Multi-Modal Perceptualization of Volumetric Data and Its Application to Molecular Docking

Ross Maciejewski Seungmoon Choi David S. Ebert Hong Z. Tan
School of ECE, Purdue University, USA
E-mail: {rmacieje, chois, ebertd, hongtan}@purdue.edu

Abstract

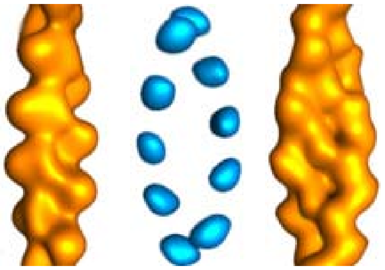
In this paper, we present a multi-modal data perceptualization system used to analyze the benefits of augmenting a volume docking problem with other perceptual cues, particularly stereoscopic vision and haptic rendering. This work focuses on the problem of matching complex three-dimensional shapes in order to reproduce known configurations. Specifically, we focus on the docking of two proteins, actin and cofilin, responsible for cellular locomotion. Users were shown examples of cofilin combining with actin and asked to reproduce this match. Accuracy of the match and completion time were measured and analyzed in order to quantify the benefits of augmenting tools for such a task.

1. Introduction

Starting in the early nineties, a push for data perceptualization resulted in the development of many haptic rendering systems. Early systems [1, 5] used the local gradient as the surface normal and force transfer functions[1]. Newer systems have incorporated proxy-based haptic rendering techniques for volumetric data rendering [4, 6]. A pioneering work where both visual and haptic perceptual cues were used was Project GROPE [3] developed at the University of North Carolina, Chapel Hill. While recently, one of the most popular molecular visualization packages, Visual Molecular Dynamics (VMD), has also been augmented with force feedback [10].

Despite the significant progress in computational models and techniques for data perceptualization, much work remains for the quantitative evaluation of data perceptualization systems in terms of their effectiveness in transmitting information to the user. Moreover, researchers and scientists have been slow in adopting the new technologies. In light of this, there is a pressing need to quantify the usefulness of data visualization systems in order to demonstrate their applicability to scientists.

Our research group recently developed an Interactive Volume Illustration System (IVIS) [9] to create illustrations of three-dimensional datasets. IVIS provides a graphical user interface in which the user can easily control the shape of a transfer function. By applying the user-defined transfer function to the dataset, IVIS can instantaneously update the visual representation, enabling the user to interactively explore the dataset.



(a) Actin. (b) Cofilin. (c) Combined.

Figure 1: Volumetric datasets for actin and cofilin.

We have extended IVIS into an *Interactive Volume Perceptualization System (IVPS)* providing 1) the simultaneous rendering of multiple volumetric datasets, 2) stereoscopic images using active stereo vision, 3) the sense of touch via a force-feedback haptic interface, and 4) interactive transfer functions for both vision and touch. IVPS was used to perceptualize the correct docking configuration of actin and cofilin (essential proteins for cell-motility; see Figure 1 for examples and [8] for details). The effects of the sensory modalities added to IVPS were quantitatively assessed by a psychophysical experiment where a subject was asked to use IVPS to find the best docking configuration of cofilin onto actin.

The remainder of this paper is as follows. Section 2 describes the IVPS rendering techniques. The experimental design is presented in Section 3, and results are summarized in Section 4. We conclude the paper in Section 5.

2. IVPS: Interactive Volume Perceptualization System

The visual renderer of IVPS is based on IVIS [9]. IVIS is a texture-based volume renderer that allows the user to interactively control transfer functions and explore their data. IVIS takes advantage of a graphic processing unit's

---

[1] A transfer function refers to a mapping of a data variable (eg, density at a voxel) to a display attribute (eg, opacity, force, etc.).

computational power when compared to using the CPU alone.

2.1. Visual Rendering

For data visualization, IVIS takes a scalar volumetric dataset as its input and incorporates the data into three-dimensional texture units. The gradient magnitude and direction at each voxel are preprocessed. Afterwards, an image is generated by slicing the volume with view-aligned quadrilaterals rendered in a back-to-front order. These slices are then combined to form the final image with opacity and color of each slice defined by a transfer function.

In IVPS, we augment the visualization capabilities of IVIS in two aspects: multiple volume rendering and stereoscopic vision. The stereoscopic approach used in our system is the parallel axis asymmetric frustum perspective projection method [2]. Implementation details of these aspects can be found in [7].

2.2. Haptic Rendering

Currently, IVPS can haptically render either one or two volumetric datasets. In both cases, one dataset is centered in the haptic interface workspace and remains stationary while force feedback is enabled. Since IVPS calculates and stores the gradient of each voxel of the dataset for visual rendering, forces for haptic rendering can be computed most effectively using the gradient force method introduced in [1]. We also explicitly incorporate a haptic transfer function in the force computation. For a probe of the haptic interface positioned at **x**, the force displayed was defined as:

$$\mathbf{F}(\mathbf{x}) = -Cg(V(\mathbf{x}))\nabla V(\mathbf{x}), \tag{1}$$

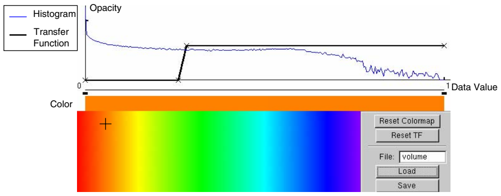
where $V(\cdot)$ is a value of the dataset at a given position, $g(\cdot)$ is a (normalized) transfer function, and C is a scaling factor. The force rendered is essentially of the form $\mathbf{F}(\mathbf{x}) = -k\mathbf{x}$, where k would be the stiffness coefficient of a spring. By inspection, k is the haptic transfer function, $g(V(\mathbf{x}))$, representing a variable stiffness coefficient dependent upon the probe's position in the dataset.

This force rendering method has been extended to multiple volumes. Of the two volumes, one is treated as a probe and the other as the volume being probed. In our example, the actin molecule was treated as a stationary object being probed in the workspace of the haptic interface. The cofilin dataset was attached to the stylus tip. The user manipulated the position of the cofilin dataset by moving the stylus. We computed the interaction forces between the two datasets by sampling a set of points representing the shape of the cofilin molecules in the dataset. Forces for each point were computed using Equation 1 and then averaged. It follows that for a given set of point samples of cofilin molecules, $\{\mathbf{x}_i | i = 1, \cdots, N\}$, our haptic rendering rule was:

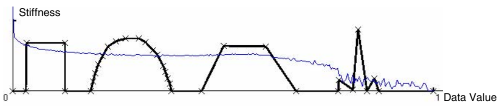
$$\mathbf{F} = -\frac{1}{N}\sum_{i=1}^{N} Cg(V(\mathbf{x}_i))\nabla V(\mathbf{x}_i), \tag{2}$$

where N is the number of sample points, and **F** is the force rendered by the haptic interface.

The sample points for a cofilin dataset were selected to be distributed evenly across the surface of each cofilin molecule. In order to achieve the typical haptic update rate of 1 kHz, the maximum number of samples for each cofilin dataset was limited to approximately 100 points. See [7] for further details on sampling.



(a) Visual transfer function



(b) Haptic transfer function.

Figure 2: IVPS interactive transfer function widgets.

2.3. Interactive Transfer Functions

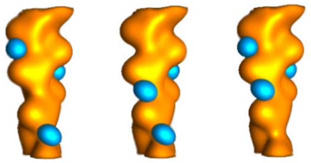
The user of IVPS can independently control transfer functions for both visual and haptic rendering using the widgets shown in Figure 2. For the visual transfer function (Figure 2(a)), the user is provided with settings for both color and opacity. The height of this function maps the associated data value to an opacity. Below this is a bar where the user may map colors, chosen from the color map, to different data values. Similarly for the haptic transfer function, users can create a piecewise linear transfer function which will map data values to a stiffness coefficient used in the force rendering equations. In Figure 2(b), we illustrate a mock transfer function showing how users may define their transfer function with any shape and value over the data range.

3. Experimental Design

In order to evaluate the benefits of added sensory modalities (stereo vision and touch) in IVPS, we performed a psychophysical experiment using the volume docking problem between actin and cofilin.

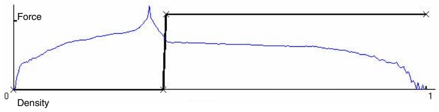
3.1. Apparatus

The IVPS used in our experiment consisted of a computer with an Intel Xeon 2.0 GHz CPU, an Nvidia Quadro FX 3000 graphics card, a stereo-capable monitor, stereo

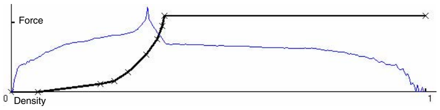


(a) Model 1. (b) Model 2. (c) Model 3.

Figure 3: Actin and cofilin models used in the experiment.



(a) $g_h(\cdot)$: For a hard surface contact.



(b) $g_s(\cdot)$: For a volumetric soft contact.

Figure 4: Haptic transfer functions used in the experiment.

shutter goggles coupled with an infared transmitter, and a force-feedback device. The shutter goggles used were Crystal EYES3 from StereoGraphics. The monitor was a Diamond Pro 2070^{SB} running at 120 Hz. The force-feedback device used was a PHANToM Desktop model from Sensable Technologies.

3.2. Stimuli

Ten subjects (S1 – S10) participated in the experiment. We used one actin model and three cofilin models shown in Figure 3 throughout the experiment. The three cofilin models were obtained by segmenting the data seen in Figure 1. Each model consisted of a unique combination of three cofilin molecules, differentiated by their position and orientation relative to one another. The volume data were sliced to make $37 \times 37 \times 37$ voxels for both the actin and cofilin datasets. Our preliminary experiment showed that finding the best docking configuration with these models was moderately difficult (see [7]). For haptic rendering, we designed two haptic transfer functions to simulate hard and soft surface contacts. The transfer functions were used together with the haptic rendering algorithm discussed in Section 2.2 creating haptic feedback. When the cofilins, attached to the PHANToM stylus, touched the actin, the transfer function $g_h(\mathbf{x})$, Figure 4(a), results in the sensation of touching a hard and rigid object. The function $g_s(\mathbf{x})$, Figure 4(b), rendered a relatively soft and permeable object. The volume dataset was mapped to a $200 \times 200 \times 200$ mm^3 cube inside the PHANToM workspace. A displacement of 5 mm of the PHANToM stylus corresponded to 1 voxel. Under this setup, IVPS rendered visual images at approximately 4 frames per second and haptic forces at 1 kHz.

Table 1: Experimental conditions.

Condition	C1	C2	C3	C4	C5	C6
Visual	Mono	Mono	Mono	Stereo	Stereo	Stereo
Haptics	None	$g_h(\cdot)$	$g_s(\cdot)$	None	$g_h(\cdot)$	$g_s(\cdot)$

3.3. Procedures

We tested six experimental conditions summarized in Table 1. For visual rendering, we compared the two cases of mono and stereo rendering. For haptic rendering, the two transfer functions for hard and soft contacts ($g_h(\mathbf{x})$ and $g_s(\mathbf{x})$) were used along with the no force-feedback condition. The order of experimental conditions presented to each subject was randomized. For each condition the subject was presented three different cofilin models. The models were presented in random order, and 18 trials were performed for the experiment. Each trial averaged forty-five seconds, and the total experiment ran for approximately thirty minutes. The subject's task was to control the position and orientation of the cofilin model with the PHANToM stylus and to explore the actin data until a best docking configuration was found. Once the subject felt a best fit was found, the subject was asked to press the 'Enter' key of a keyboard with their free hand. The computer program would then record the docking position and time taken.

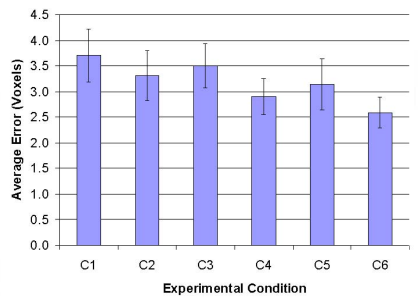
Before the experiment, each subject went through a training session to familiarize themselves with the PHANToM and learn what was considered a correct fit. Training sessions typically lasted thirty minutes. Training continued until the subject felt ready to perform the docking task. Three docking cases, different from those used in the main experiment, were used for training.

3.4. Data Analysis

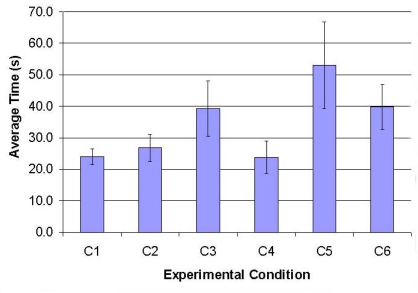
We used two metrics to evaluate performance: docking error and completion time. The docking error was defined as an average Euclidian distance between the correct configuration of the cofilin dataset and the final configuration chosen by the subject. Specifically, this is the average distance that each voxel is displaced from its correct fit position. The completion time was measured from the start of a trial to the time when the subject pressed the 'Enter' key to commit a response.

4. Experimental Results and Discussion

The results of the psychophysical experiment are summarized in Figure 5 as bar graphs. Figures 5(a) and 5(b) show the average docking error and completion time, respectively, over all trials for each condition, along with standard error. From the docking error data, we can observe that using stereo vision (C4 – C6) generally reduced the error as compared to mono vision (C1 – C3). Using stereo vision (C4) showed a statistically significant decrease in docking error compared to using only mono vision (C1). C4 reduced the error found in C1 by 0.8 voxels,



(a) Docking error.



(b) Completion time.

Figure 5: Experimental results.

which corresponds to approximately 4 mm in the PHANToM workspace. Also, using all the enhanced features of IVPS (C6 with stereo vision and $g_s(\cdot)$) showed a statistically significant decrease in docking error compared to using mono vision (C1). C6 reduced the error found in C1 by 1.1 voxels, which corresponds to approximately 5.5 mm in the PHANToM workspace. In terms of haptic rendering conditions, it is not clear which haptic rendering method produced the most accurate fit as no other conditions showed a statistical significance compared to C1. However, the results show a trend toward better accuracy using $g_s(\cdot)$ with stereo which we plan to investigate further.

From the completion time data shown in Figure 5(b), we can observe that adding force-feedback tended to increase the overall response time (C2, C3, C5, and C6) as compared to the purely visual conditions (C1 and C4). Each of C3, C5 and C6 indicated a statistically significant increase in completion time as compared to C1. This can be explained by the fact that haptic perception is achieved by local and sequential explorations of an object, while visual perception is global and parallel.

5. Conclusions

We have presented an Interactive Volume Perceptualization System that incorporates both stereoscopic and haptic rendering. Interactive visual and haptic transfer functions provided users with the ability to see and feel their data in different manners. The addition of haptic rendering to our system did not show a statistically significant benefit in reducing docking error, although it did reduce the overall mean docking errors. It also increased the time it took the subjects to find a docking position. However, we hasten to point out that our results do not imply that haptic rendering is not useful in a data perceptualization task. There were many more haptic transfer functions that could have been used for the current dataset. The state-of-the-art in haptics research does not provide general guidelines on how to design haptic transfer functions that can aid the user of a data perceptualization system to perform a task well.

Acknowledgement

The authors would like to thank Nikolai Svakhine for providing the IVIS system, and Dr. Amy McGough and her lab for providing datasets and participants in this study. This material is based upon work supported by the National Science Foundation under Grant Nos. 0081581, 0121288, and 0328984. The second author has also been supported in part by the Envision Center at Purdue University.

References

[1] R. S. Avila and L. M. Sobierajski. A haptic interaction method for volume visualization. In *Proceedings of the Conference on Visualization*, pp. 197–204, 1996.

[2] P. Bourke. 3d stereo rendering using OpenGL (and GLUT). Published Online `http://astronomy.swin.edu.au/~pbourke/opengl/stereogl/`, May 2002.

[3] F. P. Brooks, M. Ouh-Young, J. J. Batter, and P. J. Kilpatrick. Project GROPE - haptic displays for scientific visualization. *Computer Graphics*, 24(4):177–185, 1990.

[4] M. Ikits, J. D. Brederson, C. Hansen, and C. Johnson. A constraint-based technique for haptic volume exploration. In *Proceedings of the Conference on Visualization*, pp. 263 – 269, 2003.

[5] H. Iwata and H. Noma. Volume haptization. In *Proceedings of the IEEE Virtual Reality Annual International Symposium*, pp. 16–23, 1993.

[6] K. Lundin, A. Ynnerman, and B. Gudmundsson. Proxy-based haptic feedback from volumetric density data. In *Proceedings of Eurohaptics*, pp. 104–109, 2002.

[7] R. Maciejewski. Exploring the value of adding haptic and stereoscopic rendering to volume rendering. Master's thesis, Electrical and Computer Engineering, Purdue University, 2004.

[8] A. McGough, B. Pope, W. Chiu, and A. Weeds. Cofilin changes the twist of f-actin: Implications for actin filament dynamics and cellular function. *The Journal of Cell Biology*, 138(4):771–781, 1997.

[9] N. Svakhine and D. S. Ebert. Interactive volume illustration and feature halos. In *Proceedings of the 11th Pacific Conference on Computer Graphics and Applications*, pp. 347–352, 2003.

[10] W. Wriggers and S. Birmanns. Interactive fitting augmented by force-feedback and virtual reality. *The Journal of Cell Biology*, 144:123–131, 2003.