EMAN2 box files to Particle Position Assets text file conversion

Mon, 01/15/2018 - 08:16

#1

kushalsejwal

EMAN2 box files to Particle Position Assets text file conversion

Dear All,

I am looking for a simple script or bash command which can batch convert the box files generated in EMAN2 into one big Particle Position Asset text file that can be read by cisTEM accordin the format mentioned in the FAQ (https://cistem.org/documentation#tab-1-3)

Have anybody tried it already? Thanks in advance.

Best,

Kushal

timgrant

Hi Kushal,

We do not have such a script - although I think it shouldn't be too difficult to create.

If you do give it a go, I'm happy to help out with any quesitons you have. If you manage to get one working, please post your results here.

Cheers,

Tim

kushalsejwal

Hi Tim,

With my limited Python programming skills, I came up with the following script, which gets the work done (I am sure there must be more efficient way to do this in fewer lines with Bash commands) and it generates one .txt file that contains the coordinates of all the particles from all micrographs (picked in EMAN2's boxer) but when I check the particle stack generated by CisTEM it doesn't seems to pick the correct particles. I do not see my particles in the particle_stack_0.mrc, it appears to be random backround. Does CisTEM defines the coordinates differently than EMAN2?

Here is the script :

_____

```
import glob

with open('coordinates.txt', 'w') as outfile:
    for filename in glob.glob("*.box"):
        with open(filename) as infile:
            for line in infile:
                outfile.write(filename.replace("box","mrc") + " " + " ".join(line.split("\t")[:2]) + "\n")
```

_____

Best,

Kushal

timgrant

Hi Kushal,

One complication may be that the cisTEM co-ordinates are stored in angstroms, not pixels. I think the EMAN positions may be stored in pixels, so you would need to multiply the position by the pixel size.

Also, I remember that EMAN used to store the co-ordinate of the corner, and then size of the box (I'm not sure if this is still true). cisTEM needs the centre of the box.

Cheers,

Tim

Thu, 04/05/2018 - 03:30 <u>(Reply to #4)</u>

bonniemurphy

Hi,

Hi,

This is a short bash script that seems to have worked for me - maybe it will save others a few minutes. It assumes box files are named like the micrograph, but with _automatch.box in place of .mrc extension (the default naming of output files from Gautomatch). First line makes your list of mics, and removes an @ sign from mic name, since I usually have a softlink to the mic in the particle-picking directory. Don't forget to set the pixel size for your own data; boxsize is read from file, column 3. Cheers, Bonnie

```bash
#!/bin/bash

########A script for creating a text file with micrograph names and box coordinates in Angstrom, with output coordinates for the center of the box

ls *mrc | sed 's/@//' > Allmics.txt

Pix=1.053 #Pixel size in A

for i in `cat Allmics.txt`;

do

echo ${i}

cat ${i%.mrc}_automatch.box | while read line

do

 xp=`echo $line | awk '{ print $1}'`

 yp=`echo $line | awk '{ print $2}'`

 Bs=`echo $line | awk '{ print $3}'`


 xA=`echo "($xp+(0.5*$Bs))*$Pix" | bc -l`

 yA=`echo "($yp+(0.5*$Bs))*$Pix" | bc -l`


 printf "%s\t%.0f\t%.0f\n" $i $xA $yA >> Allmics_boxes.txt
```

done

done

Fri, 04/06/2018 - 14:16 <u>(Reply to #5)</u>

timgrant

Thank you very much for

Thank you very much for posting this!

Tim

Thu, 06/14/2018 - 23:59 <u>(Reply to #6)</u>

jbox

Hey bonniemurphy

I modified your script a little,

just this line

cat ${i%.mrc}_automatch.box | while read line

to

cat ${i%.mrc}.box | while read line

As I picked in eman2 and had a suffux of _sum.box and my corrected micrographs were _sum.mrc

The script seems to execute fine, and produce a list file with the following format

FoilHole_5380712_Data_5377586_5377587_20180306_101_sum.mrc
1043   1734
FoilHole_5380712_Data_5377586_5377587_20180306_101_sum.mrc
2081   2293

When I try to importr these in the Particle Positions tab of the cisTEM GUI, it progresses until the end and then shoots me an error which lists a one line error of

Line 21527, column 1 is not read as a valid Asset ID or filename, and so the line will be ignored

I assume this is because of the format of the txt file. Did you ever encounter this problem?

Tue, 06/19/2018 - 02:36 <u>(Reply to #7)</u>

bonniemurphy

empty line?

Hi,

 I don't think I've ever run into this error. Have you had a look at line 21527 of your text file? Maybe your text file has a blank line at the end?

Does it import successfully?

Cheers,

Bonnie

kushalsejwal

Hi Tim,

Thanks for the reply. Indeed I have to incoroprate these two changes in the script. For storing the coordinates in Angstrom, I simply set the Apix value to 1. When I now import this new coordinate file and generate a new refinement package based on them, I see all my particles in Display Stack GUI. But I still have some issues:

1) In the display stack, the particles are colored back. I have negative stain data and while importing the microgrpahs, I ticked "Particles are white". Is it a bug? Will this affect 2D classfication?

2) When I use this particle stack (~2000 particles) for 2D classfication, after the random start, all the classes look grey (See screensht : https://ibb.co/heQayb)

This is a small negative stain dataset and I do not wish to do the CTF correction.

Best,

Kushal

timgrant

Hi Kushal,

cisTEM expects protein to be black, so if you tick "particles are white" the contrast will be inverted. This is the expected result, so that all sounds fine.

There is no way to turn of CTF correction at the moment. What defocus values do you have for the particles that you imported?

Cheers,

Tim

kushalsejwal

Hi Tim,

I have rather large defocus for negative stain particles and the particle density is very high. I am not targetting 3D and only wish to do 2D classficiation with the dataset.

I did CTF using cisTEM now and subsequently 2d classificaiton seems to work fine.

So in principle the particles coodinates picked with EMAN2 works well with cisTEM.

Thank you and the whole cisTEM team for support and a wonderful software.

Best,

Kushal

Arne

Order of micrographs

Dear Tim and all,

we are trying something similar and it works for a single mircrograph or several if the order is exactly as stored in the database. However, this must not neccessarily always be true. If micrographs had been improted from various folders the order in the database may not correspond to the logical order (for example alphabetical).

Is there a quick way to obtain a list with micrograph id and micrograph name from cisTEM?

this would resolve our issue

many thanks

Arne

Wed, 02/14/2018 - 10:03 <u>(Reply to #12)</u>

Arne

Order of micrographs

Ok I just found that you can also import with the name - dont need the identifier.

Nevertheless, is it possible to generate a list with ID and Micrographname?

cheers

Arne

Wed, 02/14/2018 - 11:04 <u>(Reply to #13)</u>

timgrant

This is listed in the image

This is listed in the image assets panel. If you want to output it as a text the only way would be to directly access the information in the database. The following command would do this on the command line :-

sqlite3 name_of_databse.db "select image_asset_id, filename from image_assets";

Cheers,

Tim

Tue, 02/20/2018 - 17:28 <u>(Reply to #14)</u>

dovile

The list of selected micrographs

Dear Tim,

How can I modify this script (sqlite3 name_of_databse.db "select filename from image_assets") to get a list not of all micrographs, but only of the selected ones that are placed in a new image group, for example, called ctf-better-than-4?

Cheers,

Dovile

Thu, 02/22/2018 - 08:17 <u>(Reply to #15)</u>

timgrant

Hi Doville,

First you need to know the group ID for the group you want. If you run :-

```
sqlite3 my_database.db "select * from
image_group_list;"
```

you will get a list of the image groups, the ID is in the first column.

Then if you run the command below replacing $group_id (it appears twice) with the id of the group you want, you will get the filenames you want.

```
sqlite3 my_database.db "select filename from
image_assets, image_group_$group_id where
image_assets.image_asset_id =
image_group_$group_id.image_asset_id;"
```

e.g. if you want the filenames of all the images in group 1 :-

```
sqlite3 my_database.db "select filename from
image_assets, image_group_1 where
image_assets.image_asset_id =
image_group_1.image_asset_id;"
```

Thanks!

Tim

Tue, 03/06/2018 - 10:21 (Reply to #16)

dovile

Expanding the command to get defocus list

Dear Tim,

Thank you very much for the commands! That's exactly what I needed.

In our lab we are very happy with the speed, graphical visualization of the results and general performance of cisTEM. However, we often need to try different things for challenging projects, therefore, we really appreciate simple ways to go in and out from different softwares at any stage of data processing. This seems totally feasible with cisTEM when knowing the right scripts. Could you help me with this by expanding the command to generate the list of selected mics to also get the corresponding defocus values?

I would also like to know where the non-dose weighted images and the aligned movie stacks from Unblur are stored in case I want to rerun ctf estimation outside of cisTEM. I assume that Assets/Images directory contains the dose-weighted images, or am I wrong?

Cheers,

Dovile

Tue, 03/06/2018 - 15:07 <u>(Reply to #17)</u>

timgrant

Hi Dovile,

all the information is stored in an sqlite database, so you can get access to it through SQL commands.

if you run `sqlite3 my_database.db` you will get the command line access to the database. if you type `.tables` (include the .), it will list all tables, if you type `.schema name_of_table`, it will give you information about what is in that table. In general it should be fairly easy to manipulate things if you have some experience with SQL.

The image assets, and their corresponding information are held in a table called image_assets. Each group is it's own table, and it just a list of the asset_ids that are in it, for a list of all available image groups, you can look in image_group_list. The CTF parameters are in a table called estimated_ctf_parameters.

If you would like the filename followed by defocus1, defocus2, defocus_angle and phase_shift (from the active result if you have multiple ctf estimations) of all images in group 1, one way to do it would be to run the following command :-

```
select image_assets.filename,
estimated_ctf_parameters.defocus1,
estimated_ctf_parameters.defocus2,
estimated_ctf_parameters.defocus_angle,
estimated_ctf_parameters.additional_phase_shift from
image_assets, image_group_1, estimated_ctf_parameters
where image_assets.image_asset_id =
image_group_1.image_asset_id and
image_assets.ctf_estimation_id =
estimated_ctf_parameters.ctf_estimation_id;
```

This will return the phase shift as radians, which is how it is stored in the database. You can run math commands in SQLite though, so if you wanted to have the phase shift in degrees, then the command would be :-

```
select image_assets.filename,
estimated_ctf_parameters.defocus1,
estimated_ctf_parameters.defocus2,
estimated_ctf_parameters.defocus_angle,
(estimated_ctf_parameters.additional_phase_shift *
57.29578) from image_assets, image_group_1,
estimated_ctf_parameters where
image_assets.image_asset_id =
image_group_1.image_asset_id and
image_assets.ctf_estimation_id =
estimated_ctf_parameters.ctf_estimation_id;
```

These commands may seem a bit non-sensical if you don't know SQL. In future versions of cisTEM I will try to improve the export so you can get at the information better. I know it seems like it would have been easier if everything had just been stored in text files, but once you get used to accessing things with SQL there are lots of advantages, enabling you to easily make selections and do various sorting.

Cheers,

Tim

Log in or register to post comments