ChimeraX LigandRecognizer Tutorial

This document provides a step-by-step tutorial on how to use the `blob` commands from the ChimeraX LigandRecognizer bundle. The `blob` commands implement the functionality proposed in the publication "*Ligand Identification in CryoEM and X-ray Maps Using Deep Learning*" that uses a deep learning model to predict ligand types.

Installation

To download and install the latest version of the LigandReconizer bundle, go to:
https://github.com/wtaisner/chimerax-ligand-recognizer.
In the future, the bundle will also be available through the ChimeraX Toolshed.

After installation, open ChimeraX and type in the ChimeraX Command Line:
`help blob`
to check if the installation was successful.

Commands

The tool implements three basic commands:
1. **`blob validate res_id [map_id] [model_id] [xray False/True] [density_threshold]`**: validates (performs a prediction for) an existing ligand at res_id.
2. **`blob recognize [map_id] [surface_id] [xray False/True] [density_threshold]`**: recognizes (performs a prediction for) a ligand in a selected map fragment.
3. **`blob autothreshold [map_id] [withstyle False/True] [density_std_threshold]`**: sets an automatic threshold on the displayed map (volume).

All the parameters in brackets are optional, and if they are not provided, the command will use the currently active map, structure, and surface and assume that the map is a cryoEM map (xray False). The above three commands are aliases for `blobus validatus`, `blobus recognitus`, and `blobus autothresholdus` respectively.

The following sections show step-by-step how to use the `blob validate` and `blob recognize` commands in practice.

Tutorial preparation

The `blob` commands assume that both a .pdb/.cif (partial) model and a .ccp4 difference map are loaded into ChimeraX. A key requirement is that the commands work on a *difference map* (omit map) rather than a regular (model) map. The difference map should be the result of PHENIX's command `phenix.real_space_diff_map "$MODEL" "$MAP" "resolution=$RES"` for

cryoEM or the Fo-Fc map for X-ray crystallography. To obtain the cryoEM difference map, you can use the `computeMapModelDifference.sh` script, which is available in the bundle's repository.

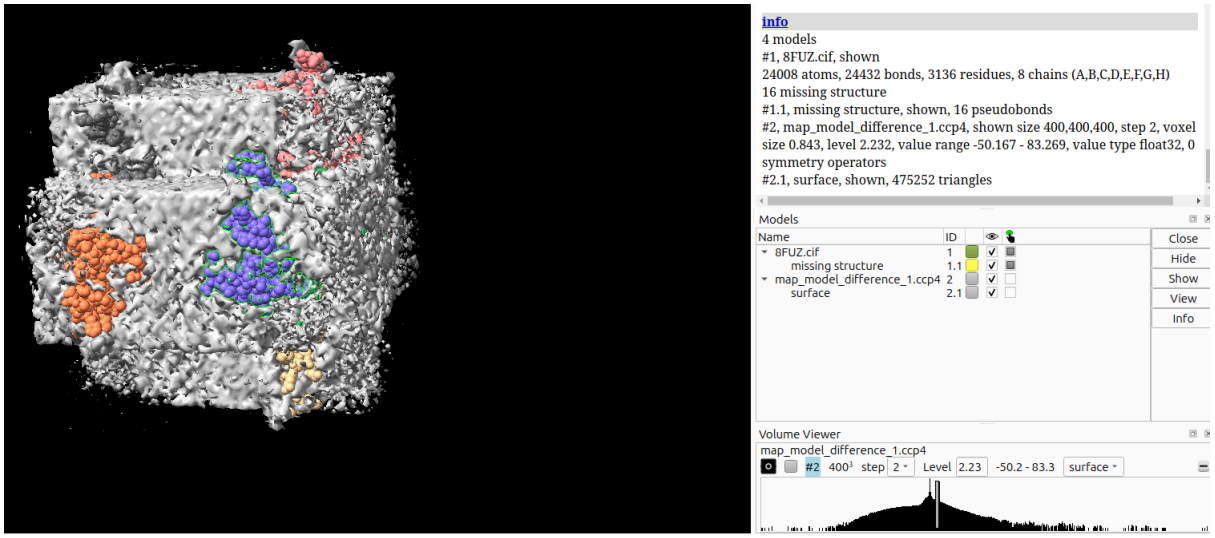
For this tutorial, we will use the PDB deposit 8FUZ as an example. You can download:

- the full 8FUZ model (8FUZ.cif)
- the partial (without ligands modeled) 8FUZ model (8FUZ_stripped.cif)
- difference map (map_model_difference_1.ccp4)

from: https://1drv.ms/f/s!Aq419F62GZU4g_tDT9LFuNWVV9Eohg?e=8vUR7W.

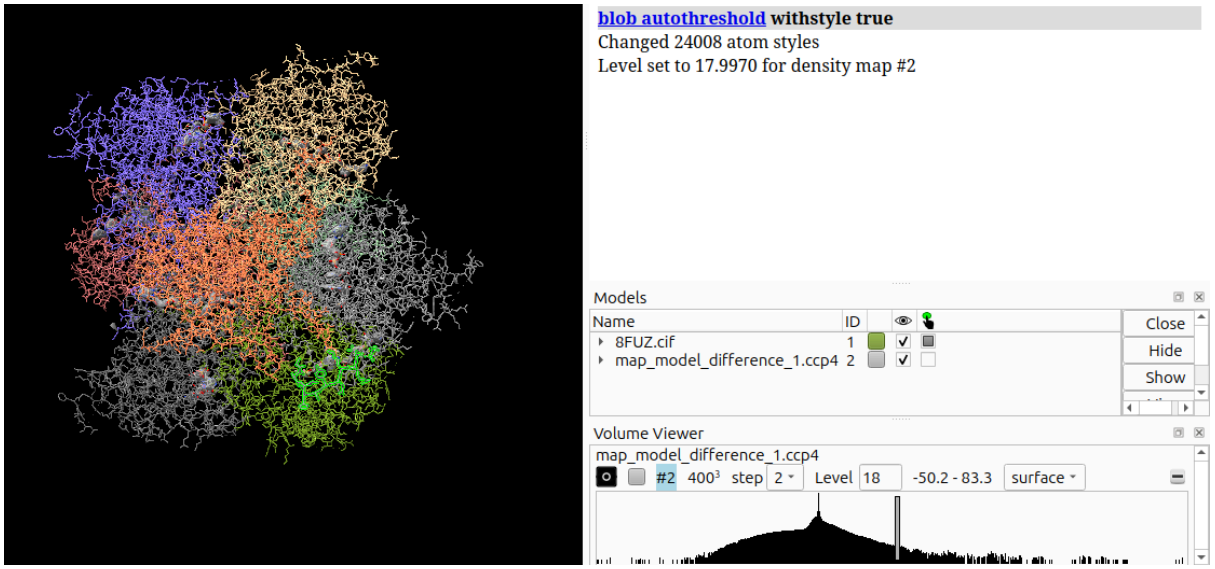
Blob validate

1. Open files
   a. Open the .cif and .ccp4 files corresponding to the map and protein.
   b. Run: **`open 8FUZ/8FUZ.cif 8FUZ/map_model_difference_1.ccp4`**
   c. You should get the corresponding outlook by running the `info` command

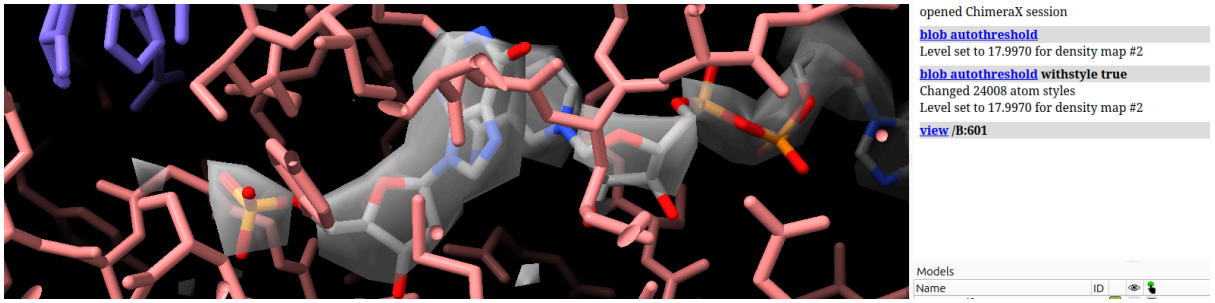


2. Adjust the view
   a. Run **`blob autothreshold`** to threshold the density map. It should see the following

b. If you want to have a clearer view of the ligands, you can also run the command **`blob autothreshold withstyle True`**, which will change the display options for atoms and maps, making them more readable. The corresponding output should look as follows:



c. Then, run **`view /B:601`** to see what part of the density map we'll be working on:



3. Run blob validate on the selected ligand:

a. To validate the residue with id 601 in chain B, run: **`blob validate /B:601`**

b. After a short while, in the ChimeraX log, you should see something similar to the following:

blob validate /B:601
Attempting to cut ligand (this may take a while)...
23 atoms, 25 bonds, 1 residue, 1 model selected
Ligand Class Predictions
(Click on ligand group name to see the full list of ligands)

Ligand Class	Probability
AMP-like	0.999
ADP-like	0.000
RARE_LIGAND	0.000
5GP-like	0.000
ATP-like	0.000
GDP-like	0.000
AGS	0.000
HEA	0.000
FMN-like	0.000
A3P	0.000

c. As mentioned in the Commands section of this tutorial, you can also manually pass the map_id, model_id, and the type of density map, e.g.: `blob validate /B:601 #2 #1` where **#2** is map_id and **#1** is the model_id.

info
4 models
#1, 8FUZ.cif, shown
24008 atoms, 24432 bonds, 3136 residues, 8 chains (A,B,C,D,E,F,G,H)
16 missing structure
#1.1, missing structure, shown, 16 pseudobonds
#2, map_model_difference_1.ccp4, shown size 400,400,400, step 2, voxel size 0.843, level 2.232, value range -50.167 - 83.269, value type float32, 0 symmetry operators
#2.1, surface, shown, 475252 triangles

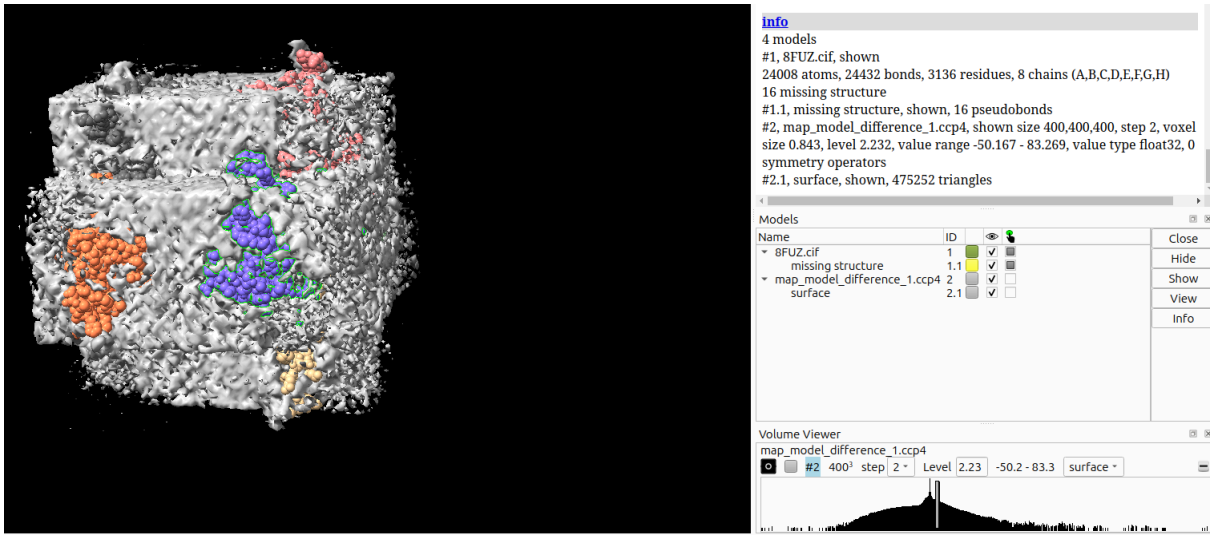
blob validate /B:601 #2 #1
Attempting to cut ligand (this may take a while)...
23 atoms, 25 bonds, 1 residue, 1 model selected
Ligand Class Predictions
(Click on ligand group name to see the full list of ligands)

Ligand Class	Probability
AMP-like	0.999
ADP-like	0.000
RARE_LIGAND	0.000

Blob recognize

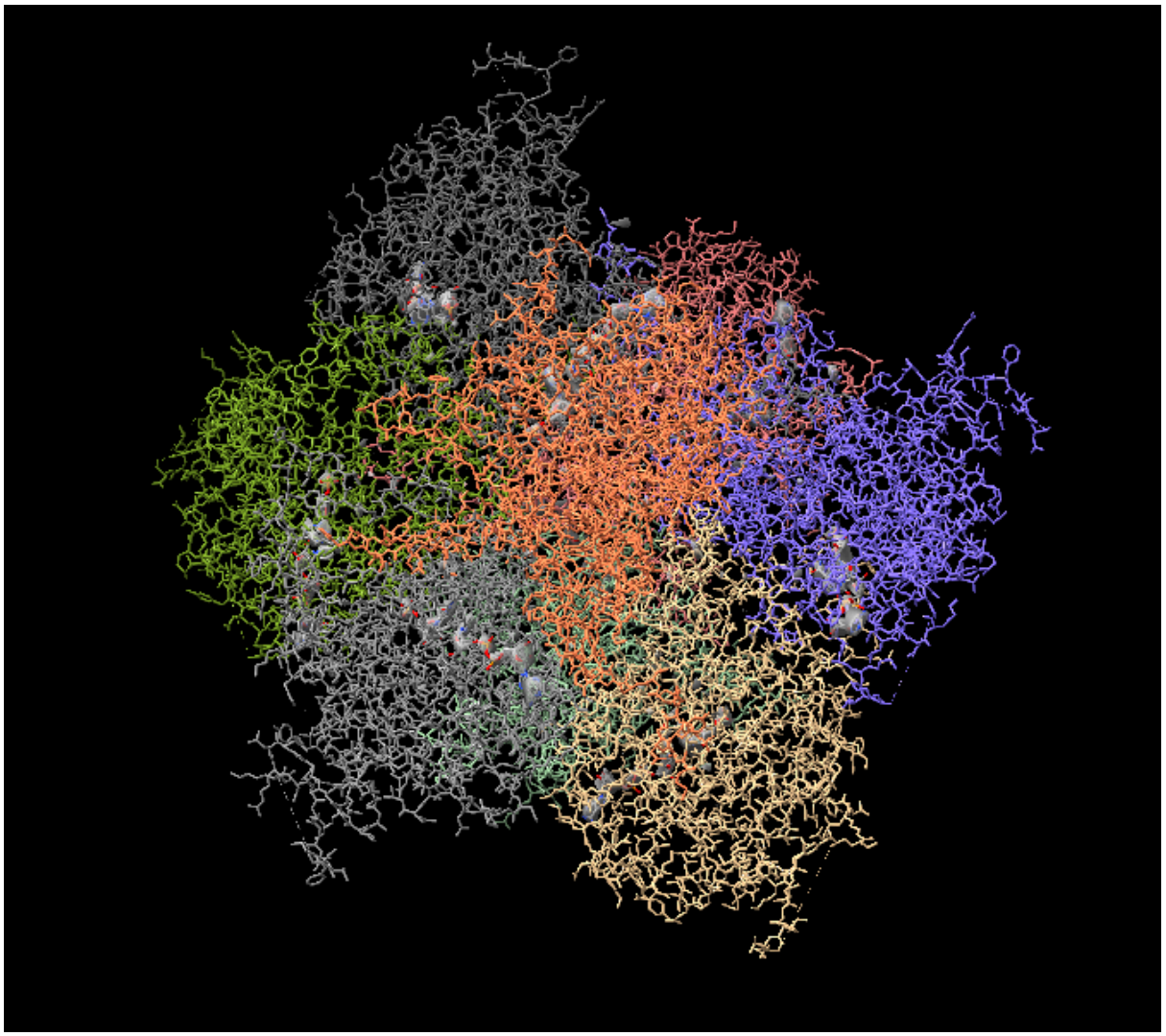
1. Open files

a. Open the .cif and .ccp4 files corresponding to the model and map.
b. Run: `open 8FUZ/8FUZ.cif 8FUZ/map_model_difference_1.ccp4`
c. You should get the corresponding outlook by running the `info` command



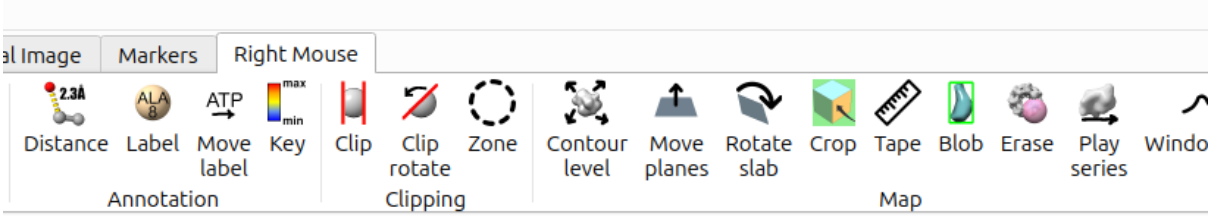
2. Adjust the display

   a. Run the command `blob autothreshold withstyle True` to get better styling:



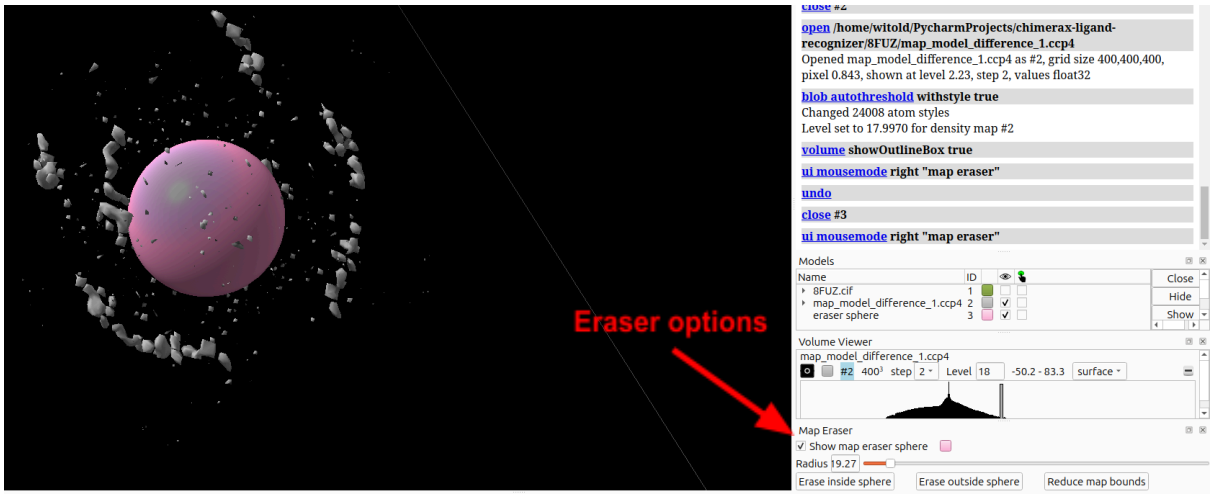
3. Crop the map – version 1 (recommended)

   a. Go to the Right Mouse menu, and choose “Erase” under Map settings:



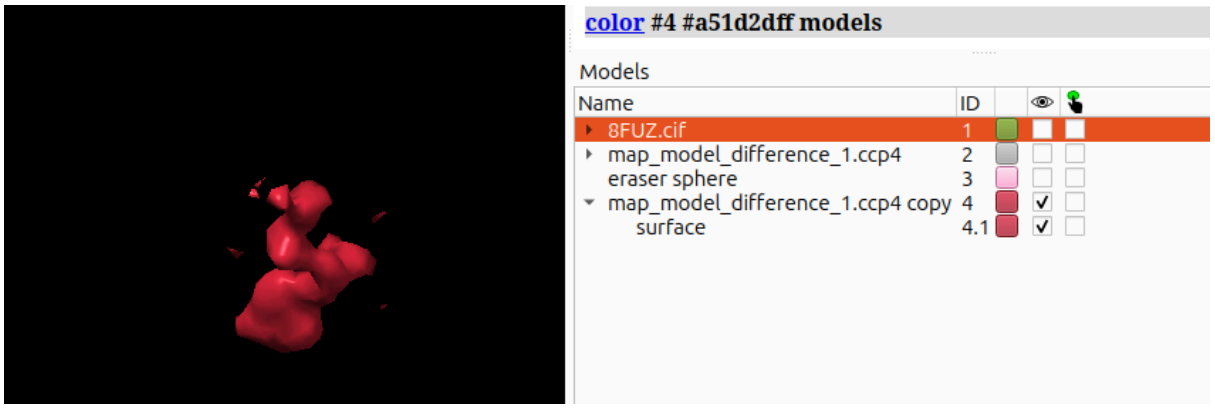
   b. Upon right-clicking on the map, a pink eraser sphere will appear

   c. Now, move the sphere using the right mouse button and use it to erase part of a map that shouldn’t be recognized. Notably, you can modify its radius using a slider in the bottom-right corner of the ChimeraX window.



d. You can use either option “Erase inside/outside sphere”. The only difference is where you will put your sphere. If the ligand to be recognized is inside the sphere, then use “Erase outside sphere”. Otherwise, you can reposition the sphere to span non-relevant parts of the map and use “Erase inside sphere.”

e. **IMPORTANT!** Using the Eraser tool will create a copy of a volume with a surface limited to the blob we positioned inside the sphere. This allows us to reposition the map eraser sphere and separate further blobs, *without* altering the original map. The created copy in our example is under id `#4` in the Models section.



f. Notably, the original volume (`#2`) remains unchanged.

g. Now, we can run **`blob recognize #2 #4.1`**, which requires the map_id of the original volume (#2) and the surface_id of the blob we want to classify (`#4.1`). We cannot use `#4` since it refers to a “volume,” not a “surface.”

blob recognize #2 #4.1

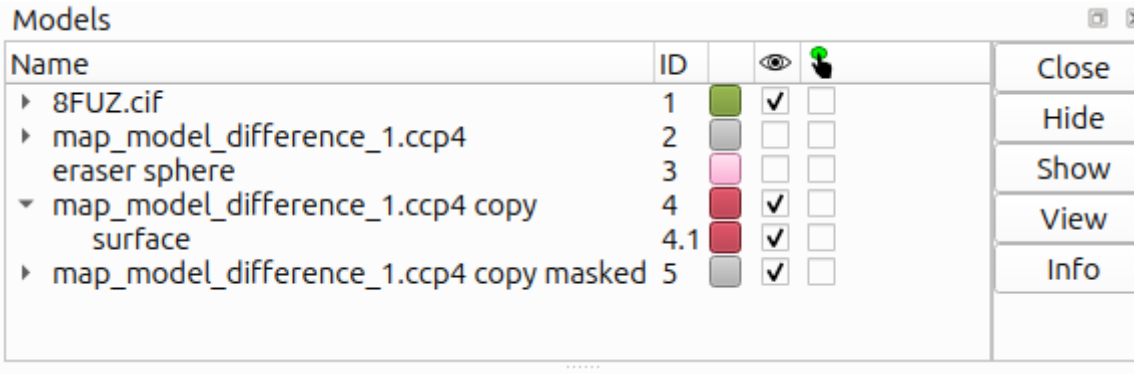
Opened map_model_difference_1.ccp4 copy masked as #5, grid size 400,400,400, pixel 0.843, shown at step 1, values float32

Ligand Class Predictions

(Click on ligand group name to see the full list of ligands)

Ligand Class	Probability
ADP-like	0.633
ATP-like	0.249
SAM	0.048
AGS	0.023
RARE_LIGAND	0.022
NAD-like	0.013
AMP-like	0.008
SAH-like	0.003
ACO	0.001
TYR-like	0.001

h. Running `blob recognize` will open another object, namely a masked copy of a volume (`#5` in the Models section):

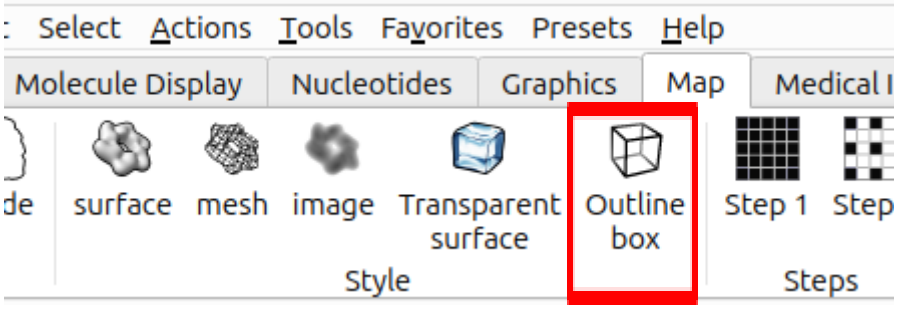


i. While it is necessary to open it to perform computations, it is redundant for any analysis and can be closed, using the “Close” button to the right, after the prediction.

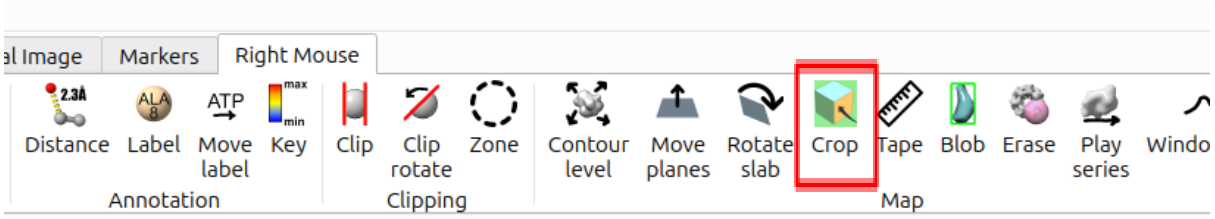
j. After predicting one ligand, the sphere can be repositioned and the process can begin anew. Make sure to toggle on/off the visibility of the entire map and/or model.

4. Crop the map – version 2 (less recommended)

a. Go to Map and toggle the “Outline Box”

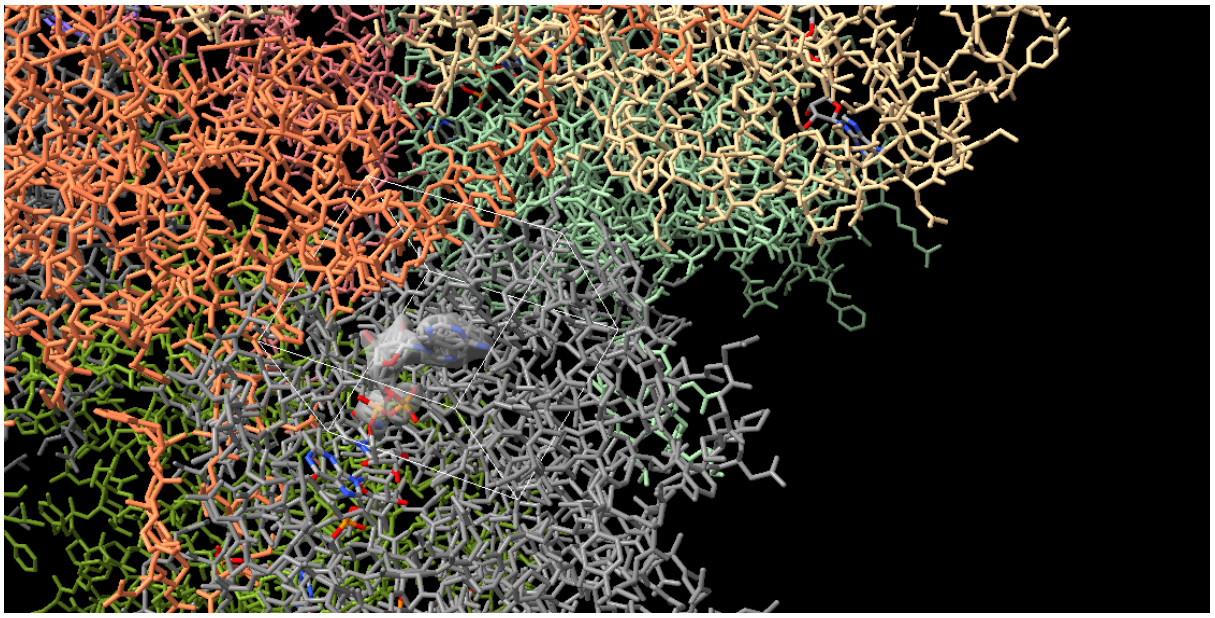


b. Go to the Right Mouse menu and toggle the “Crop” option from the Map section



c. Using the right mouse button in the map pane, limit the outline box to span a part of the density map you want to recognize

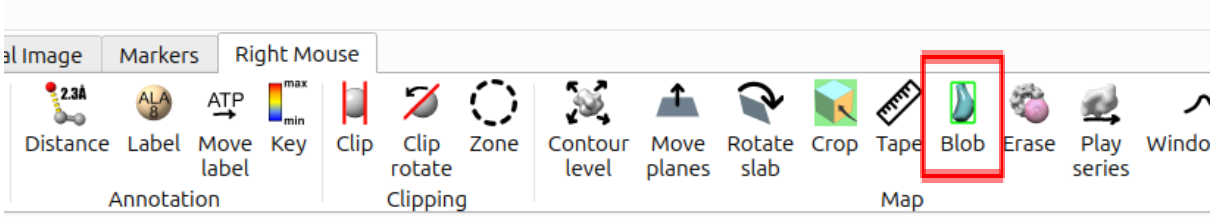
d. The result could look as follows:



e. You want to place the ligand to be recognized inside the Outline Box

f. Optionally, you can hide the protein display to work on the map only

g. You can also use the “Blob” option in the Map section of the Right Mouse settings



h. You can outline many blobs; however, the final command must be run using only one blob (one surface). To separate the blobs into distinct surfaces, you should run **`surface splitbycolor #2`**, where `#2` is the ID of the density map for which the blobs were outlined.

i. Run **`blob recognize #2 #surface_id`** to perform a prediction for a selected blob