One goal of structural biology is to understand the chemical and physical properties of macro-molecules (especially proteins) and how this enables the chemical reactions behind life’s processes. In order to study these large and complex molecules, biochemists rely on visualizations that provide various levels of abstraction. The more abstract visualizations portray a molecule’s internal structure. However, protein interactions involve the “functional surface” it presents: to a large degree, the internal structure simply exists as scaffolding to place various forces and chemical properties in proper spatial relationships with one another. While visualizations of these functional surfaces exist, they portray all of the detail and complexity of large molecules. The complexity of these visualizations is problematic as they do not afford rapid assessment, and details may obscure larger scale phenomena. To date, the degree of abstraction provided for internal structure has not been shown for the external properties.

In this paper, we introduce abstracted molecular surfaces, a visualization technique that provides for abstracted views of the boundary of a molecule and the physical and chemical properties at this boundary. Our goal is to provide simplified visual representations of molecules such that scientists can rapidly assess the most significant features of their surfaces, even when drawn at a small size. Such abstracted views are useful for rapid browsing and comparison, but also to study molecules while unencumbered by small details. Once general notions are determined from the abstracted views, a scientist can make a focused examination of a more detailed representation.

Our abstraction mechanism processes the detailed information about the molecule to provide a visually simplified representation. The shape of the molecular surface is simplified, removing small details to better convey the basic shape. Significant shape features, such as clefts and pockets, become more prominent when the visual clutter of smaller features is removed. A comparison with other display methods is shown in Figure 1 and Figure 2. Additionally, our abstracted display is more amenable to stylized rendering that accentuates the shape, retains readability at lower resolutions allowing gallery displays, and allows for the use of surface markings to display other information.

Our technique also uses abstraction on properties other than shape. Scalar fields along the surface, such as electrostatic charge, are simplified for clarity, and other properties are displayed as symbols on the surface. These presentations allow significant features to be seen quickly and clearly.

We are motivated by an increased need for tools that enable quick and comparative visual analysis. Advances in structural biology, such as high-throughput crystallography and NMR spectroscopy, together with better prediction and simulation, have lead to a marked increase in the number of proteins for which the three-dimensional atomic structure is known. Repositories for structural information, such as the PDB [2], have grown dramatically. Information about more proteins creates the need to look at large collections of molecules, requiring quick judgement.

Our technique, detailed in §3, relies on filtering of both the shape and surface fields to form abstractions. We describe how we adapt standard methods for shape smoothing by explicitly removing regions where the methods are likely to fail. §3.2 describes how surface decals can be used to display removed features, as well as other information. §3.3 describes the methods we use for more effectively portraying scalar fields on the surfaces, and §3.4 describes how stylized rendering techniques are used.

1.1 Background

Because proteins predominantly interact with other molecules non-covalently, their atomic forces effectively create shells that must fit together. The metaphor of a “lock and key” dates back over a century — [33] cites an 1894 paper by Fisher. While flexible water balloons or bean bags may be more appropriate metaphors than rigid pieces of metal, the central property governing how proteins interact is the non-penetration of their boundaries. Because the forces fall off rapidly with distance, treating the boundary as a surface (in the mathematical sense) is appropriate.

The definition of this geometric boundary surface is varied. The pioneering work of Richards (see [8] for a historical survey) introduced functional notions of these surfaces. The notion of solvent accessibility leads to molecular surfaces that provide a smooth description of the boundary. The solvent accessible surface is the surface that a spherical probe can contact without intersecting the molecule. Connolly provided practical methods for sampling these surfaces [6, 7], which have subsequently been refined in both efficiency and quality [3, 29, 39]. Even more recent works provide methods, such as in [5], that can produce high quality representations of these surfaces.

The geometric molecular surface is only one of many properties that contribute to protein interactions. There is considerable evidence that electrostatics play a critical role in protein-protein interactions (see [32] for a survey of early work). Other properties that influence interactions include hydrogen bonds (and a molecule’s ability to form them), polarity, and hydrophobicity. Because these properties fall off rapidly with distance, it is reasonable to consider them as scalar fields on the molecular surface when studying molecular interactions.
Molecules are continually in motion. In proteins, the motions range from thermal vibrations to large conformational changes. This incessant movement, coupled with uncertainty in the measurement of the atomic coordinates, means that any static configuration of a molecule is merely a snapshot of its possible state. Therefore, smaller details of the shape and other fields of the molecule are of reduced significance as they are likely changing continuously.

2 RELATED WORK

2.1 Molecular Visualization

Because of the importance of molecular shape, structural biologists have depended on visual tools from the beginning. Visual tools predate computers and continue to be developed to this day (see [8, 35, 16] for historically oriented surveys). Current state-of-the-art systems, such as Chimera [27], PyMol[11], and their competitors, provide large feature sets giving many options for the display of molecules.

Any visualization of a molecule necessarily involves some degree of abstraction. The field has developed a range of visual representations that provide different levels of abstraction; see [16] for a survey. For showing the internal composition of a molecule, many abstractions exist ranging from models that show every atom and bond, such as a ball and stick model, to highly abstracted representations, such as ribbon diagrams (Fig 2). These highly abstracted diagrams are valuable for providing a summary of a large molecule. However, because they do not indicate the “exterior” shape of the molecule, they provide less help in studying how the molecule would interact with others. Our methods provide such abstract representations for external properties.

There are two primary ways for showing the exterior shape of a molecule: space filling diagrams, where each atom is drawn as a solid sphere, and solvent-accessible surfaces, or molecular surfaces. Both views provide the shape of the molecule, however they provide it with a large amount of atomic-scale detail. Such detail is problematic as it can obscure larger scale phenomena and hinder effective portrayal of the shape. Raising the size of the probe sphere leads to molecular surfaces that exclude smaller crevices but may lose important pockets without discarding distracting bumps (Figure 3). Our methods can retain important features while reducing distracting detail.

Surface simplification (see [23] for a survey) creates approximate models with fewer polygons. These methods are useful in improving efficiency while preserving the appearance. Simplification is an essential part of large molecule surface display [13, 28]. In contrast, our approach seeks to alter the appearance to be more abstract, and does not necessarily provide a performance benefit, although abstracted surfaces are amenable to simplification. [13] applies smoothing, similar to our approach, to reduce the blocky appearance of coarse models of large molecules.

Display of other spatio-physico-chemical properties by color coding molecular surfaces became common as soon as surface representations were readily available. An early example was GRASP [24], which showed electrostatic potentials on surfaces. [4] unfolded the surfaces to better show their property distributions. Our methods provide abstracted display of these properties as well as molecular shape.

Work on displaying molecular motion shows the uncertainty in molecular shape. [21] shows uncertainty and vibrational motion by blurring standard representations, and [31] clusters states to provide visual representations of ranges of conformations. [18] uses a combination of point-based rendering and random displacement to convey surface uncertainty in volumetric data. While our work does not explicitly depict motion, it does convey a sense of uncertainty through the lack of detail.

The work of biochemist and artist David Goodsell inspires us by showing the merit of using artistically stylized depictions of molecules. His stunning figures require considerable artistic talent and effort to create. In [15] he describes a system for image processing molecular graphics that simulates a black-and-white line-art look, but which makes no abstraction of the shape.

2.2 Conveying Shape

The complexity of molecular shape is hard to convey, as some spatial cues such as size and familiarity are not applicable. To help assist in shape comprehension, biochemists often rely on motion and stereo display to enhance more standard graphics cues, such as lighting, fog, and depth-cueing.

Texture can be an effective cue for aiding in shape perception [17]. To date, it has not been applied in large-molecule visualizations as the molecular surfaces or space filling views are ill-suited for texturing. Our abstracted views make use of texture.

Clever lighting design can greatly enhance the perception of local shape [22]. To provide for better comprehension of global shape, molecular visualizations have recently begun to use both global illumination and stylized shading. QuteMol [34] demonstrates the effectiveness of these techniques and shows how they can be implemented efficiently in hardware. However, QuteMol does not provide for shape abstraction; it provides only all-atom displays or space-filling diagrams. By providing abstracted surfaces, our approach can more easily apply a range of stylized depiction effects.

[26] shows how identified features can be better presented using stylized display. We apply this idea in a new domain and extend it by explicitly removing the features and only portraying them in a stylized fashion.
In studying (i.e. larger than atomic scale interactions), not of particular interest, but need not be uniform. Properties are associated with mesh vertices; scalar fields are sampled at these points before abstraction.

The choice of the scale of “small” is chosen to be smaller than a residue, but larger than an atom. Our implementation uses Taubin’s filter [36, 37] as it is efficient and sufficiently volume-preserving. Local operations are performed on vertices lying in disc surrounding a given point, with importance falling off with the inverse of distance, as given by Fujiwara [12]. We have empirically determined the filter parameters $\lambda = .8$ and $\mu = -.87$, disc radius 4 angstroms, and 10 iterations to correspond to the desired feature size to remove. These parameters are a function of the scale at which we are interested in studying (i.e. larger than atomic scale interactions), not of particular molecules. All examples use the same parameter values.

The smoothing process creates a potential problem. “Mid-sized” features that are larger than what is removed reliably by the filter are often distorted by it. These features, such as peaks formed by protruding chemical groups and small divots, may be biologically significant. While a more sophisticated filtering mechanism might better preserve them, even undistorted, these features are undesirable for abstraction as they are still difficult to portray.

To handle mid-sized features, we apply a different strategy shown in Figure 4. Our approach identifies the features and removes them from the mesh, leaving a smoother surface. However, it remembers that a feature was removed and depicts that feature using a surface marking. This approach has the advantages that it avoids artifacts from filtering and provides more control over how features are displayed. Small surface markings are better for abstracted representations than small bumps and divots because they do not detract from the overall shape and are visible from a wide range of viewpoints (see Figure 11).

3.1 Removing Mid-Sized Features

Mid-sized features are identified and removed. At present, our methods identify bumps and bowls that are large enough to be potentially interesting, but small enough to be problematic for smoothing. Similar approaches could be applied for other shape features, such as ridges and valleys.

Our process for removing bumps and bowls consists of several steps, illustrated in Figure 4. Features are identified by finding points of that have high curvature after smoothing, which is indicative of a filtering artifact. An initial round of smoothing is applied specifically for feature detection. Vertices whose curvature are outliers are chosen as features. The system computes the 10th percentile of the absolute value of principle curvatures ($\kappa_1$ and $\kappa_2$) for all vertices over the mesh. Vertices are chosen to be outliers if their Gaussian curvature ($K = \kappa_1 \kappa_2$) is greater than $P$ times the square of the 10th percentile principle curvature. Empirically, we have chosen $P = 30$ to provide a good balance between mesh smoothness and overly aggressive feature removal. The exact value of this parameter does not matter since precise identification is unimportant; an excess or missed point is likely to be grouped with another point in a later stage.

For each of these seed vertices, our system constructs a group containing other vertices within 2.5 angstroms along the surface. This distance was empirically found to correspond to the approximate size of individual mid-size features. If groups overlap, then it is likely that they are larger aggregates of individual features on the original mesh, so the process repeatedly merges groups until no overlaps are found.

This step results in a collection of patches on the surface representing mid-sized features. These regions are then removed from the original, unsmoothed mesh. To “sand them off,” our system removes the majority of the vertices in the region and simultaneously “deflates” those that remain. To accomplish both, it first sorts the vertices according to how far away they are from their nearest seed vertex. It then takes the closest 80% and removes them, one by one, by edge contracting with their closest neighbor in the graph. This ensures that smaller triangles will be removed first, and also that vertices will be removed top-down, in the case of peaks, or bottom-up for divots.

The edge contraction process produces a mesh with its mid-sized features “sanded off” but the removal process often negatively impacts mesh connectivity, leaving many high-order vertices. Our system performs edge flips to improve the mesh by identifying high order vertices and for each one flipping its outgoing edge that is connected to the highest order neighbor. It also finds extremely low-order vertices and contracts the outgoing edge connecting to their lowest order neighbor.

After these methods remove mid-sized features, the resulting surface is smoothed.

3.2 Decaling

We would like to create surface markings that are independent of the underlying triangulation. Otherwise, a coarse or uneven triangulation might lead to jagged, irregular shaped markings.

Texture mapping provides for surface markings independent of triangulation, but requires a parameterization of the surface to provide texture coordinates. The molecular surfaces are difficult to parameterize globally. We apply the approach of [30] to place textures on regions of the surface. Their approach creates a local parameterization of a region of the surface. This approach works well for our needs because our abstracted surface is relatively smooth, and because the markings we wish to apply are local.

3.2.1 Decal Parameterization

[30] use a discrete exponential map to create a local parameterization of a surface in the neighborhood of a point. Exponential maps take a point on the surface and map the surface surrounding that point to its tangent plane (see Figure 5), in a manner that yields mappings that pre-
serve distances well. That plane serves as a local parameterization of the surface according to that map. To apply it within our molecular abstraction process, we need to automate the process of choosing the seed point, and of limiting the mapped region. These issues are challenging because poor choices can lead to parameterizations that distort the textures as they get further from the seed point.

3.2.1) and builds a parameterization around it. This parameterization determines where each of the vertices in the region lie in the texture plane, providing a 2D mesh that can be drawn on that plane. This patch is drawn to a texture such that the region outside of it is made transparent with alpha blending.

3.2.2 Choosing Decal Placement

We consider two types of markings: fixed sized glyphs centered at a point, and arbitrary shaped regions. The former are used in our system to display symbols, such as circles and checks, to denote various features on the surface. To create a glyph decal, the point position is used as the seed for creating the parameterization.

Regions are represented as a subset of the mesh vertices. To create a decal corresponding to a region on the surface, our approach selects the best vertex (using the criteria in §3.2.1) and builds a parameterization around it. This parameterization determines where each of the vertices in the region lie in the texture plane, providing a 2D mesh that can be drawn on that plane. This patch is drawn to a texture such that the region outside of it is made transparent with alpha blending.

The shape of a feature may have been distorted by both the filtering operations to create the smooth mesh and the mapping process, which may lead to patches with small holes, disconnected or poorly-connected vertices, and a jagged boundary. Removing these artifacts leads to abstracted markings that not only prevent problems in display, but also fit better into an abstracted representation and dispel any illusion that the fine details of the patch boundary are significant.

3.2.3 Using Decals

Our system abstracts patches in a number of steps. First, it applies standard binary image processing operations adapted to the non-uniform lattice of the 2D mesh. We use morphological operations [14] to remove outlying points and fill in small niches and holes. Dilation and erosion operators are defined based on the neighbors of a vertex. One step of dilation expands the patch out to include all immediate neighbors of the outermost vertices, while one step of erosion contracts the patch to remove all outermost vertices.

Rather than defining larger structuring elements, these immediate connectivity operators are applied repeatedly. We use 4 iterations of the close operator (dilation followed by erosion) to provide a good balance of problem removal and shape preservation.

Morphological operators may leave thin threads and bridges, as shown in Figure 6. These are removed by eliminating vertices with only one connected neighbor, and by expanding the patch around bridge vertices, which are defined as any vertex in the patch that has at least two neighbors not in the patch, and which do not themselves share a neighbor that is not in the patch.

After these cleaning steps, our system then finds all closed loops that lie on the border of the patch. This boundary is then smoothed (in the 2D map) by applying a low-pass filter to the 2D positions in the chains. This boundary is drawn with a stroke around its edge and the enclosed region filled, either with a flat color or with a texture defined over the plane. See Figure 7 for an example.

Fig. 4. The molecular surface (a) is first smoothed, then “mid-sized” features are identified (b). Those features are removed from the original surface (c), then smoothing is applied and decals are used to represent the removed features (d).

Fig. 5. We texture the surface using local parameterizations generated using an exponential map. First, a plane is constructed tangent to the desired position of the texture. Next, points surrounding that point (here in dark red) are mapped to that plane. Finally, the texture is placed on the surface according to that map.

Fig. 6. At left, a vertex in the patch has only one other neighbor that is also in the patch. We remove these. At right, a vertex, denoted by a “*”, joins two otherwise locally disconnected sets. We will add its neighbors, denoted by a “+”, to the patch.

Fig. 7. At left, a patch before boundary smoothing. Nodes on the boundary are placed directly on vertices on the mesh, leaving a jagged exterior. At right, after smoothing.
3.3 Abstracting Surface Fields

Many important properties beyond the atomic forces that form the molecular surfaces, such as electrostatic charge and hydrophobicity, are typically represented as scalar fields on the molecular surface and displayed using pseudo-coloring. Such properties suffer from the same profusion of small details as the shape itself, with similar issues in comprehension and display at small size.

Therefore, we abstract the scalar fields on the surface of the molecule. The scalar properties are attached to vertices before any simplification. This is particularly important because the true geometry determines the value (i.e., the position used to sample a volumetric scalar field such as electrostatic charge). Simplification should account for the real geometry in finding features (such as regions) in the scalar field as property values attached to vertices can be moved by the shape simplification. While this will distort the shape of the field, significant features of the field, such as regions of large magnitude, will remain with approximately the same shape and value.

The field abstraction process aims to remove small, less relevant details but to also preserve the coherent regions where the field has a definite value. Therefore, we apply a boundary-preserving low-pass filter to the surface scalar field. Specifically, we adapt the bilateral filter[38] to the irregular lattice of the triangle mesh.

As in the image case, the bilateral filter computes a new value at a vertex by taking a weighted average of the other vertices in its neighborhood, where these weights are determined by using both the spatial and value differences. We achieve larger kernel sizes by iterated application of smaller ones.

The general formula for our bilateral filter at iteration $i$ for vertex $v$ with value $val_i(v)$, and where $d(v,w)$ represents Euclidean distance from vertex $v$ to $w$, is:

$$val_i(v) \propto \sum_{w|w \text{ neighbors } v} \frac{val_{i-1}(w) \cdot e^{-\frac{d^2(v,w)}{2\sigma^2}} \cdot e^{-\frac{1}{2} \text{weight} \cdot \frac{1}{\epsilon^2} \sum_{w|w \text{ neighbors } v} \text{val}(w) \text{val}(v)}}{\text{weight}}$$  \hspace{1cm} (1)

Thus, our system applies a Gaussian filter for both distance and value-similarity weights. For the latter, though, we have found that by using a larger kernel for the first few iterations and then progressively reducing it at later iterations, we can prevent areas of uniform value from completely diffusing into areas of different value.

Our method to adapt kernel size is to use a kernel proportional to the standard deviation over the values at each vertex. Since the variance of the values themselves will be converging as a result of smoothing, this results in a progressively smaller sigma, which in turn gives higher weight to the value-similarity kernel.

Our system iterates until asymptotic convergence, which is reached when the average over all vertices $v$ of $val_i(v) - val_{i-1}(v)$ is less than $\epsilon$. In our experiments, $\epsilon = .005$.

When a mid-sized feature is removed, scalar field information contained in that feature is lost. This information is potentially important: for example, highly charged protrusions may be biologically significant. To preserve this important field information, the removal process associates a field value with the decal representing a removed feature. The value is determined by averaging the vertices from the removed features.

3.4 Display of Abstracted Surfaces

The resulting abstracted surfaces still provide a shape display problem. The visual presentations must not impede the use of the surface markings indicating small shape features. Also, we prefer a visual style consistent with the abstraction, rather than the “realistic” shiny plastic more commonly used to display molecular surfaces. Therefore, our system’s primary display is stylized. Quatem [34] showed the utility of stylized shading for molecular depiction. We apply several of their concepts to molecular surfaces. Our system also applies the stylized rendering to non-abstracted surface models (seen in many figures throughout the paper).

To enhance shape portrayal, we apply per-pixel silhouette shading [20], which sets $\text{brightness} = 1 - p^{-\frac{\text{area}(v,w)}{\epsilon^2}}$, where $\epsilon$ is the z component of the surface normal and $p$ is a tunable constant that we have set to .3 in our figures. This shading sets pixels on faces that orient directly toward the viewer at maximal brightness, with decreasing brightness as the face normal orients away.
Ambient occlusion (AO) lighting [20] is applied as it accentuates global shape. As pointed out by [34], the regions made darker by ambient occlusion because of lower lighting accessibility are related to the regions with lower chemical accessibility. Interior points of clefts and pockets are made darker. Our implementation of AO uses the graphics hardware to sample light directions.

As a final step, our system strokes along contours of the mesh, which can be defined as those edges that border both a front-facing and a back-facing face. This not only enhances shape perception, but gives a stylized look that gives a constant reminder of the degree of abstraction in the representation. The smoothness of abstracted surfaces makes more sophisticated contouring methods unnecessary. Experiments with Suggestive Contours [10, 9], show that they add few contours beyond the simple ones on our surfaces, and that these additional contours were typically small enough to be difficult to notice.

As in traditional molecular surface display, scalar fields are indicated on the surface by pseudo-coloring. For the examples shown in this paper, electrostatic charge is displayed using the red to blue scale that is commonly used. Colors for other decals are chosen such that they can be seen when overlayed on these colors.

4 RESULTS

We have implemented our molecular surface abstraction techniques in our bespoke visualization testbed that runs under Windows on PCs. Our system relies on standard tools for computing the surfaces and other properties. For all examples in this paper, we use MSMS [29] to generate the molecular surface meshes, APBS [1] to compute electrostatic charge distributions, and an implementation of Ligsite [19] to identify putative binding pockets.

We emphasize that abstraction is not the same as changing the probe size. Larger probe sizes will fill in crevices that may be important pockets, while leaving bumpy details that detract from comprehension as shown in Figure 3. Indeed, abstraction can be applied to surfaces generated with any probe size. All examples in this paper were generated using a probe size of 1.5 Angstroms, (except Figure 3b).

Figures throughout the paper show the results of our methods applied to various molecules. Figure 10 shows how important aspects of the molecule are made clear by abstraction. Figure 12 provides a gallery of examples.

Major shape features, such as pockets and clefts, become very clear on abstracted surfaces because there are fewer small details to distract the viewer, and the smoothness allows silhouette shading, contouring and ambient occlusion lighting to emphasize the shape.

The asymptotic complexity of our method scales linearly with the number of vertices of the input mesh, which scales (at worst) linearly with the number of atoms in the molecule. This was confirmed empirically in our performance evaluation.

To assess performance of our prototype, we randomly selected 60 proteins from the Astex test set [25]. When determining the timings for the abstraction process, we consider smoothing, decal construction, and surface field relaxation; but not the preprocessors to determine the initial mesh, electrostatic charges, or binding pockets. We also exclude the time required to load data from disk and compute ambient occlusion lighting. Timings were performed on a PC with an Athlon 4400 CPU, 2GB of RAM, and NVidia 7900GT graphics.

On our test set, the time needed to perform abstraction ranged between 7 and 113 seconds. These correspond to the smallest molecule in the set (1CBS, 137 residues, 1092 atoms) and the largest (1CX2, 2200, 21764). The expected linear performance scaling was observed.

Once abstracted, the models are displayed in real time using the graphics hardware. Our system maintained at least 30fps on all molecules in the test set. Our ambient occlusion precomputation implementation is not realtime (about 10 seconds for the largest molecule in the test set), but is computed in parallel and displayed progressively as abstracted model is displayed initially. The abstracted model, including ambient occlusion lighting, could be precomputed and stored.

5 DISCUSSION

The symbolic display of smaller features has a number of advantages. It makes shape features more readily apparent from a wider range of viewing directions (Fig 11). Symbols are visible in static displays, while shape features often are only obvious in regular displays when the object is moving. At small sizes, sampling issues may make small geometric features difficult to display. With surface textures, texture sampling hardware can perform sampling using mip-mapping, and decals can be omitted at very small sizes.

Displaying symbols as decals on the surface provides a mechanism for indicating a variety of properties about the underlying molecule. Such decaling would be difficult on non-abstracted surfaces: their non-smoothness would make parameterization difficult, and the small features would obscure the symbols with clutter and occlusion. Because the symbolic display is imprecise, they fit in better with the abstracted surfaces as abstracted surfaces imply a lack of positional precision.

Abstracted surfaces are also more amenable to mesh decimation for performance enhancement. This can be particularly important when displaying multiple molecules simultaneously in an interactive system.
Together, these features of abstracted surfaces suggest that they will be useful in creating gallery views that allow a number of molecules to be shown simultaneously for browsing and comparison. Without a reduction in detail, a gallery of surfaces would be overwhelming. In a gallery display, the size and resolution of each molecules depiction is limited. There is less opportunity for interactively rotating each molecule to find views that show shape features.

Many of the the limitations of our initial prototype should be addressed. Better shape feature finding would allow us to translate more of the shape into surface symbols to provide further abstraction. Incorporating dynamics would help us better target unstable features for removal. Better decal design should include textures that help convey shape more precisely. And methods are required to apply decals correctly on decimated meshes and patches with non-planar topology. To be truly useful, abstracted surfaces will need to be integrated into existing tools and workflows.

Our abstraction has been applied to study features at a specific scale. Exploring other scales would require retuning the methods, and possibly designing a new set of feature detectors. Studying molecules at scales much different than the atomic level interactions we consider, such as macro-molecular assemblies, would provide more challenges, including performance.

The most important step for our work is to assess how effective these representations are for scientists. To date, our testing of abstracted molecular surfaces with our biochemist collaborators has been limited, so our observations are anecdotal. In all cases, their initial reactions were extremely positive. They immediately appreciated the simplified views. On molecules familiar to them the views matched what they “expected” them to look like. In several cases they would make comments like “I never noticed that before, I wonder...” which is particularly encouraging as it implies a new way of looking at things might lead to new hypotheses. In viewing unfamiliar molecules, they were able to spot important functional areas quickly.

ACKNOWLEDGEMENTS

We thank George Phillips Jr. and his lab for their guidance in this project. Aaron Bryden helped with software pipeline issues and PyMol figures, and Cody Robson contributed some of our mesh processing implementation. Nick Reiter provided molecules and feedback. Rachel Heck assisted with video production. Cipriano was supported by NIH training grant NLM-5T15LM007359.

REFERENCES

Fig. 12. A gallery of example proteins of various sizes, shown before and after abstraction. Traditional images, rendered with [11], show molecular surfaces for the proteins, and spheres for the ligands. Stylized images use our rendering techniques on non-abstracted surfaces.