Journal of
Applied Crystallography
ISSN 0021-8898

Received 24 March 2004
Accepted 22 April 2004

MolViewX: a molecular visualization program for the Macintosh OS X system

Thomas James Smith

The Donald Danforth Plant Science Center, 975 North Warson Road, Saint Louis, MO 63132, USA.
Correspondence e-mail: tsmith@danforthcenter.org

MolViewX is a molecular visualization tool written for the Macintosh OS X operating system with a graphical user interface designed for occasional or novice users, but which can also be used to create high-quality images rapidly for publication.

1. Introduction

MolViewX is a molecular visualization tool written for the Macintosh OS X operating system. This represents a significant update from a previous application, *MolView* (Smith, 1995), that ran on the 'classic' operating system. The main purpose of this application is to bring an intuitive graphical interface to the user who only occasionally needs to view atomic structures or for classes where it is not efficient to teach new scripting commands. From either PDB files or *O* (Jones *et al.*, 1991) output files, the user can analyse structures and output publication-ready PICT images.

While a number of Unix-based programs for displaying protein structures have been ported over to the new OS X operating system, few employ the new graphical interface or run native under the new graphical user interface. The original OS 7–9 version of the program was developed in the Metrowerks (http://www.metrowerks.com) environment, but was entirely reworked for OS X using the *Project Builder* (Apple Corporation; http://www.apple.com) package. Here, some of the new features are reviewed.

2. Key features

Input files. Currently *MolViewX* reads several different types of coordinate files. The main types of files are: standard Protein Data Bank (PDB) files, *MolViewX* files, and output plot files from the Unix-based program *O* (Jones *et al.*, 1991). In the case of PDB files, *MolViewX* reads the coordinates and looks for secondary structure and polypeptide chain information. The secondary structure information is written out to a `*.struc` file, which can be later read to generate accurate ribbon models. For each polypeptide chain, the residue number is incremented by 1000 so that the user can more easily delineate the various chains for display purposes. It is important to note that the dynamic array allocation within the program permits the display of extremely large molecules, such as entire viruses. *MolViewX* files contain the original coordinates and can also contain 'snippets of atomic structures' ('MOL' objects) along with various display parameters. This is particularly useful for being able to return to a particular view. In the case of the *O* plot files, the user can use this formatted file to read in electron density (see Fig. 1*b*) and various objects from the powerful program *O*, for either demonstration or publication purposes. Currently, two structure files can be opened simultaneously for comparison purposes.

Output files. Several types of files can be written out by *MolViewX*: *MolViewX*, PICT, 3DMF, MOL Object, PDB coordinate, and VRML files. *MolViewX* files are as described above. *MolViewX* can write out PICT formatted object-oriented images of the various windows and analysis outputs. Emphasis has been placed on keeping the images object-oriented to maintain the highest possible resolution for display and printing purposes. 3DMF files contain descriptors of three-dimensional objects that are read by Apple's now defunct *QuickDraw* three-dimensional routines. This option will remain in the program since a few applications still recognize this format. This option is now largely replaced by VRML formatted files. VRML (Virtual Reality Modeling Language, version 2.0) files contain three-dimensional descriptors of the various objects shown on the screen. When this output is selected, *MolViewX* culls the various colors and writes out appropriate three-dimensional descriptors for the objects shown on the screen. While this is a standard format, it has been particularly tested to be read by the program *Strata 3D Pro* (version 3.9; http://www.strata.com). With this file, high-resolution images can be rendered for publication interactively without writing any scripts (see Figs. 1*b*–1*d*). MOL object files contain 'snippets' of atomic structures that can be read in at a later date. *MolViewX* is now able to perform three-dimensional alignments of protein structures. To facilitate this process and to make cross-platform structure files, the user can write out the coordinates as PDB formatted files.

3. Interface

In keeping with previous versions of the program, the user is presented with a number of display options and analysis tools *via* floating palettes and buttons (see Fig. 1*a*). The goal of *MolViewX* is not to recreate tools such as the powerful *O* program, but rather to implement some of the features likely to be important to biologists and to present the user with a transparent interface. Indeed, *MolView* and *MolViewX* have been tested in classrooms and students can be taught to perform sophisticated analyses within minutes. For structural biologists, the interface can greatly facilitate setting up 'scenes' for publication images. Some of the more important interface components are described below.

Tool box. This floating palette contains buttons for all of the major features of the program. These features are discussed below.

Object window. All objects are represented as buttons in this window. Each object can be toggled on/off and all of the display parameters for these objects are controlled by buttons immediately beneath the toggle switches. For example, line drawings have three buttons that control line color, thickness and pen pattern, while ball-and-stick objects have a preference button that controls coloring and diameter parameters.

Ribbon button window. One of the more powerful features of *MolViewX* is fine control of the display aspects of ribbon models. For each polypeptide chain, the first button is an on/off toggle switch and beneath is a button to change the color of the entire chain. Following this button are toggle and color buttons for each secondary structural element. The color buttons are numbered according to the first amino acid found in that secondary structural element. If the user clicks on this button while holding down the 'apple' key, then another window opens and the colors of individual amino acids within this structural element can be changed.

Color palette window. The Color Palette contains more than 100 standard colors for the user to pick from. At the bottom is a row of blank colors that the user can click on to fill with their own choices of colors, which are saved upon exiting the program.

Analyses windows. When various analyses are performed, separate windows are generated. For example, when a portion of the structure is analysed with Ramachandran plots, each data point on the graph can be selected and the corresponding residue displayed.

4. Types of models created

The emphasis here was to allow the user ultimate control over the color and display schemes in an interactive way. For all of the following models, the user can display regions, control colors and side chains only.

C-α backbones. By default, only the backbone of the structure is displayed when a file is first opened. This model is not written out to 3DMF or VRML files. For that, the user selects the MOL button in the Tool Box window and creates a C-α model.

Ball and stick. This is a traditional ball-and-stick routine. Here, the user can color the balls according to the atom type or give various segments uniform color. The user can customize the ball color scheme as well as change the ball-and-stick radii.

Stick models. There are many different types of stick models that can be created. The user can display a region of the protein with or without main-chain atoms and assign a particular color to it. The user can also create an object where the colors of the sticks are based on the atom type. To facilitate analysis, the user can choose residues with a particular name in a region of the protein. The user can also create sticks showing potential hydrogen-bonding patterns.

Space-filling models. There are several ways to make space-filling (CPK) models. The spheres can be displayed with a metallic appearance or as flat circles. As with the stick models, the user can select the color, residue range, residue type, and can display just the side-chain atoms. In addition, the user can use the MOL option to place a sphere at an atom in a particular residue. When writing to 3DMF or VRML files, these colors and radii are carried over to the output file.

B-value models. The *B* values of the atoms in the model can often show the flexible regions of the protein. To display this flexibility, the user can create a stick model of either the backbone or entire zones of the protein colored according to the *B* values.

Water accessibility. *MolViewX* can create a three-dimensional array of dots or 'stipples' that represents the accessibility of the atoms to solvent. The user can define the probe radius and can color the stipples according to charge, atom type, or hydrophobicity.

Symmetry operations. This option allows the user to read in symmetry matrices and apply them to the current model. In the case of oligomeric proteins such as viruses, these matrices can be the non-crystallographic relationships. For other proteins, these matrices can be crystallographic operators so that the user can examine crystallographic contacts. Further, the user can create a three-dimensional cage that represents the crystallographic unit cell.

DNA/RNA display. There are a couple of options specific for displaying DNA and RNA. The user can display the phosphate-ribose backbone as a ribbon, color the planes of the nucleotides according to base type, and display the hydrogen-bonding pattern for DNA base-pairing. These can be written out to either PICT files or to VRML files for further rendering (see Fig. 1*d*).

Ribbon diagrams. As mentioned above, ribbon diagrams are not only created, but can also be colored in numerous ways. For the creation of ribbon diagrams, a simple text file is read that contains the secondary structural information (`*.struc` files). *MolViewX* can create this file using information read from the header of the PDB file or it can be calculated from the structure using the Ψ–Φ values and hydrogen-bonding patterns. The latter tends to work better with higher resolution structures. This `*.struc` text file can be easily edited by the user and fixed if necessary. After creating the ribbon diagram, the user can use a Tool Box button to increase the number of planes in the ribbon diagram to obtain smoother ribbons. By being able to control this parameter, the user can maintain faster rotation rates until a final figure is needed.

5. Analysis options

There are a number of options that allow the user to analyse various aspects of the protein structure.

Labeling. The user has many options for labeling the model. The user can create the kind of label that they want using the atom type, residue number and residue type. The user can select some or all of this information in any order. The atoms to be labeled are then selected by either a mouse click on the atomic model or from a scrolling list of all of the atoms. Unlike most other applications, these labels can be individually deselected in the same manner.

Hydrogen bonding. The user can look at potential hydrogen-bonding patterns within a region of the protein and can also change the parameters that define what constitutes a hydrogen bond.

Atomic distances. The user can measure the distances between atoms in two different ways. In one option, the user selects two atoms and a vector is drawn between them; the distance is measured, and the user is prompted to create a label for this interaction. To ensure that first atom is correctly selected, a small box is drawn around it. If it is not the correct atom, the user can deselect it by clicking on it again with the mouse. The other way distances can be measured is *via* the 'neighbors' routine. Here, a mouse click on an atom causes vectors to be drawn to all adjacent atoms within a user-defined distance. The length of each vector (in Å) is displayed immediately adjacent to it.

Ramachandran plots. This option allows the user to display a Ramachandran plot of the Ψ–Φ values for particular regions of the protein. This can be saved as an object-oriented PICT file for publication. In the colorized versions of the plot, glycine residues are displayed as filled orange circles, prolines as mauve filled squares, and the remaining residue types as open, black circles. Unique to *MolViewX*, the user can click on points in the plot and the residue number for that point is displayed on the plot.

Hydrophobicity plots. The user can select regions of the protein and graph the hydrophobicity of the residues for this region according to the Kyte–Doolittle index (Kyte & Doolittle, 1982). While this is normally used when only the amino acid sequence for a

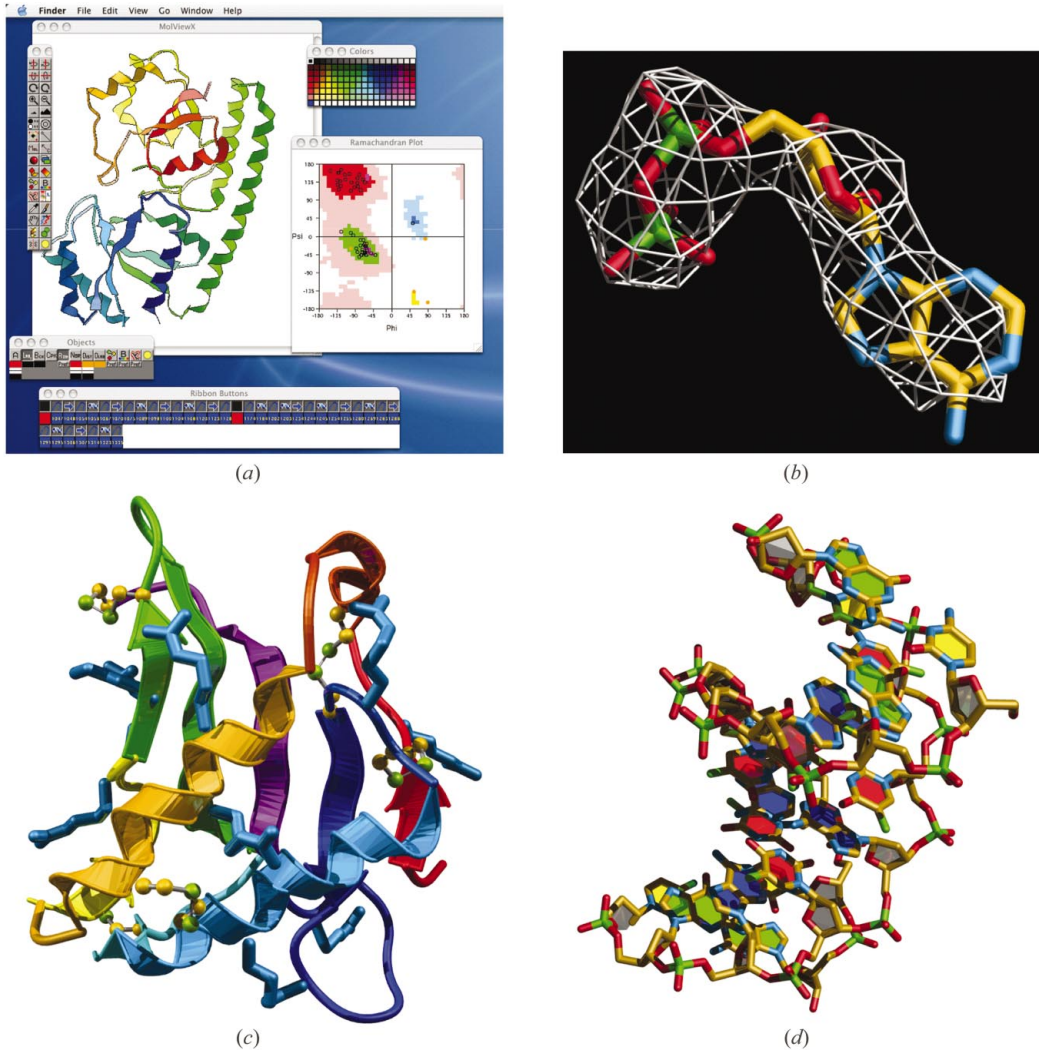

Figure 1
Examples of *MolViewX* images. (*a*) A screen capture of *MolViewX* in use. The main window in the middle of the image contains a ribbon diagram. The thin window to the left of it shows the Tool Box. The shorter strip on the bottom is the Object Window and the longer one is a Ribbon Window. The upper right window is the Color Palette and the one beneath is an example Ramachandran Plot. (*b*) An example of a final, rendered portion of electron density that was written out as a plot file by the program *O* (Jones *et al.*, 1991), displayed and clipped in *MolViewX*, written out as a VRML file, and finally rendered in the program *Strata 3D Pro* (http://www.strata.com). (*c*) In this example, *MolViewX* was used to create a ribbon diagram where the color ramps from red to purple as the backbone traces from the N to C termini. The side chains of the basic residues are represented by blue sticks and the cysteine residues are represented by ball-and-stick models. The model was created in *MolViewX*, written out as a VRML file, and rendered in *Strata 3D Pro*. (*d*) An example of how DNA can be displayed. Here the base and ribose planes are filled with colors to facilitate identification and the atoms are represented by sticks colored according to atom type. As with the other images, this was rendered by *Strata 3D Pro via* VRML output files.

protein is known to help ascertain buried regions of proteins, it is implemented in *MolViewX* for educational purposes.

Edmundson wheels. Edmundson wheels (Schiffer & Edmundson, 1967) are useful in teaching students the concept of amphipathic helices. To that end, the user can create Edmundson wheels where the colors represent the relative hydrophobicity of residues in that region.

B value plots. With this option, the user can plot the *B* values *versus* the residue number in order to more easily find the mobile loops and the absolute values of these *B* values.

Three-dimensional alignments. This new version of *MolViewX* allows the user to input a second molecule either to create additional display objects and/or for this alignment routine. When this option is selected, a dialog box opens up and the user is able to select two regions of the two proteins that should align well. *MolViewX* then uses a least-squares routine that rotates the second molecule onto the first. *MolViewX* performs the calculation, opens another dialog box that shows the user the quality of the fit in terms of root-mean-square deviation, and then asks whether this alignment is to be refined further. If selected, *MolViewX* takes the current solution and looks for pairs of atoms that are within the user-defined distance criteria for a second round of least-squares refinement. As with all other three-dimensional alignments, the better the homology, the better the fit. However, this method should allow alignment of proteins where the cores are well conserved but the loops are not.

6. Display options

There are numerous ways in which the user can manipulate the image.

Rotations. There are three ways to perform rotations. The user can use *X*, *Y*, *Z* rotation buttons in the Tool Palette. There is a 'hand' rotation tool where the user can interactively rotate the object with

the mouse. Finally, the user can use a dialog box from a menu bar selection to input precise rotations about the three axes. There is also a centering tool in the Tool Palette that is used to redefine the center of rotation. By default, the center of rotation is the center of mass of all of the objects.

Stereo. The user can toggle stereo pairs off and on, and can choose either convergent or divergent stereo from the Preferences menu bar selection.

Coloring. Colors for the various objects are selected *via* dialog boxes or by the Color Palette. To facilitate this process, an 'eye dropper' is in the Tool Box that 'picks up' a color from anywhere on the screen and a 'paint brush' is also there to drop this color onto a particular object color box. In addition, the user can select any background color for display.

Depth cueing. Writing the images out in an object-oriented manner yields the highest possible resolution for printing. However, for the object to be drawn correctly, the vectors have to be sorted along the Z direction. This can lead to slow refresh rates for complex images. To facilitate this process, there is a sorting toggle switch in the Tool Box that can be used immediately prior to writing out the PICT files. In addition, the objects on the screen can be colored according to their Z position and faded into the background color.

Scaling. There are four ways in which the user can zoom in and out on the image. There are two magnifying glasses in the Tool Box that will zoom in and out on a particular atom. There are also two 'mountain' buttons that will expand or shrink the entire image about the current centering point. The scaling factors can be changed in the preferences. Finally, the user can input a particular scale factor in a dialog box selected from the menu. In all cases, these operations are handled better than they were in previous versions by keeping the object centered on the left half of the window.

7. Distribution

MolViewX is distributed free of charge from my laboratory Web site: http://www.danforthcenter.org/smith/molview.htm. This package includes example files.

This work is supported by an NIH grant, GM10704.

References

Jones, T. A., Zou, J.-Y. & Cowan, S. W. (1991). *Acta Cryst.* A**47**, 110–119.
Kyte, J. & Doolittle, R. F. (1982). *J. Mol. Biol.* **157**, 105–132.
Schiffer, M. & Edmundson, A. B. (1967). *Biophys. J.* **7**, 121–135.
Smith, T. J. (1995). *J. Mol. Graphics*, **13**, 122–125.