Gremlin: An Interactive Visualization Model for Analyzing Genomic Rearrangements

Category: Research

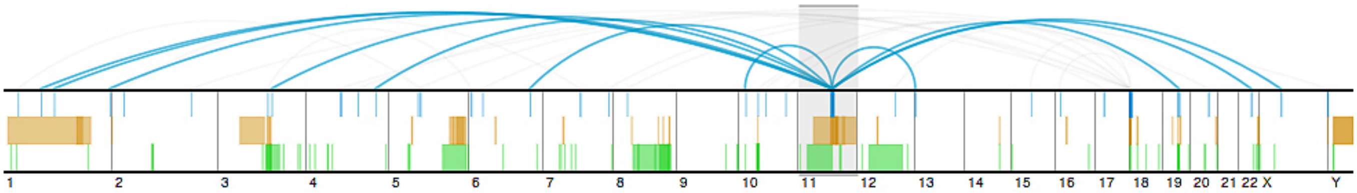


Fig. 1. Our new method for visualizing genome rearrangements: deletions (green), inversions (brown), and inter-chromosomal translocations (cyan) classified from a cancer genome with respect to the reference human genome are depicted. Arcs drawn in the upper region of the display depict inter-chromosomal translocation relationships. A region-of-interest (grey window) is selected, highlighting translocations that intersect the region.

Abstract—In this work we present, apply, and evaluate a novel, interactive visualization model for comparative analysis of structural variants and rearrangements in human genomes, with emphasis on data integration and uncertainty visualization. To support both global trend analysis and local feature detection, this model enables explorations continuously scaled from the highest-level, complete genome perspective, down to the lowest-level, nucleotide base-pair view, while preserving global context at all times. We have implemented these techniques in Gremlin, a genomic rearrangement explorer with multi-scale, linked interactions, which we apply to four human cancer genome data sets for evaluation. Using an insight-based evaluation methodology, we compare Gremlin to Circos, the state-of-the-art in genomic rearrangement visualization, through a small user study with computational biologists working in rearrangement analysis. Results from user study evaluations demonstrate that this visualization model enables more total insights, more insights per minute, and more complex insights than the current state-of-the-art for visual analysis and exploration of genome rearrangements.

Index Terms—Information visualization, bioinformatics, insight-based evaluation.

1 Introduction

We present, apply, and evaluate a novel, interactive visualization model for comparative analysis of structural variants and rearrangements in human genomes, with emphasis on data integration and uncertainty visualization. To support human geneticists and computational biologists in performing both global trend analysis and local feature detection, this model enables explorations continuously scaled from the highest-level, complete genome perspective, to the single chromosome perspective, down to the lowest-level, nucleotide base-pair view, while visually preserving global context at all times. We have applied our model to four human cancer genomes that have undergone rearrangement detection, and validated its effectiveness through a comparative, quantitative, insight-based user evaluation.

Specifically, the contributions of this work to the research community are four-fold:

1. A novel, interactive visualization model for exploring structural variants and rearrangements in the human genome,
2. Gremlin (Genome Rearrangement Explorer with Multi-scale, Linked INteractions), a freely available, web browser-based implementation of our visualization model,
3. A quantitative, insight-based comparative evaluation of our visualization model and Circos [5], the current state-of-the-art in visualizing genomic rearrangements,
4. A discussion of design space decisions and associated tradeoffs in developing an interactive framework for visual genome rearrangement analysis.

The need for effective, available, visual exploration and analysis tools for genomic rearrangement data is becoming increasingly apparent as genome sequencing technologies continue to become more efficient and widespread. This notion is supported most recently in a *Nature Methods* article [7], where the case is made that "*data analysis is replacing data generation as the rate-limiting step in genomics studies.*" And while recent cancer genome studies [9] support the notion that rearrangement analysis may lend insight on the factors that contribute to cancer, most breakthroughs have occurred in cancer types where the number of genome variations are limited and highly localized. In situations where rearrangements are more densely distributed throughout the genome, like breast and ovarian cancer, more sophisticated methods and analyses will be required. As such, a goal of this work is to increase the adoption of visualization tools by computational biologists and human geneticists to aid in the development of rearrangement analysis algorithms and promote scientific discovery.

We begin with an overview of genome biology and continue with a discussion of related works and the introduction of our interactive visualization model. We then describe our evaluation methodology and results from insight-based user evaluations. We provide a discussion of our results and design decisions and finish by presenting conclusions.

2 Genome Biology

Many types of variations have been observed in human genomes, from single nucleotide polymorphisms (SNPs) to larger structural variants and rearrangements that span thousands of nucleotide bases. For the purposes of this paper, we focus on these larger rearrangements. In particular, we aim to explore three types of variants: deletions, inversions and translocations. These rearrangements are detected using advanced computational techniques, which infer the variants by mapping paired-end genome sequence fragments to the reference human genome. An illustrative example depicting this taxonomy of genome variations is given in Figure 2.

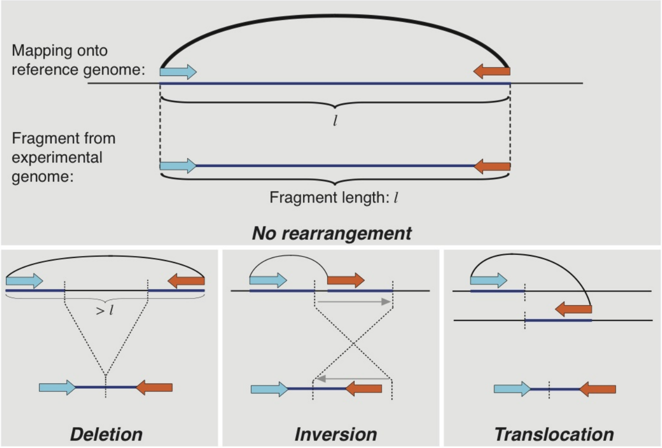


Fig. 2. An illustrated taxonomy of the genome rearrangements explored in this work.

3 Related Works

There exist several tools for visualizing the reference human genome, most notably the UCSC Human Genome Browser [4]. These tools, however, do not support the display of structural variants or rearrangements, and are not designed for the purposes of comparative genomics.

Circos [5] has become increasingly popular in the genetics community, and has recently been utilized in [9] for displaying genomic rearrangements like deletions, insertions, and inter-chromosomal translocations. An example of this style of visualization is given in Figure 4. While the circular ideogram-based model is designed to support comparative viewing across multiple genomes, its interface is not interactive, and does not support the exploration of genome details in context. Cinteny [12] and GenomeComp [14] are other rearrangement visualization tools, but neither supports multi-scale browsing of details in context, the type of analysis required to understand and refine computational methods. Seevolution [3] takes a 3D approach to visualizing genome rearrangements, though the focus of the tool is on the exposition of evolution and not exploratory analysis studies like those in cancer genome research. MizBee [6], a genome synteny browser takes an approach similar to that presented in this work, in that its visualization model also takes an interactive, multi-scale approach to exploring the genome. The focus of their visualizaton, however, is on exploring the conservation between two genomes, which is, in some sense the complement of our focus.

In the computational biology community, several algorithms exist for detecting single nucleotide polymorphisms in the human genome, though relatively few have begun to focus on the larger structural variants and rearrangements that we aim to explore in this work. Here, we utilize output from two recent computational approaches known as Breakdancer [2] and GASV [11].

4 Interactive Visualization of Genomic Rearrangements

In this section, we present the first contribution of this paper: A novel, interactive visualization model for exploring structural variants and rearrangements in the human genome, with emphasis on multi-scale data integration and uncertainty visualization. This model consists of three linked perspectives enabling explorations continuously scaled from the highest-level, complete genome perspective, down to the lowest-level, nucleotide base-pair view: a complete genome view, a region-of-interest view, and an isolated rearrangement view.

4.1 Displaying the Genome

As a point of reference, the human genome consists of over 3 billion nucleotide base pairs, which invariably requires a many-to-one mapping of bases to pixels on modern displays. When considering genome mappings, it should also be noted that chromosome ordering is an artificial construct related solely to the size of each chromosome (with the exception of the sex chromosomes). Biologically speaking, there is no structured order in the existence of chromosomes.

In our model, the reference human genome coordinates are presented according to a 1D, horizontal linear mapping to screen space. Though biologically abritrary, the conventional chromosome numbering is adopted to order chromosomes from left to right, with the sex chromosomes given at the rightmost end of the mapping. Grey lines are used to delineate chromosomes. An example of this approach is given in Figure 1.

4.2 Visualizing Rearrangement Features

Each genome rearrangement consists of several features to be visually encoded. These characteristics include the location of detected rearrangement breakpoints, uncertainty in breakpoint locations, the type of rearrangement, and the support or confidence in the rearrangement. An additional feature, specific to translocations, is the inter-chromosomal relationship between detected breakpoints. Here, visual cues are assigned to each rearrangement parameter.

Location To indicate the location of a rearrangement along the complete genome view, a semi-transparent, rectangular glyph is drawn with its left-most edge aligned to the pixel that is mapped from the starting breakpoint index of the rearrangement. In the case of deletions and inversions, the rightmost edge of the glyph is aligned to the pixel mapped from the ending breakpoint. This results in a rectangular glyph with width proportional to the difference between detected breakpoints. This may be interpreted as the *size* of the rearrangement. In the case of inter-chromosomal translocations, current sequencing technology does not enable a sense of rearrangement size to be determined. As such, each translocation breakpoint is depicted with uniform width.

In the region of interest view, rearrangement location is encoded via a semi-transparent wedge, as seen in Figure 3. The left, inner edge of the wedge is aligned to the pixel mapped from the starting breakpoint of the rearrangement, while the inner, right edge is aligned to the pixel mapped from the ending breakpoint of the rearrangement. The height of wedge is proportional to the size of the rearrangement.

Through additive effects, the semi-transparent glyphs utilized to depict rearrangement location creates salient visual cues at positions along the genome that are dense with variants or rearrangements. These salient cues are intended to draw the user's attention to potentially interesting localized regions. Examples of this effect are evident in chromosomes 1, 5 and 11 in Figure 1.

Type Glyphs from each type of rearrangement are differentiated through unique colors chosen from a perceptual color space. Additionally, to reduce blending artifacts and visual clutter, each type of rearrangement is depicted at a different height along the linear genome view. In our implementation, deletions are placed at the bottom of the view, with inversions in the middle and translocations on top. Under this arrangement, we chose the coloring of inversions to be uniformly spaced between that of deletions and translocations with respect to perception. An example of the visual encoding of all three rearrangement types in the same region can be seen in chromosome 11 of Figure 1.

Inter-chromosomal relationships In the case of translocations, inter-chromosomal relationships are depicted by arcs of uniform thickness, connecting one breakpoint glyph to its complimentary paired glyph. To reduce confusion in arc-crossings, the height of each arc is determined by a logarithmic scaling of the distance between rearrangement breakpoints. For instance, in Figure 1, the arc from chromosome 11 to chromosome 6 is drawn with greater height than that from chromosome to chromosome 10. In our implementation, the maximum arc height is specified to be one half the total height of the complete genome view. In the region-of-interest view, translocations are depicted using small wedges that point in the direction of their paired chromosome. This can be seen in Figure 3

Confidence In genome rearrangement analysis, each detected rearrangement is supported by some number of sequenced genome fragments. The greater the number of fragments that supports a rearrangement's detection, the more confident users are in its existence. In our model, this parameter is visually encoded in the region-of-interest view, where the thickness of the wedge is proportional to the number of fragments that support its existence.

Uncertainty in location Some rearrangement algorithms, like GASV [11], produce uncertainty measures, or error bars, about each detected breakpoint location. This measure of uncertainty is depicted in the isolated rearrangement view. In this view, each rearrangement breakpoint is indicated by a thin bar, while a semi-transparent wedge is drawn about each breakpoint, indicating the range of uncertainty about each detected location. An example of this display is given at the bottom of Figure 3.

In addition, contextual information is integrated into the visualization including gene copy count and cytogenetic bands. Binned copy count information is rendered along the bottom of the complete genome view, colored red where copy number is high, and blue where copy number is low. This information is also rendered at the appropriate scale in the region-of-interest view. Due to their small size, cytogenetic bands are rendered along the bottom of the region-of-interest view only, and colored according to conventions from the UCSC Human Genome Browser [4].

4.3 Multiscale, Linked, Interactive Perspectives

Our interactive visualization model consists of three persistent views: a complete genome view, a region-of-interest view, and an isolated rearrangement view. The complete genome view is displayed at the top of the visualiation, providing global context for the zoomed views at all times. A semi-transparent selection window is rendered atop the the complete genome view, defining a localized region of the genome to be displayed in the region of interest-view. Users may interactively translate and resize this selection window, which dynamically populates the region of interest view with information from the appropriate region of the genome. Additionally, the coloring of translocation arcs at the top of the visualization dynamically updates according to the selection window. When one breakpoint from a translocation lies within the region defined by the selection window, the arc associated with that translocation is rendered in blue, creating visually salient clusters of translocation arcs to enable trend recognition.

Within the region-of-interest view, users may point and click on a rearrangement glyph to populate the isolated rearrangement view with information specific to the individual rearrangement. In addition, mousing over translocation glyphs brings up a tooltip indicating which chromosome contains the rearrangement's paired breakpoint. Mousing over copy number icons displays a tooltip with the specific gene name at that location.

In the isolated rearrangement view, users are provided with information including breakpoint indices, rearrangement size, and the number of supporting fragments. In our web browser-based implementation of this visualization model, clicking on the rearrangment glyph in the isolated variant view links the user to the UCSC Human Genome Browser [4] positioned at the appropriate location in the reference human genome, enabling fast access to comprehensive reference information.

4.3.1 Interaction

To support interface flexibility, several navigation-based interactions are employed in our visualization model.

Genome zooming: When mousing over the complete genome view, the cursor displays a cross-hair icon, indicating users may click-and-drag to define and resize a region of interest. Selecting smaller regions results in zooming in, while subsequently selecting larger regions results in zooming out.

Genome walking: When mousing over the selection window, the cursor displays a double-arrowed icon, indicating users may click-and-drag to translate the selection window along the genome.

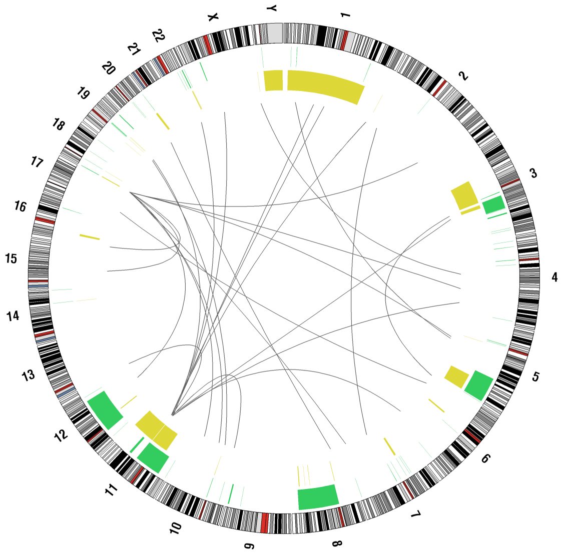


Fig. 4. Circos visualization of genome rearrangements in a cancer genome, given in the style presented in [9].

Genome jumping: When mousing over the complete genome view, and presented with the cross-hair icon, users may perform a single click to center the selection window about the cursor position.

Chromosome iteration: By pressing the up or down arrows on the keyboard, users may snap the selection window to the chromosome currently selected. Pressing the left and right arrows snaps the selection window to the chromosome immediately to the left or right of the current chromosome, respectively. In this way, users may treat the region-of-interest view as a chromosome viewer.

5 Gremlin

In this section, we present the second contribution of this paper: Gremlin (Genome Rearrangement Explorer with Multi-scale, Linked INTeractions), a freely available, open source, web browser-based implementation of our visualization model. Gremlin was built using Protovis [1], a Javascript extension that allows for declarative programming of visualizations. As such, this tool runs in any web browswer that supports scalable vector graphics, and does not require knowledge of source code or installation procedures. Gremlin and all of its source materials are available at http://gremlinViz.org.

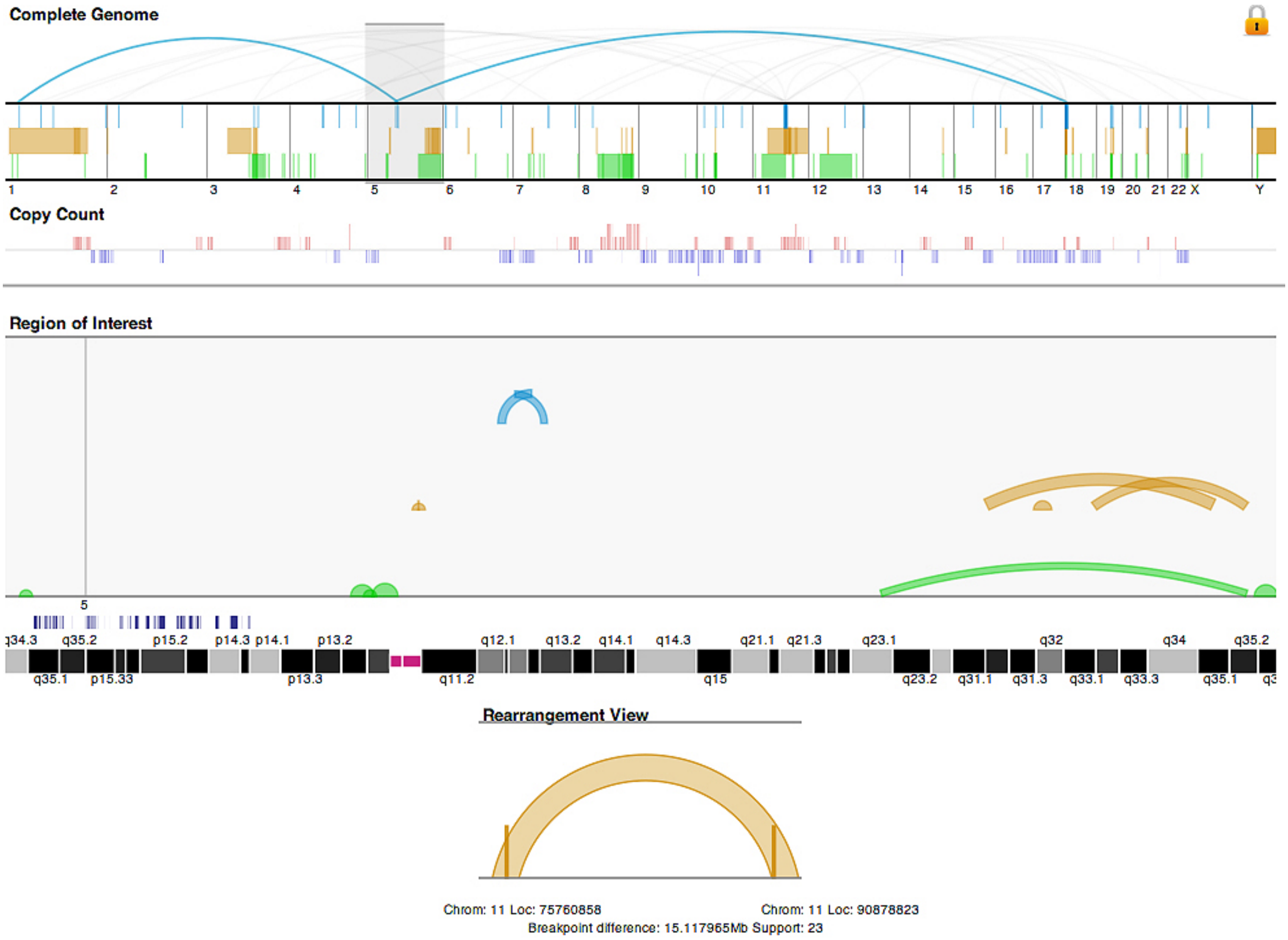
6 Evaluation Methodology

To validate the interactive visualization model implemented in Gremlin, we perform a comparative, quantitative, user study evaluation with Circos, using an insight-based methodology similar to those given in [8, 10]. In addition, we obtain and analyze qualitative feedback from participants to provide further differentiation of the effectiveness of these visualization models.

6.1 Quantifying Insight

Meaningful quantitative analysis of insight requires a clear definition and a consistent set of supporting metrics. In accordance with [10], we define an insight to be a unique, individual observation about the data by a participant: a unit of discovery. We then use the following criteria to quantify the degree to which each visualization enables insight:

Total insights: Cumulative total of individual observations by a participant.



Chrom: 11 Loc: 75760858 Chrom: 11 Loc: 90878823
Breakpoint difference: 15.117965Mb Support: 23

Fig. 3. Example of the multi-scale, details in context view. At the top of the display is the global view of the entire genome. A sliding window rendered atop the global view defines the bounds for the region-of- interest view given below. Here, an inter-chromosomal translocation has been selected, shown in the isolated variant view at the bottom of the display.

Total hypothesis-driving insights: Cumulative total of hypothesis-driving insights, as determined by an expert. Here, we consider insights that promote investigation beyond the scope of the data to be hypothesis-driving.

Insights per minute: Total insights divided by the duration of the visual exploration session.

Insight complexity: Categorization of the complexity of each insight by an expert, where categories are defined as:

Type A: Simple observation, could have been discerned from textual analysis.
e.g. "There are more rearrangements in the output from one algorithm than another."

Type B: Detailed observation, not readily apparent through textual analysis.
e.g. "Many of the rearrangements predicted in chromosome 4 lie within a centromere," *or*, "There is a distinctly patterned distribution of deletions across the genome."

Type C: Detailed observation with context, involving cross-referencing of observations, knowledge base.
e.g. "The pattern of rearrangements in chromosome 11 is interesting, because genes in chromosome 11 are known to play a role in cancer," *or*, "The difference between rearrangements in chromosome Y taken from two algorithms differ in a particular way, which suggests one of the algorithms is doing something wrong and warrants investigation."

6.2 User Study

In this section we describe the experimental details of our user study design and execution. The ultimate goal of this study is to compare the visualization model implemented in Gremlin to the current state-of-the-art genome rearrangement visualization tool, Circos, in terms of the degree of insight provided to users.

6.2.1 Participants

Five experts in computational biology volunteered to participate in the study. The group included one Post-doc, three Ph.D. students, and one Masters student. Three of the participants were female, and two male. All of the participants were familiar with genome rearrangement analysis, and three of the participants were familiar with Circos. Two of the participants had seen conceptual prototypes of Gremlin prior to the study.

To allow for posterior analysis, all sessions were recorded on video with the consent of the participants.

6.2.2 Genome Rearrangement Data

The data sets used in this study originated from four cancer genomes obtained through clinical studies performed by Washington University

Table 1. Expected insight questions provided by participants of our study. If the visualization model enabled the insight, it is marked with an X, otherwise its entry is left blank. If both models enabled the insight, but participants suggested one model significantly outperformed the other, it is marked with a red X+. The first grouping, questions 1-6, relate to rearrangement features. The second grouping, questions 7-9, relate to confidence and uncertainty in results, and the last grouping, questions 10 and 11, relate to comparative visualization.

	Initial Questions	Circos	Gremlin
1	What is the distribution of rearrangements across the genome?	x	x
2	What are the locations of rearrangements throughout the genome?	x	**x+**
3	Which types of rearrangements are predicted?	x	x
4	What is the distance between the breakpoints predicted for a rearrangement?	x	**x+**
5	Are any rearrangement breakpoints detected near a centromere or telomere?	x	**x+**
6	Are patterns exhibited in inter-chromosomal translocation relationships?	x	**x+**
7	In which rearrangement predictions are we most confident?		x
8	Are any rearrangement predictions misleading or implausible?	x	**x+**
9	To which degree of uncertainty is each breakpoint location predicted?		x
10	Do two processed genomes exhibit the same rearrangements?	x	x
11	In which ways and in which regions do two processed genomes differ?	x	**x+**

at St. Louis Genome Center[1]. Each genome was processed using two distinct algorithms, GASV [11] and Breakdancer [2], designed for locating and classifying genome rearrangements such as deletions, inversions and inter-chromosomal translocations, providing a total of eight unique sets of genome rearrangement data for visualization.

6.2.3 Visualizations

Visualizations of the rearrangement data sets were produced in both Gremlin and Circos. Interactive Gremlin visualizations were generated according to the methods prescribed in this paper. High resolution Circos visualizations were created in the style of those presented in [9], focused on rearrangement analysis. An example of this visualization is given in Figure 4. While Circos does not produce interactive visualizations in a strict sense, we chose to display the high resolution Circos images in a viewer that readily enabled mouse-based zooming and panning.

6.2.4 Protocol

To support insight-based evaluation, we employed an open-ended, think-aloud protocol in our study. Each participant was assigned to explore four pairs of visualizations: two pairs in Circos, and two pairs in Gremlin, in alternating order. A random selection of three participants was shown Circos first, while the remaining two saw Gremlin first. Each pair of visualizations, shown simultaneously on the same screen, consisted of two distinct rearrangement analyses taken from the same genome but produced by different algorithms. The decision to display two rearrangement analyses at the same time was made to promote the think-aloud nature of the study by providing a real-world use-case: comparing the output of two different algorithms.

At the start of each session, the participant was first asked to provide a series of questions one would expect a visualization of genome rearrangements to answer. These questions were revisited at the end of the the session for comparative analysis of the success of the two visualization models in providing *expected* insights.

According to the random ordering of visualization models, each participant was first shown either Gremlin or Circos. Upon first viewing one of the models, a thorough tutorial was given to the participant, in which all visual elements and interactions were explained. Once the participant was comfortable with the tutorial information, the think-aloud session commenced. The participant was instructed to speak out any observations from the data that came to mind. When the participant determined he or she had finished making novel observations, the session was completed, and the participant moved on to the next pair of visualizations.

7 Results

Results from each user study are reported in terms of quantitative insight-generation measures, expected-insight questions provided by users, and additional qualitative feedback from the participants.

7.1 Evaluation of Insight-Generation

Using the criteria given in Section 6.1, video recorded from each think-aloud study was analyzed to quantify insight-generation. Overall, these measures strongly suggest that the visualization model implemented in Gremlin enables greater degrees of insight than that of Circos.

Across all data sets in the study, each participant produced more total insights using Gremlin than Circos, by an average of 2.16 times or 15.6 insights; more hypothesis-driving insights, by an average of 2.4 times or 2 hypothesis-driving insights, and more insights per minute by an average of 1.39 times, or 0.44 insights per minute. Complete details of these results are given in Figures 5 and 6.

With respect to the types of insights enabled by the two visualization models, results from our evaluation show that the majority of insights enabled by Circos are of Type A, while the majority of insights enabled by Gremlin are of Type B. In general, however, Gremlin outperformed Circos across all data sets and participants with respect to each insight category. Complete details of these results are given in Figure 7.

7.2 Expected Insights

The initial questions provided by participants at the start of each user study session were recorded and later revisited after the participant completed the usage study. Participants were presented with their questions and asked to indicate whether one, both, or neither of the visualization models were able to provide the expected insight. In the case that both models enabled an expected insight, participants were asked to indicate if one model performed significantly better than the other.

Results from this portion of the study show that Gremlin enabled all 11 of the unique expected-insight queries provided while Circos enabled 9. Of the expected insights enabled by both visualizations, the consensus of the participants suggests Gremlin performs significantly better in 6 of the 9 cases. A complete listing of the questions is given in Table 1, along with results of the participant evaluations.

7.3 Participant Feedback on Visualization Tools

In general, user comments suggested a strong preference for the use of Gremlin for both in-depth analysis of a single genome data set, and for comparing two data sets. While three participants indicated Circos was effective in providing immediate visual feedback on global rearrangement trends, all five participants agreed that only Gremlin enabled the type of detailed analysis necessary for understanding and developing new computational methods.

[1] Because these data are unpublished, we are currently unable to disclose specific biological details about any findings in these genomes. To request the data, please contact Washington University Genome Center.

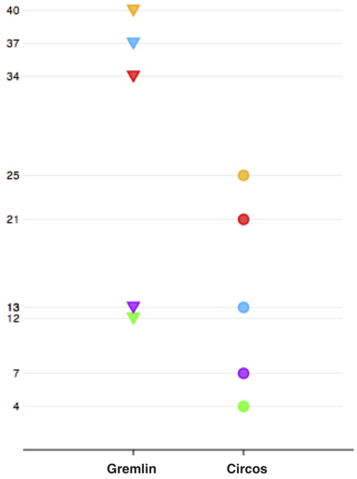


Fig. 5. Total insights from each of the five participants over the two data sets explored with each visualization model. Each participant is colored uniquely. Circos results are denoted by circles, while Gremlin results are denoted with triangles.

The majority of positive feedback for Gremlin was focused on interaction and the dynamic region-of-interest view. In particular, the chromosome-iterator interaction scheme was indicated to be most useful. One participant noted, "*I didn't think I'd find the chromosome-jumping feature to be useful, but I realize now it's what I'm using the most. It lets me flip through the whole genome very quickly, but I still get a detailed view of everything that's going on.*" While Circos produces high resolution images that can be zoomed and panned, participants commented that once they had zoomed to a level sufficient for analysis, landmark cues in the visualization were lost off-screen, like the chromosome numbers. In Gremlin, however, global landmarks persisted despite any degree of zooming.

In addition, four of the five participants noted that the radial layout of the genome in Circos was confusing at times, requiring more mental effort to keep track of the starts and ends of rearrangements. For instance, on the upper semicircle of the ideogram, rearrangements begin and end from left to right, while the opposite is true on the lower semicircle. Similarly, four of five participants noted that keeping track of rearrangement type in Circos was at times confusing. For example, due to the placement of rearrangement information on concentric circles, inversions are denoted "above" deletions on the upper semicircle, while the opposite is true on the lower half of the ideogram. These participants commented that the linear mapping of the genome employed in Gremlin was more intuitive than the radial mapping in Circos and allowed for direct comparisons more easily at specific locations on the genome.

8 Discussion of Results

In this section, we offer a discussion on results and observations from the insight-based user evaluation study.

8.1 User Strategies

In analyzing the user study evaluations, it became apparent that distinct sets of usage strategies emerged from Circos and Gremlin, respectively, which likely played a role in promoting or constraining insight generation. In Circos, nearly all participants proceeded in the same manner: first, view the entire genome for a few seconds; second, zoom in on a particular region along the perimeter of the genome, cross-referencing rearrangement predictions with copy number data; and third, pan along the perimeter, periodically zooming back out to view contextual landmarks. In this sense, it is not surprising that the insights enabled by Circos were often of the same complexity, and often of Type A.

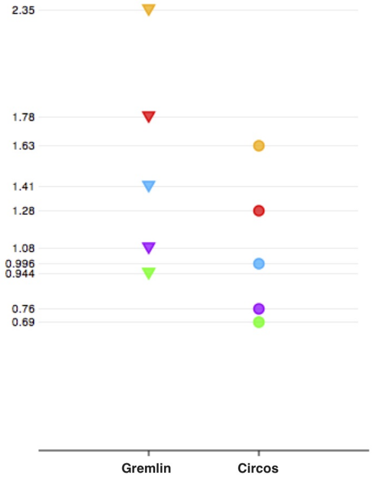


Fig. 6. Insight per minute measures from each of the five participants. Each participant is colored uniquely. Circos results are denoted by circles, while Gremlin results are denoted with triangles.

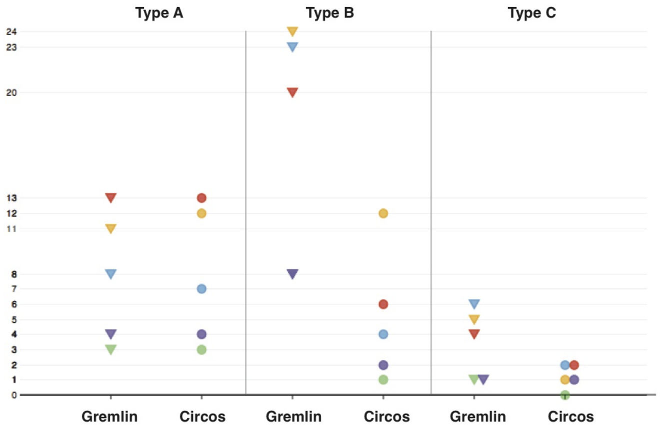


Fig. 7. Total complexity-categorized insights from each of the five participants over the two data sets explored with each visualization model. Each participant is colored uniquely. Circos results are denoted by circles, while Gremlin results are denoted with triangles.

In Gremlin, however, various strategies emerged. A few participants began in the same manner as described from Circos: absorb information from the complete genome view, zoom in on a region and drag the selection window along the genome, looking for interesting features. More commonly, however, participants would make use of the chromosome-iterator interaction scheme, quickly flipping through each chromosome, scanning for intriguing rearrangement information. Another approach in Gremlin was to make heavy use of the point-and-go interaction feature, making quick jumps from one region-of-interest to another. Lastly, one participant focused on inter-chromosomal translocations, and used the translocation-specific selection window to explore rearrangement relationships between chromosomes. In addition, in each of these cases, users would periodically highlight individual rearrangements in the region-of-interest view to obtain more specific information in the isolated rearrangement view. Given the vast array of strategies which emerged from a small group of participants, it is interesting to note that Gremlin consistently provoked insights of varying complexities in our study.

It appears that the diversity of emerging strategies fostered by a visualization correlates with the diversity and degree of insights enabled by that visualization. Thus, while we concede visual perception is integral to insight generation, we posit that flexible, interactive interfaces within a visualization are paramount with respect to sparking insight.

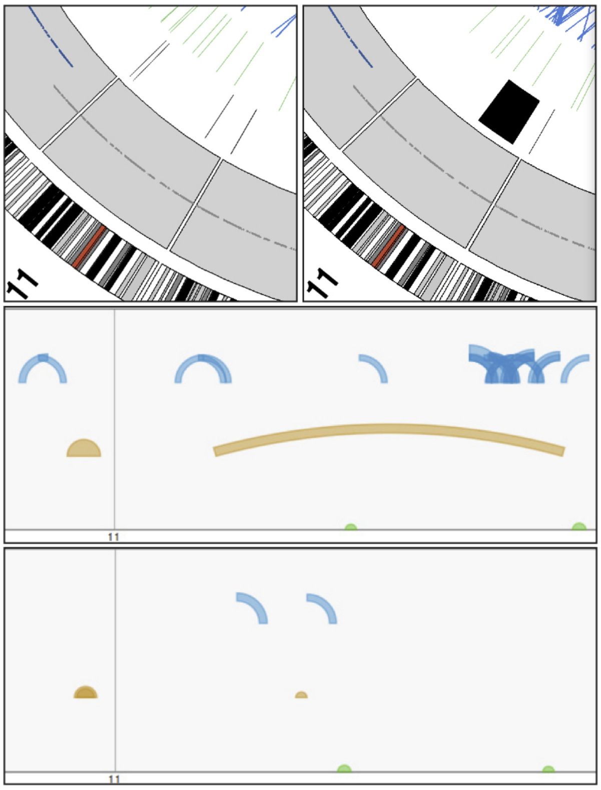


Fig. 8. Comparative visualizations using Circos and Gremlin, representative of the strategies observed in user study evaluations.

8.2 Exploration versus Exposition

To this point in the paper, we have focused solely on the ability of visualization models to enable insight through exploration and analysis. Here, however, we discuss the utilities of Gremlin and Circos with respect to *sharing* insight through exposition.

Many current genome research papers, talks, and presentations rely on Ciros diagrams for visual communication of results. Our findings support this decision for certain data situations. Based on feedback from our user study, we have found Circos to be effective in depicting general trends across the genome quickly, without any action from the user. However, when asked which visualization would be preferred in terms of sharing or presenting findings, participant responses varied, though each user echoed the notion that their choice of visualization would depend on the type of result they wished to demonstrate. To illustrate sparse, global patterns, the majority of participants argued that Circos would be the proper choice of visualization. For more localized patterns, participants suggested creating a static image of Gremlin's region-of-interest view would be a better choice. In the case of larger, seemingly unwieldy sets of rearrangements across the genome, participants supported the full interactive use of Gremlin in demonstrating results. For printing these results, it was suggested that a single, complete genome view image be displayed in conjunction with multiple regions-of-interest. Studying the visualizations toward this end was not a primary focus of our study, and as such, warrants more rigorous examination.

8.3 Comparative Visualization

Developing strategies for visually comparing multiple sets of genome rearrangement data in a concise, legible fashion is a challenging design problem. In our study, participants were provided with visualization displays of two data sets simultaneously for the purpose of enabling comparisons. In nearly all cases, users aligned the visualizations on a horizontal axis and attempted to achieve the same viewpoints in both displays to ensure consistency in visual comparison. While many users believed this strategy was effective, it does not provide a scalable solution for comparing structural variants and rearrangements across larger populations of genomes.

In the style of Circos, rearrangements from multiple genomes could be compared by placing each genome on a concentric circle. However, given the radial mapping, each genome would be mapped to a ring with different circumference and area, making consistent spatial comparisons difficult.

Using the linear mapping of the genome from Gremlin, several complete genome views could be displayed simultaneously, allowing users to perform consistent spatial comparisons by shifting their gaze direction along straight lines. To leverage common screen-space aspect ratios and support comparison of larger numbers of genomes, it may be more appropriate to use a vertical mapping. Resolving multiple regions-of-interest views and isolated rearrangement views in a scalable version of Gremlin presents an additional challenge. An illustration of the scalable comparison approaches discussed here is given in Figure 9.

9 Discussion of Design Decisions

We now present the fourth contribution of this paper: a discussion of design space decisions and associated tradeoffs in developing an interactive framework for visual genome rearrangement analysis.

9.1 Linear versus Radial Mapping

The most glaring distinction between Gremlin and Circos is the choice of genome-coordinate mapping. In Circos, genome coordinates are wrapped around the perimeter of a circle, implying a 2D radial mapping. Our visualization model, on the other hand, employs a 1D, horizontal, linear mapping of the genome to screen space. Both of these mappings have strengths and associated tradeoffs, which we discuss here.

As noted in Section 7.3 some users found the radial mapping in Circos confusing and sometimes misleading when performing spatial comparisons across rings of varying radii in the visualization. Feedback suggests users maintained two separate mental mappings: one for the upper semicircle, and another for the lower. In Gremlin, the linear mapping did not suffer these drawbacks. In fact, several users noted that the linear mapping was "intuitive" and readily enabled spatial comparisons. Furthermore, our choice of horizontal linear mapping is supported by Tufte [13]: "*Graphics should tend toward the horizontal ... Our eye is naturally practiced in detecting deviations from the horizon, and graphic design should take advantage of this fact.* In this sense, we believe a linear mapping requires a lesser degree of cognitive overhead than a radial mapping, allowing users to devote more of their efforts to the data itself.

With respect to inter-chromosomal translocations, the radial mapping in Circos preserves less variance in chromosome-linking arc lengths than does the linear mapping. Given that the assignment of chromosome positions is in some sense arbitrary, this feature may present less visual bias than in the linear mapping, where an arc drawn from chromosome 1 to chromosome 22 is significantly longer, and perhaps more visually distracting, than an arc drawn from chromosome 11 to chromosome 14, for instance. Similarly, however, in Circos, diametrically opposed translocation breakpoints result in straight, visually salient linking-*lines*, despite the fact that such a translocation may be of no particular importance. We believe a more optimal solution for rendering inter-chromosomal translocations would rely on re-ordering the chromosomes in such a way as to minimize arc crossings and prevent misleading visual artifacts. This research direction warrants further investigation.

The final discussion point on genome-coordinate mappings is related to the design of an interactive, multi-scale visualization with linked perspectives. Maintaining multiple views at different scales using a radial mapping is a challenging design problem. In a related visualization, MizBee, focused on genomic synteny, a hybrid approach is taken. The global genome view is presented using a radial mapping, while smaller, scaled regions are depicted using a linear mapping. Though this approach offers a solution to providing scaled

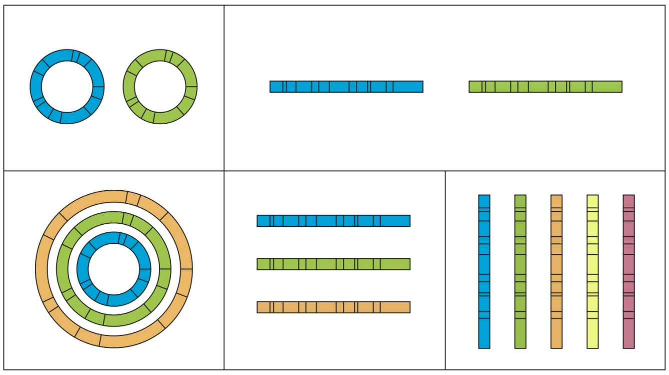


Fig. 9. Comparative visual analysis approaches with radial and linear mappings of the genome. The top row depicts strategies that emerged from our user studies. The bottom row offers scalable approaches for comparing multiple genomes using each mapping type.

views from a radial mapping, we believe it introduces additional steps of mental mappings for users, increasing their cognitive burden.

9.2 Sizing Linked Perspectives

From the results of user studies, the majority of insights enabled by Circos were of Type A (simple observations, often global), while the majority of insights enabled by Gremlin were of Type B (detailed observations, often localized). This analysis agrees with the design of the two visualizations. Circos devotes its entire screen space to a global view of the genome, while Gremlin allocates the most screen space to the region-of-interest view, relative to the other views.

As noted by users, the region-of-interest view was the most useful portion of the Gremlin visualization model, though the reasoning for this result is not unequivocal. In developing our model, we targeted the region-of-interest view to be the focal point of the visualization, enabling continuously zoomed views of portions of the genome. As such, the complete genome view was allocated less area on-screen. We believe permuting the ordering of relative sizes assigned to each view may have interesting effects on insight-generation. For instance, had more screen space been dedicated to the complete genome view, making the region-of-interest view less central to the visualization, we would expect different sets of insights to be generated. In keeping with our previous findings, allowing dynamic resizing of the views may enable a larger set of usage strategies to emerge and provoke novel insights.

10 Conclusions

We have presented and evaluated a novel, interactive visualization model for exploring and analyzing structural variants and rearrangements in the human genome. Through comparative quantitative and qualitative evaluations, we have demonstrated that this visualization model enables more total insights, more insights per minute, and more complex insights than the current state-of-the art for visual analysis and exploration of genome rearrangements. Additionally, we have presented an accessible, open source, web browser-based implementation of our visualization model, Gremlin, which is made freely available at http://gremlinViz.org.

Ultimately, we believe the adoption of effective visualization tools for exploration and analysis of genome rearrangements is integral to the development of improved rearrangement analysis algorithms and the furtherance of comparative genomics studies and scientific knowledge.

References

[1] M. Bostock and J. Heer. Protovis: A graphical toolkit for visualization. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1121–1128, 2009.

[2] K. Chen, J. W. Wallis, M. D. McLellan, D. E. Larson, J. M. Kalicki, C. S. Pohl, S. D. McGrath, M. C. Wendl, Q. Zhang, D. P. Locke, X. Shi, R. S. Fulton, T. J. Ley, R. K. Wilson, L. Ding, and E. R. Mardis. Breakdancer: an algorithm for high-resolution mapping of genomic structural variation. *Nature Methods*, 6(9):677–681, August 2009.

[3] A. Esteban-Marcos, A. E. Darling, and M. A. Ragan. Seevolution. *Bioinformatics*, 25(7):960–961, 2009.

[4] D. Karolchik, G. Bejerano, A. S. Hinrichs, R. M. Kuhn, W. Miller, K. R. Rosenbloom, A. S. Zweig, D. Haussler, and W. J. Kent. Comparative genomic analysis using the ucsc genome browser. 395, November 2007.

[5] M. Krzywinski, J. Schein, A. a. Birol, J. Connors, R. Gascoyne, D. Horsman, S. J. Jones, and M. A. Marra. Circos: An information aesthetic for comparative genomics. *Genome Research*, 19(9):1639–1645, September 2009.

[6] M. Meyer, T. Munzner, and H. Pfister. Mizbee: A multiscale synteny browser. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):897–904, 2009.

[7] C. B. Nielsen, M. Cantor, I. Dubchak, D. Gordon, and T. Wang. Visualizing genomes: techniques and challenges. *Nature methods*, 7(3 Suppl), March 2010.

[8] C. North. Toward measuring visualization insight. *IEEE Comput. Graph. Appl.*, 26(3):6–9, 2006.

[9] E. D. Pleasance, R. K. Cheetham, P. J. Stephens, D. J. McBride, S. J. Humphray, C. D. Greenman, I. Varela, M.-L. Lin, G. R. OrdÃ?Ãez, G. R. Bignell, K. Ye, J. Alipaz, M. J. Bauer, D. Beare, A. Butler, R. J. Carter, L. Chen, A. J. Cox, S. Edkins, P. I. Kokko-Gonzales, N. A. Gormley, R. J. Grocock, C. D. Haudenschild, M. M. Hims, T. James, M. Jia, Z. Kingsbury, C. Leroy, J. Marshall, A. Menzies, L. J. Mudie, Z. Ning, T. Royce, O. B. Schulz-Trieglaff, A. Spiridou, L. A. Stebbings, L. Szajkowski, J. Teague, D. Williamson, L. Chin, M. T. Ross, P. J. Campbell, D. R. Bentley, P. A. Futreal, and M. R. Stratton. A comprehensive catalogue of somatic mutations from a human cancer genome. *Nature*, advance online publication, December 2009.

[10] P. Saraiya, C. North, and K. Duca. An insight-based methodology for evaluating bioinformatics visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):443–456, 2005.

[11] S. Sindi, E. Helman, A. Bashir, and B. J. Raphael. A geometric approach for classification and comparison of structural variants. *Bioinformatics (Oxford, England)*, 25(12):i222–230, June 2009.

[12] A. Sinha and J. Meller. Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. *BMC Bioinformatics*, 8(1):82, 2007.

[13] E. R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, Connecticut, 2001.

[14] J. Yang, J. Wang, Z.-J. Yao, Q. Jin, Y. Shen, and R. Chen. Genomecomp: a visualization tool for microbial genome comparison. *Journal of Microbiological Methods*, 54(3):423 – 426, 2003.