NAME

PyMOLVisualizeCryoEMDensity.py - Visualize cryo-EM density

SYNOPSIS

PyMOLVisualizeCryoEMDensity.py [--align <yes or no>] [--alignMethod <align, cealign, super>] [--alignMode <FirstChain or Complex>] [--alignRefFile <filename>] [--allowEmptyObjects <yes or no>] [--BFactorChainCartoonPutty <yes or no>] [--BFactorColorPalette <text>] [--chainIDs <First, All or ID1,ID2...>] [--densityMapFiles <file1,file2,file3,...>] [--ignoreHydrogens <yes or no>] [--ligandIDs <Largest, All or ID1,ID2...>] [--labelFontID <number>] [--meshCarveRadius <number>] [--meshComplex <yes or no>] [--meshChainComplex <yes, no, or auto>] [--meshColor <text>] [--meshLevel <number>] [--meshWidth <number>] [--PMLOut <yes or no>] [--pocketContactsLigandColor <text>] [--pocketContactsLigandHydrophobicColor <text>] [--pocketContactsSolventColor <text>] [--pocketContactsCutoff <number>] [--pocketContactsInorganicColor <text>] [--pocketDistanceCutoff <number>] [--pocketLabelColor <text>] [--pocketSurface <yes or no>] [--selectionsChain <ObjectName,SelectionSpec,...>] [--selectionsChainStyle <DisplayStyle>] [--surfaceComplex <yes or no>] [--surfaceChainComplex <yes, no or auto>] [--surfaceTransparency <number>] [--volumeCarveRadius <number>] [--volumeComplex <yes or no>] [--volumeChainComplex <yes, no, or auto>] [--volumeColorRamp <text>] [--overwrite] [-w <dir>] -i <infile1,infile2,...> -o <outfile>

PyMOLVisualizeCryoEMDensity.py -h | --help | -e | --examples

DESCRIPTION

Generate PyMOL visualization files for viewing electron microscopy (EM) or cryo-EM density around chains, ligands, and ligand binding pockets in macromolecules including proteins and nucleic acids.

The supported input file formats are: Macromolecule - PDB (.pdb) or CIF(.cif), Cryo-EM Density - Collaborative Computational Project Number 4 (CCP4) (.map)

The supported output file formats are: PyMOL script file (.pml), PyMOL session file (.pse)

The cryo-EM density and header files along with PDB files may be downloaded from appropriate servers using DownloadPDBFiles.pl script.

A variety of PyMOL groups and objects may be created for visualization of cryo-EM density present in map files. These groups and objects correspond to maps, volumes, meshes, surfaces,chains, ligands, inorganics, ligand binding pockets, polar interactions, and pocket hydrophobic surfaces. A complete hierarchy of all possible PyMOL groups and objects is shown below:

```
<PDBFileRoot>
    .Complex
        .Complex
        .CryoEM
            .Map
            .Volume
            .Mesh
            .Surface
    .Chain<ID>
        .Complex
            .Complex
            .CryoEM
                .Volume
                .Mesh
                .Surface
        .Chain
            .Chain
            .BFactor
            .Selections
                .<Name1>
                    .Selection
                    .CryoEM
                        .Volume
                        .Mesh
                        .Surface
                .<Name2>
```

```
                    ... ... ..
            .Solvent
            .Inorganic
            .Ligand<ID>
                .Ligand
                    .Ligand
                    .CryoEM
                        .Volume
                        .Mesh
                        .Surface
                .Pocket
                    .Pocket
                    .CryoEM
                        .Volume
                        .Mesh
                        .Surface
                    .Polar_Contacts
                    .Hydrophobic_Contacts
                    .Surface
                .Pocket_Solvent
                    .Pocket_Solvent
                    .CryoEM
                        .Volume
                        .Mesh
                        .Surface
                    .Polar_Contacts
                .Pocket_Inorganic
                    .Pocket_Inorganic
                    .CryoEM
                        .Volume
                        .Mesh
                        .Surface
                    .Polar_Contacts
            .Ligand<ID>
                .Ligand
                    ... ... ...
                .Pocket
                    ... ... ...
                .Pocket_Solvent
                    ... ... ...
                .Pocket_Inorganic
                    ... ... ...
        .Chain<ID>
            ... ... ...
            .Ligand<ID>
                ... ... ...
            .Ligand<ID>
                ... ... ...
        .Chain<ID>
            ... ... ...
    <PDBFileRoot>
        .Complex
            ... ... ...
        .Chain<ID>
            ... ... ...
            .Ligand<ID>
                ... ... ...
            .Ligand<ID>
                ... ... ...
        .Chain<ID>
            ... ... ...
```

The meshes, volumes, and surfaces are not created for complete complex in input files by default. A word to the

wise: The creation of these mesh, volume, and surface objects may slow down loading of PML file and generation of PSE file, based on the size of input complex and map files. The generation of PSE file may also fail. In addition, you may want to interactively manipulate the contour level for meshes, volumes, and surfaces. The recommended value for contour level is automatically retrieved from header files available from EM density server. The recommended value may not always work.

OPTIONS

-a, --align <yes or no> [default: no]

Align input files to a reference file before visualization along with available cryo-EM density map files.

--alignMethod <align, cealign, super> [default: super]

Alignment methodology to use for aligning input files to a reference file.

--alignMode <FirstChain or Complex> [default: FirstChain]

Portion of input and reference files to use for spatial alignment of input files against reference file. Possible values: FirstChain or Complex.

The FirstChain mode allows alignment of the first chain in each input file to the first chain in the reference file along with moving the rest of the complex to coordinate space of the reference file. The complete complex in each input file is aligned to the complete complex in reference file for the Complex mode.

--alignRefFile <filename> [default: FirstInputFile]

Reference input file name. The default is to use the first input file name specified using '-i, --infiles' option.

--allowEmptyObjects <yes or no> [default: no]

Allow creation of empty PyMOL objects corresponding to solvent and inorganic atom selections across chains, ligands, and ligand binding pockets in input file(s).

-b, --BFactorChainCartoonPutty <yes or no> [default: yes]

A cartoon putty around individual chains colored by B factors. The minimum and maximum values for B factors are automatically detected. These values indicate spread of cryo-EM density around atoms. The 'blue_white_red' color palette is deployed for coloring the cartoon putty.

--BFactorColorPalette <text> [default: blue_white_red]

Color palette for coloring cartoon putty around chains generated using B factors. Any valid PyMOL color palette name is allowed. No validation is performed. The complete list of valid color palette names is a available at: pymolwiki.org/index.php/Spectrum. Examples: blue_white_red, blue_white_magenta, blue_red, green_white_red, green_red.

-c, --chainIDs <First, All or ID1,ID2...> [default: First]

List of chain IDs to use for visualizing cryo-EM density. Possible values: First, All, or a comma delimited list of chain IDs. The default is to use the chain ID for the first chain in each input file.

-d, --densityMapFiles <file1,file2,file3,...> [default: auto]

CryoEM density map file names. The EMDB ID is retrieved from PDB and CIF file to set the cryo-EM density file name during automatic detection of density files. The format of the file name is as follows:

```
emd_<EMDBID>.map.gz or emd_<EMDBID>.map
```

The density files must be present in the working directory.

-e, --examples

Print examples.

-h, --help

Print this help message.

-i, --infiles <infile1,infile2,infile3...>

Input file names.

--ignoreHydrogens <yes or no> [default: yes]

Ignore hydrogens for ligand and pocket views.

-l, --ligandIDs <Largest, All or ID1,ID2...> [default: Largest]

List of ligand IDs present in chains for visualizing cryo-EM density across ligands and ligand binding pockets. Possible values: Largest, All, or a comma delimited list of ligand IDs. The default is to use the largest ligand present in all or specified chains in each input file.

Ligands are identified using organic selection operator available in PyMOL. It'll also identify buffer molecules as ligands. The largest ligand contains the highest number of heavy atoms.

--labelFontID <number> [default: 7]

Font ID for drawing labels. Default: 7 (Sans Bold). Valid values: 5 to 16. The specified value must be a valid PyMOL font ID. No validation is performed. The complete lists of valid font IDs is available at: pymolwiki.org/index.php/Label_font_id. Examples: 5 - Sans; 7 - Sans Bold; 9 - Serif; 10 - Serif Bold.

--meshCarveRadius <number> [default: 1.6]

Radius in Angstroms around atoms for including cryo-EM density.

--meshComplex <yes or no> [default: no]

Create meshes for complete complex in each input file using corresponding density map file.

--meshChainComplex <yes, no, or auto> [default: auto]

Create meshes for individual chain complex in each input file using corresponding density map file. By default, the meshes are automatically created for chain complexes without any ligands.

--meshColor <text> [default: blue]

Line color for meshes corresponding to density maps.. The specified value must be valid color. No validation is performed.

--meshLevel <number1,number2,...> [default: auto]

Comma delimited list of contour levels in sigma units for generating meshes for each input file using corresponding density map file. The default is to automatically retrieve the recommended contour levels for each input file. The header file emd-<EMDBID>.xml corresponding to an input file must be present in the working directory to automatically retrieve recommended value for mesh contour level. Otherwise, the default contour level is set to 1.

You may want to interactively manipulate the contour level for meshes and surfaces. The default recommended value may not always work.

--meshWidth <number> [default: 0.5]

Line width for mesh lines corresponding to density maps.

-o, --outfile <outfile>

Output file name.

-p, --PMLOut <yes or no> [default: yes]

Save PML file during generation of PSE file.

--pocketContactsLigandColor <text> [default: orange]

Color for drawing polar contacts between ligand and pocket residues. The specified value must be valid color. No validation is performed.

--pocketContactsLigandHydrophobicColor <text> [default: purpleblue]

Color for drawing hydrophobic contacts between ligand and pocket residues. The specified value must be valid color. No validation is performed. The hydrophobic contacts are shown between pairs of carbon atoms not connected to hydrogen bond donor or acceptors atoms as identified by PyMOL.

--pocketContactsSolventColor <text> [default: marine]

Color for drawing polar contacts between solvent and pocket residues. The specified value must be valid color. No validation is performed.

--pocketContactsInorganicColor <text> [default: deepsalmon]

Color for drawing polar contacts between inorganic and pocket residues. The specified value must be valid color. No validation is performed.

--pocketContactsCutoff <number> [default: 4.0]

Distance in Angstroms for identifying polar and hyrdophobic contacts between atoms in pocket residues and ligands.

--pocketDistanceCutoff <number> [default: 5.0]

Distance in Angstroms for identifying pocket residues around ligands.

--pocketLabelColor <text> [default: magenta]

Color for drawing residue or atom level labels for a pocket. The specified value must be valid color. No validation is performed.

--pocketSurface <yes or no> [default: yes]

Hydrophobic surface around pocket. The pocket surface is colored by hydrophobicity. It is only valid for proteins. The color of amino acids is set using the Eisenberg hydrophobicity scale. The color varies from red to white, red being the most hydrophobic amino acid.

--selectionsChain <ObjectName,SelectionSpec,...> [default: None]

Custom selections for chains. It is a pairwise list of comma delimited values corresponding to PyMOL object names and selection specifications. The selection specification must be a valid PyMOL specification. No validation is performed.

The PyMOL objects are created for each chain corresponding to the specified selections. The display style for PyMOL objects is set using value of '--selectionsChainStyle' option.

The specified selection specification is automatically appended to appropriate chain specification before creating PyMOL objects.

For example, the following specification for '--selectionsChain' option will generate PyMOL objects for chains containing Cysteines and Serines:

```
Cysteines,resn CYS,Serines,resn SER
```

--selectionsChainStyle <DisplayStyle> [default: sticks]

Display style for PyMOL objects created for '--selectionsChain' option. It must be a valid PyMOL display style. No validation is performed.

--surfaceComplex <yes or no> [default: no]

Create surfaces for complete complex in input file(s) corresponding to density map.

--surfaceChainComplex <yes, no or auto> [default: auto]

Create surfaces for individual chain complexes in each input file using corresponding density map file. By default, the surfaces are automatically created for chain complexes without any ligands.

--surfaceTransparency <number> [default: 0.25]

Surface transparency for molecular and cryo-EM density surfaces.

--overwrite

Overwrite existing files.

--volumeCarveRadius <number> [default: 1.6]

Radius in Angstroms around atoms for including cryo-EM density.

--volumeComplex <yes or no> [default: no]

Create volumes for complete complex in each input file using corresponding density map file.

--volumeChainComplex <yes, no, or auto> [default: auto]

Create volumes for individual chain complex in each input file using corresponding density map file. By default, the volumes are automatically created for chain complexes without any ligands.

--volumeColorRamp <text> [default: auto]

Name of a volume color ramp for density map files. The specified value must be a valid name. No validation is performed. The following volume color ramps are currently available in PyMOL: default, 2fofc, fofc,

rainbow, and rainbow2.

The default is to automatically create a new volume color ramp for the first input file using recommended contour level with an offset of 0.3 around this value. The header file emd-<EMDBID>.xml must be present in the working directory to automatically retrieve recommended contour level and generate a volume color ramp. Otherwise, PyMOL default volume color ramp is employed to color volumes.

The volume color ramp automatically created for the first input file is used for all other input files.

`-w, --workingdir <dir>`

Location of working directory which defaults to the current directory.

EXAMPLES

To download structure and cryo-EM data for 5K12, 5UMD, 5W81, and 5UAK before running the following examples, type:

```
% DownloadPDBFiles.pl --DensityMap yes 5K12,5UMD,5W81,5UAK
```

To visualize cryo-EM density at recommended contour level for the first chain complex in a PDB file using corresponding density map and header file, and generate a PML file type:

```
% PyMOLVisualizeCryoEMDensity.py -i 5K12.pdb -o 5K12.pml
```

To visualize cryo-EM density at recommended contour level for the first chain complex in a PDB file and highlighting densities for all cysteines and serines using corresponding density map and header file, and generate a PML file type:

```
% PyMOLVisualizeCryoEMDensity.py -i 5K12.pdb -o 5K12.pml
  --selectionsChain "Csysteines,resn cys,Serines,resn ser"
```

To visualize electron density for the largest ligand in chain K, and ligand binding pocket to highlight ligand interactions with pockect residues, solvents and inorganics, in a PDB and using corresponding map files, and generate a PML file, type:

```
% PyMOLVisualizeCryoEMDensity.py -c K -i 5UMD.cif -o 5UMD.pml
```

To visualize cryo-EM density for all chains along with any solvents in a PDB file and using corresponding map files, and generate a PML file, type:

```
% PyMOLVisualizeCryoEMDensity.py -c all -i 5K12.pdb -o 5K12.pml
```

To visualize cryo-EM density at a specific contour level for the first chain complex along with volume and surface in a PDB file using corresponding to a specific density map file, and generate a PML file, type:

```
% PyMOLVisualizeCryoEMDensity.py -d emd_8194.map.gz --meshLevel 1.0
  --surfaceChainComplex yes --volumeChainComplex yes -i 5K12.pdb
  -o 5K12.pml
```

To align and visualize cryo-EM density at recommended contour levels for the largest ligand in the first chain along with pockets or the first chain complex in input files using corresponding maps and header files, type:

```
% PyMOLVisualizeCryoEMDensity.py -a yes -i "5W81.pdb,5UAK.pdb"
  -o SampleOut.pml
```

To align and visualize cryo-EM density at recommended contour levels for all chains and ligands in input files using specified density files, type: in input files using corresponding maps and header files, type:

```
% PyMOLVisualizeCryoEMDensity.py -a yes -i "5W81.pdb,5UAK.pdb"
  -o SampleOut.pml -c all -l all -d "emd_8782.map.gz,emd_8516.map.gz"
```

AUTHOR

Manish Sud(msud@san.rr.com)

SEE ALSO

DownloadPDBFiles.pl, PyMOLVisualizeCavities.py, PyMOLVisualizeElectronDensity.py, PyMOLVisualizeInterfaces.py, PyMOLVisualizeMacromolecules.py, PyMOLVisualizeSurfaceAndBuriedResidues.py

COPYRIGHT

Copyright (C) 2026 Manish Sud. All rights reserved.

The functionality available in this script is implemented using PyMOL, a molecular visualization system on an open source foundation originally developed by Warren DeLano.

This file is part of MayaChemTools.

MayaChemTools is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.