3D Chromosome Rendering from Hi-C Data using Virtual Reality

Yixin Zhu Siddarth Selvaraj Philip Weber Jennifer Fang Jürgen P. Schulze Bing Ren
University of California, San Diego, La Jolla, CA, USA

ABSTRACT

Most genome browsers display DNA linearly, using single-dimensional depictions that are useful to examine certain epigenetic mechanisms such as DNA methylation. However, these representations are insufficient to visualize intra-chromosomal interactions and relationships between distal genome features. Relationships between DNA regions may be difficult to decipher or missed entirely if those regions are distant in one dimension but could be spatially proximal when mapped to three-dimensional space. For example, the visualization of enhancers folding over genes is only fully expressed in three-dimensional space. Thus, to accurately understand DNA behavior during gene expression, a means to model chromosomes is essential.

Using coordinates generated from Hi-C interaction frequency data, we have created interactive 3D models of whole chromosome structures and its respective domains. We have also rendered information on genomic features such as genes, CTCF binding sites, and enhancers. The goal of this article is to present the procedure, findings, and conclusions of our models and renderings.

Keywords: 3D Visualization, Data Exploration, Chromosome, Hi-C, Virtual Reality.

INTRODUCTION

DNA is a very simple yet complicated molecule. Even though it only consists of four basic nucleotides (A, C, G, and T), these molecules serve as the genetic blueprint for all living organisms. There are about 3 billion nucleotides in the human genome, and only a small fraction of that (estimated 2-3%) is believed to directly influence gene expression. The vast remainder is vaguely understood with many unknown areas. Thus, the identification of new genomic areas is extremely important and helpful for biologists trying to figure out the intricate workings of our DNA. The identification of new genomic areas, however, depends a great deal on the representation of the genome and the application. For most applications, linear sequences suffice [2, 13]. However, for other applications, more complex representations are needed [9]. We believe that three-dimensional models can be extremely beneficial for some of these other applications. For example, in order to determine how distal genomic areas on a chromosome interact and affect its structure, three-dimensional structures become essential. This is because proteins sometimes bind to distal parts of a chromosome which affects DNA transcription, ultimately influencing protein creation. When interactions between distal areas of a chromosome occur, it can be difficult to represent on a linear model. Three-dimensional structures are also very useful for identifying new patterns that are common or appear repetitively during inter-chromosomal interactions. These newly identified patterns are potential target genomic areas for further research that could lead to the discovery of some new functionality of that area or even the discovery of new genes. These are some of the driving motivations for creating a three-dimensional viewer.

The primary purpose of our application is to facilitate the identification and verification of the relationship between structure and function in chromosomes.

RELATED WORK

The article by William Stafford Noble et al. [1] establishes the importance of chromosome structure and interaction and its influence on function. This article also outlines three key ways to determine chromosome structure, one of which is the Hi-C method [17] that we used to obtain our data sets.

The work done by Rie Kawamura et al. [6] found that the visualization of the spatial positioning of SNRPN, UBE3A, and GABRB3 in the human genome suggests that the distances between alleles are related to nuclear organization and

gene expression. This relationship is difficult to represent with a linear model, but much simpler in three-dimensional space. Few generic three-dimensional viewing software applications exist and those that do (ADN-Viewer, RasMol, Swiss-PDB Viewer) are hard to use, highly technical, and not widely used.

INFRASTRUCTURE

The infrastructure of our system is quite complex with many different components. The two most notable are CalVR [14], our OpenSceneGraph-based [18] software framework for developing graphical applications, and the NexCAVE [15], the virtual reality system we use to run our application. In this section, we give a brief overview of both components.

CalVR is our own open source virtual reality engine. It is object oriented and written in C++. Functionality can be added through a simple plug-in system which allows compiling new modules separately from the main code. CalVR has built-in navigation algorithms, a 3D menu system, support for a variety of 3D display and tracking systems, as well as support for collaborative work over the internet.

The Calit2 NexCAVE is a cluster of monitors that simulates a virtual reality environment. It extends to a semi-circle that covers the users' field of view. The NexCAVE consists of 17 JVC polarized 3D TVs, arranged into five columns, and produces data resolution close to human visual acuity. The Calit2 NexCAVE has an effective resolution of roughly 10,000 x 1,500 pixels per eye. It is powered by 9 high end graphics PCs running under CentOS 6, with dual Nvidia 480 graphics cards. For user tracking and interaction, we use an optical tracking system from ART, which consists of two infrared cameras and wireless tracking targets.

DATA SETS

In this section, we discuss the data sets our project utilizes. A large part of the problem for 3D chromosome visualization is the generation of proper data. As mentioned before, we use Hi-C to first obtain interaction data about the target chromosome [5]. In a broad sense, Hi-C sequencing consists of first crosslinking the target chromosome, second utilizing restriction enzymes to cut up the target chromosome, and finally ligating the resulting pieces of chromatin that are closest to each other in space in order to generate interaction numbers. Next, we take the data resulted from Hi-C sequencing and applying the BACH/BACH-MIX algorithm [16]: a Bayesian approximation technique developed in our lab that generates three-dimensional coordinates from Hi-C. The result of the BACH algorithm is very detailed and highly accurate. Consequently, the output coordinates of the BACH algorithm are forty-four times the size of the input. Thus, if the Hi-C data is large, the resulting coordinate set will be even larger. Because of this, our rendering procedure needs to handle this accordingly and make optimizations where necessary. Figure 1 illustrates our workflow.

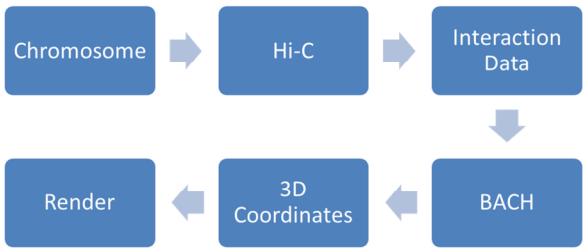


Figure 1: High level workflow of how we go from chromosome to 3D coordinates.

PROCEDURE

The rendering process contains a three-tier model of visualization (see Figure 2) where each subsequent tier is a more detailed segment of the preceding tier. At the highest level, the first tier models entire chromosomes consisting of interconnected cylinders. Each cylinder in the first tier represents a chromosomal domain that links to the second tier of visualization. The second tier models chromosome domains consisting of cylinders representing bins of 20,000 and 40,000 base pairs. The second tier also depicts genomic features – such as genes, enhancers, and CTCF binding sites– with different color schemes that elucidate interactions between regions. Each cylinder in the second tier links to the third tier, which is a genome browser showing the detailed linear annotations of the corresponding bin. As a stretch goal, the third tier could also render a helix structure of color coded base pairs. Each base pair will ideally consist of 20 triangles and will depict the linear annotations in a three-dimensional helix structure. Currently, this third tier is unimplemented.

In each tier, our models emphasize genomic areas of interest with an emphasis on promoters, enhancers, and CTCF binding sites. Each of these three genomic areas provides clues to the determination of structure, interaction, and function in chromosomes. For instance, promoters are typically distally close to identified genes and have even been seen as folding over genes since genes require promoters in order to begin transcription. Thus, if promoters are found in non-identified genomic areas, more interest should be given to those areas for further research. Enhancers enhance transcription by either making the process faster or more efficient in that it results in more mRNA. It, too, is closely related to identified genes. CTCF binding sites are the most interesting as they directly influence the structure of DNA and, consequently, directly influence gene expression. While CTCF binding sites have been known to both promote and repress gene expression, it is still unknown how or why [10]. By emphasizing and highlighting these three genomic areas we had hoped to find some interesting results that relate to the structure of chromosomes.

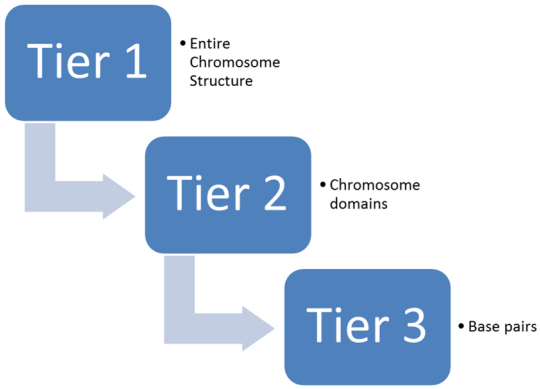
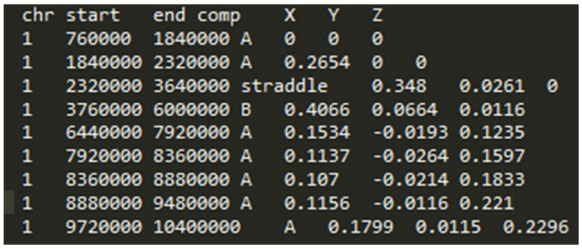


Figure 2: High level summary of the three tiers of visualization used in our models.

IMPLEMENTATION

This section aims to describe how our application transforms the data sets we receive to actual polygons that are rendered on the screen. The data files we use as input for our application are received in ASCII text files. An example of such input is shown in Figure 3.



```
chr start   end comp    X   Y   Z
1   760000  1840000 A   0   0   0
1   1840000 2320000 A   0.2654  0   0
1   2320000 3640000 straddle    0.348   0.0261  0
1   3760000 6000000 B   0.4066  0.0664  0.0116
1   6440000 7920000 A   0.1534  -0.0193 0.1235
1   7920000 8360000 A   0.1137  -0.0264 0.1597
1   8360000 8880000 A   0.107   -0.0214 0.1833
1   8880000 9480000 A   0.1156  -0.0116 0.221
1   9720000 10400000    A   0.1799  0.0115  0.2296
```

Figure 3: Snippet of a sample input file for our rendering algorithm.

Each row corresponds to a single data point coordinate that is to be represented in 3D space. The first column of each row determines which chromosome this data point belongs to. The next two columns define the start and end indices that depict which base pairs this data point represents. The fourth column specifies what type of chromatin the data point represents. The classification of type is discussed further in [5]. Finally, the last three columns represent the location of the data point in 3D space.

In order to represent these data points graphically, we decided to render an OpenSceneGraph sphere (osg::Sphere) for each data point location specified per row of the input files. These spheres are then connected with OpenSceneGraph cylinders (osg::Cylinder) to create a connected, chromosome-like structure. The colors of the spheres and cylinders are determined by the type of chromatin that the data point represents. In order to handle large data, we have an optimization that renders a capsule-shaped object (osg::Capsule) for every pair of data points to reduce the number of geometry and triangles that needs to be rendered. This methodology is used for both the tier 1 and the tier 2 models.

The implementation for transitioning between the two tiers of views has changed quite a bit from the initial conceived procedure. Initially, we had tried to implement a Google Maps-type transition where more detailed views appear as you zoom in closer to a target area. We had tried to implement this but were faced with the complication of how tier 2 domains were connected with each other. We have data for specific domains but not for how to join them because we do not have a means of measuring the 3D orientation of the boundaries. There is a lot of repeat and junk linker DNA between the domains that is hard enough to sequence let alone determine its 3D structure (if it even has a structure). This is still a hot area of research in the field of bioinformatics. For the purpose of our project, we decided to compromise by adding a separate loading menu where domains can then be loaded into the scene.

The highlighting of specific genomic areas in both tiers is determined via a separate input file that describes exactly which indices of the corresponding input files are what type of genomic area. For instance, active genomic areas are highlighted green, target genomic areas are highlighted blue, and the rest of the chromosome is colored red. The green highlight is emphasized further by rendering an extra sphere around the area with an opacity level of 0.4 so the content within the sphere can still be distinguished. The active genomic areas are the areas that influence structure such as CTCF sites, promoters, and enhancers. The target genomic area is an area of interest whose structure we wish to observe. We choose this implementation methodology so as to clearly and distinctively highlight which areas are potentially affecting structure and which areas are potentially being affected. We also thought about using labels to distinguish even further, but after a few prototypes we discovered that the labels often times intrude on the rendered structure of interest, blocking out important features. Thus, we decided to stick to color highlighting as it is less obtrusive.

During implementation, we found that the input files remain static for the most part so reading input files in each time became inefficient as data files got larger and larger. We started to save the scene that has already been rendered into an OBJ file that can be loaded in and read by any framework that accepts OBJ files. This became a standard menu option and has led to some interesting use cases that will be further discussed in later sections.

FINDINGS

After generating a number of different data sets for both mouse and human sequenced chromosomes we have found three distinctive patterns that are worth noting. The first (briefly mentioned before) is that the pattern of promoters

folding over genes was found to be a recurring model. The second is a new pattern that was noticed in at least three distinct models. In these models, enhancers were seen as if they were folding over genes, which is something that has not been noticed before. This could either be a coincidence or an area for further research that could potentially lead somewhere very interesting. The last pattern is perhaps the most interesting. For specific domains in male and female mouse chromosomes, we have found a direct correlation between the presence and absence of CTCF binding sites and the structure of the chromosome. See Figure 4 below for further details.

In this figure, the CTCF sites (highlighted in green) is seen to directly influence the structure of the chromosome in the target genomic area (blue). On the left, we have the absence of CTCF sites creating amalgamation of the target area whereas the presence of CTCF binding sites is creating a divergence of the target genomic area (right).

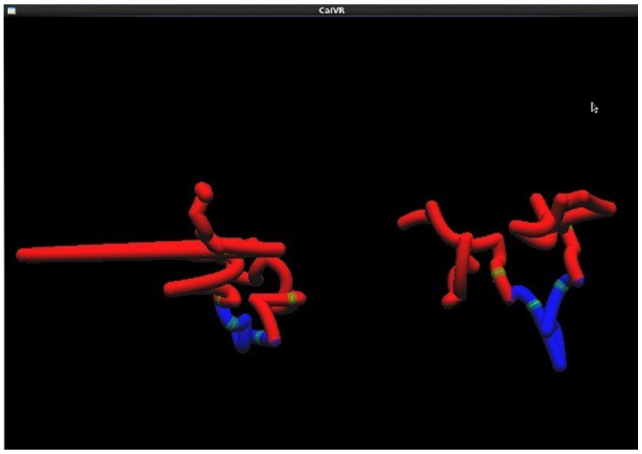


Figure 4: This figure shows a domain of chromosome 7 of male and female mouse chromosomes. The blue area is the target genomic area and green highlighted spheres depict the CTCF binding sites of interest. This image was rendered on a local desktop version showing the optimized low geometry setting.

The three patterns noted above are also just recurring patterns that in no way imply causation. These patterns are noted as useful and interesting for further investigation. The purpose of our application is to facilitate, not mediate, the verification and identification of the relationship between structure and function in chromosomes. These patterns were all discovered using only two distinct data sets of mouse and human chromosomes and their domains. We predict that there are many more useful patterns that will arise given more and/or different types of data sets.

DISCUSSION

While we would like to believe that our application is quite useful, there are inevitably many limitations and considerations that we wish to address in this section. One big limitation is the lack of visualization flow as defined by the three-tier model. What we mean by this is that ideally we would have the Google Map-type feel where zooming in and out of a chromosome would lead to higher or lower detailed views that depict all aspects of structure. However, as touched on briefly before, knowing how different domains are connected in 3D space is hard and thus, achieving this would consequently be hard as well. Our solution was to separate the loading of domains in another menu option, but in doing so we lose some of the benefits that virtual reality and immersive environments have to offer as it limits our scene. A good example would be if we spotted an abnormality in structure on the chromosome-tier view and wanted to examine further, we could not zoom in and view all the target domains by zooming in and selecting the area. We would have to load each of the affected domains as a distinct entity and place them side-by-side as best as we can.

Another limitation is the type and amount of data that our application can reasonably accept. Currently, the data sets are primarily generated using the Hi-C sequencing coupled with the BACH/BACH-MIX algorithm to gather coordinates. However, other sequencing methodologies can be accepted granted that the output data files match the format of our input files discussed earlier. If the format is met, then our application will be able to render a scene for any other

sequencing methodology granted that the data set is not too large. What we mean by too large is on the scale of hundreds of thousands. Although our application has optimizations to reduce the amount of geometry rendered given the size of our data set, there is still a point where the frame rate becomes an issue. All of our CalVR applications are measured for efficiency using a built-in system monitoring tool that shows frame rate, number of geometries, and memory usage (among many other things). When we stress tested our application, we found that as the size of our data sets got larger (hundreds of thousands) rendering became very slow and almost unusable. It is worth noting that for the majority of Hi-C + BACH generated data sets, we have not reached this threshold.

While we cannot make guarantees about other sequencing methodologies, we can safely say that our application accepts OBJ files. Briefly mentioned earlier in the implementation section, OBJ files were used as an optimization to prevent having to read in input files and calculating the geometry each time a chromosome needed to be loaded. Rather, after the initial read and calculation, the input file can be associated with an OBJ file that saves the data of the scene such that the next time the same model needs to be loaded, the application will just load from the OBJ file, thus avoiding redundant calculations. Unexpectedly, this led to a number of new extensions that tremendously improve availability. First, our application can be used to accept any OBJ file that has been pre-rendered either by our application or any other application. Second, the ability to output OBJ files (and other supported file types) can also be used to link to other applications or viewers. Third, we could reasonably extend our application to be able to run on a website as well as on specialized hardware. We cover this further in the next section, but the increase in availability allows us to provide an application that reaches a wider audience, one that does not have to be as tech savvy.

POTENTIAL APPLICATIONS

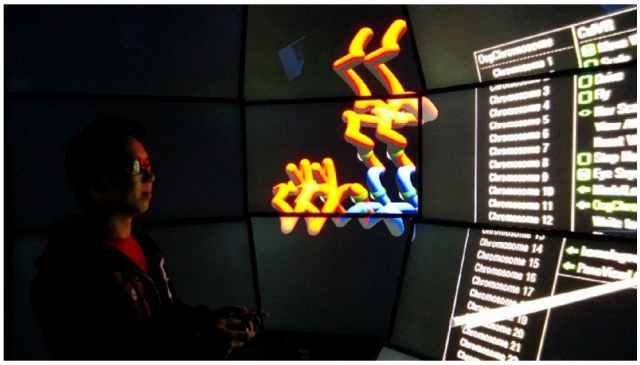


Figure 5: User in NexCAVE viewing domains of mouse chromosome 7. This is the same model that is shown in Figure 4 only rendered on the NexCAVE with full geometry rendering.

Our primary application is a data visualization tool to facilitate biologists understanding of our DNA. Figures 5 and 6 show a typical usage scenario. There is still much biologists have yet to understand about the function of the different parts of the chromosomes. Many theories exist in the literature for how to best proceed in identifying these remaining parts. One particular theory is that of the relationship between structure and function. There is a reason why chromosomes fold and unfold during different stages of meiosis, mitosis, and transcription. Different parts of chromosomes can interact with each other dynamically during runtime that can constitute a gene. If we think of a chromosome as a linear array from 0 to 3 billion where each element is an A, G, C, or T, then traditionally we would think of indices 500-1000 to be a gene for black hair. But what if, during transcription, due to present CTCF sites, indices 1000-1500 (originally thought to be junk DNA) folded over indices 5000-5500 and directly influenced what proteins were created as a result. This new structure can be identified by our application as a pattern and could potentially be discovered by biologists later on after some further research as a new gene. Or instead of a gene, it could be some inhibitor or enhancer of protein creation for a certain process that explains some causation between structure and function. There are many potential purposes our application can serve. But even if we stick to just existing functions that we already understand, our application can provide a means to visualize and verify the direct relationship between that function and the structure. An extension of this is to animate the models in order to show how the structure is affected during the entire process of some cellular activity such as transcription. This would, however, require the data to

be available. This would also bring up new problems but is still viable in the future as a possible extension.

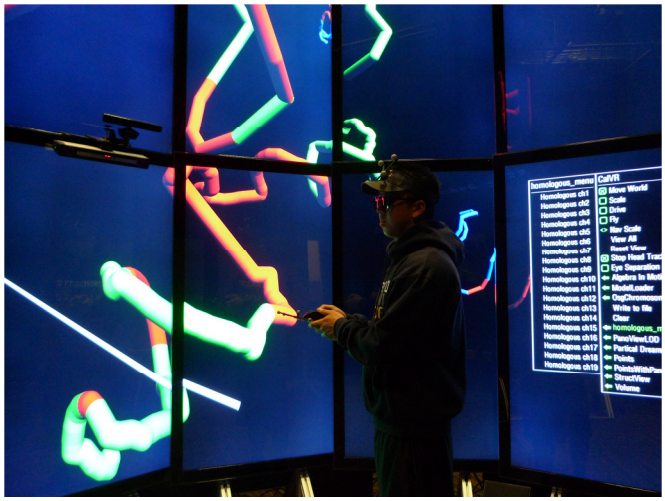


Figure 6: User with a number of different human chromosomes.

Another potential application is to apply our tool to cancer research. If a causation can be established between a distinctive structure with a distinctive function (cancerous cellular behavior), then the use of our application can help identify and verify cancerous structures. Ideally, if there are preventative patterns that can be recognized as well, our application can be used to identify those as well. In one study, secondary chromosome structures have been shown to lead to cancer and other diseases [3].

CONCLUSION

There are still many things we do not know about the human genome. Our application aims to facilitate spawning new areas of research to uncover potentially meaningful results that will add to our understanding of our genome. With the handful of data sets we had available we were already able to discover some potentially interesting patterns. Consequently, we have also uncovered many areas to improve on when interfacing with biologists during project demonstrations and usability tests. One of the biggest work items is to introduce more interactivity to allow more flexibility during visualization. An example of this is to be able to label and/or select certain regions of interest and have the information about that section saved for later research. Finally, we want to increase availability by extending our application to web-sites/browsers, a work item that was addressed earlier. Overall, we believe our application serves as a good initial tool for facilitating biologists' understanding of the relationship between chromosome structure, and function and we hope to work with more labs to see how else our tool can be improved to better serve their purposes.

ACKNOWLEDGEMENTS

We want to thank Calit2 programmer Andrew Prudhomme who was extremely helpful and supportive in providing the tools and knowledge necessary to complete this project. We would like to thank Alfred Tarng for the initial VR software environment setup. And we want to thank the researchers and scientists of the Ren lab for their constructive comments.

REFERENCES

[1] Blau, C. A., Noble, W. S., Mao, Y., Dekker, J., and Duan, Z.-J. "The Structure and Function of Chromatin and Chromosomes". Ch. 42, 434–440.

[2] Derks, S., Bosch, L. J., Niessen, H. E., Moerkerk, P. T., Van Den Bosch, S. M., Carvalho, B., Mongera, S., Voncken, J., Meijer, G. A., De Brune, A. P., Herman, J. G., and Van Engeland, M. 2009. "Promoter CPG island

hypermethylation- and h3k9me3 and h3k27me3-mediated epigenetic silencing targets the deleted in colon cancer (dcc) gene in colorectal carcinogenesis without affecting neighboring genes on chromosomal region 18q21". Carcinogenesis 30, 6, 1041– 1048.

[3] Dillon, L., Pierce, L., Ng, M., and WANG, Y.-H. 2013. "Role of DNA secondary structures in fragile site breakage along human chromosome 10". Hum Mol Genet. 22, 1443–1456.

[4] Goetze, S., Mateos-Langerak, J., Gierman, H. J., De Leeuw, W., Giromus, O., Indemans, M. H. G., Koster, J., Ondrej, V., Versteeg, R., and Van Driel, R. 2007. „The three-dimensional structure of human interphase chromosomes is related to the transcriptome map". Molecular and Cellular Biology 27, 12, 4475–4487.

[5] Hu, M., Deng, K., Qin, Z., Dixon, J., Selvaraj, S., Fang, J., Ren, B., and Liu, J. S. 2013. „Bayesian inference of spatial organizations of chromosomes". PLoS Comput Biol 9, 1 (01), e1002893.

[6] Kawamura, R., Tanabe, H., Wada, T., Saitoh, S., Fukushima, Y., and Wakui, K. 2012. "Visualization of the spatial positioning of the SNRPN, UBE3A, and GABRB3 genes in the normal human nucleus by three-color 3D fluorescence in situ hybridization". Chromosome Research 20, 6, 659–672.

[7] Lieberman-Aiden, E., Van Berkum, N. L., Williams, L., Imakaev, M., Ragoczy, T., Telling, A., Amit, I., Lajoie, B. R., Sabo, P. J., Dorschner, M. O., Sandstrom, R., Bernstein, B., Bender, M. A., Groudine, M., Gnirke, A., Stamatoyannopoulos, J., Mirny, L. A., Lander, E. S., and Dekker, J. 2009. „Comprehensive mapping of long-range interactions reveals folding principles of the human genome". Science 326, 5950, 289–293.

[8] Miele, A., and Dekker, J. 2008. „Long-range chromosomal interactions and gene regulation". Mol. BioSyst. 4, 1046–1057.

[9] Mistelli, T. 2007. "Beyond the sequence: cellular organization of genome function". Cell 128, 120, 787–800.

[10] Phillips, J., and Corces, V. 2009. "CTCF: Master weaver of the genome". Cell 137, 94, 1194–1211.

[11] Rousseau, M., Fraser, J., Ferraiuolo, M., Dostie, J., and Blanchette, M. 2011. "Three-dimensional modeling of chromatin structure from interaction frequency data using markov chain monte carlo sampling". BMC Bioinformatics 12, 1, 414.

[12] Taddei, A., and Gasser, S. M. 2012. "Structure and function in the budding yeast nucleus". Genetics 192, 1, 107–129.

[13] Teves, S. S., and Henikoff, S. 2011. "Heat shock reduces stalled RNA polymerase II and nucleosome turnover genome-wide". Genes and Development 25, 22, 2387–2397.

[14] Schulze, J.P., Prudhomme, A., Weber, P., and DeFanti, T.A., "CalVR: An Advanced Open Source Virtual Reality Software Framework", In Proceedings of IS&T/SPIE Electronic Imaging, The Engineering Reality of Virtual Reality, San Francisco, CA, February 4, 2013, ISBN 9780819494221

[15] DeFanti, T.A., Acevedo, D., Ainsworth, R.A., Brown, M.D., Cutchin, S., Dawe, G., Doerr, K.-U., Johnson, A., Knox, C., Kooima, R., Kuester, F., Leigh, J., Long, L., Otto, P., Petrovic, V., Ponto, K., Prudhomme, A., Rao, R., Renambot, L., Sandin, D.J., Schulze, J.P., Smarr, L., Srinivasan, M., Weber, P., and Wickham, G. "The Future of the CAVE", Central European Journal of Engineering, 1(1), 2011, ISSN 1896-1541

[16] Hu, M., Deng, K., Qin, Z., Dixon, J., Selvaraj, S., Fang, J., Ren, B., Liu, J.S. "Bayesian Inference of Spatial Organizations of Chromosomes", PLoS Computational Biology 9(1): e1002893, 2013

[17] Belton, J.M., McCord, R.P., Gibcus, J.H., Naumova, N., Zhan, Y., Dekker, J. "Hi-C: A Comprehensive Technique to Capture the Conformation of Genomes", Methods. 2012 Nov;58(3):268-76. doi: 10.1016/j.ymeth.2012.05.001. Epub May 29, 2012

[18] Wang, R., Qian, X. "OpenSceneGraph 3.0: Beginner's Guide", Packt Publishing, December 2010, ISBN 9781849512824