 5

O

overall_fourier_completeness() (*relion._parser.class2D.Class2DParticleClass property*), 6
overall_fourier_completeness() (*relion._parser.class3D.Class3DParticleClass property*), 7

P

particle_sum() (*relion._parser.class2D.Class2DParticleClass property*), 6
particle_sum() (*relion._parser.class3D.Class3DParticleClass property*), 7
Project (*class in relion*), 5

R

reference_image() (*relion._parser.class2D.Class2DParticleClass property*), 6
reference_image() (*relion._parser.class3D.Class3DParticleClass property*), 7

T

total_motion() (*relion._parser.motioncorrection.MCMicrograph property*), 6