Bacterial Gene Neighborhood Investigation Environment: A Scalable Genome Visualization for Big Displays

BY

JILLIAN AURISANO
B.A., University of Chicago, 2006

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Chicago, 2014

Chicago, Illinois

Defense Committee:

Jason Leigh, Chair and Advisor
Andrew Johnson
Barry Goldman, Monsanto

This work is dedicated to Adam.

ACKNOWLEDGEMENTS

I'd like to thank:

- My advisors, Jason Leigh and Andy Johnson. Thank you for bringing me into the EVL, fundamentally changing the course of my life and teaching me everything I know about visualization, graphics, computer science research and THE FUTURE! It has been fun and I look forward to doing my PhD in the best possible place, the lack of oceans and illegality of 'on-top-of-Jeep' riding notwithstanding.

- Barry Goldman, David Bush, Niran Iyer, Shawn Stricklin, Arun Krishnan, Saritha Kuriakose and the rest of the computational biology team at Monsanto for bringing me onto your team and helping me put together this project. I learned so much from all of you and your ideas will carry my research forward for a long time!

- Everyone at the EVL for providing me with ideas, support, inspiration, and laughter. Special thanks to Khairi Reda for helping me pull together resources for perception and cognition, and its impact on visualization for big displays. You have given me lots of research ideas and been a great role-model for how to do this whole research thing. Special thanks to Dennis, Victor, Arthur, Alessandro, Brad, Steve, Tommy, Krishna, Luc and Lance for making me laugh even in the midst of buggy code. Also, for all the sushi and Munchkin®.

- Tanya Berger-Wolf for being such a wonderful mentor and teaching me that there is just about nothing better than spotting zebras at Mpala from atop a Jeep.

- My parents for endless, boundless, life-long encouragement and support. Also, thanks Dad for convincing me to (finally) take a CS course. You were right. I loved it and it was exactly what I was meant to do.

- Alessia, thank you for being such an amazing little kid! Most of this project was completed during the course of your existence, so you are likely imperceptible woven into the code and this text. In conclusion: BUUUUS!

- And Adam. I cannot hope to capture your contribution to all this in words here. Thank you!

JMRA

TABLE OF CONTENTS

CHAPTER PAGE

TABLE OF CONTENTS (continued)

CHAPTER PAGE

TABLE OF CONTENTS (continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (continued)

SUMMARY

In this thesis, I present a novel genome data visualization targeting an important area of genomics research: comparative bacterial gene neighborhood analysis. This approach demands scalable visualization designs that accommodate simultaneous comparison of hundreds of genomes at once.

Decreases in genome sequencing costs have driven a proliferation in the volume of genomic data. Thousands of complete genome sequences have been compiled on public databases and even larger volumes of data have been generated privately by independent research groups. In particular, rates of bacterial genome sequencing have accelerated, due to low sequencing costs. The comparative analysis of these genomes provides a new approach to the study of novel proteins and protein interactions. While automated analysis plays a significant role in the generation and analysis of this data, visualization is needed to bring experts into the data-mining loop, to verify the results of automated analysis and to detect patterns that are difficult to find through computation alone.

At the same time, advances in display hardware have enabled similarly rapid growth in display resolution and the development of large, high-resolution environments. These environments present an opportunity to visualize big data in new ways and better integrate expert judgment in the computational analysis of big data. Recent research suggests that big displays present benefits to visualization designers and enable the analysis of complex data sets. However, more research is needed to understand the design decisions, such as perceptually scalable design, that best take advantage of these benefits.

SUMMARY (continued)

Genomic data visualizations have largely failed to keep pace with this growth in genomic data generation and display resolution. Existing visualizations are not designed to enable the comparison of more than a few genomes at once, and are built to work on moderate to low-resolution environments. While it might seem reasonable to simply 'scale-up' these visualizations to fill available screen space, in many cases the design of these approaches do not work on big displays.

In this thesis, I present Bacterial Gene Neighborhood Investigation Environment, or BactoGeNIE, a new comparative gene neighborhood visualization designed to address large volumes of bacterial genome sequences and explore the design decisions that best take advantage of large, high-resolution environments. I will describe the design of this approach and will characterize the ways in which this design scales to large numbers of comparisons, is suited to high-resolution environments, and adopts perceptually scalable encodings. My high-density genome data visualization approach relies on interactive visual queries to transform large data volumes into high-resolution comparative genomic maps, that use pre-attentive visual cues to address analytic tasks for comparative gene neighborhood investigations across large volumes of complete bacterial genomes. In addition, I present a visual algorithm that transforms the data into a view which simultaneously provides detail-up-close and context-from-a distance, allowing researchers to simultaneously access data at these two scales.

The implementation of this approach is an interactive application that can run on single-machine, tiled-display walls as well as high-resolution personal displays. I will describe the program design and architecture, along with several examples from visualizing draft *Escherichia coli* (*E. coli*) genomes.

Preliminary results of this work suggest the novelty and significance of this approach, as well as potential areas of extension for this work. This approach adopts encodings that scale more effectively to large displays and large data volumes, enabling the rapid performance of analytic tasks across large data volumes. This approach also enables the simultaneous comparison and analysis of hundreds to thousands of genomic sequences, which greatly exceed the volumes possible in any existing tool.

Future work in this area includes generalizing the approach to other sub-fields in comparative genomics. In addition, I will adapt this approach to be fit within an ecosystem of multiple-coordinated visualizations for tiled-display walls. I will also explore parallelization by adapting the rendering for the graphics processing unit, to achieve better interactivity.

The primary contributions of this thesis are as follows:

1) BactoGeNIE is a novel visualization design that is the first scalable visualization for comparative analysis of hundreds to thousands of gene neighborhoods. The state of the art competitor visualizations handle no more than 9 gene neighborhoods.

SUMMARY (continued)

2) BactoGeNIE is the first interactive ‘thousand-genome’ comparative visual approach on the gene neighborhood scale combining navigation, details-on-demand, contig sorting, contig density control, ‘zoom’ and application of color tags to genes of interest

3) BactoGeNIE is the first to employ a dynamic ‘gene targeting’ interaction which combines on-the-fly alignment and sorting by user selected ortholog clusters with the application of a color ramp for the target gene and contig, allowing for rapid hypothesis testing and the pre-attentive identification of commonly recurring neighbors, deletions, insertions, inversions, truncations, and potential errors in data processing.

4) BactoGeNIE is the first to unify overviews with gene neighborhood details that can be accessed through physical movement.

5) BactoGeNIE is the first gene neighborhood comparative visualization to be implemented for large, high-resolution environments.

1. INTRODUCTION

Science has historically been a story of direct human observation and experimentation to understand nature. Galileo's telescopic observations and calculations challenged the geocentric orthodoxy and founded observational astronomy (1). Darwin's study of finches in the Galapagos established evidence for biological evolution (2). Mendel's study of crosses in pea plants provided the basis for genetics (3). Rosalind Franklin's x-ray crystallographic images enabled the discovery of the structure of DNA (4). These direct visual observations of data paved the way for further experimentation, observation, data collection, analysis and insight. Historically, scientists were embedded in the process of collecting and interpreting data and translating these interpretations into new knowledge. Humans were inextricably 'in the loop'.

Big Data, a broad term which describes massive volumes of data arising from digitization, challenges this traditional scientific process by removing human experts from data collection and analysis. Big Data is generated on massive scales by sensors, digitized instruments and vast participation in internet resources (5). Big Data problems require advanced computational approaches to store, access, process, reduce, and analyze data (6). In the process, however, the domain expert may be left out of the loop, confronting data he or she did not personally collect, with potential for errors that cannot be directly observed, with opaque automated methods that process and analyze the data. An important question for computational researchers is how to bring experts with years of experience and wide knowledge of the domain into the loop to transform computational data and automated analysis methods into knowledge (7).

Visualization is one potential approach to this problem of bringing the human into the loop. Human visual systems utilize 1/3 of human brain, giving humans powerful pattern recognition and visual processing abilities. Effective visualizations can provide an access point for these visual systems (8).

The complexity in volume, variety and veracity of Big Data requires new 'lenses' to bring this information into focus so that human eyes and insight can be brought to bear on a problem. Visualization is needed to allow researchers to interface with their data and the computational methods that produce, process, and analyze big data, and then communicate results (9).

1.1 Genome sequencing boom

Genomics is one such field grappling with Big Data. A decade ago, genome sequencing was a monumental undertaking. The 3.3 billion base-pair sequence of the human genome, released in 2003, took 13 years and $2.7 billion dollars to complete. Along the way, researchers were able to identify over 3 million human genetic variations and catalog 70% of the 30,000 putative genes in the human genome (10, 11, 12).

This project of sequencing, annotating and analyzing human genomic data has driven the development of technologies that have in turn reduced the cost to complete a genome sequence faster than Moore's law. These diminished sequencing costs have produced a genome sequencing boom, filling online databases with draft genome sequences (13). The result is an abundance of valuable data that researchers are eager to leverage in order to answer longstanding questions in diverse fields.

Diminished sequencing costs have also driven research groups to define new avenues of research that depend on large-scale sequencing. From the 1000 genome project, which sets out to catalogue critical variations in human genomes, to bacterial genome sequencing in industry to improve crop yields, biofuel production and drug manufacture, to the sequencing of metagenomes derived from diverse aquatic environments, these custom genome sequence data sets pose new challenges (14, 15, 16, 17, 18).

Based on current trends, in the next decade we can expect continued growth in genome data generation. As a result, researchers in the biological sciences are facing new research opportunities and new big-data challenges. Given the vital role of visualization in bringing expert judgment to bear on complex data, it is important to consider whether current genome visualization approaches will scale to handle these volumes of sequence data.

1.2 Big lenses for complex data

Biology researchers have relied on a variety of lenses to visualize their data. Rosalind Franklin's X-ray crystallographic photographs proved critical for the determination of the structure of DNA (4). Photographs of amplified and separated DNA fragments, treated to fluoresce under UV light, drove many subsequent discoveries in genetics (19). With the advent of complete genome sequencing in the early 2000's, visualization made the leap onto a digital ‘lens’, with the development of genome browsers built for the personal desktop monitors of the day (20).

In the past decade, advances in graphics hardware have enabled scientists to engineer new digital lenses that present opportunities to visualization engineers. From personal workstations with several high-resolution display monitors, to tiled-display walls these environments provide an unprecedented resolution to display and juxtapose complex data sets (21). Recent studies have documented an array of perceptual and cognitive benefits to the adoption of these environments for visual analytics (22). However, the implications of these environments for big data visualization remain inadequately explored. More research is needed to examine the design decisions that will capitalize on the benefits of big displays for big genomic data analysis (23).

1.3 Summary of thesis

In this thesis, I will explore a particular problem in genomics that depends upon, and is complicated by, large volumes of complex data: comparative bacterial gene neighborhood analysis. This problem arises from an abundance of genome sequences and stands as an opportunity to explore visualization design for big data and big displays. I will present a novel genomic data visualization approach, which is the outcome of several years of close work with a genomics research team.

In the first three chapters of this thesis, I will establish the basis for the design approach. In Chapter 1, I will provide a description of the domain problem. I will describe the new questions in comparative genomics that arise from an abundance of complete bacterial genome sequences. I will discuss the role of automated analysis in addressing these questions and will justify a need for scalable visualization design, to complement these automated approaches. These visualization goals fall into 3 areas, which I will call **'the 3 E's'**: **Errors, Exploratory Analysis** and **Expertise**.

In Chapter 2, I will describe advances in graphics hardware that have produced an increase in display size and resolution. From high-resolution personal displays to multi-monitor desktop environments to tiled-display walls, limitations in display resolution for visualization designers are diminishing rapidly. This presents an opportunity to address visualization design in new ways. I will discuss these advancements and potential benefits to visualization design. I will discuss the limitations inherent in porting applications to these environments, and will describe 4 types of scalability that need to be considered when evaluating the scalability of a particular design: **Pixel-Density Scalability**, **Display-Size Scalability** and **Analytic Task Scalability**.

In Chapter 3, I will describe the 'state of the art' in genomic data visualization. I will engage in a critical analysis of related genomic visualization approaches, to understand the representation expectations for researchers in the field. I will consider the degree to which these representations address the domain problem, by satisfying the "3 E's", as well as the degree to which they will scale to large display sizes (pixel-density scalability and display-size scalability) and the performance of analytic tasks across large numbers of genomes (analytic task scalability). This section will provide concrete examples against which to compare my visualization approach.

Then I will describe my design, implementation, and the results of my work, in Chapters 5, 6 and 7. I will conclude by addressing the implications of this work for other genomic data visualization problems, and will discuss future areas of research in this domain.

1.4 Summary of contributions

In this thesis, I describe a design that is implemented in a program called 'BactoGeNIE', which is novel in the following ways:

1) BactoGeNIE is a novel visualization design that is the first scalable visualization for comparative analysis of hundreds to thousands of gene neighborhoods. The state of the art competitor visualizations handle no more than 9 gene neighborhoods.

2) BactoGeNIE is the first interactive 'thousand-genome' comparative visual approach on the gene neighborhood scale combining navigation, details-on-demand, contig sorting, contig density control, 'zoom' and application of color tags to genes of interest

3) BactoGeNIE is the first to employ a dynamic 'gene targeting' interaction which combines on-the-fly alignment and sorting by user selected ortholog clusters with the application of a color ramp for the target gene and contig, allowing for rapid hypothesis testing and the pre-

attentive identification of commonly recurring neighbors, deletions, insertions, inversions, truncations, and potential errors in data processing.

4) BactoGeNIE is the first to unify overviews with gene neighborhood-details that can be accessed through physical movement.

5) BactoGeNIE is the first gene-neighborhood comparative visualization to be implemented for large, high-resolution environments.

2. DOMAIN PROBLEM

In this section I will present the results of close work with domain scientists to understand an important area of research arising from growing volumes of complete bacterial genome sequence data. I will describe **comparative gene neighborhood analysis**, an area of research that depends on large volumes of sequence data (24). I will describe the relationship between automated methods and human analysis, in order to arrive at a specific set of priorities to address through visualization. I will present the genomic data generation pipeline, which impacts strongly on the way in which researchers understand this data. I will discuss the ways in which expert insight is needed to verify the output of automated approaches, engage in exploratory analysis that permits spontaneous discovery and integrate expertise with observed subtle relationships.

2.1 Bacterial genome sequencing boom

DNA sequencing technologies have undergone a revolution over the past few decades. When first invented in the 1970s, determining the sequencing of a short stretch of DNA was an expensive and time-intensive operation coupling chemical reactions with direct human data interpretation. In the 1980s, sequencing short fragments of DNA was automated and digitized. In the 1990s, advances in computational approaches allowed researchers to determine complete genome sequences for a wide array of model organisms. In the 2000's technical innovations in automated data processing accelerated the sequencing process and reduced the costs to sequence a genome faster than Moore's Law (13) (Figure 1). This accelerated pace of genome sequence generation presents challenges in managing, storing, processing, and analyzing these big and complex datasets.

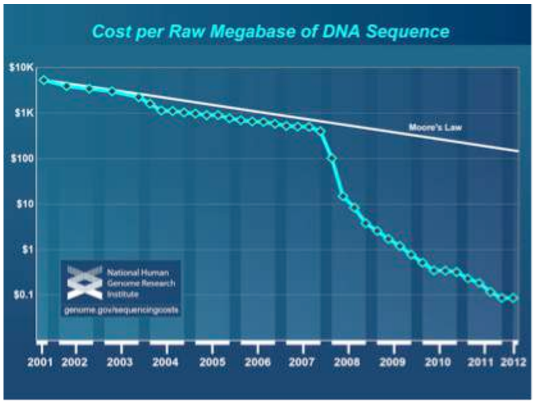


Figure 1. Genome sequencing costs are decreasing faster than Moore's Law. Reprinted with permission from (1).

One field that is particularly impacted by the increased rate of genome sequencing is bacterial genomics. Containing anywhere from 139 thousand base pairs to 13 million base pairs, bacterial genomes are several orders of magnitude smaller than the human genome, which contains 3.2 billion base pairs (25). Bacterial genomes can be more rapidly and inexpensively sequenced and, as a result, thousands of draft microbial genomes can be found on public databases, such as PubMed (26).

As the availability of completely sequenced bacterial genomes grows, researchers are finding new avenues of investigation that benefit from this data. From evolutionary genetics to metagenomics, large volumes of bacterial sequence data raises new questions and provides an opportunity to answer standing questions in new ways. In response to this opportunity, researchers are working to develop and parallelize automated methods to make use of large volumes of complete genome sequences (27).

In some respects, however, genome visualization approaches have not kept pace with the growing scale of genomic data (28). As a result, it is difficult for experts to interface with the output of these new automated processing methods, an essential step in turning data into knowledge in any data-intensive research domain.

2.2 Comparative gene neighborhood analysis

Abundant bacterial genome sequences provides researchers with a unique opportunity to address a long-standing research challenge: characterizing the function of novel proteins. In the next section, I will describe the role complete genome sequencing plays in predicting novel protein function. I will then discuss the role of automated data analysis in this problem and the role for visualization in connecting researchers to automated output and processing.

2.2.1 Characterizing the function of novel proteins

Bacteria contain useful genes that researchers have long sought to identify, characterize and, finally, leverage to meet important needs. For instance, genetic studies in *Bacillus thuringiensis*, a soil bacteria toxic to several orders of insects, led to the identification of crystal ('Cry') proteins that have subsequently been employed to limit insect infestations in agriculture (29). Identification of similar gene products from bacteria could continue to decrease chemical pesticide use and prevent the spread of resistance in insect populations. Other uses of bacterial-genome exploration include the identification of gene products from bacterial strains such as *E. coli* and *Clostridium acetobutylicum* to improve the efficiency of biofuel production (15). Methods that enable researchers to identify and characterize novel bacterial genes have the potential to enable new approaches to a variety of industrial, medical and environmental problems.

Prior to the bacterial genome sequencing boom, predicting the function of a novel protein was an expensive and time-intensive process. Traditional 'wet-lab' research methods in reverse genetics, where scientists modify gene sequences and observe phenotypic differences, have proven extremely effective in predicting the impact of a gene-product on bacterial cells. However, these studies require enormous resources, requiring years of intensive work by teams of researchers (30).

Sequencing and computational methods can help accelerate the process. Since protein function is tightly coupled with gene sequence, highly similar gene sequences across genomes, termed '**orthologs**', likely perform similar functions even in distinct species (31). Computational approaches, such as BLAST, can be used to find similar sequences to a target sequence, across distinct species, and can serve as a basis for predicting the function of an unknown gene-product (32).

However, in some cases these approaches identify orthologs that are themselves uncharacterized. Further study of gene sequence can reveal protein-family domains, or sequences that produce 3-dimensional protein structures which operate in a known manner within a cell, but this does not give researchers the complete picture needed to understand the precise role of an uncharacterized protein in the context of a cell. Approaches in machine learning, such as support vector machines, can also be employed to combine measures of identity to classify novel gene products (33). However, many novel proteins remain unclassified after the application of automated methods based on gene sequence analysis.

Even in cases where computational methods yield potential functions of an uncharacterized gene-product, the relationship between this protein and other proteins, as well as its specific role in complex cellular systems, remains unknown. While it may be possible to computationally search for genes with sequence-level similarity to a novel gene, there are few ways to computationally search for sets of genes that participate together in a particular cellular process. Additionally, when researchers are studying a particular cellular process, there are few ways to computationally identify unknown proteins that participate in that process. This significantly limits the ability of researchers to characterize and understand bacterial genes and the processes that drive bacterial behavior. Given the volume of sequence data now available, researchers are examining new computational approaches to this problem (Goldman, 2012, personal communication).

2.2.2 Exploiting bacterial genome organization to understand function

Bacterial genomes possess unique properties that, when coupled with high volumes of complete genome sequences, may provide researchers a new way to generate hypotheses about uncharacterized protein function and find genes whose products contribute to important cellular processes. Researchers increasingly believe that bacterial genomes are organized such that groups of genes that produce proteins which are involved in similar activities or that functionally interact are located in close proximity in the genome. There are two potential reasons for this organization. First, genes in bacteria are 'expressed', or turned into protein products, in tandem with neighbors residing in 'operons' (Figure 2). Researchers suspect that functionally related genes might be placed near each other, since co-expression and conservation of functionally related genes is evolutionary advantageous. This means that the function of novel proteins in bacteria might be inferred from its genomic 'neighborhood', by looking at the function of the gene-products of its neighbors (30, 34, 35).

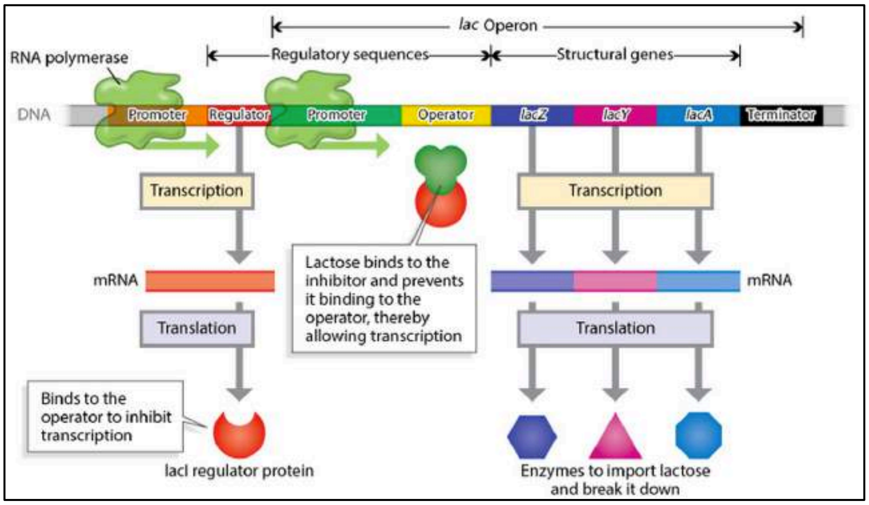


Figure 2. Genes in operons are expressed in concert. Reprinted with permission from source: (36) ©2013 Nature Education Adapted from Pierce, Benjamin. Genetics: A Conceptual Approach, 2^{nd} ed.

In the late 90s, researchers at Argonne exploited this property and developed a methodology for identifying conserved gene clusters across a set of 30 sequenced genomes. In this methodology, a set of genes is defined as a 'run' when they occur on the same strand and are separated by 300 nucleotide base pairs or less. These researchers used a metric to define when two genes, *Xa* and *Xb*, from two genomes, *Ga* and *Gb*, are a 'bidirectional best hit' (BHH), based on sequence-level similarity and the absence of a better match. Using these two measures, they were able to find that around 35% of genes with a known role in a particular pathway occurred in a run with another gene with a different role in a similar pathway. Extending this methodology to uncharacterized genes, researchers were able to identify sets of genes with potential functional similarities.

Researchers note that these results are only statistically significant if sets persistently recur in large numbers of genomes or in more distantly related strains of bacteria. As the volume of genome sequences grows, it is clear that these sorts of automated methods, that combine sequence-level comparisons with distance metrics, may help predict the function of many novel genes in bacterial genomes (35).

2.2.3 Evolutionary biology and gene neighborhood analysis

In addition to using gene context to predict the function of uncharacterized gene-products, comparative analysis of gene neighborhoods is important for research questions in bacterial evolution.

Bacterial genomes possess remarkable plasticity, with high-rates of horizontal gene-transfer, modifications in gene order and gene-loss. As a result, constructing evolutionary relationships can be a significant challenge, particularly when studying the evolution of systems that undergo rapid changes. One such rapidly evolving system is in genes involved in pathogenicity. Many researchers are interested in studying evolution of bacterial plant pathogens to better understand the relationship between genotypic changes and the mechanisms of pathogenicity (37).

The ‘red-queen’ hypothesis is one model for understanding the evolution of bacterial pathogenicity and resistance in host plants. As bacterial pathogens develop a new approach to suppress plant defense systems, plant hosts then develop adaptive responses to this suppression. In response, bacterial pathogens develop new approaches to overcome this resistance and attack a host plant, and so on (38).

In bacteria, horizontal gene transfer, where genes are passed-on through mechanisms other than reproduction, is relatively common. However, genes inherited this way are selected and retained in subsequent generations when they provide a competitive advantage. Such genes are formed into operons and are co-localized, through genetic rearrangements within a bacterial species. Once co-located, these genes can be transmitted through horizontal gene transfer as a group, which then provide the recipient bacterial genome with a set of genes with all of the functions needed to provide a competitive advantage. As a result, sets of genes involved in a particular function, such as pathogenicity, often reside in pathogenicity islands or on plasmids within a bacterial genome. By examining these genomic islands, researchers are able to uncover new modifications to pathogenicity systems, which can then be used to understand the cycle of pathogen/resistance evolution in bacteria and plant hosts (37).

In another domain, complete genome sequences have had a significant impact on the study of operons in bacteria. Early research in *E. coli* drove the development of the operon model for bacterial genetics. Researchers assumed *E. coli* was an ideal model organism for bacteria. However, more recent studies on sequenced bacterial genomes have shown differences in gene clustering and operon structure. In some species, such as *Helicobacter pylori*, operons are relatively unconserved, with genes involved in related processes distributed throughout the genome. Studying these variations could help us understand the adaptive advantage conferred by clustering in some bacterial species as opposed to others (39).

The new abundance of complete genome sequences has the potential to contribute significantly to these questions. Visualizations that present gene context across related bacterial strains, stands to significantly contribute to our understanding of conservation and evolution in bacterial genomes.

2.3 Role for visualization

As described in the previous section, the growth in volume and quality of genomic sequences provides an opportunity for the development of new automated analysis approaches in genomics research. This work has led to the development of a new methodology in identifying commonly recurring sets of genes and predicting the function of novel gene-product by examining its genomic context.

While computational methods are essential in addressing this problem, to generate, process, compare and analyze genomic sequence data, these methods alone are insufficient. Visualization serves to bring a researcher 'into the loop' and leverage the significant portion of the human brain that is dedicated to processing visual information.

However, to understand the specific role for visualization in addressing this problem, and define the visualization design goals, it is valuable to study specific ways in which automated methods alone fall short. Automated methods could be used to retrieve commonly recurring sequences of related genes around a target gene, the output of which would be lists of frequent gene neighbors. While this might be sufficient for some scenarios, this approach would not be adequate in this case. There are 3 primary reasons that visualization is needed to supplement automated approaches for this specific domain problem, which I call the **'3 E's':**

1) Verify the output of computational methods, catch potential **errors** in automated processing, and take these errors into account during subsequent data analysis.

2) Find patterns and relationships through **exploratory analysis** that can be difficult to target with computational approaches, because they require advanced knowledge of the data.

3) Bring researcher **expertise** into the analysis to find subtle patterns or relationships that depend on wider domain knowledge.

In the following sections, I will describe the unique role for visualization in this domain, elaborating on the '3 E's'. I will describe the data generation pipeline, to motivate the need for visualization to identify errors. I will then describe the need for exploratory analysis and the ways in computational methods alone might be insufficient. I will finish by describing the need to bring expertise to bear on the problem through visualization.

2.4 Errors: Verifying the output of computational methods

In broad terms, complete genome sequences are generated through methods that bring together chemical reactions with digital instrumentation and computational methods. At each step there is a potential for error. While these errors could be screened through more computation, a good visual representation will not hide these errors. Visualization provides a lens into the data generation process, allowing for expert verification and data correction. In this section, I will review the data generation pipeline and describe the sources for error and the ways in which a visual representation might address the limitations and uncertainties in the genomic datasets.

2.4.1 Data generation pipeline

Since the 1970s, short genome sequences could be generated through chain termination sequencing. In this process, short segments of DNA are replicated in four separate reactions in the presence of four chain-terminating amino acids which prematurely halt the DNA replication process. The strands that are produced from each replication reaction are of varying lengths, terminating at each site where the chain-termination amino acid can be found. A separating gel and electric current can be used to separate these fragments by length, visualized by a dye, and then observed by the human eye under UV light to piece together the sequence. In this process, sequencing is a combination of chemical reactions and analysis of data through direct visual processing (40).

A major transformation in genomic sequencing occurred with the development of **‘shotgun sequencing’**. This process involves the same chemical replication reaction on a short sequence of 800-900 base pairs, only in a single reaction where fluorescent chain terminators are employed and sequences can be read by digital instruments without requiring the human eye. (Figure 3) These ‘short reads’ do not span a complete genome, even small bacterial genomes. To generate a complete genome sequence, short reads need to cover the genome and be ‘stitched’ together through computational methods. This process is called **‘genome assembly’**, which stitches together short reads into contiguous sequences, called **'contigs'** (40) (Figure 4). As the human genome project ramped up in the mid-90s, genome assembly computation improved in speed and fidelity (11). Over the course of the 2000s, parallelization further increased the speed, lowering the compute time of genome assembly and effectively decreasing the cost (41). Today, bacterial genomes, requiring 20,000-200,000 short reads can be assembled on a single computer (39).

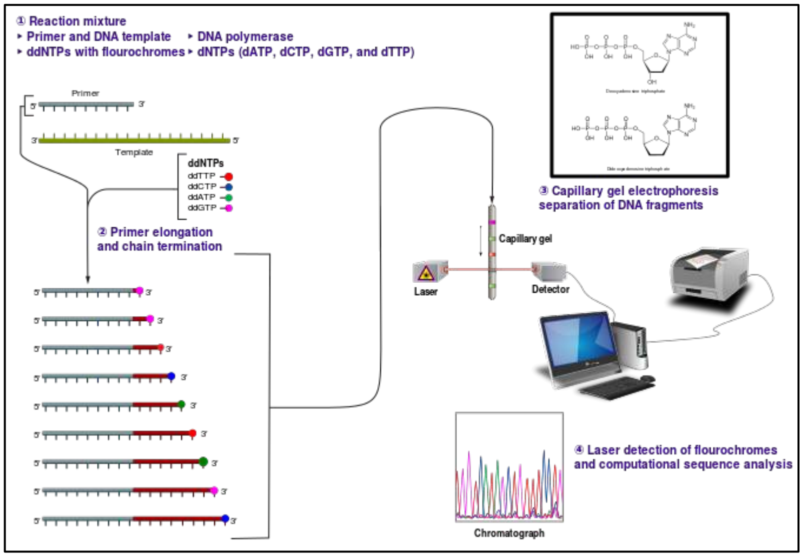

① Reaction mixture
‣ Primer and DNA template
‣ DNA polymerase
‣ ddNTPs with flourochromes
‣ dNTPs (dATP, dCTP, dGTP, and dTTP)
Primer
5'
3'
3'
5'
Template
ddNTPs
ddTTP
ddCTP
ddATP
ddGTP
② Primer elongation and chain termination
③ Capillary gel electrophoresis separation of DNA fragments
Capillary gel
Laser
Detector
④ Laser detection of flourochromes and computational sequence analysis
Chromatograph
NH₂
HO
OH

Figure 3. Sanger chain-termination sequencing methods accelerate complete genome sequence production. Reprinted with permission from source (43).

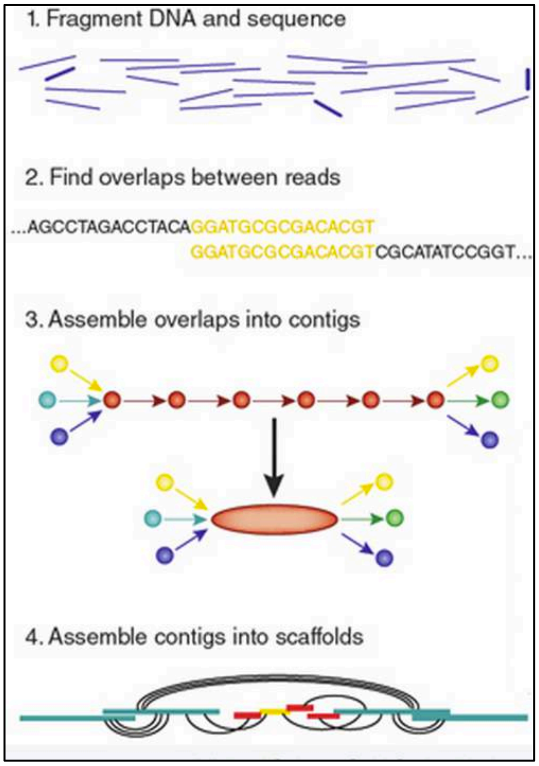

1. Fragment DNA and sequence
2. Find overlaps between reads
...AGCCTAGACCTACAGGATGCGCGACACGT
GGATGCGCGACACGTCGCATATCCGGT...
3. Assemble overlaps into contigs
4. Assemble contigs into scaffolds

Figure 4. Reads are assembled into contigs. Reprinted by permission from Macmillan Publishers Ltd: Nature Methods source: (44), ©2012

While efficiency and speed have increased in genome sequencing and assembly, there are a variety of sources of error in these processes. Short reads of sequence are not always accurate, because sometimes nucleotides are incorrectly identified or stretches of bases are not recorded. Genome assembly is computationally complex and cannot always generate a complete and accurate sequence, particularly in the presence of repeated nucleotide sequences. These errors in assembly result in breaks in contigs, such that multiple contigs are generated for a single chromosome. These errors can be very difficult to correct through computational methods alone, and can hinder efforts to examine gene neighborhoods (42).

The next step in genome data generation is **genome annotation.** Sometimes genome annotation is community-driven, and based on BLAST searchers and manual annotations of known gene sequences. However, this process is often performed computationally, where automated methods use common markers to distinguish features in the genome, such as gene boundaries. Once the gene boundaries are identified, computational approaches can be used to predict the identity of the element, typically involving sequence comparisons with other known elements (42).

Due to errors in sequencing and assembly, sometimes gene boundaries are missed during genome annotation. In addition, mistakes in gene identification can arise, which results in missing gene labels and a lack of information as to gene identify, which compromises the activity of examining gene neighborhoods for commonly recurring genes and functional activities (42).

Once data has been annotated, it typically resides in files that contain either a) raw sequences in some format, such as fasta format, or b) genome feature lists, with varying degrees of annotation detail, such as genome feature file format.

These data sets typically are not designed for comparative analysis without additional automated analysis. There are a variety of methods that can transform this non-comparative data into a comparative format. Some methods compute similarities between genomes as a whole and others compute similarities between gene sequences to find potential orthologs. These algorithms rely on fast gene sequence comparisons. Refining the comparative method to be sufficiently sensitive can be difficult, resulting in missed orthologs or poorly identified orthologs. This error can complicate gene neighborhood analysis by making it difficult to distinguish true and false orthologs (46).

Finally, computation can be used to find commonly recurring subsequences of genes, which directly touches on the comparative gene neighborhood analysis problem. These methods, relying on the identification of commonly recurring sequences around a target, would in the absence of visualization produce lists of genes in textual format (35). In section 2.5, I will describe the ways in which this question cannot be directly answered through computational methods alone.

All of these computational methods have the potential for errors that can be difficult to catch without deep engagement with the data. Data visualization can provide such a mechanism for deep engagement that capitalizes on human perceptual strengths and expert judgment.

2.4.2 Using visualization to make errors in computation evident

As described in the previous section, genome data generation is not an error free process, complicating gene neighborhood comparative analysis. To enable researchers to effectively compare across many gene neighborhoods, it is essential to provide a representation that does not obscure these potential errors in the data. Data mining, while an efficient means of computing sequences of gene neighbors, will not typically take into account the potential for errors and, as a result, the output may fail to reflect the ground truth. Visualization can bridge the gap between data which arises from automation and the ground truth by bringing human pattern recognition and judgment into the loop. There are 3 primary types of errors in data generation that complicate gene neighborhood comparisons.

1) Unexpected gaps: Bacterial genes are typically placed close together with short stretches of intervening sequence, particularly when in a common operon. A large gap is unexpected and might indicate errors in the automated methods that generate the genome sequence data, either in sequence reads, assembly or in annotation. In other words, though the data doesn’t list a gene in the region with a gap, a gene might be found there, but was missed due to errors in data generation (47).

If a researcher were to use purely automated approaches to identify commonly recurring sequences of genes around gene targets, additional analysis would be required to factor in these gaps, and indicate this discrepancy. This is difficult to do in concert with common gene sequence identification algorithms, but without this additional analysis, lists of commonly recurring gene neighbors around a target gene would misrepresent the ground truth. (Figure 5).

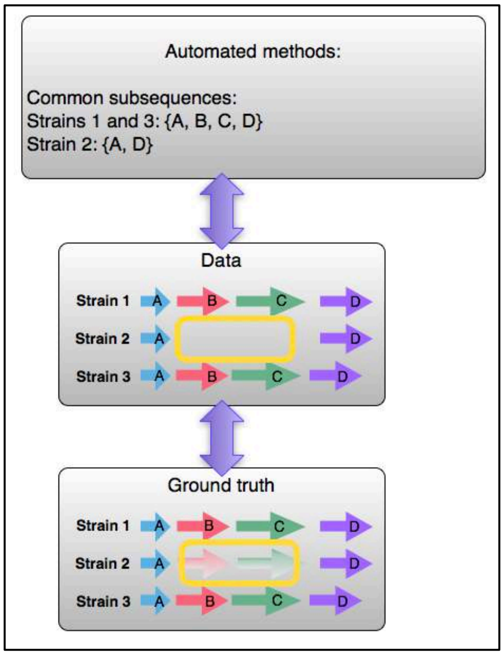


Figure 5. Errors in automated processing result in unexpected gaps that are difficult to catch through automated methods

Visualization can help with this problem. A good visual representation will make these gaps immediately evident, and may even highlight potential candidate genes to fill the gaps. It may also cue researchers in to potentially interesting variations, if these gaps represent the ground truth.

2) Breaks in assembly: Errors in assembling complete genomes often results in several distinct contigs that cannot be stitched together. As a result, contiguous genes in a sequence will not appear to be contiguous in the data. This complicates gene neighborhood analysis, because automated analysis algorithms can only compute on the data as given, and, as a result, some genes that occur close to gaps will have missing neighbors (42). (Figure 6)

A good visual representation that highlights similarities between related bacterial strains, might suggest methods to assemble contigs together, or provide insight as to why a particular gene in a strain of interest lacks expected gene neighbors.

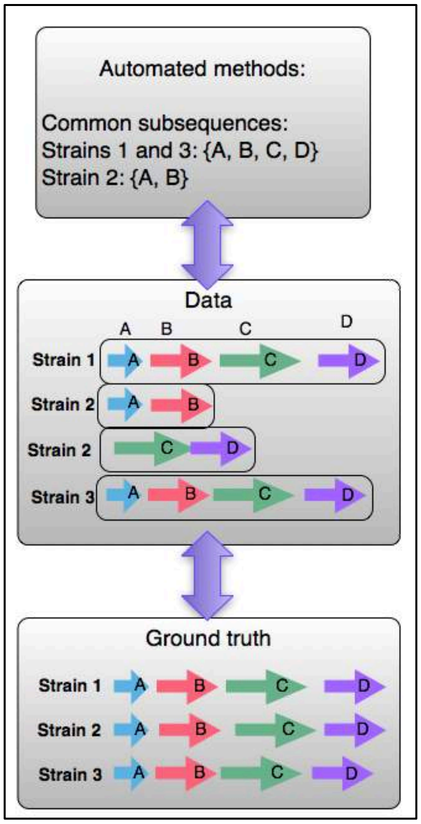


Figure 6. Breaks in genome assembly complicate gene neighborhood analysis

3) Missed annotations: As described in the previous section, the genome annotation step in data processing identifies gene boundaries and uses automated methods to assign identities to genes. For instance, this step may label a particular gene as "exonuclease 3", and then assign a description to that gene. This step is essential, because it allows researchers to examine gene neighborhoods and look for common functionality in gene-products. The annotation algorithms sometimes fail to find a candidate match for a gene-product (45).

This error in annotation complicates gene neighborhood analysis because the output of automated analysis methods may contain many genes that are labeled as unknown. While a researcher could dig into this data and attempt to identify these genes, this is a potentially lengthy process that limits data analysis of many gene neighborhoods and reduces flexibility to explore data.

To efficiently sequence complete genomes, these sequences are generated, assembled and annotated without direct human involvement. Visualization plays an important role in bringing experts into the data mining loop, to provide verification to these automated methods and identification of errors that complicate data mining.

2.4.3 Implications for visualization design

The visualization can't simply show lists of commonly recurring genes around a target. The full genomic context is important and needs to be shown. This context needs to show gene boundaries, mapped to the nucleotide position, otherwise unexpected gaps between genes will not be evident and may obscure the analysis of gene neighborhoods around a target gene. This context needs to show the start and end of contigs, so that breaks in assembly can be identified and factored into an analysis. This context needs to make descriptive and identifier information available, so that missed gene labels can be seen.

In addition, errors in gene annotations mean that this data cannot be the basis for comparative analysis of gene neighborhoods. For instance, it might be tempting to look for all genes labeled as 'exonuclease 3', and consider these to be orthologous. However, due to errors in annotation, a more robust ortholog identification algorithm is needed.

2.5 Exploration: Visualization brings human pattern recognition and judgment into data exploration

In analyzing gene neighborhoods and identifying commonly recurring gene neighbors, several automated methods could be used. These methods would involve the identification of common sets of genes, along with the strains that exhibited these sets and strains with variations. Statistics could also be generated to measure the 'fit' of a particular sequence of genes. This approach would be effective for capturing common sets of genes around a gene target (35).

Such lists of commonly recurring gene neighbors found through data mining could suggest functional relationships between genes, or functional roles for novel genes, but this summary excludes data that could be meaningful to experts. To find these subtle variations or relationships, researchers would need to know in advance what to look for. The unexpected observations that arise through exploration are important, and need to be considered in visualization design. To enable exploration, what is needed is a view that calls attention to potentially significant variations in the neighborhood and structure of orthologous genes. These **analysis tasks to enable through exploratory visualization** are: identification of **gene truncations** and sequence **deletions, insertions, duplications** and **inversions.**

2.5.1 Gene truncations

Two novel genes *Xa* and *Xb* might be identified as orthologous, due to a high percentage of shared nucleotide sequence, but one of these genes, *Xa*, may be truncated. Truncation means that the gene is missing large stretches of nucleotide sequences, which results in a smaller protein product that may lack domains critical for that protein's function (48). Essentially, there might be significant differences between the function of *Xa* and the function of *Xb*, even if they share large stretches of nucleotide sequences. Further, if these genes are not distinguished in data mining, significant similarities or differences in *Xa* and *Xb*'s neighborhood may go unnoticed. To account for truncated genes, researchers would need to know in advance to search for these situations, and would need to then carefully design sequence comparison parameters to account for this possibility. An effective visual encoding will allow researchers to quickly find truncations in genes of interest, allowing them to use these sequences in subsequent automated searches. This situation is shown in Figure 7.

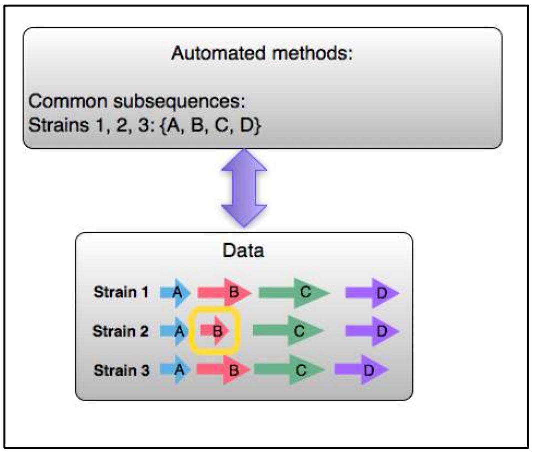


Figure 7. Gene truncations will not be detected through automated methods without knowing in advance to look for them.

2.5.2 Insertions and deletions

Mining for commonly recurring sequences of genes in the neighborhood of a gene of interest will return sets of genes with potential functional similarities. However, what may be lost are unusual variations in gene presence and order. For instance, an **insertion** event has occurred when a new gene is included in an otherwise conserved set of genes in a few strains of related bacteria. This insertion might have important consequences for the function of that strain. For instance, a new gene might complement the activity of a neighbor, when co-expressed. Alternatively it may repress a neighbor's function, when co-expressed. The presence of a new gene might signal an anomaly that has significant molecular consequences, when co-expressed with other genes. Conversely, a gene deletion might indicate a loss of function or a loss of repression, which might be significant (30). When studying a gene's neighborhood, differences on content might be significant, and summary representations of commonly recurring sets of genes may miss these anomalies.

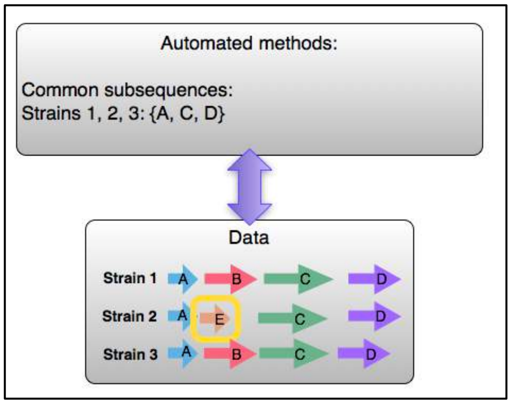


Figure 8. Rare gene insertions will not be detected by common subsequence detection methods.

2.5.3 Duplications

Another interesting variation that will not be picked up by data mining without deliberate searchers are duplications of genes or sets of gene. This occurs when a stretch of DNA is duplicated in a bacterial genome. This duplication could indicate an error in data processing, where short reads are erroneously stitched together in more than one position in a genome. Alternatively, this duplication could indicate an evolutionary adaptation, particularly if the duplicated region contains variations in sequence that represents a functional divergence. Further, a duplication event may arise through horizontal gene transfer from another bacterial strain. These events are significant to researchers, but difficult to pick-up through data mining alone (30).

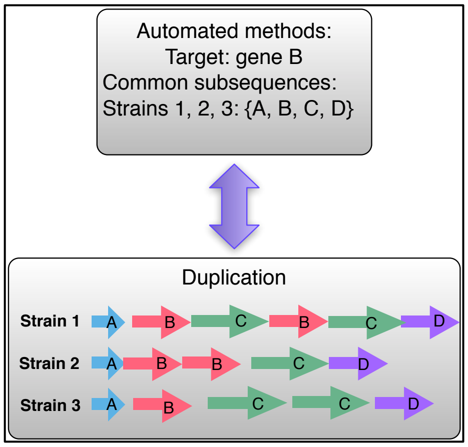


Figure 9. Gene duplication events can be missed through automated analysis

2.5.4 Inversion

Inversion events occur when a set of genes are ‘flipped’, such that they both arise on a different strand of DNA. Inversions are significant because genes are not simply co-expressed with neighbors in an operon, but are expressed in only one of the two directions. This is due to the molecular mechanisms of gene transcription, which limit the order of transcription along one strand of DNA. An inversion event therefore potentially breaks co-expression patterns, having potentially significant consequences for molecular pathways and interactions (30).

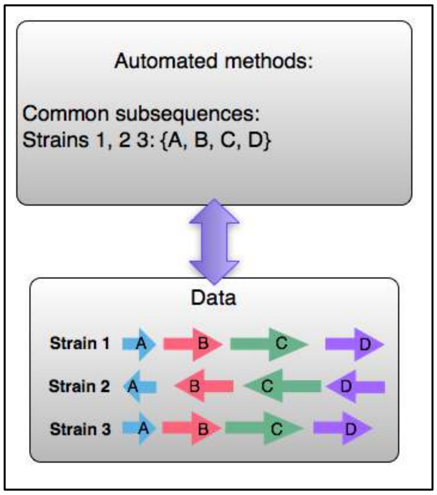


Figure 10. Sequence inversion events can be missed in automated analysis approaches.

2.5.5 Exploratory visualization: Implications for visualization design

In conclusion, it is essential to consider insertions, deletions, duplications and inversions when constructing a visual design for gene neighborhood comparative analysis. This suggests a visual representation that shows: gene boundaries, including size, position and strand of genes, in a genomic context, with some encoding of orthology between genes across genomes. This representation must accommodate display of many such gene neighborhoods, to permit exploration of existing volumes of genome sequence data.

2.6 Expertise: Expert judgment recognizes patterns and connections

Data mining can assist in the identification of commonly recurring sequences of genes, which can suggest the function of novel proteins. However, to capture subtle relationships and patterns of conservation, it is necessary to bring experts into the loop through the development of good representations of data and interactions that enable interactive drill-down into genomic data.

For instance, data mining algorithms might pick up commonly recurring sequences of genes, what it will fail to pick-up variations that are significant given the large body of knowledge about the genes and bacterial strains under consideration. For instance, sometimes a particular variation strikes a research as particularly significant because it arises in a bacterial strain with unusual properties. Or a researcher will spot an interesting variation based on what they know about a particular functional pathway or low-level molecular process. To find this using automated methods, it is necessary to know in advance what you are looking for. Even with sophisticated data retrieval algorithms, it is likely that researchers want to use what they know in analyzing the data and not simply rely on automated methods.

A good visual representation will provide researcher with a platform for using what they know in analyzing data. Representations must provide sufficient context for expertise to be useful. High-level summaries or overviews without relevant details will likely be insufficient. What is needed are representations that make relationships and patterns quickly accessible, but also allow for access to textual details that allow researchers to make connections between their knowledge and the information in the visualization. A representation that provides **both details and context** will better enable the integration of expertise in this domain problem.

2.7 Summary: Visualization design goals for comparative bacterial gene neighborhood analysis

Given the extent to which genome sequences are generated, assembled and annotated without direct human involvement, visualization must help bring experts into the data mining loop, both to provide a verification to these automated methods, as well as to identify patterns that are less accessible to data mining. For this particular domain problem, this entails a visualization design that:

1) Helps experts identify sources of **error**, such as **gaps**, **breaks in assembly** and **missing annotations**

2) Allows experts to engage in **exploratory analysis** of the data, by looking for **gene truncations** and sequence **deletions, insertions, duplications** and **inversions**, and support for subtle pattern identification

3) Enable researchers to integrate **expertise** in data analysis, by making textual details available on demand, and by providing access to **overviews and context as well as low-level details.**

Since inferring gene-product function from gene-neighborhoods relies upon the comparative analysis of many genomes for statistical significance, visualization must address growing volumes of sequence data and allow for comparisons between hundreds to thousands of genomes. This scalability challenge will likely require new modalities for viewing and interacting with larger volumes of genomes in a way that addresses research questions of interest.

In the following Chapter, I will describe how these design goals dovetail with developments in graphics hardware, presenting an opportunity for scalable visualization design. In Chapter 4, I will then discuss the state of the art in genomic data visualization, to illustrate the extent to which current approaches fail to meet the above goals.

In Chapter 5, I will elaborate on the implications for visualization design that follow from the need to bring experts into the loop to identify **errors**, **explore** the data for patterns that may be difficult to find through automated methods, and integrate researcher **expertise** in the analysis.

3. LENSES FOR BIG DATA

In Chapter 2, I described technological advances in genome sequencing technology that have enabled rapid growth in the volume of complete genome sequences. These technical developments present new opportunities to researchers, enabling new approaches to long-standing problems. Technical developments in graphics hardware have followed a similar trajectory: rapid advances in display and graphics hardware have are enabling research groups to build low-cost, low-infrastructure, large, high-resolution display environments. From personal multi-monitor displays to collaborative tiled-display walls, these environments present an opportunity to visualize big, complex datasets in new ways (21).

In this Chapter I will describe the developments in display technology, in order to illustrate the ways in which the technical landscape for big data visualization has shifted over the past decade. I will use this discussion to explain the ways in which the increased size and resolution in large displays impacts big data visualization in general, and genomic data visualization in particular. This discussion will be used as the basis for evaluating genomic data visualization in Chapter 4.

3.1 Display technology

Display technology has evolved significantly over the past decade, giving individual researchers increasing resolution and space to display data. This evolution can be seen both in personal workspaces as well as collaborative workspaces.

Standard desktop computing systems in the early 2000s had variable graphical capabilities. Graphics cards were used in gaming systems and high-end machines, but had not been incorporated in standard desktop environments. A single machine typically could drive only one or two moderate resolution displays, but this technology was not widely adopted. (Since that time, GPUs and graphics hardware have become standard features of desktops and laptops. With developments in LCDs and LEDS, monitors became thinner, allowing analysts and researchers to place multiple displays on a standard desk without losing work-space. Single machines can drive these multi-monitor systems at increased resolutions (21).

At the same time, advanced graphical systems, including wall-sized collaborative environments have become increasingly accessible to researchers in lab and office environments. In the 1990s and early 2000s, many research groups investigated the complexities in building ‘wall-sized’ displays. Projector-based systems, were expensive and difficult to maintain. These systems required large spaces and to accommodate compute clusters and display hardware. Tiling, aligning, calibrating and maintaining projector-based systems were non-trivial challenges. To maintain these systems, a dedicated staff was needed to replace light bulbs and calibrate colors across display nodes (23).

Tiled-display walls were beginning to be adopted in response to the challenges arising from projector-based systems. Tiled-display walls were easier to maintain, but still required clusters to drive high-resolution graphics. In addition, wide ‘bezels’ on the edges of displays limited the ability to coherently present high-resolution data. For all these reasons, there was a high 'barrier to entry' for groups interested in adopting advanced graphical systems for working spaces, such as offices or research labs (49). Much of this has changed in the past decade, bringing advanced graphics environments into standard work and research settings (figure 11).

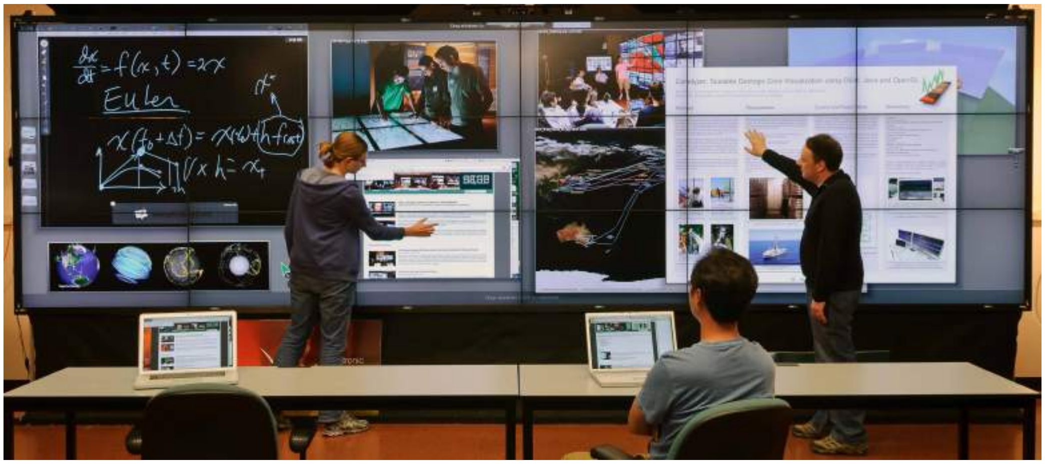


Figure 11. Large, high-resolution environment. Image courtesy of the UIC Electronic Visualization Laboratory (Photo: Lance Long, UIC).

3.2 Large, high-resolution displays present an opportunity for big data visualization

The decreasing cost and increasing availability of large and high-resolution displays in research settings presents an opportunity to visualization designers. These environments have a variety of perceptual and cognitive benefits, particularly for **externalizing large volumes of information** and **leveraging embodied cognition**. The question, however, is whether these environments can be utilized to more effectively visualize big data sets. In this section, I will examine recent research on the perceptual and cognitive benefits of big displays as well as research on design considerations for effectively visualizing information in these environments.

3.2.1 Externalize, perceive and process more

Hegarty describes visualizations as systems that allow users to externalize memory, by providing an external store for information capable of handling a larger volume of data than internal stores, which are limited by working memory capacity. This information is then rapidly accessible through the visual system. By offloading memory onto an external, visual representation, cognitive resources are freed for other, higher-level activities. Internal representations in the mind of the user are still needed, for instance to store the target of a search or the locations of important information on a display, but these representations can be sparse, with volume and details accessible on the display (50).

In addition to storing information, visualizations allow researchers to offload cognition onto perception, which given a proper representation can be automatic, effortless, parallel, and pre-attentive, which means that emergent features and patterns will be accessible to a viewer through the visual system (51). Card et al refers to this as ‘using vision to think’ (52).

On a big display, more information can be externalized and accessed perceptually, there is evidence to suggest that users take advantage of increased resolution on large displays, and are able to scale-up their perceptual processing to perform visual queries over a larger volume of data. Given the value of external representations in sensemaking, big data problems stand to benefit from visualization on big displays. In essence, visualizations on big displays have the potential to support scalable perception of visual patterns, allowing researchers to perform scalable visual queries of big datasets (53).

However, for these benefits to be realized, several perceptual limitations must be considered. For visualizations to be accessible directly through perception, it must rely on **pre-attentive cues**. For instance, the identification of a particular pattern or relationship will take place pre-attentively if it stands out immediately to a viewer. A visualization that relies on pre-attentive cues eases the cognitive load on the user, allowing for other high-level tasks to take place in parallel. Color is often used to present information pre-attentively, but these representations will vary depending on the data type and the ways in which users wish to engage the data (54).

Sometimes information will not be immediately available to a user, and a visual search will be required. A visual search takes place when a user scans the scene for a target or pattern held in working memory. For instance, a researcher may search for a connection between two specific nodes in a graph, and will search the representation to find this relationship (55).

As the number of entities on display increases, the search time also increases. This means that visual searches over large datasets are potentially challenging for a researcher. It is vital to consider ways to ease the demands of a visual search or query. Since working memory is limited, it is ideal to minimize the time to perform a visual search or offload this task onto perception or interactivity. Visualizations that **aid visual search**, will help address this problem and will be essential for big data visualization on big displays (55, 56).

Visualization designers often seek to enable the rapid identification of patterns. Often this involves the identification of a pattern in the visualization, say a particular group of elements, holding this target in working memory and then performing a search for similar targets across the visualization. Working memory is limited, so this consumes cognitive resources that could be used for higher cognition. If a representation can enable **direct visual comparison**, that can be carried out through eye movements pre-attentively, then the user will be able to make more subtle connections (55).

Visual clutter describes problems that arise when the presentation of information hinders perception of information and the performance of analytic tasks (57). This includes encodings and layouts that obscure relationships or make it difficult to detect patterns. While clutter can arise for small-data visualization problems, it is particularly problematic as the number of entities on display increases. Layouts and encodings that work for small numbers of entities may fail for larger numbers of entities (50).

3.2.2 Leverage embodied cognition

Andrews describes large, high-resolution displays as environments at *human scale*, in that the size and resolution of the display matches the perceptual and physical scale of the human body. This provides a variety of benefits to the viewer. One of the most significant benefits is that these environments permit users to explore a large dataset through physical navigation instead of virtual navigation, allowing visualization designers to exploit embodied cognition, such as spatial memory, in complex data analysis scenarios. In addition, queries can be performed rapidly and directly through head and eye movements, which can be faster than virtual navigation (58).

On conventional displays, visualization researchers have developed a variety of effective virtual navigation techniques, such as pan and zoom navigation or Focus+Context views (55). While these techniques are beneficial, several studies suggests that performance of tasks through physical navigation is more effective than through these virtual navigation techniques. Researchers have studied visualization search tasks on detailed maps on small, low-resolution displays and large, high-resolution displays. On small, low-resolution environments during a search task which required extensive virtual navigation, users frequently reported a single target more than once, suggesting that users find virtual navigation techniques to be disorienting. On a larger display, physical navigation could be used to locate targets in addition to virtual navigation. In these situations, researchers found that memory of a target's position was improved with physical navigation. Confidence about a target increased when searching on a large display. Though virtual navigation techniques were available to users, they generally chose to navigate physically over a big display. Feedback indicated that users are better able to maintain awareness of spatial context of targets why navigating physically. This study indicates that performance improves for basic visualization tasks over detailed data when performed on large, high-resolution displays. Targets were found twice as quickly. Users felt more confident in their findings and reported less frustration in completing the tasks. This suggests a strong benefit to viewing and exploring complex data sets in high-resolution environments (59).

Following up on these results, researchers have investigated whether large, high-resolution displays allowed for physical navigation, and examined the consequences for user performance in tasks within a large, 2D information space. In this study, users performed 4 tasks: navigating to a target, searching for a target, pattern finding for a set of targets, and open ended insight on a group of targets. In performing these tasks, users reported a strong preference for physical navigation over virtual navigation. When given the choice, users chose physical navigation 100% of the time. In terms of performance, virtual navigation was found to have a significant negative impact when compared to physical navigation. Number of zooms and pans negatively correlated with performance and increased task completion time. Display size had an impact on the choice of physical navigation over virtual navigation. As display sizes increased, virtual navigation decreased and physical navigation increased. This study demonstrated that increased display size and resolution allowed users to perform visual searches, pattern matching and target identification through multiple levels of scale through physical navigation, and that this type of navigation was a more efficient and effective (60).

However, many of these studies were performed on geospatial datasets, which possess properties that are particularly suited to physical navigation and data exploration. What is not known is what qualities a visualization must possess in order to properly leverage the advantages afforded by embodied cognition.

3.3 Addressing big data volumes with big displays: visualization scalability challenges

As the preceding section indicated, there are a variety of benefits that arise when viewing complex data on large or high-resolution display environments. It might be tempting to conclude from these research results that big data visualization challenges primarily boil down to a lack of space to display information. In some instances, for example in analyzing high-resolution maps or pixel-based visualizations, a major limitation to understanding a big dataset is lack of sufficient space and resolution to perceive the data. (60, 58) In these cases, benefits may be realized by simply porting existing visualizations to a big display so that it shows more data on a large space, without fundamentally changing the visualization design or application code.

However, this is not the case for all big data problems. In many instances, visualizations designed for small data sets and small, low-resolution displays will not effectively 'scale' without modification to the visualization design. Even in cases where the application does scale to fill the available space or scale to show more data, I believe sense making around this data is negatively impacted.

While there are a variety of potential places where a visual design might fail to adequately scale to accommodate big data and large, high-resolution environments, in this thesis I will focus on four major types of scalability that are required to effectively showing more information across a larger display surface. The goal is to enable the performance of a particular analytic task across a larger number of entities, on larger and higher resolution space. These types of scalability are interrelated, but raise important questions when considering porting an existing approach to a big display or designing a new approach for a big display. These types of scalability will serve as the basis for my discussion of previous visualization approaches in comparative genomics, and the basis for my visualization design described in Section 5.

3.3.1 Pixel density scalability

As display pixel density increases, does a visual approach take advantage of increased resolution to show more entities, relationships or to show data at higher detail? Some visual approaches are low-density, in that the visualization is not equipped to show information at high-detail even when given the pixels to do so. Other approaches require minimal spatial dimensions to sensible to a human eye, such as a glyph that requires at least ½ inch of screen space for the user to see the information. In contrast, **high-density visualizations** are ones that can be compressed to show more data across the display, or take advantage of increased detail available to show information.

Some visualizations are also not oriented around displaying **detail and overviews simultaneously**, often because these approaches assume low to moderate-resolution display environments. Focus+Context is a technique developed by visualization researchers to accommodate the need to view details and overviews simultaneously on a conventional display (61). These Pixel-Density Scalability criteria are listed in Table I.

TABLE I:
Pixel Density Scalability Criteria

Pixel-Density Scalability Criterion	**Description**
High-density representation	High-density representations have the potential to exploit increase pixels per inch to show more entities and relationships at higher detail. Low-resolution visualizations are difficult to use at high pixel density, or fail to exploit the available detail to show more information.
Detail and overview	High pixel density displays are capable of showing details and overviews at the same time. Low-pixel density visualizations are not geared to showing detail and overviews simultaneously without Focus+Context approaches.

3.3.2 Display size scalability

As display size increases, could a visual approach be expanded to take advantage of the increased space to depict more entities or relationships? Some applications do not take advantage of increased display sizes, and instead simply increase the volume of 'whitespace' that does not encode information. These applications do not use the increased display size to show more information.

In other cases, an increase in display size hampers perception of data and relationships. For instance, users may have an easy time performing a direct visual comparison between data points shown on a small display, because both data points are in view. On a big display, if that visualization is not adapted to the environment, these related data points may no longer be simultaneously in view in the user's area of focus. Instead, this information may be in peripheral vision or out of sight, on the other side of the display. Users may then travel to opposite sides of a big display to see information, which means that a comparison that formerly took place visually, now requires working memory and additional cognitive resources. This visualization fails to use **clustering techniques** to bring together elements that need to be directly compared (22).

In other cases, a visualization fails to capitalize on spatial memory and the potential for embodied cognition, which have been found to positively benefit visualization tasks on big displays. Spatial memory is a powerful aid to exploring a bit dataset, for example, but visualizations that do not **encode information through spatial positioning**, may not adequately take advantage of the opportunity to exploit spatial memory in an analysis. This idea is in line with graphically scalable encodings, as presented by Andrews et al. Spatial positions scale with display size, and are therefore graphically scalable encoding. In contrast, discernable colors do not scale with display size, and are not graphically scalable (58).

Other visualizations rely primarily on virtual navigation, and do not **take advantage of physical navigation**, which provides a variety of perceptual and cognitive benefits in visual analytics, as described previously. Zoom and pan, on a big display, can be supplemented by physical movement around the visualization, stepping back to see overviews and forward to view details (60).

Some visualizations are designed with a single distance from the display and a fixed visual acuity in mind. Visual acuity for distinguishing two points is approximated 1/60th of the visual angle between those points. As the user moves their eyes closer to a display with high pixel density, more points can potentially be resolved. Some visualizations take advantage of this property by providing **multiple levels of accessible detail,** permitting users to see overviews at a distance and details up close to the display through physical movement. Many representations do not provide additional detail on closer inspection, and thus fail to capitalize on this benefit of large displays (59). These criteria are listed in Table II.

TABLE II:
Display Size Scalability Criteria

Display Size Scalability Criteria	**Description**
Encode big data spatially	Use additional space to show more information
Cluster related elements	Encode similarity of items through spatial clustering to facilitate formation of spatial memory and allow direct, visual comparisons to be performed.
Physical navigation	Multiple levels of detail accessible through physical navigation. Allow overviews to be seen at a distance and details to be seen up close.

3.3.3 Analytic Task Scalability

As screen size and resolution increases, and the data volumes scale to fit this new space, a visualization may not scale to accommodate a particular analytical task across more data and more space. In some cases, a visualization is designed to scale for one analytic task and not others. For instance, a social network visualization might be designed to enable the comparison of personal networks between two individuals, but it may not be designed to accommodate comparisons between more individuals, even when given more space to do so. However, more granularity in the two-way comparison may be available. But if the goal is to compare across more individuals, this approach will not be effective.

Other approaches may technically accommodate the presentation of more data, but it may fail to enable the performance of an analytic task across larger volumes of data. For instance, some approaches fail to rely on **pre-attentive cues** to show patterns or relationships of interest so that an analytic task can be easily performed on the data. These cues would allow the user to automatically and directly perceive the information of interest, as discussed in the previous subsection (54).

In addition, many approaches lack the interactivity to enable visual queries on large data sets, such as brushing or selecting elements of interest in order to identify areas of interest. Scalable visual queries permit researchers to engage in an interactive loop with the data. Not all visualizations attempt to enable large-scale interrogations of data (22).

In this event, visual search will be needed. But visual search across large data volumes and big displays can be complicated. Users may not be able to see all parts of the displayed data at once. In addition, users may need to perform a more complex search and retain more pieces of information in working memory when analyzing a big data set. Unless **aids to visual search** are used, it may be difficult for researchers to search large volumes of information on a big display (55). These criteria are listed in Table III.

TABLE III:
Analytic Task Scalability Criteria

Analytic task scalability criteria	Description
Analytic tasks performed pre-attentively	Encode data such that pre-attentive processing can be used to perform analytic tasks
Analytic tasks aided by visual queries	Visual queries enabled to direct scalable exploration of data
Aids to visual search for performing analytic tasks	Enable visual search across large volumes of data

3.3.4 Perceptual scalability

A visual approach might not scale perceptually to accommodate sense-making around the increased number of entities and relationships, even when given more space and resolution for visualization. The central question here is: as the number of entities on display increases, does the encoding still work, or is there a negative impact on perception and sense-making of that encoding? For instance, a graph visualization might work wonderfully for a small number of instances on a small display, but fail to be sensible to researchers as the number of entities and relationships are increased across a larger, higher pixel-density display. Text labels may be fine on a small display, with a small number of entities but text is not accessible at a distance and it is slow to search across text for patterns and relationships (58).

As the density of a representation increases, there is a higher potential for **visual clutter**. Some representations are particularly vulnerable to clutter as pixel density increases and data volumes scale to take advantage of the resolution. In addition, as items are represented at higher density, visual search may be compromised, as targets become harder to resolve and the volume of the search entities increases (54).

As the display size increases, there is the potential for increased distance between items that need to be compared or analyzed together. A visual comparison that formerly took place directly, for instance, by shifting focus, will then require a user to remember the pattern and do a visual search against the mental image of the pattern. This utilizes cognitive resources that could be extended to higher-level operations (58).

As number of entities on display increases, visual search becomes complicated. The user needs to scan more items and compare a mental image against a larger volume of items. Search time will increase and the potential for error will also increase. Aids to visual search can help address this problem by interactively transforming a visual search problem into a pre-attentive processing problem (58).

An example of a representation that may lack display size scalability is a highly detailed graph visualization. If nodes and edges extend across the display, it is difficult to effectively perceive those relationships at a distance without additional help. Items that are connected on opposite sides of the screen may be difficult to resolve. As the number of items increases, it can become more difficult to identify and search for meaningful relationships without aid. Different layouts that bring related elements together, or visual approaches that apply color appropriately, which can be pre-attentively perceived at a distance, might address these challenges (58).

These features define several aspects of how visualizations could be scaled up in data, resolution and display size to show more information usefully. In Chapter 4, I will use these features to examine whether existing approaches will port successfully big display environments.

3.4 Summary

In this Chapter, I described advances in display technology, and the potential benefits of these technical advances for big data visualization. I then described four potential features needed to port an existing visualization or design a new visualization for high-resolution or big display environments: **pixel-density scalability**, **display size scalability**, **analytic task scalability**, and **perceptual scalability**.

In the Chapter 4, I will examine existing approaches to comparative genome data visualization and will evaluate whether these approaches scale to big displays, focusing on the **four scalability dimensions**. I will also discuss whether these tools support the specific analytic tasks required by the researchers and bring the 'human into the loop' for the '**3 E's**': identify **errors**, **explore** the data for patterns that may be difficult to find through automated methods, and integrate researcher **expertise** in the analysis.   Finally, in Chapter 5, I will describe my design and will discuss both the scalability and 'human in the loop' issues presented in the previous chapters.

4. STATE OF THE ART IN GENOMIC DATA VISUALIZATION

An examination of previous work in the field will help clarify the design requirements for future genome visualizations as well as the potential pitfalls of repurposing existing approaches to larger volumes of data and big displays.

There are many powerful genome visualization approaches that have been developed since the first complete genomes were published. In working with genomics researchers it became clear that these approaches did not enable the comparative analysis of large numbers of gene neighborhoods. In general, few of the approaches appeared to be directed toward comparative genomic analysis of more than a few genomes.

Reviewing the literation on genome data visualization supports the observation that scalability is not a priority for the vast majority of comparative genome data visualization approaches. Nielsen et al reviews genome visualizations broadly, with a section on different approaches to alignment visualization. This review does not explicitly discuss issues arising from scaling visualization designs to large numbers of genomes and over large, high-resolution displays (28). In addition, most of publications for comparative genome data visualization tools do not discuss scalability issues, or present case studies over more than a few genomes. Few of these tools are designed with large, high-resolution environments in mind.

I will engage in a critical review of existing ‘state-of-the-art’ approaches in comparative genomic data visualization, paying particular attention to the degree that these approaches scale to large data volumes and large, high-resolution displays. Assessing the scalability of existing approaches is difficult, since most tools will not immediately permit the visualization of more than a few genomes at once. To get around this limitation, I will consider whether the approach adopted by a tool could be scaled-up, and, if so, would it remain effective. I will consider the facets of scalability discussed in Chapter 3, **pixel-density scalability** and **display size scalability, analytic task scalability** and **sense-making scalability** paying particular attention to perceptual issues such as **leveraging pre-attentive processing**, **enabling visual search over large numbers of gene neighborhoods**, and **avoiding visual clutter**. I will also consider whether this approach would take advantage of **embodied cognition**.

4.1 Non-comparative tools

In this section, I will discuss some of the prominent non-comparative tools that visualize gene neighborhoods and related data. This work is significant to this thesis because it provides an understanding of basic approach to visualize genomic data for a single genome. We will study the visual approach to presenting gene position, size and direction in the genome, as well as gene identity and other descriptive information.

The first genome browsers were developed to visualize the growing volume of complete genome sequences. Genome browsers serve as both an access point to public genomic datasets as well as a simple map providing a visual representation of annotations and research results.

Genome browsers are widely used and well supported by the genomics research communities. Genome browsers serve as tools for publishing genome sequences and research results in a simple and accessible format.

A variety of browsers have been deployed, specializing in the display of genomic data from different organisms or providing different types of interfaces and analysis features. Some browsers are stand-alone applications, allowing private organizations to view proprietary data, but the majority are web-based data portals to public datasets.. Some browsers are dedicated to showing data from a single species (TAIR, MGI, FlyBase, WormBase) and others house data from multiple species (UCSC Genome Browser, Ensembl, NCBI Map viewer, Phytozome and Gramene). Many browsers are based off of common back and front-end architectures, including Generic Model Organism Database (GMOD), the GBrowse framework, which is html-based, or JBrowse, a javascript-based framework (62, 63, 64, 65, 66, 67, 68, 69, 70, 71).

Genome browsers are general purpose tools designed for a variety of research tasks and do not specifically address the comparative domain problem described in Chapter 2. However, it is valuable to study their design and analysis goals, as elements of this design motivates the work in subsequent genome visualization tools, including that employed in this thesis.

4.1.1 Genome browser visualization design

Genome browsers are, by and large, designed to integrate heterogeneous data from a single genome by displaying different data types in distinct ‘tracks’. Tracks include genomic annotation data such as predicted coding sequences, regulatory element binding sites, mutations, epigenetic modification sites as well as genomic data reliability and uncertainty data such as coverage. Sequence conservation data is often presented as well, which provides an comparative overview for researchers, indicated regions of high and low conservation across related species.

To coordinate the presentation of data in these tracks, all elements are mapped to a single, fixed, reference coordinate system so that they can be displayed and related simultaneously. By placing multiple data types in distinct tracks, users can juxtapose information mapped to the reference coordinate system and make connections between different data types.

In figure 12, the JBrowse framework is used to show a variety of data at a single coordinate in a single genome. By placing annotations along sets of parallel tracks, users are able to integrate different data types as well as research from a variety of sources together when considering a region in the genome (20, 71).

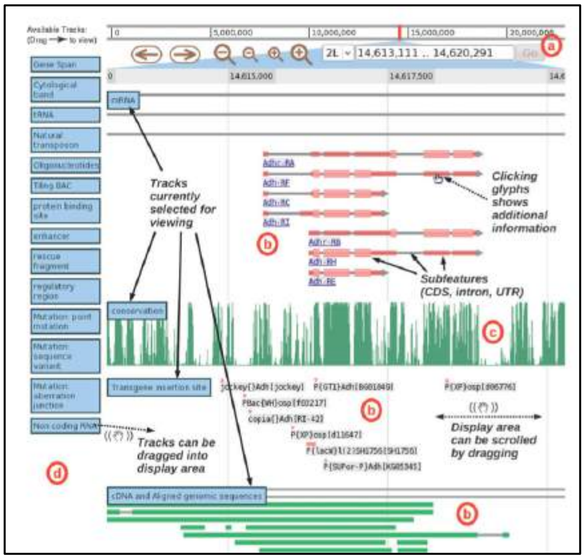


Figure 12. JBrowse framework juxtaposes multiple genome data types mapped to a common reference coordinate system. Reprinted with permission from source: (71) © 2009 Cold Spring Harbor Laboratory Press

Basic navigation is generally provided, either through paging or a ‘Google-maps’-like interaction approach, with click and drag to move and scroll to zoom to different levels of detail. Users are able to decide which tracks to display, and can customize colors, track-order and other graphical features. Labels are generally shown by default over genome features. Further details are generally provided ‘on-demand’, either through small pop-up box or through linking-out to reference sites, as demonstrated in the UCSC browser (20).

Gene representation in genome browsers takes several forms and serves as a standard in genome visualizations depicting gene boundaries. Genome browsers typically depict gene boundaries on several tracks. Due to the biological mechanisms of transcription, there are several useful ways to define a 'gene boundary' in a generic organism. As a result, separate but related tracks will typically show gene sequence information to researchers, by default.

Generally an icon is used to indicate the position and size of a gene. The browser may have two closely spaced tracks to distinguish the strand of DNA on which a gene is found. Alternatively, the browser may use arrows or color to indicate strand and direction of transcription of a gene.

4.1.2 Genome browser paradigm adapted to comparative tasks

The genome browser approach is not targeted to comparative tasks, but integration of heterogonous data from within a single genome. However, several browsers have made adaptations to the visualization design in order to enable comparisons between tracks of data from distinct genomes.

Other browsers show alignments within or between species. GBrowse_syn, SynView and SynBrowse are examples of this approach. Built upon GBrowse, these tools align 2-3 genomes against a signal reference coordinate system and then show orthology or sequence similarity though color and/or line connections (72, 73, 74). I will discuss these approaches in section 4.2

4.1.3 Critical analysis of genome browsers

Genome browser's primary strengths lie in a robust data management and visualization framework for diverse data types. These tools are designed to enable analytic tasks that involve integrating different data at a single site along a reference coordinate system, or moving between the visualization and database entries about interesting genes and features. For instance, a research can go from browsing journal titles, to searching for relevant features, to viewing these features in their genomic context in a reference genome, mapped in parallel to other relevant data.

Browsers are not oriented around comparative tasks, so they cannot address the domain problem described in Chapter 2. However, in the next section we will consider some of the ways in which browser back and front-end architectures have been repurposed for comparative tasks.

4.2 Gene neighborhood comparative approaches: Comparative track visualizations

Comparative track-based visualizations draw a track between genomes in which line connections between orthologous genes are drawn. Many of these approaches are based off of established genome data visualization frameworks that have been re-purposed to work in for comparative tasks, These visualizations generally follow this approach: Each genome is laid-out against its own coordinate system. Genes are generally drawn as arrows, or as boxes on a 'track' with two distinct tracks for each DNA strand. Sometimes the initial positioning a genome is based on alignment with a target sequence or gene of interest. Individual genes are generally labeled with text. Genomes are labeled with text. Sometimes, individual genes are distinguished with distinct colors. Or genes are colored by other attributes. There are two basic subtypes of this approach: two-way and three-way comparisons.

4.2.1 Two-way comparisons

Two-way comparisons between related gene neighborhoods are a common approach to comparative genome data visualizations. In this approach, two genomes are aligned against each other for a gene target of interest. In the center you have the comparative track. Directly above the center are gene boundaries for one genome. Above that are other genomic data tracks for genome 1. Directly below the center you have gene boundaries for genome 2. Below that you have genomic data tracks for genome 2. In the comparative track either lines or bands of color are used to connect related elements.

An example of the two-way comparative track approach is SynBrowse, SynView, and GBrowse_syn. SynBrowse, shown in figure 13, builds upon GBrowse and uses BioPerl modules to construct the visualization (70, 74).

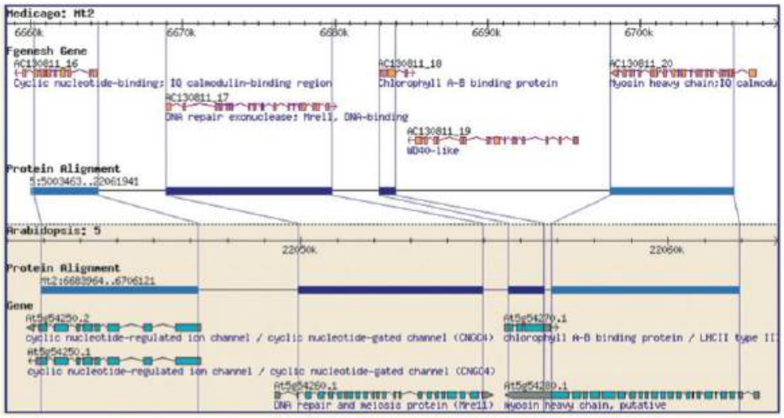

Medicago: Mt2
6660k
6670k
6680k
6690k
6700k
Fgenesh Gene
AC130811_16
Cyclic nucleotide-binding: IQ calmodulin-binding region
AC130811_17
DNA repair exonuclease: Mre11, DNA-binding
AC130811_18
Chlorophyll A-B binding protein
AC130811_19
WD40-like
AC130811_20
Myosin heavy chain:IQ calmodu
Protein Alignment
5:5003463..22061941
Arabidopsis: 5
22050k
22060k
Protein Alignment
Mt2:6683964..6706121
Gene
At5g54250.2
cyclic nucleotide-regulated ion channel / cyclic nucleotide-gated channel (CNGC4)
At5g54250.1
cyclic nucleotide-regulated ion channel / cyclic nucleotide-gated channel (CNGC4)
At5g54260.1
DNA repair and meiosis protein (Mre11)
At5g54270.1
chlorophyll A-B binding protein / LHCII type II
At5g54280.1
myosin heavy chain, putative

Figure 13. SynBrowse application comparing two gene regions. Reprinted with permission from source: (74) ©2005 BMC Bioinformatics.

Other examples of this approach include MizBee, which is a multi-scale visualization approach that includes a gene-neighborhood level scale (75). Other examples of this include Combo, which has both a reference and query genome, alignment track and 'dot plot' panel. (76). This approach is discussed in Section 4.2.4.

Another example of a two-way comparative approach is found in CGAT, except this approach draws bands between genes and colors genes by degree of similarity (77). This approach is shown in figure 14.

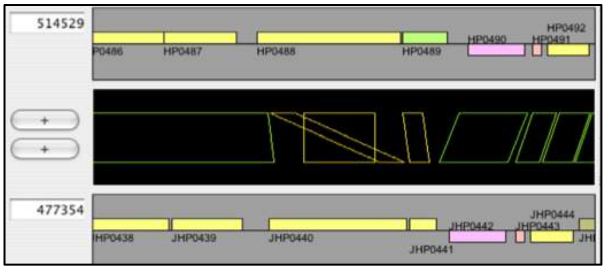


Figure 14. CGAT comparative browser. Reprinted with permission from source: (77) ©2006 BMC Bioinformatics.

4.2.2 Three-way comparisons

A variation on the two-way comparison approach is a three-way comparative approach. In this approach, one genome is taken as a reference, and is drawn in the center. 2 additional genomes are aligned against it, one drawn above and one drawn below. Lines connect orthologs between the central genome and the top genome, and the central genome and the bottom genome. The top and bottom genomes are not compared directly to each other, but indirectly through comparison with the central genome.

An example of the three-way comparative track approach is SynView. SynView is another GBrowse-based framework that allows for comparisons between 2-3 genome at once. Like the above approaches, it aligns 2-3 genomes against a single reference employs lines and bands to represent synteny between regions or orthology between genes (73, 70). This approach is shown in figure 15.

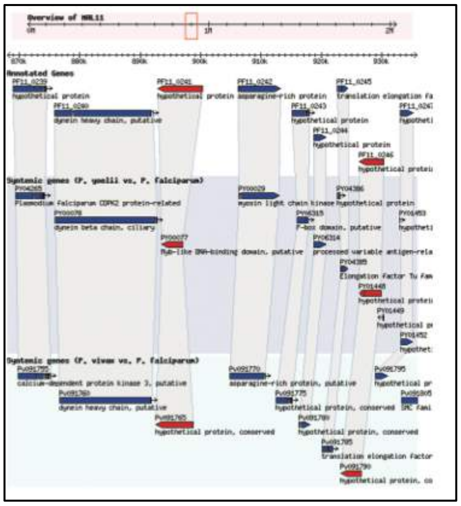


Figure 15. SynView framework comparing 3 gene neighborhoods. Reprinted with permission from source: (73) ©2006 BMC Bioinformatics.

Another tool that adopts this approach is the Artemis Comparative Tool (ACT), which shows features from 3 genomes alongside bands showing synteny or orthology between regions of those 3 genomes (78). This approach is shown in figure 16.

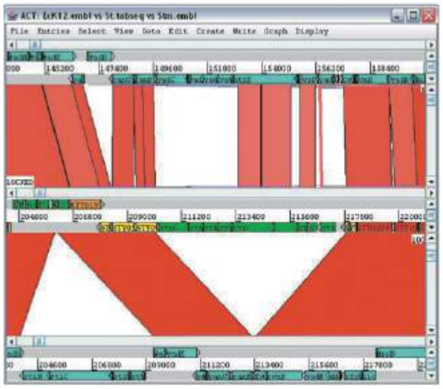


Figure 16. ACT visualization tool comparing 3 genomes. Reprinted with permission from source: (78) ©2005 BMC Bioinformatics.

Expanding this approach to more than 3 genomes raises several problems. If one genome is taken as the target, and other genomes are aligned against it and placed in a stack above or below the target, line connections are difficult to draw. For instance, if a gene is missing on one genome, it is difficult to know how to connect a line from the adjacent genomes. If a new gene appears in one genome, but is unaccounted for the in the reference genome, it will not be connected to genes in other genomes.

4.2.3 Multi-way comparative-track approaches

Several tools attempt to get around this problem by stacking 2 and 3 way comparisons in a single view, each aligned against a common genome. This allows researchers to see how many genomes align against a common reference. For instance, GBrowse-syn is capable of showing 4 gene neighborhoods compared against one reference genome (72).

Other approaches use line connections between genes or blocks of DNA, adopting a methodology similar to parallel coordinate plots. Mauve is one such visualization approach that supports the comparative analysis of collinear blocks across less than a dozen related genomes. Color encodes co-linear blocks of genes and line connections encode synteny (79). This approach is shown in figure 17.

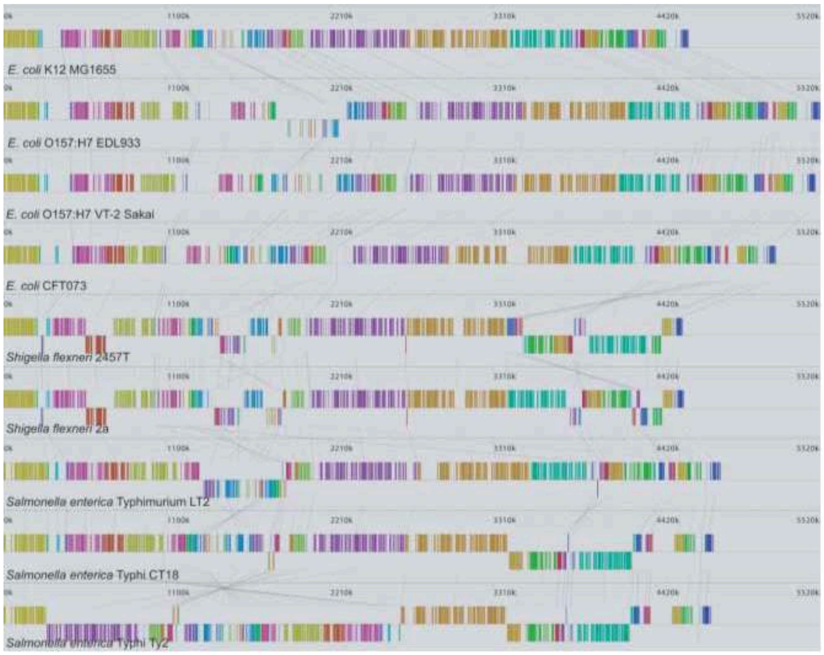


Figure 17. Mauve showing 9 related bacterial genomes, with co-linear blocks sharing colors across genomes and line connections between related elements. Reprinted with permission from source: (79) ©Cold Spring Harbor Laboratory Press 2004

These approaches that use line-connection approaches for more than 2 genomes do not do so for more than several genomes at once. There is no indication in these publications that such tools will permit researchers to load data from more than several genomes at once. Fundamentally, these tools are not implemented to accommodate more data.

4.2.4 Critical analysis of comparative track approaches

These tools address many of the domain task requirements. However, they are not fundamentally designed to accommodate more than a few comparisons. Even if the approach were modified to accommodate more date, there are significant limits to scalability of these approaches.

For instance, considering Mauve as the best-in-class for line connection-based comparative genome visualization, visual clutter is a significant problem even for a small gene regions from a few genomes. If this approach were to be scaled up both in data volume and display resolution to show wider regions from more genomes, visual clutter would likely get worse.

Pixel density scalability is limited in Mauve, because it is difficult to follow line connections between closely spaced genomes in high-density. Line connections require space to distinguish and follow, so perception of relationships across genomes would likely suffer.

Display size scalability is limited in Mauve as well. Since elements are drawn relative to a fixed coordinate system, it is possible that related genes will arise on opposite ends of the screen. Line connections cannot be easily perceived at a distance, so users will likely step-up to the display to see the relationships between genomes. As they do so, they effectively will be unable to see relationships between genes on opposite ends of the screen. Line connections are an encoding that are not display size scalable.

Analytic task scalability is also limited. As more genomes are fit onto a larger display, the question that must be asked is whether the researcher can efficiently compare many gene regions at once. One of the reasons that Mauve has limited analytic task scalability is that the relationships between genes are not encoded in a way that permits pre-attentive processing. Effectively, users must conduct a visual search over the representation, following lines and remembering relationships across genomes. This taxes cognitive resources, and places significant limits on analytic task scalability.

Mauve is an excellent tool for comparing small regions of a few genomes, but it does not scale to large volumes of data, large numbers of comparisons or big, high-resolution displays. Other approaches in this task face similar limitations, indicating that a different approach is needed to address the domain-problem presented in Chapter 2.

4.3 Gene neighborhood comparative approaches: Spatial alignment and color to represent orthology

4.3.1 GeneRiViT Application

Several gene neighborhood comparison tools use color and alignment to depict orthology relationships between genes in gene neighborhoods. GeneRiViT is one such tool. GeneRiViT adopts a circular approach, with each genome displayed in one layer of concentric circles. This approach avoids adopting a fixed reference coordinate system, and each genome can rotate and be aligned against an arbitrary target, such that a target ortholog cluster will be positioned in a single line in the center of the view (80). This approach is shown in figure 18.

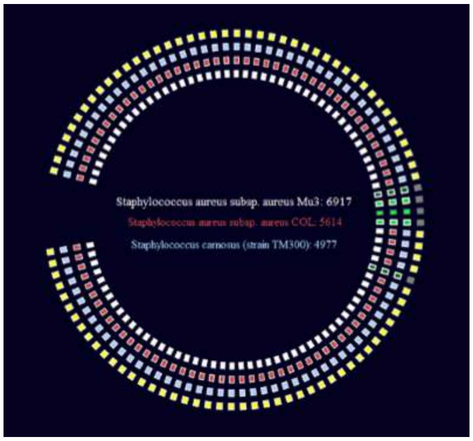

Staphylococcus aureus subsp. aureus Mu3: 6917
Staphylococcus aureus subsp. aureus COL: 5614
Staphylococcus carnosus (strain TM300): 4977

Figure 18. GeneRiViT showing comparisons between gene neighborhoods in 4 genomes. Reprinted with permission from source: (80) ©2012 IEEE

The circular layout employed in GeneRiViT places a limit on scalability. While it is technically possible to scale the approach up to larger data volumes on big displays, there are several notable obstacles. Display size scalability is less of a significant problem as in the linear, line-connection based approaches. Those approaches would likely suffer on a large display, since related elements might be placed on opposite ends of the display without a scalably perceptible means to connect them. In GeneRiViT, positioning similar genes in space takes advantage of clustering. In addition, pixel-density scalability is less of a problem, since the representation of a single genome is already fairly compact and could be further compressed.

However, there are limits to the number of genomes that can be fit in a given display. Genomes near the center of the circle will eventually have insufficient space to represent a given genome. At the outer ring, a genome will have a large diameter to display this information, and the base-pairs to pixel ratio will be sufficient. At the center of the circle, a genome will have insufficient pixels per base pair, and will be unable to show that genome at sufficient detail. At the center, fewer genes will be shown, and effectively the window over which the comparison will take place in a center genome will be extremely small.

In addition, color is used to identify genomes, or gene insertions and deletions. However, the identity of other genes remains unknown. Effectively. GeneRiVit is a tool to identify deletions and insertions, not investigate gene neighborhoods for the full range of potentially interesting variations.

4.3.2 PSAT Application

PSAT is, in many respects, the tool that most closely addresses the domain problem by encoding orthology through color and spatial positioning. In this approach, orthologs are pre-computed and stored in a database, retrieved, and genomes are presented with a target gene in the center of the display and related genomes aligned to the gene target. Genes are shown on one of two strands, indicated by parallel tracks for each genome. Each gene on display is given a color, and orthologs are given the same color (81). This approach is shown in figure 19.

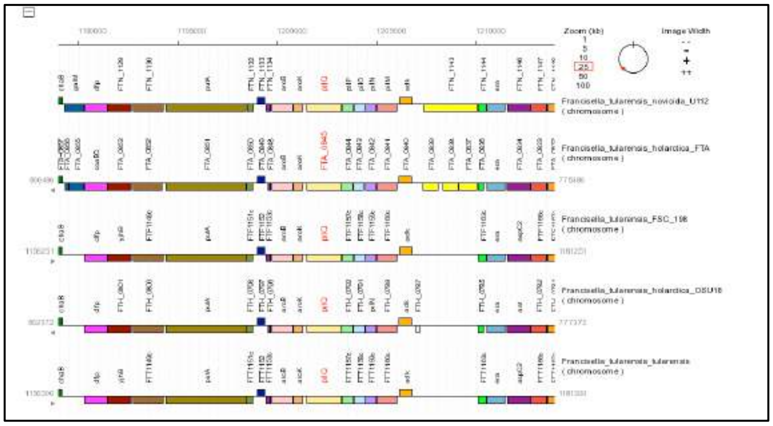

Zoom (kb)
1
5
10
25
50
100
Image Width
--
-
+
++
Francisella_tularensis_novicida_U112
(chromosome)
Francisella_tularensis_holarctica_FTA
(chromosome)
Francisella_tularensis_FSC_198
(chromosome)
Francisella_tularensis_holarctica_OSU18
(chromosome)
Francisella_tularensis_tularensis
(chromosome)
pdpA
FTA_0845

Figure 19. PSAT visualization comparing several gene neighborhoods. Reprinted with permission from source (81). ©2008 BMC Bioinformatics.

This approach has the benefit of drawing attention to relationships between genes pre-attentively, but since there are limits to the number of colors that can be distinguished, the automatic application of color across larger volumes of genomic data is problematic. In addition, the use of text labels by default limits the pixel-density scalability and thus the amount of data that can be displayed. These labels also increase the potential for visual clutter, which can hamper understanding.

4.4 Gene neighborhood comparative approaches: Dot plots

4.4.1 Dot plots description

Dot plots are a representation that shows alignment between genes in several genomes in a format that scales to permits a complete genome to be shown in a single view. The x-axis is a coordinate space for one genome and the y-axis is the coordinate space for a second genome. Genes in these and other genomes are positioned based on the location of the ortholog in genome

A and genome B. Color is used to distinguish genomes, permitting multiple genomes to be mapped simultaneously.

In two way-comparisons, that of the x-axis genome and y-axis genome, the interpretation of these plots is straightforward. Regions of a genome that have identical order to the genomes in the x and y-axis will form a diagonal line with slope of 1 and y intercept of 0. When a gene deviates from the order in either genome, it jumps out of the diagonal. If a segment of a genome has been translocated, it will form a diagonal line with slope of 1and a different y-intercept. If a segment has been inverted, it will form a diagonal line with negative slope. In multi-way comparisons, diagonal bands of different colors indicate alignment of genes in multiple genomes with the two reference genomes. The GeneRiViT dot plot is shown in figure 20.

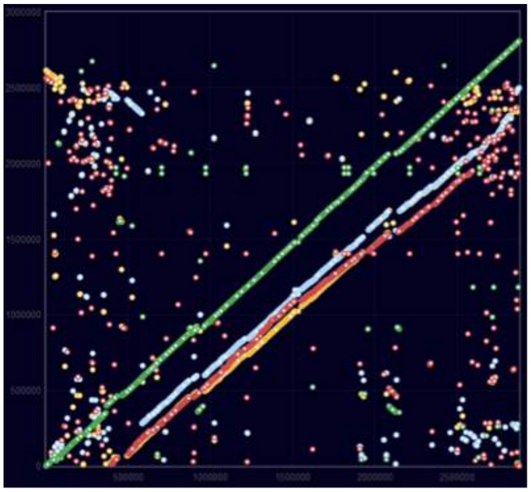


Figure 20. Multi-way dot plot from GeneRiViT application. Reprinted with permission from source: (80) ©2012 IEEE

Dot plots do not show the viewer the genomic context of the genes displayed. For this reason, these may be considered overview, comparative representations, not gene-neighborhood representation. However, some tools integrate a dot plot with a gene-neighborhood-level representation, as in GeneRiViT and Combo (80, 76). When users click on a gene in the dot plot, they can pull it up in a genomic context comparative view, and vice versa. The Combo tool is shown in figure 21.

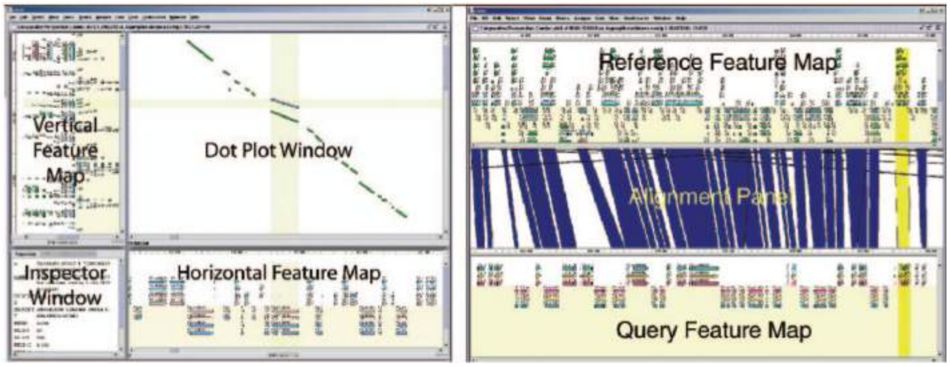


Figure 21. Combo integrates a dot plot with a feature map and a line-connection based alignment track. Reprinted with permission from source: (76) © BMC Bioinformatics, 2006.

4.4.2 Dot plots critical analysis

Dot plots allow researchers to quickly spot regions of alignment and identify genes in locations that differ between 2 genomes. Diagonal lines and dots out of line are easy to spot. However, once more genomes are added to the scene, it becomes more difficult to mentally map the dot plot representation to the genomic context, particularly if these genomes differ markedly from one or both of the reference genomes.

By removing the data from its genomic context, more genes can be simultaneously displayed. So this representation might scale for the analytic task of identifying regions of similarity across several complete genomes. However, it does not address many of the analytic tasks identified in Chapter 2, even with a linked genomic context view.

Further, it is not particularly scalable in comparing across many genomes. This representation works well for several-way comparisons of lots of genes, but is more difficult to follow as the number of colors used to distinguish genomes increases, and the number of differences between the query and reference genomes accumulate.

4.5 Comparative Overview Visualizations

In this section I will describe several 'overview' comparative genomics visualization approaches. While these tools do not address the domain problem, in that they do not show gene neighborhood details to researchers, they provide valuable context for studying encodings of genome similarity as well as potential approaches for scalable genome data visualization.

4.5.1 Whole-genome circular comparative tools

A variety of whole-genome circular comparative tools have been developed to address comparative genomic questions. This approach takes two genomes and places them the diameter of a circle. Similar regions, known as syntenic regions, are indicated by drawing bands from one genome to another through the center of the circle. These bands are colored, to distinguish between regions more clearly. Visualization designers claim that the circular layout gives the visualization increased space to indicate regions of similarity.

Some prominent examples of this approach include Circos and MizBee. Circos permits both whole-genome comparisons between 2 genomes, and the depiction of data in parallel circular 'tracks'. MizBee utilizes a similar approach to whole-genome comparison, but also includes a set of windows that permit analysis at multiple scales simultaneously (75, 82). This approach is shown in figure 22.

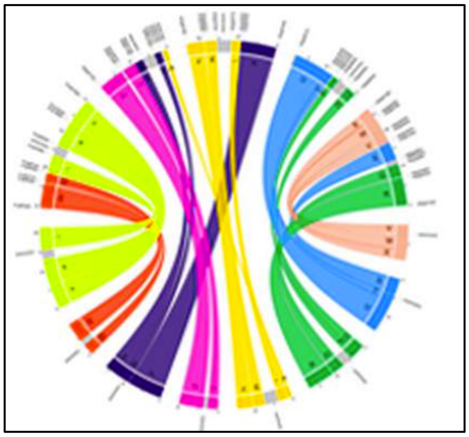


Figure 22. Circos comparative visualization tool for comparing two complete genomes. Reprinted with permission from source: (82) ©Cold Spring Harbor Laboratory Press 2009

Theoretically, this approach could be adapted to the gene-neighborhood comparative task, by drawing lines between orthologous genes rather than syntenic regions in the two genomes under comparison. Small multiples could then be used to relate many genomes to a single reference.

In a two-way comparison, this approach could allow more genes to be shown in any single genome, as compared to linear line-based comparative approaches described in section 4.2. This is because the diameter has greater numbers of pixels than a line across the screen. In addition, this view resolves the problem of many line connections, because the entire center of the circle is available to indicate connections between similar genes.

However, this solution has many of the problems that are present in line-based comparative approaches. There is the potential for many edge crossings, even with greater space in the center of the circle. In addition, there would not be enough distinguishable colors to separate distinct orthologs, which may complicate the perceptibility of the encoding.

While this task may scale to show many genes on a large display, it would not easily permit the comparison of many genomes. Some researchers have used a ‘small-multiples’ approach, showing relationships between two distinct genomes in each small multiple. While the small multiples approach may help address this problem, relating similarities and differences between small multiples would be challenging. In addition, a large proportion of the space in a circular representation is dedicated to showing orthology relationships, leaving less space for showing gene neighborhoods. For these reasons, adapting this approach to a gene-neighborhood comparative task would not be effective.

4.5.2 Sequence surveyor

Sequence surveyor is an overview visualization for large-scale genome alignment data. It is designed to enable the comparative analysis of many genomes in one view. After computing orthology or synteny across related genomes, this application displays this data in parallel using color, position and aggregation encodings. The goal is to give researchers an overview of the similarities and differences in related genomes. They apply techniques to develop a scalable encoding, which permits them to visualize 100 bacterial genomes in a single view (83). This approach is shown in figure 23.

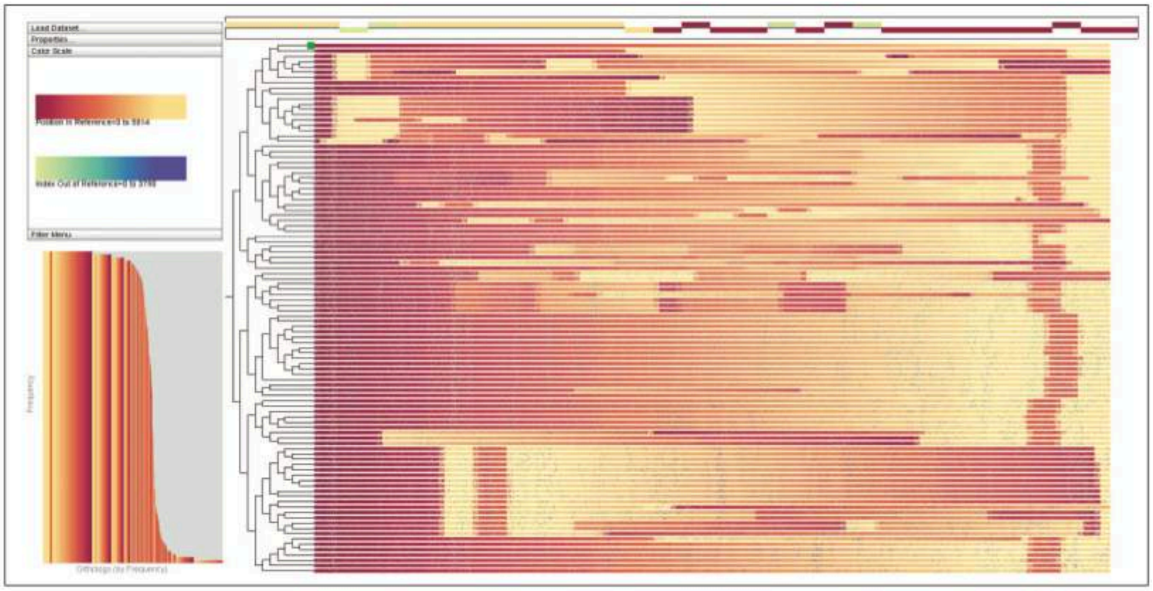


Figure 23. Sequence surveyor showing orthology relationships across 100 synthetic genomes. Reprinted with permission from source: (83) ©2013 IEEE

Figure 23 shows orthology relationships across 100 synthetic genomes generated by an evolution simulation. Genes are placed directly adjacent to each other, and are order by position in the genome and colored by position in the reference genome. A dendrogram on the left shows a phylogenetic tree depicting relationships between the simulated genomes. A histogram on the left shows a frequency distribution of orthology group sizes.

In addition, there are several alternative overview visualizations presented, which position and size orthologous genes based on different criteria, to give researchers a high-level view of the similarities and differences between genomes.

They primarily utilize color and spatial positioning to encode information about the genomes being compared. Color and position of a gene can be mapped to: gene index (order in the genome), gene start position, orthologous gene position in a reference genome and several frequency properties. Several different color schemes are provided, from Color Brewer ramps and a gray coloring scheme (84).

Albers claim that the color mappings provide visual patterning over the data, making it easy for users to spot similarities and differences between genomes. It supports pre-attentive processing, by creating fields of similarity encoded by color. Spatial positioning provides a spatial clustering utilizes the tendency of the visual system to group similar items together.

While this approach would not address the domain problem, since it only provides an overview of the data, it presents several valuable visualization principals for scalable design. While these researchers did not directly present this information on a big display, their design would be scalable to large, high-resolution environments and larger data input sizes. I believe that this design could scale to big displays, overcoming all 4 roadblocks to display and data scalability. This design achieves pixel density scalability by encoding data in such as way as to minimize space per-genome. The linear design for each genome, relying primarily on color and spatial positioning of genes to convey information is highly compact, maximizing the number of genomes and genes that can be displayed in a single view. This design achieves display size scalability by using spatial positioning to cluster information, such that a user looking at one portion of a big display will see related elements. Researchers would be able to step back to see a high-level overview, and step closer to see more details in a region of interest. Analytic task scalability is achieved since this visualization supports the task of viewing broad relationships between genes in many related genomes. Sense-making scalability is achieved by making information available pre-attentively to users in a scalable format.

This approach could take advantage of embodied cognition, by enabling researchers to encoding information spatially, such that spatial memory and proprioception can be used over physical navigation. However, this approach does not appear to capitalize on the possibility for viewing multiple levels of detail, depending on distance from a large, display.

4.6 Sequence visualizations

In this section, I will describe raw-sequence comparative visualization approaches. While these approaches only show short stretches of sequences, not gene boundaries, and focus on nucleotide-level differences, it is valuable to examine the scalability of these approaches.

4.6.1 Color and Accordion

Sequence Juxtaposer is a raw-sequence comparison visualization. This approach utilizes color and accordion drawing, which is a subset of the ‘focus+context’ approaches in information visualization. This approach provides a context for details of interest, stretching the selected sequence regions and species, while compressing the remainder.

While the application of color may be somewhat visually overwhelming as the volume of sequence data is scaled up, or this visual approach is scaled to fit a larger display, up-close the data can be pre-attentively interpreted to identify broad patterns in sequence conservation.

4.7 High-resolution and large-display based genome visualizations

Orchestral is a visualization application built for large, high-resolution display walls showing copy number variations across 100 or more related genomes. While it does not address this domain problem, it does demonstrate several features that point the way toward scalable genome visualization for the comparative gene neighborhood problem.

The visual approach is high-density, packing in the relevant information into a compact encoding relying primarily on color and alignment to communicate relationships between genomes. This visual approach is sense-making scalable, by avoiding visual clutter as the data volume and display size increases. It utilizes pre-attentive processing, instead of visual search to convey relatedness between genomes, since information is available without needing to scan or use cognitive resources. This approach is also display size scalable, by clustering the sections of genomes to be compared and providing overviews at a distance and details up-close (86). This approach is shown in figure 24.

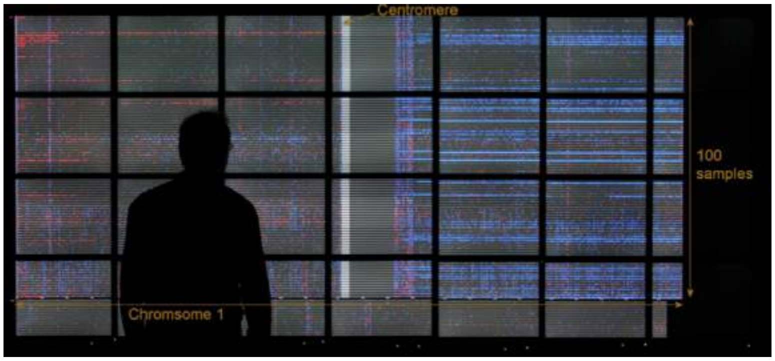


Figure 24. Orchestral large, high-resolution environment visualization comparing copy-number variations across many genomes. Reprinted with permission from source: (86) © 2013 IEEE

This approach is not particularly interactive, and does not provide visual queries across the dataset. However, the use of color, alignment, high-density presentation appears to be effective for this domain problem.

4.8 Summary

Of the visualization approaches that provide detailed views of gene neighborhoods along with comparative information showing orthology between genes in distinct genomes, none address visual scalability or display on large, high-resolution displays. Mauve is the only visualization approach that provides examples of comparisons across more than 3-6 genomes, with one example showing the comparative analysis of 9 genomes (79).

In addition, none of the approaches that address the domain problem could be scaled effectively to larger data volumes and larger displays. The line-connection based approaches would struggle with pixel-density scalability, since these approaches cannot be compressed without significant perceptual issues. These approaches also would have difficulty with display size scalability, since line connections across the display can be difficult to follow and would require visual search or working memory. The lack of pre-attentive cues, and the high potential for visual clutter make line-connection based approaches ill-suited to the domain problem where scalability is needed.

While color and alignment are used effectively in PSAT, the automatic application of color limit scalability in comparisons, since users can only effectively distinguish a limited number of colors in one visualization. In addition, the heavy reliance on text limits pixel-density scalability, since the representation can only be compressed to a point where the user can effectively read the text and much of the display is occupied by whitespace (81).

GeneRiViT is relatively pixel-density scalable and display size scalable, by encoding data primarily through selective application of color and alignment. But the lack of complete gene neighborhood details and lack of details-on demand, limit the utility of this approach for the domain problem (80). As discussed in Chapter 2, gene boundaries and identification-on-demand are essential components of the analysis problem.

Of the approaches not geared to this domain problem, the key characteristics are: high-density genomic data presentation, user applied color or color ramps to identify relationships between elements in distinct genomes, and spatial alignment to bring objects of comparison into a single region of focus or to take advantage of spatial memory. These approaches would be well suited to display over large volumes of genomic data and on large, high-resolution displays, and will serve to motivate my design.

5. DESIGN

This chapter presents BactoGeNIE, Bacterial Gene Neighborhood Investigation Environment, an interactive genomic data visualization approach for comparing large numbers of bacterial genomes in high-resolution environments. In Chapter 2, I described gene neighborhood comparative analysis, a current research area in bacterial genomics that arises from growing volumes of sequence data. This is an area that requires new visualization approaches and which stands to benefit from display in large, high-resolution environments.

To address this problem and design a new visualization approach, I worked closely with genomics researchers over the course of several years. I discussed the existing approaches adopted by researchers to address this research area and identified their limitations, described in Chapter 3. I then worked with researchers to formulate a set of goals to motivate the visualization design, which are introduced in Chapter 2.

Design goal 1: Visualization must help **bring experts into the data mining and automated processing loop**, verify the output of computational methods and to enable the identification of relationships and patterns that would be difficult to catch through computation alone. For this domain, this needs to be accomplished by enabling the following analytic tasks:

a) identification of automated data processing **errors** including: making **evident breaks in genome assembly**, potential **errors in annotation** and missed **identification of orthologous genes**

b) enabling **exploration** involving the identification of **gene truncations** and sequence **deletions, insertions, duplications** and **inversions**, along with support for subtle pattern identification

c) bringing **expertise** to bear on the analysis, by providing sufficient details about the genes and strains on display

Design goal 2: Visualization must **address growing volumes of sequence data**, providing new modalities for viewing and interacting with larger volumes of genomes in a way that addresses research questions of interest. A more scalable design is needed, to enable the comparison of hundreds of gene neighborhoods simultaneously.

Design goal 3: Visualization ought to run on large, high-resolution displays as well as high-resolution personal workspaces, to accommodate

In Chapter 3, I described the potential benefits of addressing scalable design using large, high-resolution display environments, and the challenges that arise in adopting this approach. The following types of scalability must be considering in constructing a novel visualization approach for showing big genomic data on big displays:

1) Pixel-Density Scalability: As pixel-density increases, ensure that the design is compatible with a high-density display, taking advantage of increased resolution of display to show more gene neighborhoods in a given portion of the screen.

2) Display-Size Scalability: As display size increases, create a visual approach that takes advantage of the large display space to display more gene neighborhoods.

3) Analytic Task Scalability: As screen size and resolution increases, create a visual approach that enables the performance of the analytic tasks described in Chapter 2 across more genomes.

4) Perceptual scalability: Ensure that the visual approach adopted to enable comparisons between many genomes on large, high-resolution displays scales perceptually to accommodate sensemaking around the increased number of genes and genomes on display.

In Chapter 4, I described the primary ways in which existing visualization approaches failed to scale, finding that many approaches did not even consider scalability in their design, only accommodating comparisons between a few genomes at once. I found that even if these designs were adapted to accommodate more genomes, the encodings adopted to display relationships between related genes did not take advantage of pre-attentive pattern identification, and would face serious challenges with visual clutter.

In the following Chapter, I will provide a detailed description of the visualization design and the techniques employed to enable the identified analysis tasks across large numbers of completely sequenced genomes. I will highlight the ways in which this design takes advantage of high-resolution environments for displaying large volumes of bacterial genomic data.

5.1 Visualization design: Addressing analytic tasks

To address the analytic tasks and build a high-density, scalable design, it is necessary to determine what information needs to be shown to the researcher. Genomic datasets typically contain many pieces of related information and if each were shown for all genomes, the display would quickly be overwhelmed with visual clutter.

1) Data needs to be shown at 'gene neighborhood' scale, by showing **gene boundaries**. Raw sequences or whole-genome comparisons are not priorities for this particular set of tasks.

2) Visualization must show gene order: researchers were interested in not just sets of genes commonly co-located in close proximity, but also in the order of these genes. Variations in gene order indicate potential inversion, deletion or duplication events, which have potential significance.

3) Visualization must show gene relative distances, not just gene order. Simply depicting the order of genes is not enough, because gaps or variations in distance are significant. A gap indicates a potential error in annotation, failing to pick up a gene. In bacteria you do not expect to find large gaps between genes. A gap could indicate a break in an operon transcription unit, which has implications for the function of the genes in that operon.

4) Visualization must show gene size. Researchers wanted to be able to detect truncation events, because these indicate potential modification or loss of function in a gene, compared to the complete counterpart.

5) Visualization must show orthology in genes of interest, so that researchers can compare gene neighborhoods across genomes.

8) Visualization must make annotation data accessible to the researcher. Once interesting genes or relationships are observed, the identity and description of a gene will be needed to continue the investigation.

7) Visualization must allow many genomes to be compared at once. This was the key feature lacking in many visualizations: failure to accommodate contemporary volumes of genomic data and failure to take advantage of resolution and scale of modern graphics environments. Encodings that did not minimize space per genome and space between genomes would detract from this. Scalable design will be described in the following subsection.

5.2 Visualization design: Addressing scalability

Through close work with bacterial genomics researchers, it became clear that existing visualization tools did not adequately enable the analysis tasks described in the preceding section. In some cases, the visualization does not address the analytic tasks described in Chapter 2. For instance, some tools show data at the wrong scale, focusing on raw sequences of whole-genome comparisons. Other approaches are not fundamentally oriented around enabling comparisons between distinct genomes. In others, summaries or high-level overviews are utilized, leaving out necessary information such as gene size or distance between genes.

Even approaches that are geared toward the analysis tasks described in Chapter 2 are generally not designed to be scalable and only allow a few genomes to be compared in one view. This lack of scalability is a significant problem, since rates of data generation are outpacing the capacity of visualization designs and new analysis problems require visualizations that present this data.

Further, visualizations are failing to take advantage of increased resolution provided by high-resolution displays and large display walls, as described in Chapter 3. Even on traditional desktop monitors, resolution and size has increased, allowing high-resolution visualization approaches to be adopted to visualize large datasets.

In Chapter 4, I discussed the reasons that existing designs could not be simply 'scaled-up' to accommodate comparisons between more genomes or 'expanded' to fill large-display walls. Neither modification to these designs will usefully show more data or make this data perceptually accessible to researchers.

In this section, I will describe and justify such a set of principals underlying scalable genome visualization design. I worked with researchers to build upon the low-density, low-resolution visualization approaches that researchers have found useful for low-volume genomic data analysis into high-density, high-resolution visualizations suitable for comparing of genomes in one view.

First, I will discuss my approach to **high-density genome visualization design,** focusing on pixel-density scalability, analytic task scalability and perceptual scalability. Using effective geospatial visualizations as a model, I adopted a selective encoding to enable **'high-density' genome display**. This entailed shifting some data attributions onto interaction events. I explored ways to use **scalable encodings that enabled pre-attentive pattern finding**, such as color and spatial positioning, to represent data in ways that limited confusion in presenting large data sets and took advantage of large-display environments. I designed a new interactive visualization algorithm called **'orthologous gene targeting'**, that rapidly pulls together relevant information and creates high-resolution, comparative genomic maps for rapid analysis. This approach allows researchers **to view overviews and details through physical movement**.

5.2.1 Selective encoding and high-density layout

Genome visualizations typically employ a variety of approaches for laying out genomic data. Browsers show data in linear 'tracks'. Comparative viewers show relationships in a 'comparative track' situated between two genomic data tracks. Circular comparative viewers move the comparative track into the center of a circle, or place genomic data in concentric circles that can be aligned based on ortholog similarity. While these layouts are effective in showing comparisons between a few genomes, they will not scale to hundreds of genomes and will not permit high-density representation.

In my approach, contigs are stacked closely together, maximizing the number of genomes that can be displayed per pixel. This allows more genomes to be visualized in one view. This layout eliminates a 'comparative' track and limits space traditionally used to display gene identities or other data. This brings a new challenge: how to selectively encode information in a way that is compatible with a high-density layout.

Effective visualizations establish priorities in which data to display at a given time. In constructing a high-density encoding for genomic data, we had to make choices in what was displayed. Genome browsers excel at showing multivariate data of a single genome, showing coding sequences, mutations, regulatory factor binding sites, and epigenetic modification sites in separate but parallel tracks. However, adopting a similar approach would lead to a cluttered display that could only fit a few genomes at once. Further, the representation would not enable rapid pattern identification across many elements or across a large display. To create a 'high-density' genome visualization approach, we needed to select one data type to show in a single 'track' for each genome.

Based on the list of features that need to be shown we adopted a selective encoding that shows in the default view 1) Contig boundaries and identities 2) The position, size and strand of genes. I adopted the convention of showing genes as arrows, whose length corresponds with length in nucleotides of the coding strand and whose direction corresponds with the DNA strand on which the coding sequence is located and direction of transcription.

In other comparative gene neighborhood representations, gene identities were shown through text labels. These text labels, however, take up valuable space, and will not be particularly useful as the number of elements on-screen increase. Given the volume of data on display, it is critical to think about ways to compress the volume of information on-display, particularly non-perceptually scalable elements such as text.

5.2.2 Pre-attentive cues in representations of gene identity and orthology

In constructing a high-density genomic data visualization for large, high-resolution spaces, it is important to consider whether the adopted encodings will be perceptually scalable. Will data and relationships be perceived by the user even as the number of data elements on screen are scaled up? In considering this question, several issues needed to be considered: how to give users access to gene identity and how to represent orthology? This is a critical question, because the user needs to analyze gene neighborhood composition around target genes across hundreds to thousands of genomes. To achieve analytic task scalability, pre-attentive cues need to be utilized so comparisons can take place directly and not through visual search. When visual search is need, aids to this search may be used.

Gene identity is typically shown in one of two ways, drawn with an adjacent text label or a unique color. This representation will not ultimately scale to a large-volume of genes. Text labels are not perceptually scalable as more text is added on-screen, it becomes less sensible to the viewer, cluttering the scene and distracting from representations of data. Similarly, it is not possible to apply a unique color to every gene on-screen, or even to apply a unique color to every set of orthologs on-screen. Research has shown that users can only distinguish a limited number of colors at a time, so unique colors cannot be applied in an automated fashion across so many entities.

I described in the preceding section how orthology between genes could not be shown through a distinct ‘comparative track’ with line connections drawn between similar elements, due to the way that this track limited the density of genomic data display. An additional reason to avoid this encoding comes from the limit it places on scalability, both in pixel-density and display size scalability, as well as perceptual scalability. Line connections can indicate relationships, but do not work effectively with large numbers of connected entities or across large-displays.

To address this issue, we focused on interactivity, to present non-scalable data for selected attributes ‘on-demand’. To provide gene identity and other relevant details, we decided to employ a **‘details-on-demand’** approach, where user selections bring up a menu with gene id, description and other available information. This approach has been documented to be an effective way to present information while avoiding overwhelming the visualization with detail (87).

While finding details about a particular gene, we used **coordinated highlighting** to show orthologous genes. When a user hovers over a gene, all orthologous genes on screen are highlighted. This provides a window into a set of contigs, allowing the user to get a quick sense of which genes may be of interest and which patterns might be worth considering. This technique is perceptually scalable, because highlighted elements will be visible even with many elements on-screen (87).

Once the user has identified genes of interest through exploration, they can apply colors to genes, essentially **tagging orthologous genes of interest**, which serves as a persistent and scalable identifier for genes of interest. Once a tag is applied, all orthologs should be colored the same way, to enable pattern identification across genomes. This use of color to identify genes is scalable, because users can quickly spot their tagged elements, even in a scene with many elements.

With a high-density view, it can be difficult to 'hover' over genes with precision. To get around this problem, I added an 'accordion' function, which expanded the height of the contig under the mouse, and all the genes within that contig as well. Contigs around the target are repositioned, so that no elements are obscured.

5.2.3 Spatial positioning as a perceptually scalable encoding of orthology and gene content

Analysis of geographic data can be relatively intuitive because such data is spatially presenting in a meaningful coordinate system. Elements near each other on a map are near each other in a physical space. Spatial position depicts important information, and interpreting an effective mapping of data is relatively natural, or requires a small amount of training.

While researchers may desire to analyze genomic data with the same ease as analyzing data on a high-resolution map, genomic data is different in many respects. Unlike physical geography, where there is a fixed coordinate system around which data is organized, different genomes will have distinct mappings of elements within different coordinate systems. The coordinate position of a gene in one genome is not comparable to the coordinate position of a similar, or orthologous gene in another genome. Through evolution genes are shifted around. Sections of genomes are duplicated and placed elsewhere. Genes are reordered and moved onto different strands of DNA. Relating elements across many genomes cannot be effectively done relative to a single coordinate system.

For instance, a gene involved in pathogenicity in one strain of *E.coli* might map to nucleotide 1000. An ortholog in a related strain might map to two places in that genome, nucleotide position 130,000 and 560,000. The fact that these orthologs are in different absolute positions can be meaningful for some analysis tasks, however in comparing gene neighborhoods what matters is the composition of adjacent genes.

However, from a visualization standpoint, it is possible to design the visualization that allows users to control the placement of elements such that the spatial positioning of genomes is meaningful for the comparative task at hand. To accomplish this, we created interactive features that enabled researchers to 'sort' genomic segments so that their order on screen would relate to researcher queries. For instance, sorting genomic segments by length of the contig helps researchers isolate breaks in genome assembly. Sorting by species, brings together segments from the same strain of bacteria, so that genomic content could be compared across species. Sorting by gene allows researchers to bring together genomic segments with genes of interest, effectively filtering data by relevance. This ordering of genomic segments is essential to bringing meaning to 'nearness' in the y-coordinate space.

Spatial positioning is also particularly important on a large display, because detailed information close in space can be directly compared. Detailed information that is related, and ought to be compared, but on opposite ends of the display will require user memory and cognition. In addition, operations that could occur directly will instead require a visual search across a big display surface and over large volumes of information, which will take time and effort on the part of the user.

Similarly, we enabled the users to position genomes in x-space such that similar genes are aligned, which is a form of clustering of related elements. Alignment brings a gene into the same x-position as its orthologs, and flips the segment so all orthologs point in the same direction. Alignment in this manner allows a user to quickly discern differences in gene neighborhoods. Coupled with sorting, researchers can isolate similar and dissimilar neighborhoods, and create high-resolution comparative maps, where x and y space are meaningful to the investigation at hand.

5.2.4 Ortholog cluster targeting: New comparative genomics visualization algorithm

The creation of high-resolution genomic maps is made possible through a function we termed **'ortholog cluster targeting'**, which combines genome sorting, alignment and coloring to transform genomic data into a more meaningful form. In this operation, the user first selects a 'gene target' and this target is highlighted. Then adjacent genes on that contig are given a color on a gradient from yellow, closest to the target gene, to either green or blue, farthest from the target gene in the direction of transcription and in the direction opposite to transcription for up to 15 genes on either side. Color ramps were effectively used in a similar manner in Sequence

Surveyor, described in Chapter 4. The color applied serves as an indicator of distance from the target gene and orientation to the target gene.

Once a gene is given a color, potential orthologs in different genomes are given the same color. A sorting algorithm is then applied to pull to the top contigs with the target ortholog cluster. Finally, contigs are aligned against this target cluster and placed in the center of the screen. These actions together, reconfigure the data to make it easy for researchers to make rapid comparisons between gene neighborhoods, with gene conservation and variations presented in a perceptually scalable format. Gene neighborhoods with identical gene content will have an identical color gradient. Genes not present in the targeting neighborhood will remain the default color, signaling to the user the occurrence of an insertion or deletion event. The gradient is directional, and as a result inversion events, where segments of a genome migrate onto a different strand, will be quickly visible by a reversal of the color gradient. This novel visualization algorithm, creates a view makes high-density gene neighborhood analysis possible across hundreds of related bacterial genomes.

6. IMPLEMENTATION

This chapter describes the implementation of BactoGeNIE, an interactive visualization designed to enable the comparative analysis of bacterial gene neighborhoods. This application implements the design described in Chapter 5. In this chapter I will describe this program in detail, including the programming environment, data, algorithms, program design, visualization space, user interface, interactions and large-display environment.

6.1 Programming environment

BactoGeNIE was implemented in C++ using the Qt API for graphics and user interface elements. Qt is cross-platform API for desktop application development. I used the QGraphicsView framework, which supports display and interaction with large numbers of 2D graphical objects. Given the need for high-density display of many genes and contigs, this framework seemed suitable for implementing the design described in Section 5. Qt also includes an event propagation framework, which was used to connect orthologous genes together, enabling coordinated responses to user input.

6.2 Data and data pre-processing

BactoGeNIE requires input of at least two file types: genome feature files and fasta sequence files. These file types have been widely adopted by the genomics research community for the storage of genomic data, making this application accessible to researchers working on a variety of projects. The genome feature files store the boundaries of contigs, genes, coding sequences and other elements, along with annotation information including identifier, name, description and origin bacterial strain. Fasta files contain the identifiers of proteins and the complete protein amino-acid sequence. This data is needed for comparative analysis and the identification of orthologs.

The BactoGeNIE application will parse and display data from any gff and fasta file. However, BactoGeNIE was developed and tested on the PubMed draft bacterial datasets, and is optimized for parsing annotation data from PubMed files. To obtain sample datasets, thousands of draft *E.coli* genomes containing thousands of contigs were parsed into hundreds of gff and fasta data files, one for each strain.

6.3 Cd-hit ortholog clustering algorithm

For BactoGeNIE to enable comparative analysis of gene neighborhoods, a gene clustering algorithm was used to group the genes in uploaded gff and fasta files by sequence similarity. The cd-hit algorithm is widely used for comparing protein or nucleotide sequences because it is fast and able to handle very large datasets (89).

When the data is loaded into BactoGeNIE, the fasta filename is passed to the cd-hit executable, which runs on the dataset. The output file contains a list of cluster ids and gene ids. Each cluster id represents a group of highly similar protein sequences. This similarity is used to forge links between related elements.

6.4 BactoGeNIE program design

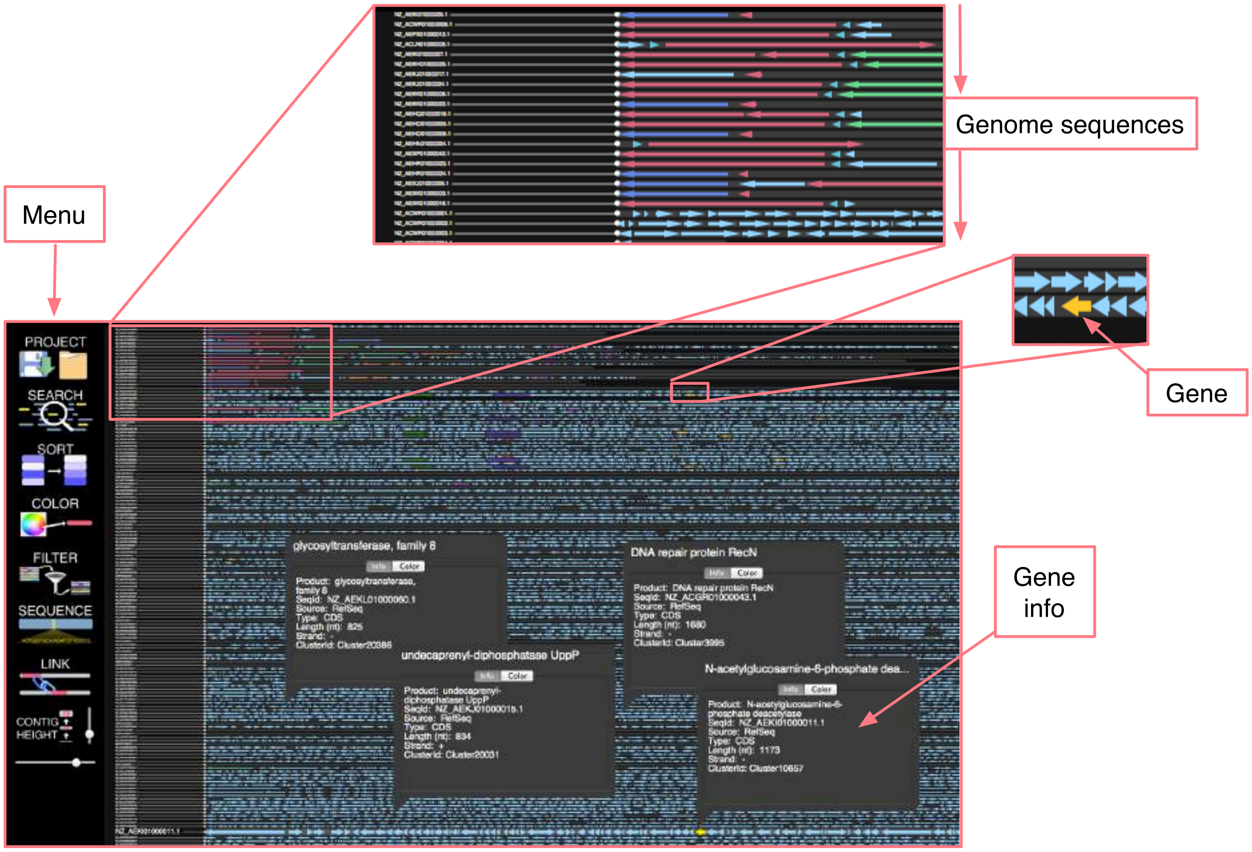
First, gff data is read and used to build the basic visual elements, genes and contigs, and populate the database. Then fasta data is loaded into the comparative analysis program, called cd-hit, described in the previous subsection. This produces lists of cluster ids each of which are associated with lists of gene ids with similar sequences. These cluster ids are then associated with the genome feature file data and used to connect genes through a set of signals and slots designed to enable coordinated responses to user inputs.

Only the first 1000 contigs are stored in memory, with the remaining contigs stored in a MySQL database. This enables the analysis of relatively large datasets, since all elements in memory and the database can be accessed and displayed. Multiple threads are used for data upload, and data entry into the SQL database.

Custom graphics items are created to display genes, contigs, species labels and gene pop-up menus. Contigs and contig labels are stacked vertically using a custom layout manager. Initially, contigs and labels are positioned in the original order in the uploaded dataset, with default contig heights and start x-positions. Genes are positioned within contigs using a custom layout manager that maps their x-position based on nucleotide distance from the start nucleotide. Genes arrows are drawn based on DNA strand and the direction of transcription.

Each gene was connected to other on-screen genes in the same cluster using signals and slots. Genes were connected to parent contigs through an additional set of signals and slots.

This signal and slot architecture formed the basis for interactive exploration of the dataset. Figure 25 depicts this design.



glycosyltransferase, family 8

Product: glycosyltransferase, family 8
Seqid: NZ_AEKL01000060.1
Source: RefSeq
Type: CDS
Length (nt): 825
Strand: -
Clusterid: Cluster20386

undecaprenyl-diphosphatase UppP

Product: undecaprenyl-diphosphatase UppP
Seqid: NZ_AEKJ01000016.1
Source: RefSeq
Type: CDS
Length (nt): 834
Strand: +
Clusterid: Cluster20031

DNA repair protein RecN

Product: DNA repair protein RecN
Seqid: NZ_ACGR01000043.1
Source: RefSeq
Type: CDS
Length (nt): 1680
Strand: -
Clusterid: Cluster3995

N-acetylglucosamine-6-phosphate dea...

Product: N-acetylglucosamine-6-phosphate deacetylase
Seqid: NZ_AEKI01000011.1
Source: RefSeq
Type: CDS
Length (nt): 1173
Strand: -
Clusterid: Cluster10657

Figure 25. Application design

6.5 Visualization Space

The majority of the visual space is dedicated to the data visualization. In the center of the screen are a series of contigs, which are drawn as long linear segments containing a series of blue arrows. Each blue arrow is a gene, whose position, size and strand are found in the coding sequence feature from the uploaded gff file. The direction of the arrow corresponds to the DNA strand on which the gene can be found. The length of the arrow corresponds to the size in nucleotides of the gene. The distance between arrows corresponds to the distance between the genes in nucleotides. On the left side, in small text, contig ids are provided for each contig on display. Other non-perceptually scalable visual elements, such as text to identify the genes on screen, are available on-demand. Interaction with the genes or through the UI is required to transform the scene and analyze the data.

Initially, all genes are given a default color. Typically, with 100-300 contigs onscreen, there could be anywhere from 100-1000 unique gene clusters on screen, so it would not be possible to give each cluster a unique color. Interactivity is needed to transform the scene so that researchers can make sense of this data.

6.6 User Interface

The screen space is divided into visualization space and a menu bar on the left. This menu bar provides several options for reconfiguring the scene to display data according to user interests.

There is a menu bar on the side of the display that provides global operations that can be performed on the entire dataset, such as modifying contig density or nucleotides per pixel or sorting contigs. In addition to the side bar menu, a menu is displayed to users on hovering over a gene. This menu fulfills the design goal of presenting non-scalable textual information 'on-demand'. The information provided includes an identifier and description, along with a variety of analysis options. These features will be described in more detail in section 6.7.

6.7 Interaction

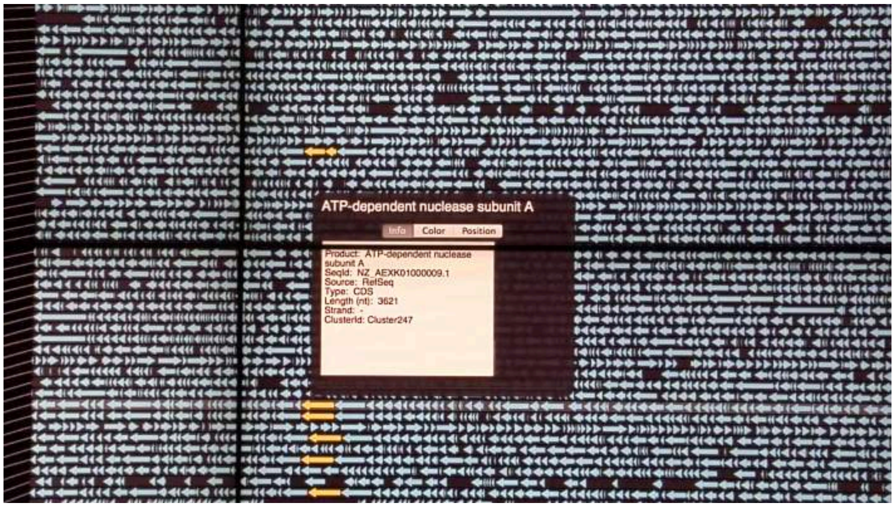
Interactions were designed to enable the analysis design goals described in Chapter 5. In the following subsection I will describe these interactions and will discuss their importance to the analytic tasks and scalability.

6.7.1 Navigation

Users are able to navigate through the scene by clicking and dragging in the direction they wish to move. Clicking and dragging up and down, moves the users through different subsets of contigs. Click and dragging right and left, shifts these contigs to show different subsets of genes. This is important because even with a scalable design not all genomes in a given dataset will be able to fit onscreen.

6.7.2 Coordinated Highlighting

Orthologous genes clustered together by the cd-hit algorithm, are connected together by a signal and slot system that enables coordinated highlighting. When a user hovers over a gene, all genes on-screen which are of the same cluster are highlighted. Additional information about the selected gene is shown in the pop-up menu, described in section 6.6. Coordinated highlighting is a method to provide scalable representations of gene orthology. This approach passes the pixel-density scalability criteria, because it is does not require additional space to encode orthology. Secondly, it fits display size scalability criteria, because highlighted genes can be spotted quickly and pre-attentively, and is visible at a distance, up-close and in peripheral vision. This is shown in figure 26.



Product: ATP-dependent nuclease subunit A
SeqId: NZ_AEXK01000009.1
Source: RefSeq
Type: CDS
Length (nt): 3621
Strand: -
ClusterId: Cluster247

Figure 26. Coordinated highlighting allows for scalable visual queries

Coordinated highlighting surveys as a means of performing a scalable visual query across the visible subset of the data. It helps in the identification of commonly recurring orthologs and sets of orthologs, which can then be used in future interactions.

6.7.3 Color assignment

Users are able to tag genes by giving the gene a color through its menu. Once a color is assigned to a gene, the same color is assigned to genes within its ortholog cluster. Color is used to connect similar genes together and serves as a scalable encoding that does not take up additional space and allows for a larger number of genomes to be displayed. By giving users the ability to assign colors to genes of interest, researchers are essentially able to define landmarks across the display, and search for relationships between tagged genes.

User applied color allows researchers to define neighborhoods of interest, and directly perceive similar neighborhoods in the data set. Since the colors are not automatically assigned, users can choose colors that are perceptually distinct to them, as well as apply color encodings to group similar sets of genes together, for instance genes with similar functions. This feature is shown in figure 27.

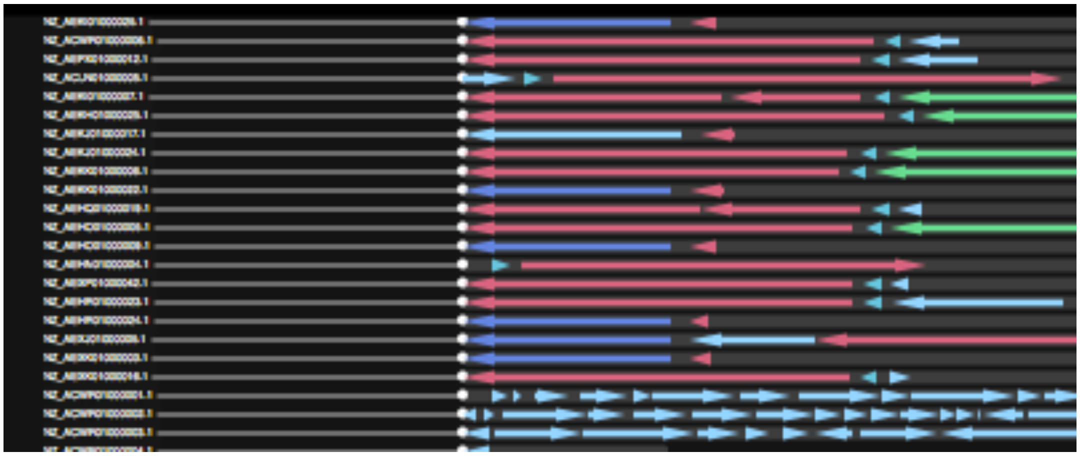


Figure 27. Close-up view of color assignment allows the user to design scalable queries that can be pre-attentively processed

6.7.4 Sorting

Initially, contigs are displayed in the order in which they are loaded, which typically clusters contigs by species. To implement the design goal of mapping contig placement in y to meaningful parameters, several sorting algorithms were implemented. The goal of this sorting is to bring into view the contigs containing genes that are relevant for the analysis. Particularly in large datasets, without this ability to cluster contigs by relatedness users would have to rely on memory to compare gene neighborhoods. Thus this feature is needed for analytic task scalability.

Contigs may be sorted by length, from longest to shortest and vice versa. This is a useful view because it tends to cluster similar stretches of sequence together from different species. As contigs are assembled, breaks occur due to difficulties in computational analysis. These breaks will tend to occur in similar sequences as well. As a result, contigs with similar content from different species tend to 'break' at similar points in assembly. So, sorting by contig size tends to sort by contig content. This view, then, provides a convenient way to begin surveying for patterns or groups of genes in similar sequences from distinct bacterial strains.

Contigs may also be sorted by gene content. There is a function to search and bring to the top of the scene contigs with a particular gene, or cluster of orthologous genes, to the top of the view. This allows researchers to effectively 'search' for genes of interest and remodel the scene to reflect the outcome of this search.

An additional goal of sorting is to use spatial positioning of contigs as a way to provide meaning. Spatial positioning is scalably perceptible, so sorting is an appropriate operation in a high-density genomic visualization. This approach could be extended in future work on this application, as described in the 'conclusions' section.

6.7.5 Alignment

Like sorting, the alignment feature is designed to enable users to use spatial positioning of contigs as a way to map information to the visualization space in a perceptually scalable manner. As described in Chapter 3, display-size scalability relies on spatial positioning techniques, such as clustering, to pull related information together allowing for comparisons and pattern recognition using eye movements, not working memory.

In this feature, users are able to select a gene, select the ‘alignment’ option and effectively reposition all contigs containing this gene cluster such that the clustered genes are positioning in the same place in x, and are flipped to show the same direction of transcription. All on-screen contigs containing this gene are brought into line, which, coupled with user applied color, allows researchers to rapidly spot variations in gene order.

Alignment enables researchers to compare genomes without the need for a ‘comparative track’, allowing more genomes to be compared simultaneously. Spatial positioning is a more scalable representation of orthology than line connectors.

Users can align against any on-screen gene. While most comparative gene neighborhood approach allowed for alignment between related genes on upload, the scene could not by dynamically re-aligned. This limits the ability to identify genes of interest through the visual interface and then reconfigure the scene to explore a new gene target.

6.7.6 Ortholog Targeting

The targeting operation couples the application of color with spatial positioning through sorting and alignment. This can be done on the fly, with an arbitrary gene target. This visualization is the first to implement an approach of this type. The algorithm proceeds as follows:

1. User hovers over a gene to bring up the menu

2. User selects the ‘target cluster’ option

3. Database is called to get a list of contigs containing genes in the same cluster as the target

4. Contigs present in memory are moved to the top, and drawn on top of the display

5. Remaining contigs are retrieved from the database, and populated with child genes

6. Added contigs and genes are connected to appropriate signals and slot mechanism

7. Added contigs are drawn at the top of the display

8. Target gene and neighbors are assigned color on a gradient based on proximity to the target

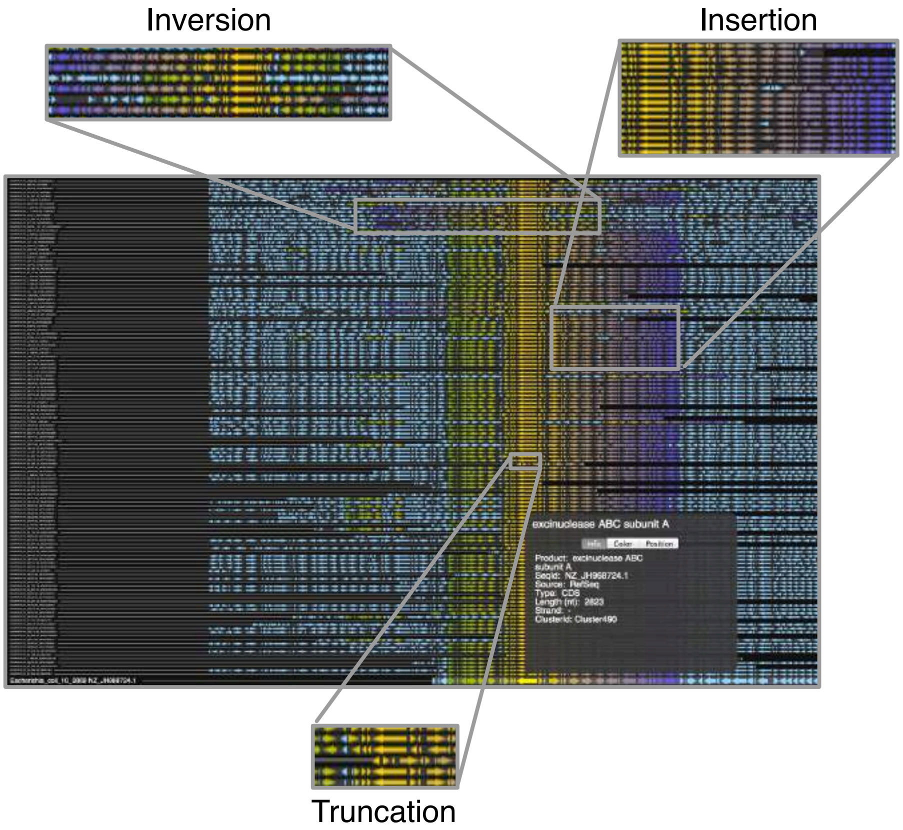
9. Orthologs to the target gene and the gene neighbors receive a signal to change color to match members of the cluster

10. Target cluster is shifted such that the gene is positioned at the center of the display space

11. Contigs containing orthologous genes are aligned to this centered gene

12. Contigs containing orthologous genes in the opposite orientation are flipped

The results of this algorithm are a seen in figure 28. This algorithm achieves several things. In one click, the user can cluster contigs by gene content, and position the target neighborhood in a single area on a large display. This means that comparisons can happen by moving the eye or the head, without needing to navigate virtually or view the data at a distance.



excinuclease ABC subunit A

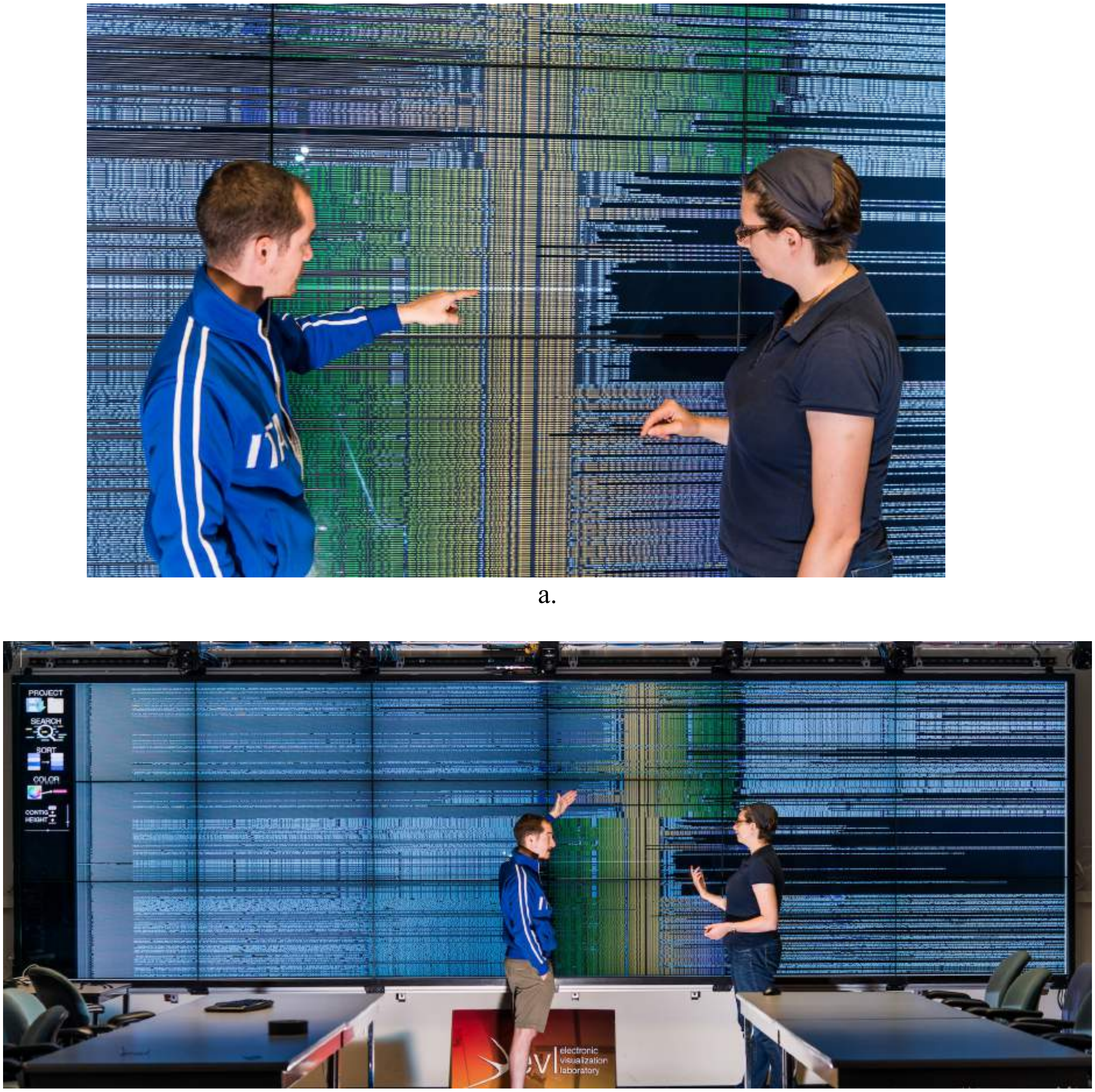
Product: excinuclease ABC subunit A
Seqid: NZ_JH968724.1
Source: RefSeq
Type: CDS
Length (nt): 2823
Strand: -
Clusterid: Cluster490

Figure 28. Gene targeting algorithm highlights similarities and differences across genomes

The application of color in an automatic gradient allows the user to see variations in gene neighborhoods around targets pre-attentively. The target gene and target contig will have a continuous application of the color gradient. Other contigs, may have genes that are not colored in the gradient. These genes are insertions. Other contigs may have genes that are colored in reverse direction, with green gradient going left as opposed to right. This indicates an inversion event, where gene order is no longer conserved and, therefore, are no longer coordinately transcribed. Variations in gene order, seen from breaks in the gradient, will also be quickly apparent. This algorithm allows the user to identify relevant variations for further investigation.

6.8 Large, High-Resolution Environment

BactoGeNIE runs both on traditional display environments and tiled-display walls driven by a single machine. Tiled display walls traditionally needed a cluster of computers to provide high resolution graphics across all displays. Today the compute power from a single machine is in many cases sufficient to drive a big display. Development of applications for this configuration is advantageous because it is easier for academic and research institutions to set up and maintain. Figures 29a and 29b show the results of the gene targeting algorithm in this environment.



a.

b.

Figure 29a, 29b. BactoGeNIE running on a large, high-resolution environment after running the gene-targeting algorithm. Image courtesy of the UIC Electronic Visualization Laboratory (Photo: Lance Long, UIC).

7. RESULTS

In this section, I will describe three measures of the success of this approach. First, I will talk about the ways in which this application expands upon features present in other genome visualization approaches, and includes features that help address the domain problem. Then, I will discuss the scalability of the approach compared to existing visualization approaches, with respect to resolution and display size. Finally, I will present user feedback on the usefulness of the approach for addressing the domain problem.

7.1 Features

Based on my analysis of the domain problem, I identified a set of visualization requirements, presented in Chapter 5. In the following tables, I will situate my visualization with the state of the art in comparative gene neighborhood visualization approaches.

Table IV shows components of pixel density scalability and my assessment of the tools described in Chapter 4. Notably, none of the gene-neighborhood applications fit the criteria I have defined for a pixel-density scalable application.

TABLE IV:
Pixel-Density Scalability and High-Density Representation

Visualization	Level of Detail?	Pixels per genome? Fixed or Dynamic?	High-Density Data Encoding?	High-Density Similarity Encoding?	High Density Identification?
SynView	Gene Neighborhood	150 Fixed	No	No (lines)	No (text by default)
ACT	Gene Neighborhood	160 Fixed	No	No (lines)	Somewhat (text compressed)
GBrowse_syn	Gene Neighborhood	100 Fixed	No	No (lines)	No (text by default)
Mauve	Gene Neighborhood	106 Fixed	No	No (lines)	Yes (on demand)
GeneRiViT	Gene Neighborhood Overview	Circular, Fixed	Yes	Yes	Yes (on demand)
PSAT	Gene Neighborhood	90 Fixed	No	No (color universally applied)	No (text by default)
Sequence Surveyor	Genome Overview	20 Fixed	Yes	Yes (color ramp and positioning)	Yes (on demand)
Orchestral	Copy Number Variation	20 Fixed	Yes	Yes (color intensity and alignment)	Not available
BactoGeNIE	Gene Neighborhood	4-10 Dynamic	Yes	Yes (user applied color and color ramps, alignment, coordinated highlighting)	Yes (on demand)

Table IV shows criteria for display size scalability and my assessment of the tools described in Chapter 4. The lack of dynamic clustering and overview at a distance, limits the display size scalability of gene-neighborhood comparative visualizations.

TABLE V:
Display-Size Scalability and High-Density Representation

Visualization	Level of Detail?	Dynamically cluster related genes	Details up close	Overview from a distance?
SynView	Gene Neighborhood	No	Yes	No
ACT	Gene Neighborhood	No	Yes	No
GBrowse_syn	Gene Neighborhood	No	Yes	No
Mauve	Gene Neighborhood	No	Yes	No
GeneRiViT	Gene Neighborhood Overview	Yes	Somewhat	Somewhat
PSAT	Gene Neighborhood	No	Yes	No
Sequence Surveyor	Genome Overview	No	No	Yes
Sequence Juxtaposer	Gene sequence		Yes	Yes
Orchestral	Copy Number Variation	No	Yes	Yes
BactoGeNIE	Gene Neighborhood	Yes	Yes	Yes

Table VI shows criteria for analytic task scalability and my assessment of the tools described in Chapter 4. Most applications designed to address the domain task fail to encode relationships in a way that would make comparative information accessible pre-attentively. In addition, line connection approaches encounter problems with visual clutter. Visual queries are limited for these applications as well, since users are unable to dynamically target genes of interest.

TABLE VI:
Display-Size Scalability and High-Density Representation

Visualization	Level of Detail?	Similarity pre-attentive or visual search	Avoid visual clutter	Visual Query for arbitrary neighborhoods
SynView	Gene Neighborhood	Visual search (lines)	No	No
ACT	Gene Neighborhood	Visual search (lines)	No	No
GBrowse_syn	Gene Neighborhood	Visual search (lines)	No	No
Mauve	Gene Neighborhood	Visual search (lines)	No	No
GeneRiViT	Gene Neighborhood Overview	Pre-attentive (alignment, overview colors)	Yes	Yes
PSAT	Gene Neighborhood	Pre-attentive (color, alignment)	No	No
Sequence Surveyor	Genome Overview	Pre-attentive (color)	Yes	No
Sequence Juxtaposer	Gene sequence	Pre-attentive (color, alignment)	Yes	No
Orchestral	Copy Number Variation	Pre-attentive (color, alignment)	Yes	No
BactoGeNIE	Gene Neighborhood	Pre-attentive (color, alignment)	Yes	Yes

As you can see from the above chart, BactoGeNIE meets the task requirements better than other genome visualization tools described in Chapter 4.

7.2 Resolution and number of genomes that can be compared

One of the most important analysis tasks required for this visualization is comparison across large numbers of genomes. This is both because lots of genomic sequences are being generated, demanding new genome visualization approaches, and because comparative gene neighborhood analysis requires comparisons across many genomes simultaneously.

I performed an analysis of the scalability of BactoGeNIE relative to other approaches, shown in figure 30. Purely based on the pixels per genome, I mapped out the number of genomes that could be fit on screen and found the following relationship. None of these approaches are designed to accommodate the scales of data displayed in the chart, but this is a basic assessment of the pixel-density scalability of the approach.

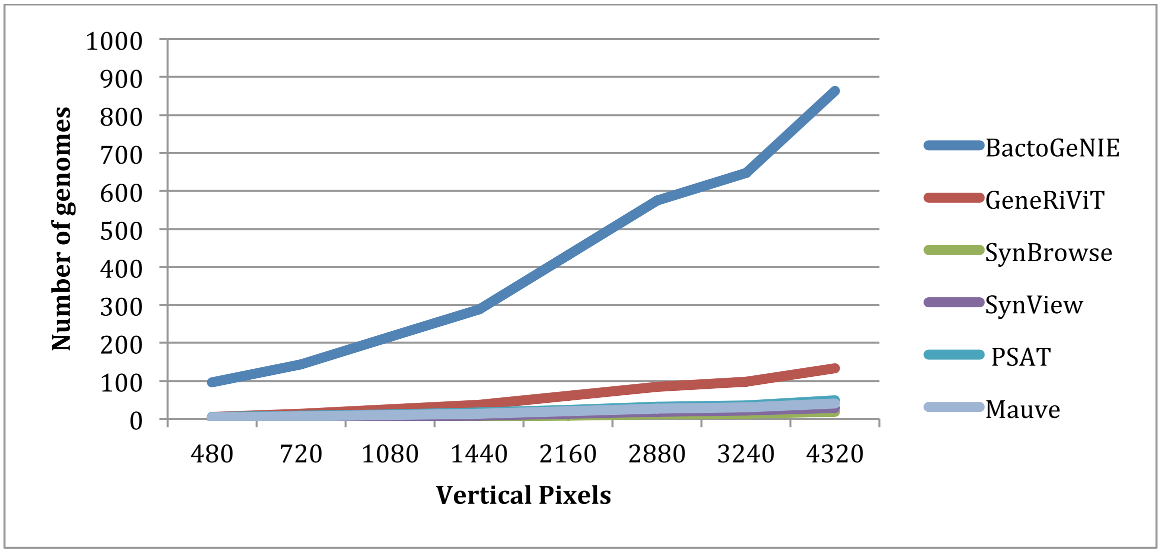


Figure 30. Resolution vs number of genomes displayed. BactoGeNIE is capable of displaying far more than its competitors.

Based on this chart, it is clear that BactoGeNIE is a more high-density representation of genomic content. It will scale more effectively as resolutions increase. Further, other approaches adopt encodings of identity and orthology that are not perceptually scalable. In the following chart I summarized the perceptually scalability of the orthology representations.

7.3 User feedback

BactoGeNIE was created through close contact with bacterial genomics researchers. To capture the effectiveness of this approach, I solicited feedback from a central team member. While these are preliminary results, it illustrates the ways in which my approach is useful to a genomics researcher. Based on other feedback from researchers who have seen this approach, I believe that this basic approach has potential utility in a wide range of fields.

User survey results are as follows, with the responses in italics.

1) Usefulness of the visualization

This tool has been widely used by members of my team to show the comparative analyses of genomic context for several bacterial genomes

a) What sorts of questions does my visualization help address in bacterial genomics research?

One of the primary questions this tool helps to understand is what kinds of proteins cluster together in bacterial genomes. By comparing hundreds and thousands of *bacterial genomes we can find the pathogenicity islands that i.e. proteins that are involved in pathogenicity.*

b) Does it help in the identification of commonly recurring sets of genes? How does it help?

The current genome browsers that are available help at looking at single genome and sometimes 10 genomes, but with comparing several genomes we can look at genomic context associated with our protein of interest. Questions of what kind of proteins are co-localized with our protein of interest? Are these proteins evolving together? Inversions, deletions, translocation in genomic regions can help us answer biological question such as for eg. what makes a bacteria a pathogen vs non-pathogen.

c) Does it contribute to research on unidentified genes? How does it help?

Yes, co-evolving proteins with no known annotation help us to discover the space of unknowns. Based on genomic context, we can hypothesize biological processes these unknown proteins may be involved.

d) Does it help identify potential functional relationships between proteins? How does it help?

Linkage of proteins to known protein helps us to understand the functional relationship of these proteins. The enhancers, cofactors, regulators and other functional relationship can be hypothesized for the neighboring proteins.

2) Comparison with other approaches / Novelty of the approach

Genome browsers such as Jbrowse enable researchers to do comparative genome analyses for nearly 10-50 genomes. But fail to work when we are studying several hundreds of genomes of interest. This tool is really unique and it's the only tool that I am aware of that can scale up to any number of genome comparisons. The ability to load multiple tracks of genomes, and the zoom in and out options with color coding, annotation tracks makes it very convenient for scientists to quickly look at patterns. This tool has a potential to serve both for visualization as well as data mining needs.

a) How difficult is it to answer these questions with existing approaches?

As explained in the previous question, no tool exits that can help in doing analyses for more than 100 genomes.

b) Does the visualization complement automated approaches- data mining, BLAST, etc? How does it complement these approaches?

Bioinformatics and data mining analyses by genome comparisons are being done by researchers for answering biological questions. But for communication needs a viz tool such as this tool really complements to quickly get the message across to different audience.

c) How does this visualization compare to other genomic data visualizations? How novel is it?

Several aspect of this tool makes it novel such as

· Scalability to several genome comparative analyses

· User's ability to decide the protein clustering. The CD-hit capability helps researchers to cluster proteins with different levels of identity to find protein families

· Ability to download sequence of interest for researchers to do further downstream analyses

· Flexibility of the tool for users to make their personal decision on color choice

3) Any additional comments?

This tool has gained lot of attention from several groups in my organization. Several groups have requested access to use this tool. This speaks on how researchers see the value of this tool.

7.4 Summary

These preliminary results indicate that my approach has features that extend the functionality of the state-of-the-art competitors, and is far more scalable, both in potential data volume that can be displayed and analyzed directly, and in display-size and pixel-density. User feedback supports these claims and indicates the utility of this approach for critical research applications.

Future work on this application will concentrate on collecting more robust feedback and measures of effect and scalability in this approach.

8. CONCLUSION

The contemporary phenomena referred to as ‘Big Data’ is a product of advances in automated processes. Our understanding of Big Data will be dependent on computational solutions that can present such large volumes of complex information to human experts in a scalable way.

Automated methods are a necessary part of any Big Data analysis problem, but the output of these methods are only the starting point in the generation of new knowledge. To move from automation to knowledge, humans need to be brought into the loop. Automated methods can effectively act as a ‘black-box’ to researchers, unless they are given the opportunity to verify the output of these methods. In addition, human pattern recognition is extremely powerful, with one third of the human brain devoted to the processing of visual information. Leveraging the human visual system can bring expertise to bear on a problem, allowing disparate information and experience to influence identified results. However, producing visualizations of large data sets presents many challenges. In domains where existing visual approaches were designed for small data volumes, these approaches may fail to scale-up in data volumes.

Big displays present an opportunity to visualize big data in new ways. There are a variety of documented cognitive benefits to presenting data on big displays. The capability to utilize embodied cognition and spatial memory has been shown to benefit the analysis of complex datasets. Provided appropriate encodings, the human visual system has been shown to ‘scale-up’ to process larger volumes of information over larger display spaces. What remains to be documented are specific ways in which domain visualizations must adapt to these new environments to realize these benefits and avoid potential pitfalls.

Comparative genomics is one such domain. Recent genome sequencing advances have enabled the generation of unprecedented volumes of complete genome sequences. Existing visual approaches largely fail to scale to accommodate these new volumes of sequence data, particularly for tasks involving comparisons between genomes. In addition, existing visual approaches are designed for moderate resolution, traditional displays, and fail to consider the implications of 'scaling-up' the visual design for the environments of the future.

In this thesis I explored scalable comparative genome visualization design through the lens of a particular and pressing subfield of comparative genomics: comparative bacterial gene neighborhood analysis. Thousands of complete bacterial genome sequences are presently filling public databases and, providing researchers with a new tool in the analysis of interesting genes and biological processes. However, visualizations that address this specific domain problem fail to scale beyond small numbers of genomes and small numbers of genes.

8.1 My contributions

My approach, 'BactoGeNIE', presents a novel design that shows this comparative data in a significantly more scalable format. In addition to accommodating the presentation of hundreds of genomes in a single view, it also employs encodings that avoids visual clutter and permits the pre-attentive identification of variations in gene neighborhood composition. This approach leverages large displays, by using clustering in the form of dynamic sorting and alignment, allowing researchers to view relevant neighborhoods simultaneously. In addition, a novel ‘gene targeting’ algorithm represents neighborhoods of ortholog clusters using spatial positioning and color to provide simultaneous ‘details-up-close’ and ‘overview-at-a-distance’ views to researchers. This is the first visual design to accomplish this level of scalability for the analytic tasks in comparative gene neighborhood analysis and its design provides several lessons for future scalable comparative genomic designs. Preliminary results support the effectiveness of this approach, with feedback indicating that this tool is highly desired by members of the community and presents data more effectively than existing approaches.

8.2 Future work

There are several possible directions for future research. First, there are several ways to extend the existing framework to accommodate a larger variety of analysis problems. The visualization design might benefit from increased usage of color ramps, such as the ability to simultaneously apply multiple distinct color ramps. This would allow for the simultaneous analysis of two gene neighborhoods, which may be significant for several analysis problems. In addition, applying additional sorting criteria, such as relatedness of bacterial strains or similarity of gene neighborhoods could be beneficial in grouping together related strains. This approach could also be generalized to show whole-genome comparisons or to address comparative analysis problems for other species, which would entail different analysis tasks.

Secondly, there are several ways to improve the visualization architecture. First, an implementation that took advantage of parallelism could be beneficial for more responsive interactions. An approach that uses GLSL shaders would likely be faster and allow for more rapid hypothesis testing.

Thirdly, this approach could be ported to multi-user, multi-window systems for tiled-display walls, such as the Scalable Adaptive Graphics Environment (SAGE) and accept input from a wider array of devices (90). This is important for several reasons. The mouse is not an ideal input device for all situations with this approach on tiled-display walls. Touch input provides the ability for users to interact ‘up-close’. Secondly, porting this application to SAGE would allow for this information to be juxtaposed with distinct visualizations that provide contextual information for analyzing these datasets.

Finally, additional use cases are needed to demonstrate the utility of this approach for current genomic data analysis problems. Additional use cases, along with more rigorous measures of this approach as opposed to others would help provide evidence for the success of this tool.

CITED LITERATURE

1. Finocchiaro, Maurice A., ed. *The Galileo affair: a documentary history*. Vol. 1. Univ of California Press, 1989.

2. Sulloway, Frank J. "Darwin and the Galapagos." *Biological Journal of the Linnean Society* 21, no. 1‐2 (1984): 29-59.

3. Briggs, David. *Plant variation and evolution*. Cambridge University Press, 1997.

4. Klug, A. "Rosalind Franklin and the Discovery of the Structure of DNA." *Nature* 219 (1968).

5. Lynch, Clifford. "Big data: How do your data grow?." *Nature* 455, no. 7209 (2008): 28-29.

6. Howe, Doug, Maria Costanzo, Petra Fey, Takashi Gojobori, Linda Hannick, Winston Hide, David P. Hill et al. "Big data: The future of biocuration." *Nature*455, no. 7209 (2008): 47-50.

7. Keim, Daniel, Huamin Qu, and Kwan-Liu Ma. "Big-Data Visualization."*Computer Graphics and Applications, IEEE* 33, no. 4 (2013): 20-21.

8. Findlay, John M., and Iain D. Gilchrist. *Active vision: The psychology of looking and seeing*. Oxford University Press, 2003.

9. Ware, Colin. *Information visualization: perception for design*. Elsevier, 2012.

10. Lander, Eric S., Lauren M. Linton, Bruce Birren, Chad Nusbaum, Michael C. Zody, Jennifer Baldwin, Keri Devon et al. "Initial sequencing and analysis of the human genome." *Nature* 409, no. 6822 (2001): 860-921.

11. Venter, J. Craig, Mark D. Adams, Eugene W. Myers, Peter W. Li, Richard J. Mural, Granger G. Sutton, Hamilton O. Smith et al. "The sequence of the human genome." *Science* 291, no. 5507 (2001): 1304-1351.

12. Lander, Eric S. "Initial impact of the sequencing of the human genome." *Nature* 470, no. 7333 (2011): 187-197.

13. Wetterstrand K.A. “DNA Sequencing Costs: Data from the NHGRI Large-Scale Genome Sequencing Program.” Accessed May 7, 2014. http://www.genome.gov/sequencingcosts.

14. Via García, Marc, and 1000 Genomes Project Consortium. "An integrated map of genetic variation from 1,092 human genomes." *Nature, 2012, vol. 491, p. 56-65* (2012).

15. Davidson, S., “Sustainable bioenergy: Genomics and biofuels development,” *Nature Education*, 1(1), 2008.

16. Rubin, Edward M. "Genomics of cellulosic biofuels." *Nature* 454, no. 7206 (2008): 841-845.

17. Hazell, Stuart L. "Genomics and drug discovery." *Microbiology Australia* 23, no. 5: 17-18.

18. Steward, G.F. and Rappé, M.S., “What’s the ‘meta’ with metagenomics?,” The ISME Journal, 1, 2007, pp. 100–102.

19. Slater, Gary W., Claude Desruisseaux, Sylvain J. Hubert, Jean‐François Mercier, Josée Labrie, Justin Boileau, Frédéric Tessier, and Marc P. Pépin. "Theory of DNA electrophoresis: a look at some current challenges."*Electrophoresis* 21, no. 18 (2000): 3873-3887.

20. Kent, W. James, Charles W. Sugnet, Terrence S. Furey, Krishna M. Roskin, Tom H. Pringle, Alan M. Zahler, and David Haussler. "The human genome browser at UCSC." *Genome research* 12, no. 6 (2002): 996-1006.

21. Leigh, Jason, Andrew Johnson, Luc Renambot, Tom Peterka, Byungil Jeong, Daniel J. Sandin, Jonas Talandis et al. "Scalable resolution display walls."*Proceedings of the IEEE* 101, no. 1 (2013): 115-129.

22. Andrews, Christopher, Alex Endert, and Chris North. "Space to think: large high-resolution displays for sensemaking." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 55-64. ACM, 2010.

23. Moreland, K. "Redirecting research in large-format displays for visualization." In *Large Data Analysis and Visualization (LDAV), 2012 IEEE Symposium on*, pp. 91-95. IEEE, 2012.

24. Rogozin, I. B. (2004). Computational approaches for the analysis of gene neighbourhoods in prokaryotic genomes. *Briefings in Bioinformatics*, *5*(2), 131–149. doi:10.1093/bib/5.2.131

25. Peterson, Scott N., and Claire M. Fraser. "The complexity of simplicity."*Genome biology* 2, no. 2 (2001): comment2002.

26. Pruitt, Kim D., Tatiana Tatusova, Garth R. Brown, and Donna R. Maglott. "NCBI Reference Sequences (RefSeq): current status, new features and genome annotation policy." *Nucleic acids research* 40, no. D1 (2012): D130-D135.

27. Field, D., Wilson, G., & Van Der Gast, C. (2006). How do we compare hundreds of bacterial genomes? *Current Opinion in Microbiology*.

28. Nielsen, Cydney B., Michael Cantor, Inna Dubchak, David Gordon, and Ting Wang. "Visualizing genomes: techniques and challenges." *Nature methods* 7 (2010): S5-S15.

29. Dean, Donald H. "Biochemical genetics of the bacterial insect-control agent Bacillus thuringiensis: basic principles and prospects for genetic engineering."*Biotechnology and genetic engineering reviews* 2, no. 1 (1984): 341-363.

30. Huynen, Martijn, Berend Snel, Warren Lathe III, and Peer Bork. "Exploitation of gene context." *Current opinion in structural biology* 10, no. 3 (2000): 366-370.

31. Koonin, Eugene V. "Orthologs, paralogs, and evolutionary genomics 1." *Annu. Rev. Genet.* 39 (2005): 309-338.

32. Altschul, Stephen F., Thomas L. Madden, Alejandro A. Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J. Lipman. "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs." *Nucleic acids research* 25, no. 17 (1997): 3389-3402.

33. Cai, C. Z., L. Y. Han, Zhi Liang Ji, X. Chen, and Yu Zong Chen. "SVM-Prot: web-based support vector machine software for functional classification of a protein from its primary sequence." *Nucleic acids research* 31, no. 13 (2003): 3692-3697.

34. Dandekar, Thomas, Berend Snel, Martijn Huynen, and Peer Bork. "Conservation of gene order: a fingerprint of proteins that physically interact."*Trends in biochemical sciences* 23, no. 9 (1998): 324-328.

35. Overbeek, Ross, Michael Fonstein, Mark D'souza, Gordon D. Pusch, and Natalia Maltsev. "The use of gene clusters to infer functional coupling."*Proceedings of the National Academy of Sciences* 96, no. 6 (1999): 2896-2901

36. Ralston, A. Operons and prokaryotic gene regulation. *Nature Education.* 1, no. (2008): 216.

37. Arnold, Dawn L., and Robert W. Jackson. "Bacterial genomes: evolution of pathogenicity." *Current opinion in plant biology* 14, no. 4 (2011): 385-391.

38. Lawrence, Jeffrey G., and John R. Roth. "Selfish operons: horizontal transfer may drive the evolution of gene clusters." *Genetics* 143, no. 4 (1996): 1843-1860.

39. Binnewies, Tim T., Yair Motro, Peter F. Hallin, Ole Lund, David Dunn, Tom La, David J. Hampson, Matthew Bellgard, Trudy M. Wassenaar, and David W. Ussery. "Ten years of bacterial genome sequencing: comparative-genomics-based discoveries." *Functional & integrative genomics* 6, no. 3 (2006): 165-185.

40. Metzker, Michael L. "Emerging technologies in DNA sequencing." *Genome research* 15, no. 12 (2005): 1767-1776.

41. Wheeler, David A., Maithreyan Srinivasan, Michael Egholm, Yufeng Shen, Lei Chen, Amy McGuire, Wen He et al. "The complete genome of an individual by massively parallel DNA sequencing." *nature* 452, no. 7189 (2008): 872-876.

42. Hernandez, David, Patrice François, Laurent Farinelli, Magne Østerås, and Jacques Schrenzel. "De novo bacterial genome sequencing: millions of very short reads assembled on a desktop computer." *Genome research* 18, no. 5 (2008): 802-809.

43. Wikipedia contributors, "Sanger sequencing," *Wikipedia, The Free Encyclopedia,* http://en.wikipedia.org/w/index.php?title=Sanger_sequencing&oldid=603880078 (accessed May 13, 2014).

44. Baker, Monya. "De novo genome assembly: what every biologist should know." *Nature methods* 9, no. 4 (2012): 333.

45. Meyer, Folker, Alexander Goesmann, Alice C. McHardy, Daniela Bartels, Thomas Bekel, Jörn Clausen, Jörn Kalinowski et al. "GenDB—an open source genome annotation system for prokaryote genomes." *Nucleic Acids Research*31, no. 8 (2003): 2187-2195.

46. Li, Christian J. Stoeckert, and David S. Roos. "OrthoMCL: identification of ortholog groups for eukaryotic genomes." *Genome research* 13, no. 9 (2003): 2178-2189.

47. Salgado, Heladia, Gabriel Moreno-Hagelsieb, Temple F. Smith, and Julio Collado-Vides. "Operons in Escherichia coli: genomic analyses and predictions." *Proceedings of the National Academy of Sciences* 97, no. 12 (2000): 6652-6657.

48. Dobrindt, Ulrich, Gabriele Blum-Oehler, Gabor Nagy, György Schneider, André Johann, Gerhard Gottschalk, and Jörg Hacker. "Genetic structure and distribution of four pathogenicity islands (PAI I536 to PAI IV536) of uropathogenic Escherichia coli strain 536." *Infection and immunity* 70, no. 11 (2002): 6365-6372.

49. Reda, Khairi, Alessandro Febretti, Aaron Knoll, Jillian Aurisano, Jason Leigh, Andrew Johnson, Michael Papka, and Mark Hereld. "Visualizing Large, Heterogeneous Data in Hybrid Reality Display Environments." (2013): 1-1.

50. Hegarty, Mary. "The Cognitive Science of Visual‐Spatial Displays: Implications for Design." *Topics in cognitive science* 3, no. 3 (2011): 446-474.

51. Scaife, Mike, and Yvonne Rogers. "External cognition: how do graphical representations work?." *International journal of human-computer studies* 45, no. 2 (1996): 185-213.

52. Card, Stuart K., Jock D. Mackinlay, and Ben Shneiderman, eds. *Readings in information visualization: using vision to think*. Morgan Kaufmann, 1999.

53. Yost, Beth, Yonca Haciahmetoglu, and Chris North. "Beyond visual acuity: the perceptual scalability of information visualizations for large displays." In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 101-110. ACM, 2007.

54. Healey, Christopher G. "Perception in visualization." *Retrieved February* 10 (2007): 2008.

55. Ware, Colin, William Wright, and Nicholas J. Pioch. "Visual Thinking Design Patterns." In *Proceedings of the 19th International Conference Distributed Multimedia Systems (DMS '13)*, 2013.

56. Alvarez, George A., Talia Konkle, and Aude Oliva. "Searching in dynamic displays: Effects of configural predictability and spatiotemporal continuity."*Journal of vision* 7, no. 14 (2007): 12.

57. Rosenholtz, Ruth, Yuanzhen Li, and Lisa Nakano. "Measuring visual clutter."*Journal of Vision* 7, no. 2 (2007): 17.

58. Andrews, Christopher, Alex Endert, Beth Yost, and Chris North. "Information visualization on large, high-resolution displays: Issues, challenges, and opportunities." *Information Visualization* 10, no. 4 (2011): 341-355.

59. Ball, Robert, and Chris North. "Effects of tiled high-resolution display on basic visualization and navigation tasks." In *CHI'05 extended abstracts on Human factors in computing systems*, pp. 1196-1199. ACM, 2005

60. Ball, Robert, Chris North, and Doug A. Bowman. "Move to improve: promoting physical navigation to increase user performance with large displays." In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 191-200. ACM, 2007.

61. Baudisch, Patrick, Nathaniel Good, Victoria Bellotti, and Pamela Schraedley. "Keeping things in context: a comparative evaluation of focus plus context screens, overviews, and zooming." In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 259-266. ACM, 2002.

62. Lamesch, Philippe, Tanya Z. Berardini, Donghui Li, David Swarbreck, Christopher Wilks, Rajkumar Sasidharan, Robert Muller et al. "The Arabidopsis Information Resource (TAIR): improved gene annotation and new tools."*Nucleic acids research* 40, no. D1 (2012): D1202-D1210.

63. Blake, Judith A., Carol J. Bult, Janan T. Eppig, James A. Kadin, and Joel E. Richardson. "The Mouse Genome Database: integration of and access to knowledge about the laboratory mouse." *Nucleic acids research* 42, no. D1 (2014): D810-D817.

64. Pierre, Susan E. St, Laura Ponting, Raymund Stefancsik, and Peter McQuilton. "FlyBase 102—

advanced approaches to interrogating FlyBase." *Nucleic acids research* 42, no. D1 (2014): D780-D788.

65. Yook, Karen, Todd W. Harris, Tamberlyn Bieri, Abigail Cabunoc, Juancarlos Chan, Wen J. Chen, Paul Davis et al. "WormBase 2012: more genomes, more data, new website." *Nucleic acids research* 40, no. D1 (2012): D735-D741.

66. Hubbard, T., Daniel Barker, Ewan Birney, Graham Cameron, Yuan Chen, L. Clark, Tony Cox et al. "The Ensembl genome database project." *Nucleic acids research* 30, no. 1 (2002): 38-41.

67. Pruitt, Kim D., Tatiana Tatusova, and Donna R. Maglott. "NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins." *Nucleic acids research* 33, no. suppl 1 (2005): D501-D504.

68. Goodstein, David M., Shengqiang Shu, Russell Howson, Rochak Neupane, Richard D. Hayes, Joni Fazo, Therese Mitros et al. "Phytozome: a comparative platform for green plant genomics." *Nucleic acids research* 40, no. D1 (2012): D1178-D1186.

69. Ware, Doreen, Pankaj Jaiswal, Junjian Ni, Xiaokang Pan, Kuan Chang, Kenneth Clark, Leonid Teytelman et al. "Gramene: a resource for comparative grass genomics." *Nucleic Acids Research* 30, no. 1 (2002): 103-105.

70. Donlin, Maureen J. "Using the generic genome browser (GBrowse)." *Current Protocols in Bioinformatics* (2009): 9-9.

71. Skinner, Mitchell E., Andrew V. Uzilov, Lincoln D. Stein, Christopher J. Mungall, and Ian H. Holmes. "JBrowse: a next-generation genome browser." *Genome research* 19, no. 9 (2009): 1630-1638.

72. McKay, Sheldon J., Ismael A. Vergara, and Jason E. Stajich. "Using the generic synteny browser (GBrowse_syn)." *Current protocols in bioinformatics* (2010): 9-12.

73. Wang, Haiming, Yanqi Su, Aaron J. Mackey, Eileen T. Kraemer, and Jessica C. Kissinger. "SynView: a GBrowse-compatible approach to visualizing comparative genome data." *Bioinformatics* 22, no. 18 (2006): 2308-2309.

74. Pan, Xiaokang, Lincoln Stein, and Volker Brendel. "SynBrowse: a synteny browser for comparative sequence analysis." *Bioinformatics* 21, no. 17 (2005): 3461-3468.

75. Meyer, Miriah, Tamara Munzner, and Hanspeter Pfister. "MizBee: a multiscale synteny browser." *Visualization and Computer Graphics, IEEE Transactions on*15, no. 6 (2009): 897-904.

76. Engels, Reinhard, Tamara Yu, Chris Burge, Jill P. Mesirov, David DeCaprio, and James E. Galagan. "Combo: a whole genome comparative browser." *Bioinformatics* 22, no. 14 (2006): 1782-1783.

77. Uchiyama, Ikuo, Toshio Higuchi, and Ichizo Kobayashi. "CGAT: a comparative genome analysis tool for visualizing alignments in the analysis of complex evolutionary changes between closely related genomes." *BMC bioinformatics* 7, no. 1 (2006): 472.

78. Carver, Tim J., Kim M. Rutherford, Matthew Berriman, Marie-Adele Rajandream, Barclay G. Barrell, and Julian Parkhill. "ACT: the Artemis comparison tool." *Bioinformatics* 21, no. 16 (2005): 3422-3423.

79. Darling, Aaron CE, Bob Mau, Frederick R. Blattner, and Nicole T. Perna. "Mauve: multiple alignment of conserved genomic sequence with rearrangements." *Genome research* 14, no. 7 (2004): 1394-1403.

80. Price, Adam, Robert Kosara, and Cynthia Gibas. "Gene-RiViT: A visualization tool for comparative analysis of gene neighborhoods in prokaryotes." In *Biological Data Visualization (BioVis), 2012 IEEE Symposium on*, pp. 57-62. IEEE, 2012.

81. Fong, Christine, Laurence Rohmer, Matthew Radey, Michael Wasnick, and Mitchell J. Brittnacher. "PSAT: a web tool to compare genomic neighborhoods of multiple prokaryotic genomes." *BMC bioinformatics* 9, no. 1 (2008): 170.

82. Krzywinski, Martin, Jacqueline Schein, İnanç Birol, Joseph Connors, Randy Gascoyne, Doug Horsman, Steven J. Jones, and Marco A. Marra. "Circos: an information aesthetic for comparative genomics." *Genome research* 19, no. 9 (2009): 1639-1645.

83. Albers, Danielle, Colin Dewey, and Michael Gleicher. "Sequence surveyor: Leveraging overview for scalable genomic alignment visualization." *Visualization and Computer Graphics, IEEE Transactions on* 17, no. 12 (2011): 2392-2401.

84. Harrower, Mark, and Cynthia A. Brewer. "Colorbrewer. org: An online tool for selecting colour schemes for maps." *The Cartographic Journal* 40, no. 1 (2003): 27-37.

85. Slack, James, Kristian Hildebrand, Tamara Munzner, and Katherine St John. "SequenceJuxtaposer: Fluid Navigation For Large-Scale Sequence Comparison in Context." In *German Conference on Bioinformatics*, vol. 53. 2004.

86. Ruddle, Roy A., Waleed Fateen, Darren Treanor, Peter Sondergeld, and Phil Ouirke. "Leveraging wall-sized high-resolution displays for comparative genomics analyses of copy number variation." In *Biological Data Visualization (BioVis), 2013 IEEE Symposium on*, pp. 89-96. IEEE, 2013.

87. Shneiderman, Ben. "The eyes have it: A task by data type taxonomy for information visualizations." In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pp. 336-343. IEEE, 1996.

88. Chuah, Mei C., and Steven F. Roth. "On the semantics of interactive visualizations." In *Information Visualization'96, Proceedings IEEE Symposium on*, pp. 29-36. IEEE, 1996.

89. Huang, Ying, Beifang Niu, Ying Gao, Limin Fu, and Weizhong Li. "CD-HIT Suite: a web server for clustering and comparing biological sequences." *Bioinformatics* 26, no. 5 (2010): 680-682.

90. Jeong, Byungil, Jason Leigh, Andrew E. Johnson, Luc Renambot, Maxine D. Brown, Ratko Jagodic, Sungwon Nam, and Hyejung Hur. "Ultrascale collaborative visualization using a display-rich global cyberinfrastructure." *IEEE computer graphics and applications* 30, no. 3 (2010): 71-83.

VITA

NAME: Jillian Marie Rochat Aurisano

EDUCATION: B.A., Biological Sciences, Religious Studies, University of Chicago, Chicago, Illinios, 2006

TEACHING: Biological Sciences Collegiate Division, University of Chicago: CORE Biology Writing Program Teaching Assistant, 2005-2006

HONORS: Grace Hopper Scholarship award, 2013
Anita Borg Memorial Scholarship recipient, 2012
Bachelors degree conferred with honors, 2006
Dean’s List 2002-2006
Howard Hughes Medical Institute Summer Research Fellowship, 2004
First-Year Undergraduate Summer Research Fellowship, 2003

PUBLICATIONS: K. Reda, J. Aurisano, A. Febretti, A. Johnson, J. Leigh. “Visualization Design Patterns for Ultra-Resolution Display Environments.” Submitted Workshop on Visualization Infrastructure & Systems Technology at SuperComputing13 in September 2013.

K. Reda, A. Febretti, A. Knoll, J. Aurisano, J. Leigh, A. Johnson, M.E. Papka and M. Hereld. “Visualizing Large, Heterogenous Data in Hybrid Reality Display Environments.” Computer Graphics and Applications, IEEE Computer Society, July 2013.

M. Schumer, R. Birger, C. Tantipathananandh, J. Aurisano, M. Maggioni, P. Mwangi. “Infestation by a Common Parasite is correlated with Ant Symbiont Identity in Plant-Ant Mutualism.” To appear in Biotropica.

J. Aurisano, “Toward Systems-Level Visualizations of Molecular Networks on Large-Scale, High-Resolution Displays”. Poster presented at the IEEE Symposium on Biological Data Visualization in Providence, RI on Oct. 23rd during IEEE VisWeek 2012.