Multiscale Visualization of Structural Variants

Armita Safa, Janet Li, and Neera Patadia

Abstract—The three billion base pairs that comprise the human genome present biomedical scientists with staggeringly complex datasets to analyze. One particular area of interest is the identification and analysis of genomic variants. Structural variants (SVs) are of significance as there is extensive research associating SVs to various human diseases. SVs are annotated in bioinformatic databases with various quantitative and categorical attributes, and these annotations are a wealth of information. Given the complex and high-dimensional nature of genomic variant datasets, we implemented a visualization tool that allows for a multiscale analysis of SV data. Our visualization tool provides a multiscale representation of SV data by utilizing different views and visual encodings to represent multiple attributes for a given SV. The tool is implemented using Gosling.js, a bioinformatics visualization language, and D3.js. It consists of a Circos plot and linear genome browser view, which indicate the location of SVs within the genome, a set of bar charts to represent summary statistics of interest, and a tabular view that lists out all SVs with clinical annotations and relevant details for the user to explore. The tool is designed for bioinformaticians that are interested in investigating and understanding relationships between structural variants within a particular sample.

Index Terms—Structural variants, Genome Sequencing, Clinical genetics

✦

1 INTRODUCTION

The advent of genome sequencing in the biomedical sciences has provided researchers with massive, high-dimensional datasets that can be used for a variety of purposes such as understanding genetic variation in the human population and elucidating the genomic cause of diseases [10]. Genomic data can be considered as having a multiscale structure. This large, multiscale structure can make it difficult to interpret and understand genomic information.

Visualization tools can be used to better understand the overall structure of genomic information, as well as to gain insights into potential relationships within genomic data. A major area of interest in genomics is finding genetic variants within the genome. Genetic variants are considered to be any change in the sequence of nucleotides that make up a given DNA sequence in comparison to a reference sequence. Genetic variants can range in size, from single nucleotide variants (SNVs), to structural variants (SVs) which are any variants larger than 50 base pairs (bp) [9].

Structural variants can take on a variety of forms including deletions, insertions, duplications, inversions and translocations. These structural variants can result in a range of functional consequences, and often contribute to the occurrence of diseases. Structural variants that cause disease are considered to be "pathogenic". Pathogenicity of a variant can fall on a spectrum from being highly pathogenic or likely pathogenic to neutral or benign [1].

Biological data are often stored and shared in large, publicly available databases. Data from these databases can be downloaded as a text file and used for bioinformatic analysis [5]. The National Consortium of Biological Information (NCBI) provides clinically relevant structural variants and their corresponding pathogenicity annotations in its ClinVar database. In this work, we aim to use a curated set of ClinVar structural variants that have been annotated with pathogenicity classifications to develop a tool that visualizes a user's structural variants in relation to ClinVar's reviewed SVs. Data from ClinVar will be used to develop filtering mechanisms to allow for the visualization of variants of differing levels of pathogenicity. To demonstrate the utility of our visualization tool, we will use the Human HG002 dataset, which is a set of variants pertaining to a single individual [13]. The SVs from HG002 will be queried against the set of ClinVar SVs to identify the pathogenicity of the SVs from HG002. These results will be displayed on a global view of the genome, as well as on individual chromosomes, providing different levels of detail in the multiscale

data. Furthermore, we will provide details about associated disease information for individual SVs, based on the ClinVar annotations.

2 RELATED WORK

There are a variety of tools that have recently been introduced for the visualization of SVs. This section will discuss the implementation and utility of these approaches, as well as their benefits and limitations.

2.1 Linear Genome Browser

Linear genome browsers were one of the first classes of tools used to visualize the human genome. The UCSC Genome Browser was initially developed during the Human Genome Project and allowed for the visualization of the DNA sequences of all 23 chromosomes. Linear genome browsers typically display the nucleotide sequence of interest below a reference sequence. The nucleotides that comprise the DNA sequence of interest and the reference genome are displayed in a horizontal view. Furthermore, custom views of the genome or "tracks" can be added to linear genome browsers in order to visualize different aspects of the genome such as genomic variants [3].

The Integrative Genomics Viewer (IGV) tool can be considered as a type of linear genome browser, which allows for the visualization of diverse genomic data types. The viewer consists of a series of rectangular panels. The top panel shows the region being investigated on a chromosome in a horizontal view. Data being visualized through IGV can also include annotations in regards to phenotype, experimental label or clinical label. These annotations can be visualized in the two leftmost columns, with the annotated categories listed vertically [8].

While linear genome browsers have a wide range of utility in the visualization of genomic data, one caveat arises when considering the fact that they are based on visualizing short-read sequencing data. Short reads are not ideal for identifying structural variants, so linear genome browsers have not been optimized to visualize structural variant data [11].

2.2 Ribbon

The Ribbon visualization tool provides a similar view to linear genome browser visualization tools but is designed to be compatible with long read sequencing data. Horizontally at the top of the visualization is a representation of the reference genome segmented into chromosomes. Users can select a chromosomal section to see the relevant sequence alignments of interest lined up vertically, as well as structural variants such as translocations [7]. This visualization can be considered as an improvement over the IGV visualization tool due to its support of visualizing long-read sequencing data.

Table 1. What-why-how for our structural variant visualization instance.

Question	Description
What?	Input SV calls and associated ClinVar annotations for pathogenicity.
Why?	Explore and summarize an SV dataset to understand the distribution of variants and/or identify clinically relevant SV candidates for further analysis.
How?	Multiview presentation of variants at different levels of granularity (genome-wide, chromosomal, single SV).

2.3 MoMI-G: A Graph Based Genome Browser

MoMI-G is a web based genome graph browser that contains multiple panels that can be used to visualize different aspects of genomic structural variants. The panels contain three main views. The first view is a circos plot which provides a chromosomal level overview of the structural variants. Within the circos plot, the structural variants are represented by curved line segments on different regions within the chromosomes. The second view is a table, which contains metadata on each annotated structural variant such as the type of structural variant (insertion, deletion, translocation, duplication, inversion), the chromosome the variant occurs on and the start and end position of the SV. Finally, the browser also contains a linear genome browser view which visualizes structural variant positions in relation to a reference genome [12].

3 TASK AND DATA ABSTRACTION

We reframe our project in the what-why-how framework in Table 3.

3.1 Task

Clinical researchers usually obtain several thousands or even millions of SV calls for a single sample. Exploring the distribution of variants is a data validation step and can help generate hypotheses. Additionally, identifying the medically relevant SVs within a set is crucial for determining the cause of disease and gaining a better understanding of the role of these genomic aberrations in human health. The biological relevance of a variant is often inferred by manually querying a database of known variants (e.g. ClinVar) for matches. The presence or absence of a variant in a database can be used as a metric for prioritizing variants for further analysis, and the extra data available in the database is also used to annotate these variants.

At the analysis level, a clinical bioinformatician both consumes and produces new information from a large SV dataset. By analyzing a set of SVs, the researcher can verify, disconfirm or even generate a new hypothesis about how a disease develops or progresses. The process of annotating variants with clinical and phenotypic data produces new information. Once a bioinformatician obtains a set of SVs, they may want to examine the global distribution of variants, select a set of clinically relevant candidates from the dataset or compare several potentially relevant SVs to one another. They may also be interested in summarizing the entire dataset to get a global view of the genome or a region of it, for example "how many insertions are present on chromosome 21?" or "how many pathogenic variants are present in my dataset?" At the search level, a bioinformatician must browse through a set of SV calls to identify variants of interest. The locations of these SVs are unknown and the exact identity is unknown as well. The user will likely be browsing for SVs with specific clinical attributes, such as SVs labelled as "pathogenic" or "likely pathogenic". There may be cases where a user is simply exploring their dataset, as well, to see if any of their SVs are present in a database, and what diseases are associated with it.

3.2 Data

Structural variants are identified in relation to a reference genome. The human reference genome contains 24 distinct chromosomes, so the chromosome is a categorical "bin" within the reference. Variants are defined by their start position along a certain chromosome in the reference, making them ordered and quantitative. The sequence of the variant, also referred to as the allele, is another important characteristic of SVs. A variant is a single item within a tabular SV dataset. The data presented in our visualization was derived from two input datasets: the ClinVar database [5] and variant calls for the human individual HG002 [13]. The HG002 dataset is used as an example input. Both datasets are tabular, where one item corresponds to a single variant. The ClinVar database consists of 150,782 items in total, and attributes include SV chromosome, position, type, allele, clinical significance, associated phenotype list and associated gene list. The HG002 dataset contains 46,024 items, and attributes include SV chromosome, position, type and allele.

In order to annotate variants with clinical information, we generated a custom dataset by searching the HG002 variants for matches in ClinVar. The HG002 SVs could not be matched simply by identical start positions or sequences because these attributes often vary slightly. For a variant form HG002 to match a variant in ClinVar, we expected their respective locations to be within close vicinity of each other. We also calculated the similarity of variants as a derived variable while performing the search. The similarity score is a string metric that represents how well the sequences from the two variants match. Considering the characteristics of genomic sequences, we performed a local alignment between the two sequences and used the score from the alignment matrix. This value is then converted to a percentage. The output of this search is a set of HG002 variants with matches in ClinVar, and is a combination of the attributes in the two input datasets. While the alleles were used to calculate similarity, they were not presented in the resulting visualization. A summary of these attributes is presented in Table 2.

4 SOLUTION

Our solution is an interactive multi-view visualization tool that presents the structural variants in the HG002 SV dataset, highlighting matches and providing relevant clinical and metadata. The various views represent different levels of detail within the HG002 dataset. The top left portion of the visualization provides a global view of variants in their genomic context (Figure 1A). The SVs and their position along the reference genome is displayed using a Circos plot [4]. Intrachromosomal SVs are encoded by rectangle marks, while interchromosomal SVs (i.e. translocations) are encoded by links. The outer track contains all variants in the HG002 dataset, providing a summary of the entire variant call set. The inner track contains all variants from the custom match dataset. The inner track provides a summary of the clinically significant SVs, and allows users to explore the distribution of variants and identify patterns on a global scale. Two tracks were used because there are many more variants in HG002 that do not have matches than there are variants with matches, and showing all variants on the same track would likely cause occlusion of matches by the more prevalent non-matches.

The hue channel is used to encode the match status and pathogenicity of variants, using a custom scale that has diverging and categorical elements (Figure 2). While three of the pathogenicity levels (benign, likely pathogenic and pathogenic) have an inherently diverging order, variants without a pathogenicity annotation ("unknown significance") do not fall in this order. Therefore the benign, likely pathogenic and pathogenic variants are encoded using a diverging green to pink colour scale, while variants with unknown significance are encoded by a separate blue hue. Items on the outer Circos track, which contains all HG002 variants, are encoded by a neutral, grey hue, which allows items on the inner track to pop. This colour scale allows users to quickly identify pathogenic and likely pathogenic variants from those that are not clinically relevant.

Below the Circos plot is a linked, linear view of the genome, which encodes all matching SVs with rectangle marks (Figure 1B). When a user brushes across a region in the Circos plot, the linear view navigates to this region, providing a more detailed slice of the matches within it (see Figure 3). The brushed region is highlighted within the Circos plot to verify what region is being shown in the linear view. Also, the brushed region in the Circos plot can be modified to the visible range in the linear view by clicking and dragging. The viewpoint can be zoomed in further by double clicking on the linear track. The encoding is mostly

Table 2. Attributes in input and custom datasets.

Variable name	Description	Type	Possible values	Origin
Allele ID	Identifier/key for the ClinVar variant	Categorical	150,782	ClinVar
Chromosome	Chromosome that the variant is located on	Categorical	22 possible values: 22 autosomes + 2 sex chromosomes	HG002
Pos	Start position of HG002 SV along chromosome	Quantitative	1 - length of chromosome	HG002
Type	Type of SV	Categorical	5 possible values: insertion, deletion, translocation, microsatellite, duplication	HG002
ClinicalSignificance	Pathogenicity or clinical relevance for a variant	Categorical, with ordering between benign, likely pathogenic and pathogenic.	4 possible values: Uncertain significance, benign, likely pathogenic, pathogenic	ClinVar
PhenotypeList	Phenotypes (diseases) associated with a variant	Categorical	11,583 possible values: Up to 5 phenotypes are given listed for a single variant; if more than 5 are associated, the number of phenotypes is given instead	ClinVar
HGNC_ID	List of genes (in HUGO gene nomenclature) overlapping a variant	Categorical	7,562 possible values	ClinVar
Similarity	The similarity of the HG002 variant to the corresponding ClinVar variant. Given as a percentage.	Quantitative	0-100	Derived

shared with the Circos plot; variants are all encoded by rectangle marks, and their lengths are encoded by horizontal width. Instead of links, the breakpoints for interchromosomal regions are encoded as rectangles, because connections are more difficult to display on a linear track, especially when they are far apart within the genome. The length attribute is not encoded in the Circos plot because most variants are too small to be visible in the context of the whole genome. The linear track is automatically configured for semantic zooming (see Implementation), so marks only become visible when their calculated width takes up more than one pixel. Hovering over a variant in the linear view causes a tooltip to appear with details of its specific location, type (which is no longer encoded by different marks), similarity score and a list of phenotypes (diseases) that it is associated with.

The linked Circos plot and linear track was chosen as the genome browser idiom to provide both a global and focus view of variants. Circos plots are more space-efficient than linear tracks for presenting the whole genome, and are also more effective at presenting connections (for interchromosomal variants). The linked linear view allows users to examine variants at a smaller scale while still providing local genomic context, and has a lower barrier to entry than graph-based visualizations, which are less intuitive.

The bottom section of the panel contains a table of all HG002 matches with some visual encodings (Figure 1C). This provides granular details about each match, including their exact positions, types, clinical significance, similarity, ClinVar allele ID, associated phenotypes and overlapping genes. Clinical significance is listed and encoded by the colour hue, and similarity score is represented as a line mark, where the horizontal length represents the exact value. Since the table contains many attributes, these visual elements allow users to make judgments about the most important attributes more quickly. Rows can be sorted to rank candidate SVs based on their position, type, clinical significance or similarity, and multi-sorting is available (Figure 4). The table is paginated so it remains a reasonable size, and by default all matches are shown.

The top right section of the visualization panel contains summary plots and interactive controls for filtering and selecting variants. Stacked bar charts (Figure 1F) are used to present the number of variants per chromosome in the ClinVar and HG002 match datasets. Pathogenicity is also encoded by the hue channel in these charts. This allows users to quickly identify patterns in SV distribution within and between the two datasets. Users can select matches by pathogenicity level using checkboxes (Figure 1D), which filters variants shown in the Circos plot, linear track, table and stacked bar chart. Users can also navigate to a specific chromosome using a dropdown menu (Figure 1E). This navigates the Circos brush mark and linear track to that chromosome and filters variants shown in the table. These filtering techniques reduce the number of items shown, and are particularly useful for exploring

Table 3. Team members' contributions.

Contribution	Armita	Janet	Neera
Conceptualization	33%	33%	33%
Design	33%	33%	33%
Data Cleaning and preparation	75%	5%	20%
React app structure	0%	100%	0%
Circos plot	100%	0%	0%
Linear plot	0%	0%	100%
Stacked bar charts	0%	100%	0%
Table	0%	100%	0%
Interactions and linking of views	0%	100%	0%
Course deliverables (slides + reports)	33%	33%	33%

specific categories of variants and identifying "hotspots" within the genome. Filtering items was chosen as the method for reducing items as opposed to aggregation or dimensionality reduction because the purpose of our tool is mainly exploration. Aggregation and dimensionality reduction requires data transformations, and these transformations abstract away certain details that may be necessary for exploration or hypothesis generation.

5 IMPLEMENTATION

The ClinVar and HG002 variant datasets were preprocessed and cleaned using Python and command line tools. Python was also used to identify and score ClinVar matches in the HG002 dataset. The visualization solution was implemented as a web application using various Javascript libraries. The Circos plot and linear genome track were implemented using the Gosling.js grammar and visualization package [6], including the linking of views, brushing and tooltip. Gosling.js provides a grammar for genomic visualizations, and is automatically configured for certain idioms such as semantic zoom. The summary bar charts were implemented with D3.js [2]. The React library was used for the user interface and state management, along with the UI libraries react-table and react-select. The team members' contributions are described in Table 3 The estimated and actual hours and completion dates for our project are outlined in Table 4.

6 RESULTS

Our final visualization is depicted in Figure 1. The aim of our project was to develop a visualization tool that displays pathogenicity annotations for multi-scale structural variant data to guide users in exploring the dataset, identifying clinically relevant variants or comparing several candidate SVs. Our tool successfully displays the SV data at multiple levels using several different views and idioms. The views are linked and interactive, which reduces cognitive load and provides context
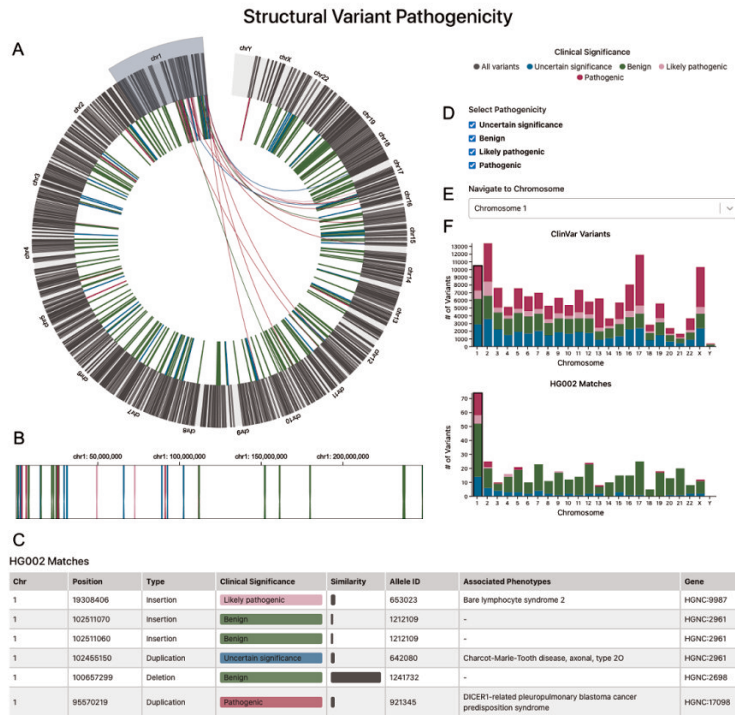
Structural Variant Pathogenicity

Fig. 1. Final visualization panel. The Circos plot (A) contains an outer track showing all HG002 variants, and an inner track showing all ClinVar matches. The Circos plot and linear view (B) are linked through brushing, shown by the blue brush mark. When the user hovers over a variant in the linear view, a tooltip will appear with details. A table (C) lists HG002 matches and also encodes clinical significance and similarity. Two stacked bar charts (F) show the total number of variants per chromosome for ClinVar (top) and for the HG002-Clinvar matches (bottom). Matches can be filtered by pathogenicity (D) or by chromosome (E), which updates the Circos plot, linear track, table and stacked bar charts.

Table 4. Expected and actual completion dates and required time for project tasks.

| Task	Estimated Completion Date	Actual Completion Date	Estimated Time (hr/person)	Actual Time (hr/person)	Description & Assignments
Project Pitch	Sep 29	Sep 29	2	2	-
Pre-proposal Meeting	-	-	-	-	All
Proposal	Oct 21	Oct 21	3	4	All
Obtain data, initial analysis for proposal	Oct 21	Oct 21	2	2	Janet
Data cleaning	Oct 27	Nov 13	2	2	Neera
Match HG002 variants to ClinVar dataset and finalize inputs	Nov 3	Nov 15	4	4	Armita
UI implementation (initial; pre-"Post-Update" meeting)	Nov 5	Nov 9	2	3	React app setup and Gosling config: Janet
	Nov 16	Nov 13	10	12	Bar charts: Janet
	Nov 16	Nov 15	12	10	Circos plot: Armita
	Nov 16	-	12	8	Linked linear view: Neera
Written Update	Nov 16	Nov 16	6	6	All
Peer Project Review	Nov 17	Nov 17	2	2	All
Post-Update Meeting	Nov 24	-	-	-	All
Make any necessary changes to plans	Nov 26	N/A	2	N/A	All
Finish implementing visualization	Dec 12	Dec 14	10	14	Circos plot: Armita
	Dec 12	Dec 14	12	8	Linked linear view: Neera
	Dec 12	Dec 16	30	24	Interactive features: Janet
	Dec 12	Dec 16	4	2	Match table: Janet
Presentation	Dec 15	Dec 15	6	6	All
Final Report	Dec 17	Dec 17	8	8	All
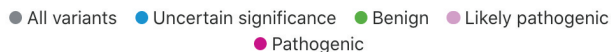
Fig. 2. Custom colour scale for variants.
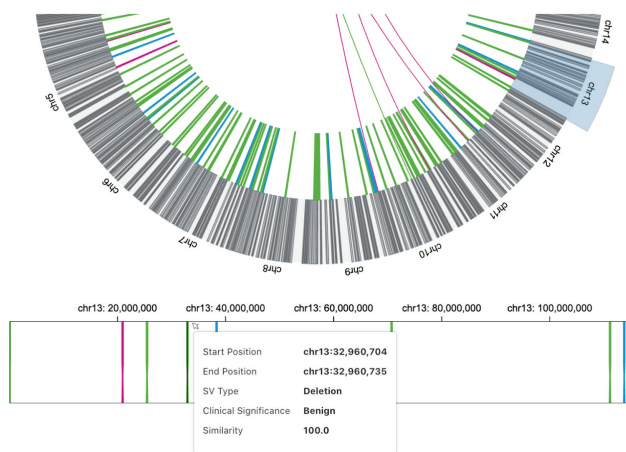
Fig. 3. Linking of Circos plot and linear track through brushing. A tooltip with SV details appears when a variant is hovered over on the linear track.

Fig. 4. Table of HG002 matches, with clinical significance and similarity scores visually encoded.

when filtering and reducing items. Item selection was not implemented in our solution, therefore individual variants cannot be compared easily (see Discussion and Future Work section). Overall, our solution provides multiple levels of detail for the HG002 structural variant dataset, allowing for summarizing patterns and identifying candidates, but falls short in the task of comparing items. We describe a usage scenario for our tool below.

Usage Scenario: HG002 Dataset Exploration

A bioinformatician is interested in using the HG002 dataset for their research, and wants to explore the dataset to get more familiar with it. They first look at the outer track of the Circos plot, and notice that HG002 variants are mostly evenly distributed across the genome. Looking at the inner track, they quickly notice that the majority of interchromosomal structural variants involve chromosome 1. Looking at the lower stacked bar chart on the right, they also notice that the chromosome has significantly more variants than the other chromosomes. Next, they choose chromosome 1 from the dropdown, which causes the linked linear plot to navigate to that region. The table of matches is also updated to reflect this selection. They notice that most variants occur at the beginning of the chromosome, or the left side of the linear track. They sort the match table, and find that most pathogenic variants have very low similarity scores. They consider that these findings may be biologically interesting, but may also suggest biases in the data, taking note that they should further analyze chromosome 1.

7 DISCUSSION AND FUTURE WORK

While Gosling.js allows for the implementation of many bioinformatic visualizations, it is a very new tool that is undergoing active development, and certain customizations are not currently available. For instance, the ability to add interactions such as click events have not been added. As a result, we were unable to implement a "variant selection" interaction, which would allow users to identify specific structural variants of interest from the genome plots for comparison. Our visually encoded table was an alternative to this interaction, and although it provides more detail for filtered variants, item selection would have been more effective for comparisons. Furthermore, the tool is very specific to the generation of high-level genomic visualizations, and is not very customizable. The tool is quite opinionated, and many design choices are default or only slightly modifiable. For instance, the Circos plot can only be configured in a counter-clockwise direction, and only eight marks and ten channels are currently supported. As an extension of this issue, custom marks and channels are not available.

We learned many lessons during the course of this project. The main lesson was the importance of understanding the full range of capabilities of tools before and while planning a design. We had to pivot our plans several times due to unexpected roadblocks, and would have been better off had we taken the limitations of our tools into account before jumping into the implementation of the visualization.

Going forward, we plan to implement additional interaction and filtering features into the visualization. For example, user interactions with the Gosling plots could be implemented so that a user can select specific variants to be visualized in the table. We would also be interested in adding custom glyphs to represent different types of variants in the table. In terms of the type of data being analyzed, we would like to add the ability for users to input their own custom structural variant datasets. Furthermore, additional tracks could be added to the Circos plot to represent the positions of human genes on the chromosomes. The addition of gene level information can allow for insights into a variety of derived attributes such as whether the SVs occur within gene regulatory elements, which would allow for users to investigate how dysregulation within these elements correspond to pathogenicity.

8 CONCLUSION

The main objective of this project was to build a tool for the visualization of structural variant data. We developed a multi-scale visualization of this data, including a Circos viewer, linear browser, and dataset summary plots, along with a tabular view of the match data. To carry out this aim, we developed the genomic visualizations on top of the

Gosling.js grammar. Gosling.js allowed for the implementation of interactive, bioinformatics-specific visualization idioms, along with the ability to add linking between these visualizations and panning to browse through the linear track. The bar plots allow the user to view various statistics relevant to a given set of structural variants, and item filtering reduces the number of items on the screen to reduce cognitive and computational load. During development, the novelty of Gosling.js proved to be a limitation for our visualization as language currently does not allow for the implementation of features such as click interactions, and is quite rigid in terms of how genome visualizations should be presented. Overall, our tool provides bioinformatic users the ability to view complex genomic structural variants, aiding in the ability to interpret this type of data.

REFERENCES

[1] L. G. Biesecker, R. L. Nussbaum, and H. L. Rehm. Distinguishing variant pathogenicity from genetic diagnosis: How to know whether a variant causes a condition. *JAMA*, 320(18):1929–1930, 2018.

[2] M. Bostock, V. Ogievetsky, and J. Heer. D³ data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011.

[3] D. Karolchik, A. S. Hinrichs, and W. J. Kent. The ucsc genome browser. *Current Protocols in Bioinformatics*, 28(1):1–1, 2009.

[4] S. J. B. I. C. J. G. R. H. D. J. S. J. . M. M. A. Krzywinski, M. Circos: An information aesthetic for comparative genomics. *Genome Research*, 19(9):1639–1645, 2009.

[5] M. J. Landrum, J. M. Lee, G. R. Riley, W. Jang, W. S. Rubinstein, D. M. Church, and D. R. Maglott. Clinvar: Public archive of relationships among sequence variation and human phenotype. *Nucleic Acids Research*, 42(Database issue):D980–D985, 2014.

[6] S. L'Yi, Q. Wang, F. Lekschas, and N. Gehlenborg. Gosling: A grammar-based toolkit for scalable and interactive genomics data visualization. OSF Preprints, 2021.

[7] M. Nattestad, R. Aboukhalil, C.-S. Chin, and M. C. Schatz. Ribbon: Intuitive visualization for complex genomic variation. *Bioinformatics*, 37(3):413–415, 2021.

[8] J. T. Robinson, H. Thorvaldsdóttir, W. Winckler, M. Guttman, E. S. Lander, G. Getz, and J. P. Mesirov. Integrative genomics viewer. *Nature Biotechnology*, 29(1):24–26, 2011.

[9] L. Tattini, R. D'Aurizio, and A. Magi. Detection of genomic structural variants from next-generation sequencing data. *Frontiers in Bioengineering and Biotechnology*, 3:92, 2015.

[10] The DNA Universe. A journey through the history of dna sequencing. https://the-dna-universe.com/2020/11/02/a-journey-through-the-history-of-dna-sequencing, October 2020.

[11] T. T. Yokoyama and M. Kasahara. Visualization tools for human structural variations identified by whole-genome sequencing. *Journal of Human Genetics*, 65(1):49–60, 2020.

[12] T. T. Yokoyama, Y. Sakamoto, M. Seki, Y. Suzuki, and M. Kasahara. Momi-g: Modular multi-scale integrated genome graph browser. *BMC Bioinformatics*, 20(1):548, 2019.

[13] J. M. Zook, N. F. Hansen, N. D. Olson, L. Chapman, J. C. Mullikin, C. Xiao, S. Sherry, S. Koren, A. M. Phillippy, P. C. Boutros, S. M. E. Sahraeian, V. Huang, A. Rouette, N. Alexander, C. E. Mason, I. Haji-rasouliha, C. Ricketts, J. Lee, and R. Tearle. A robust benchmark for detection of germline large deletions and insertions. *Nature Biotechnology*, 38(11):1347–1355, 2020.