Fachhochschul-Bachelorstudiengang
MEDIZIN- UND BIOINFORMATIK
A-4232 Hagenberg, Austria

Short Tandem Repeat Calling

Bericht zum Berufspraktikum
bei Dr. Fritz Sedlazeck, Baylor College of Medicine

Erstellt von

Damaris Lattimer

Begutachtet von Magg. Dr. Gerald Lirk

Hagenberg, August 2021

Kurzfassung

Das Praktikum fand remote im Human Genome Sequencing Center des Baylor College of Medicine in Houston, Texas statt. In meinem Team, das Experte für Varianten-Calling im Zusammenhang mit Next- und 3rd-Generation- Sequenzierung Technologien ist, wurde mir ein Projekt zu einer ganz bestimmten genetischen Variante, den Short Tandem Repeats (STRs), zugewiesen. STRs oder Mikrosatelliten, wie sie auch genannt werden, sind kurze Motive von Nukleotiden, die in der DNA mit hoher Wiederholungszahl vorkommen.

Eine solche, sich wiederholende Struktur zu analysieren, ist jedoch ziemlich schwierig. Die Motive sind nur schwer von Sequenzier Fehlern zu unterscheiden und die repetitiven Abschnitte sind in manchen Fällen zu lang, um ihre volle Länge zu bestimmen. Korrelationen mit neurologischen Defekten und anderen Krankheiten wurden in mehr als 50 Erkrankungen beobachtet und haben daher wissenschaftliches Interesse geweckt. Bisher wurden hauptsächlich forensische STRs durch Southern-Blotting nachgewiesen. Den Prozess zu beschleunigen, kann in vielen Anwendungsfällen interessant und durch bioinformatische Analysen wie Next Generation Sequencing und anschließenden Variant Callern möglich sein.

Zusammen mit Fritz Sedlazeck haben wir eine Pipeline und weitere Tools entwickelt, um einen Eindruck vom aktuellen Stand der Technik in STR Calling zu gewinnen. Die Erkenntnisse aus der Studie könnten dazu beitragen, bestehende und zukünftige STR Calling-Methoden zu verbessern.

Abstract

The internship took place remotely in the Human Genome Sequencing Center of Baylor College of Medicine in Houston Texas. In a team that is expert in variant calling of next and 3rd generation sequencing technologies I was assigned a project to a very specific variant the Short Tandem Repeat (STR), to complete during and after the official part of the mandatory Medicine and Bioinformatics bachelor internship. STRs or microsatellites as they are also called are short motifs of nucleotides that occur in the DNA with a high number of repeats.

To analyze such repetitive structure is however quite difficult. The motifs are difficult to keep apart from sequencing errors and the full patterns are in certain cases too long to determine their full length. Correlations to neurological and other disorders have been observed in more than 50 diseases, and therefore raised scientific interest. Up until now mostly forensic STRs were detected by southern blotting. To speed up the process can in many use cases be of interest and possible through bioinformatic analysis due to next generation sequencing and following Variant calling Methods.

Together with Fritz Sedlazeck we developed a pipeline and tools necessary to gain an impression of the current state of art in STR Calling. The knowledge drawn from the study will help to improve existing and future STR Calling methods.

Kurzfassung 1
Abstract 1

1. Introduction 3
1.1. Enterprise: Baylor College of Medicine 3
1.2. Project: ALL OF US 3
1.3. Research Group: Nex-Gen 3
1.4. Background 4
1.4.1. NGS Illumina, Pacific Biosciences, Oxford Nanopore 4
1.4.2. Short Tandem Repeats 6
1.4.2.1. Importance of STRs 6
1.4.2.2. Difficulty of STRs 6
1.5. Next-Gen: Short Tandem Repeat Calling 7
1.5.1. Task and Approach 7

2. Benchmarking STR Calling Tools 8
2.1. Material 8
2.2. Design 9
2.3. STR Simulation: MonSTR (Manipulation ON STRs) 12
2.3.1. Implementation MonSTR (Manipulation ON STRs) 14
2.3.2. Testing 15
2.3.3. Analysing 16
2.4. Testing of STR Callers 18
2.4.1. Concept 18
2.4.2. STR Callers 21
2.5. Evaluation 25
2.5.1. Concept 25

3. Discussion and Conclusion 27
3.1. STR Calling 27
3.2. General 27

4. Bibliography 29

1. Introduction

1.1. Enterprise: Baylor College of Medicine

Baylor College of Medicine (BCM) is a medical school that was founded in 1900 in Dallas. The aim was the improvement of the medical practice in North Texas. In 1943 Baylor College of Medicine was invited by MD Anderson Foundation to join the Texas Medical Center (TMC). (*History*, n.d.)

TMC took its first steps in the 1920s when the Herman Hospital first opened and was finally founded in 1942 by the University of Texas establishment of the MD Anderson Hospital of Cancer and Research. Meanwhile TMC counts as the largest medical complex in the world with around 9 million patients per year and covering a 5 square mile area (12.8km²). (*History Of Innovative Medical Research*, 2019)

The National Human Genome Research Institute (NHGRI) nominated Baylor College of Medicine as one of six pilot programs for the final phase of the Human Genome Project (HGP), therefore the Human Genome Sequencing Center (HGSC) was established in 1996 and counts now as a world leader in genomics and is one of the three biggest sequencing centers in the USA. It employs more than 180 workers on more than 36000 square feet, operates several sequencing platforms like Illumina, Pacific Biosciences, Oxford Nanopore, Sanger and analyzes the sequencing data in bioinformatic pipelines. This and other factors like the close cooperation with other research centers as well as companies of new technologies open opportunities to take part in large scale projects. (*About the BCM-HGSC*, 2016)

The Informatics-Nex Gen group from HGSC works in the context of the large scale project "All of US". Principal Investigator of our work group and the supervisor of the internship is Fritz Sedlazeck Ph.D., who focused the last 5 years on structural variation and focuses now with his team on the identification of coding and non-coding complex variations and their impact.

1.2. Project: ALL OF US

All of Us is a Research Program of the National Institutes of Health (NIH). Goal of the program is "Better health for all of us". It was enrolled in 2018 with a 10 year plan to generate and analyse the genetic and health data of more than 1 mio people as diverse as possible. It is so far an unique approach to consider the biology, lifestyle and environment in order to gain a better understanding of the phenotype and increase the knowledge about risk factors to develop better treatments that are precise and considerate. Each person has to be looked at as an individual and participants from all backgrounds will be considered. All of Us should connect people and clinical studies. The whole genome of 1 million people, genotyped arrays of 2 million people and circa 6,000 Pacbio genomes are getting analysed. To analyze this amount of data complex bioinformatic pipelines are necessary. Therefore the investigation of existing as well as the development of new algorithms could help make analysis faster and more accurate. More and better technologies can be steps to get to a healthier "all of us".

1.3. Research Group: Nex-Gen

The Nex-Gen group analyzes complex variants. One of those variants are Short Tandem Repeats (STR) and therefore we focussed during the last 4 months on technologies that can call those variants (more details will follow in Chapter 1.5 Next-Gen: Short Tandem Repeat

Calling). The Nex-Gen group is involved in “All of us” and several other large projects. They work with clinical as well as research data and are developing and improving biocomp tools, pipelines and processes. Everyone has clear projects and responsibilities. Every wednesday from 12pm till 1pm the team meets in a zoom-call, all participants have one slide to sum up their last week's work. Communication outside meetings works mainly over slack. Working with large and sensitive patient data everybody is mostly working via VPN connection in a cluster. The STR benchmark project was assigned to me and led by Fritz Sedlazeck. We did discuss in one or two meetings per week, the progress of the work. Whenever there were difficulties slack enabled discussing the topic with each other and other colleagues. To benefit from each other's experience is a great prospect.

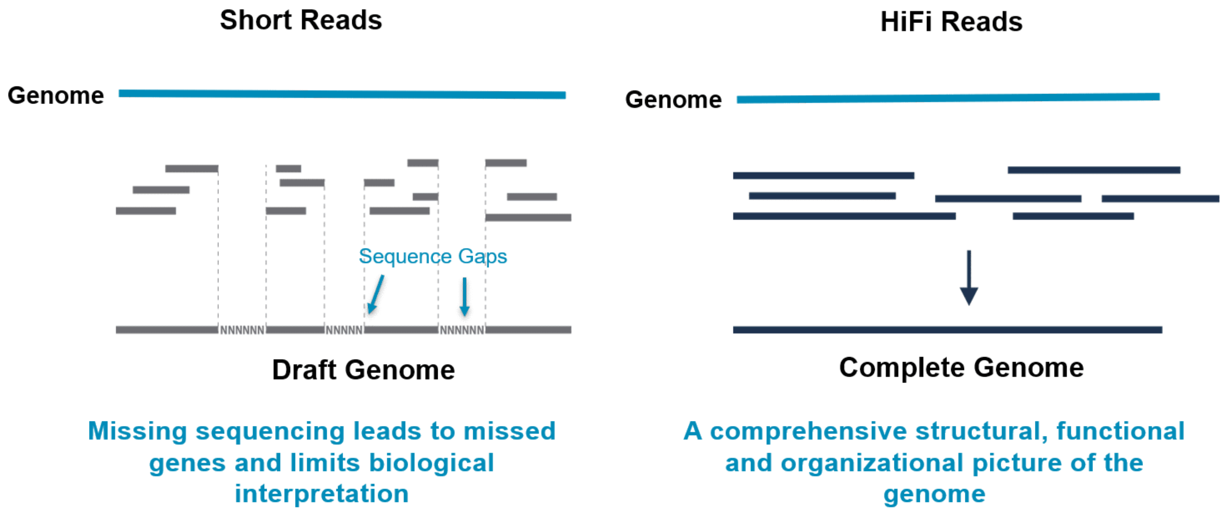
1.4. Background

1.4.1. NGS Illumina, Pacific Biosciences, Oxford Nanopore

A pursuit in variant calling follows only after the sequencing step. Therefore, in understanding and developing variant calling technologies it is vital to be aware of the technology that provided the DNA sequence.
Depending on the complexity or length of the variants of interest, some sequencing technologies might have massive advantages over other technologies that on the other hand are classed as more accurate. Each technology has its weaknesses and advantages. Little comparison see Table 01.
In our studies we used datasets from Illumina, PacBio and Oxford Nanopore.

Illumina

Illumina is an established short read technology that has been dominating the market (Logsdon et al., 2020). The approach is based on PCR amplification and is considered to be still the approach with the lowest error-rate. The limitation of this accurate method is that the amplicon can only reach a certain length.
Short Reads means around 50 till 500 base pairs / read. Assembling short reads can cause sequence gaps (see Picture 01), which make further analysis more difficult. To provide the full length of for example for tandem repeats the reads are often too short. Paired-end reads and information from flanking regions can help in length estimation. Those estimations however lack accuracy and need further validation with gold standard techniques like southern plotting. (Rajan-Babu et al., 2021)
Another difficulty of the PCR amplification is it’s tendency to stutter. Stutters are short DNA motifs that are being added or left out (Riman et al., 2019). To differentiate between stutters and STRs a higher number of reads is necessary. In research 20-30x depths, in clinical sequencing 30-60x depth across the genome and more than 100x average depth across targeted regions are necessary to statistically rule out single errors like stutters over real variants. (Koboldt, 2020).


Short Reads
Genome
Sequence Gaps
NNNNNN
Draft Genome
Missing sequencing leads to missed genes and limits biological interpretation
HiFi Reads
Genome
Complete Genome
A comprehensive structural, functional and organizational picture of the genome

Picture 01: *PacBio: The evolution of DNA Sequencing Tools (Website, n.d.)*

Long read technologies: Pacific Bioscience and Oxford Nanopore

Pacific Bioscience, with its SMARTbells and Oxford Nanopore, with its synthetic nanopore membrane, provide with their third generation sequencing technologies reads that cover entire alleles (Logsdon et al., 2020). These methods reach fragementsizes of in average tens of kilobases. Hence they fully resolve tandem repeats and are likely to expose other nucleotide modifications. This however comes at the cost of a higher sequencing error rate, but with prospect of improvement in the future (Dohm et al., 2020). Instead of using amplification the technologies are based on single molecule or single nanopore real-time sequencing, which lowers the chance of substitutions errors, low GC content and sequence complexity bias (Shin et al., 2013). Oxford nanopore senses nucleotides directly, which makes it possible to detect nucleotide modifications. Further the technology is technically not limited to a maximum read length, even keeps the quality with increasing length and promises a lower turn-around time leading to a cost-effective method. (De Roeck et al., 2019).

Table 01: *Small overview over technologies. (De Roeck et al., 2019; Logsdon et al., 2020)*

Sequencer	Illumina	Pacbio HiFi	ONT
Error Rate	<0.1%	~1%	~8%
Error Rate specific	Stutter	8% insertions	4% insertion 4% substitutions 4% deletions
Read length in kb	0,05-0,25 *	10-20	10-1000
	*(2x bei Paired-end)		

1.4.2. Short Tandem Repeats

1.4.2.1. Importance of STRs

Short tandem repeats (STRs) or microsatellites as they are otherwise known as are repetitive units with a motive length around 1-6bp, depending on the source even associated with longer motifs like 1-9 bp, for example in Depiennes paper (Depienne & Mandel, 2021). They occur in around 3% of the human genome. Of their total count 8% are located in coding regions, the rest in non-coding regions where they could play roles in the regulation of gene expression and transcription. (Fan & Chu, 2007; Raz et al., 2019) They are classed as highly mutable, polymorphic, multiallelic and correlating to an increasing length they become more unstable. Therefore association with human disorders seem evident. Currently 50 disorders with expansion loci are described. (Depienne & Mandel, 2021)
Due to the polymorphic nature of STRs they supply a chance to tell different people statistically significant apart from each other, for example in a forensic context. STRs are in routine-diagnostics and development of interest. A development of an accurate and quick STR Caller can therefore lead to improvement in forensics and medicine.

1.4.2.2. Difficulty of STRs

Calling STRs means finding and counting differences in a region with a repetitive pattern. If the number of repeats is as important as e.g. in forensic, any repeat that is not natural but due to the sequencing technology, is a problem (Riman et al., 2019). Stutters differ from STRs through the coverage. Running multiple reads gives the opportunity to identify true alleles, as they will be present in most reads, whereas sutter alleles will only be present in a few reads. It is not always as easy, considering that the DNA could be from multiple contributors. (“Stutter Analysis of Complex STR MPS Data,” 2018) and the equation is not getting any easier considering the chances of additional sequence gaps. If the TRs are longer than the read length, tandem repeat length estimations are necessary. This is not always possible, especially if flanking regions are as well very repetitive. Gaps and low coverage occur often in regions with high GC-content or DNA degradation. Even in PCR-free protocols those issues are difficult to overcome. (Mousavi et al., 2019). Looking then at technologies that might overcome problems discussed, through for example long read technologies, other factors have to be taken into consideration. As those sequencers are known for their still high error rates around 1-10% (see Table 01). Considering however that the difference between human and human is around 0.1% or between human and chimpanzee is less than 5%, these error numbers are not that small (Pflanzer & Lee, 2018; *What Does the Fact That We Share 95 Percent of Our Genes with the Chimpanzee Mean? And How Was This Number Derived?*, n.d.).
The sequencing errors and mutation rate correlate both positively with a rising length of TR expansion, which increases the difficulty of analysis of STRs and other variants (Fungtammasan et al., 2015). Further, sequencing errors and mutations can cause shifts. They increase the chance of another error: the alignment errors (“Characterization of Pairwise and Multiple Sequence Alignment Errors,” 2009). A high number of errors, mutations and large STRs challenge assemblers to align the reads (Pightling et al., 2014). Therefore error rate and assembler have to be considered (Li, 2018) and correctly chosen, when working with such data. All steps in in situ analysis can cause issues. Another issue less biologically and technically is:
The definition of what is classed as an STR, as it can cause issues when developing and testing STR calling tools. Choosing a differently defined set of repeats, than the STR calling method was designed for, could obviously lead to poorer results. After creating a repeat data set by UCSC (*Table Browser*, n.d.), we identified too few overlaps with an STR data set generated by GangSTR, an STR calling method(Rajan-Babu et al., 2021). This could be caused by different definitions of the term “microsatellites".

1.5. Next-Gen: Short Tandem Repeat Calling

Short tandem repeats can stretch over hundreds of base pairs, even though some Tools have accurate length estimation approaches, they might face difficulties with complex STRs like for example forensic STRs. Therefore long read technologies and the STR Callers that specialized for those might seem a reliable choice. However they come as well with their disadvantages. Short Read technologies were considered in the last decade as the more reliable choice and STR Callers might be more established and advanced in their development.

1.5.1. Task and Approach

Aim of this project is to get a good overview over current STR Callers, understanding their approaches and observing their qualities. A comparison of the tools developed for both sequencing technologies can give insights for possible future improvements. How good can short read STR callers overcome named difficulties and are long read STR Callers the answer, or are they still very challenged by higher error rates?
To test the different STR Callers, GIAB data sets and forensic data sets were available, however to test the de novo calling abilities of the tools or their stability towards higher mutations/sequencing errors, data simulations were necessary. Hence a simulation algorithm had to be implemented. Later the evaluation of the distinct result files had to be addressed. The full testing process will after successful run, summed up into a pipeline.
Throughout the entire course knowledge from sequencing to variant calling with its typical file formats had to be gained. For important bioinformatic steps like sequence assembly, mapping and read simulation training in common shell commands and bioinformatic tools were mandatory.

2. Benchmarking STR Calling Tools

2.1. Material

- **Grch 38: Genome Reference Consortium Human Build 38**
  is the unique and unambiguous assembly identifier all our tests were referenced to or based on. (*Frequently Asked Questions*, n.d.)

- **GIAB HG002 is the human control sample we used as a real data set.**
  Usually the first step to gain real data is the preparation of DNA for the sequencers in a laboratory. This step took place outside of this project and our work group, as we worked mainly with reference datasets and genome in a bottle datasets (GIAB). As the consortium states on their homepage: "The priority of GIAB is authoritative characterization of human genomes for use in benchmarking, including analytical validation and technology development, optimization, and demonstration.“ (*Genome in a Bottle*, n.d.). They create reference samples and benchmark variant calls as well as regions with high-confidence under strict guidelines. This enables further research with the certainty of a well analysed, trustworthy genome, as otherwise an NGS run is only at 20-30% reproducible which would make the reliability of later generated data questionable.

- **Region file hg38_ver13.bed.gz by GangSTR** (Rajan-Babu et al., 2021)
  is the file with the coordinates and motifs describing variants, in this case STRs.
  Using this dataset in a benchmark study might bias the results for that tool itself. However when creating an independent UCSC data set of STRs, we got only little overlap with the datasets by GangSTR. Even though the UCSC dataset was filtered to only tandem repeats of the size of typical STRs, the rate of overlapping regions with known STRs from GangSTR was so insignificant that we withdrew the dataset to not cause negative bias for one or more tools. Therefore this dataset was used in the simulation test runs.

- **Forensic dataset** (Gettings et al., 2018)
  Forensic STRs are far more complex than normal STRs. Forensic STRs are not simple repeats of trinucleotides. They consist of consecutively connected different motifs and repeat numbers. As the altering motives are highly polymorphic, the flanking regions are what makes them identifiable. Forensic STRs are planned in our study, to test the tools on extreme cases. However this needs further preparation, and will be a future improvement of the analysis and will occur outside the official part of the internship.

- **The short read STR calling tools**
  GangSTR (Mousavi et al., 2019), HipSTR (Willems et al., 2017), STRetch (Dashnow et al., 2018), ExpansionHunter (Dolzhenko et al., 2019), RepeatSeq (Highnam et al., 2013), STRling (quinlan-lab, n.d.) were observed and of them only GangSTR, ExpansionHunter and STRling were executed in our tests of short read callers.

- **The long read STR calling tools**
  TideHunter (Gao et al., 2019), TandemGenotypes myfrith (Mitsuhashi et al., 2019), NanoSatellite (De Roeck et al., 2019), TRiCoLOR (Bolognini et al. 2020), PacmonSTR (Ummat & Bashir, 2014), NCRF (Harris et al., 2019), STRique (Giesselmann et al., 2019) and STRaglr (Chiu et al., 2021) were explored. The analyses of the long read

tools PacmonSTR and STRaglr were processed. Maybe, one or two of the other named tools might be included into the pipeline when it is finished.
Some of the tools are limited to only gene-wide STR calling or only search for a small countable number of motifs, others require a Nanopore specific fast5 raw data format. Such tools can not be further included into our pipeline, but still be listed and explained. In forensic or clinical investigation, where the demand might be to only look for certain STRs or only analyse certain areas of the DNA, tools we excluded might be a perfect fit, as that's the demand some of the tools were developed for. In our study, we are aiming for the qualities of tools that analyse genome wide, and with unlimited assigned STR motifs. Tools that are able to identify de novo STRs would be ideal for us.

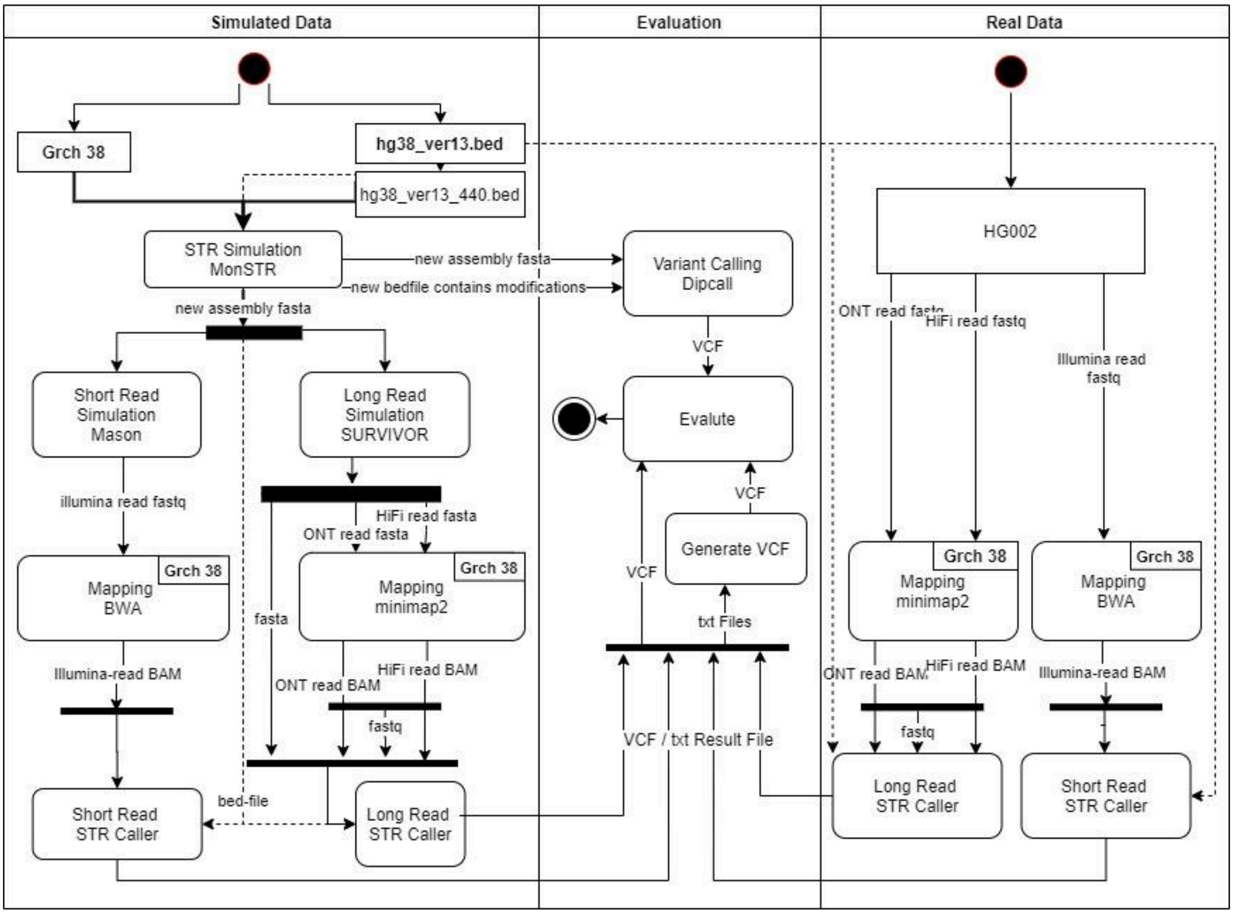
- **Long Read Variant Caller**
  "Clair3 - Integrating pileup and full-alignment for high-performance long-read variant calling" (Luo et al., n.d.). Additionally we computed the long read data sets (BAM files after simulation) on Clair3, to get an impression how well a non-STR specific Caller analyses STRs. Before the project started tests have shown that some insertions and deletions of STRs don't go unnoticed in standard variant callers. This opens an interesting comparison to the STR specific tools. However the Clair3 step is not part of our pipeline. It was executed from other experts of our workgroup and will only be a feature for the evaluation.

2.2. Design

STRs are regions that still beare a lot of secrets. As they are so polymorphic, finding and analysing them might help understand phenotypes better. In order to find them quicker and even in routine analysis STR Callers could help a great deal. There are only a few STR Callers around by now, and most of them are limited in a certain way. Our aim was to get an understanding of current STR Calling methods.

Therefore we designed a pipeline to analyse the STR calling tools looking at their ability to find known and unknown (de novo) STRs. We tested the tools on a few simulated regions, and a few blindfolded simulated regions. Further we tested them in a full genome approach focussing on the GIAB genome HG002.
Last step planned is as well to see how they deal with extremely difficult STRs like forensic STRs, unfortunately this step is quite work intensiv, and will take place outside the official part of the internship.

The pipeline to analyse the STR Callers is outlined in Picture 02.
Several different steps are necessary to prepare the genomes to make them available as input data for the STR Callers. The STR Tools result in very individual formats. Some use the widely known VCFs, which enables the utilization of publicly available tools for the evaluation process. Other formats need first be translated into VCF files or other evaluation methods have to be developed and included into the pipeline.



Picture 02*: Activity Diagram. Gross overview over the pipeline. Missing in this diagram is the forensic approach and the Clair3 step, which was briefly mentioned in the material section.*

About the pipeline steps:

The output format of a sequencer (f.e. illumina, ONT, HiFi) can be different raw data formats, which usually are getting transformed into fastq files. They are then either getting assembled or mapped.

Fastq files and BAM files are ideal formats to allow further analysis, as they not only give information about the DNA sequence, they contain information about the quality of each base too, and in this format the genome is still in the form of several reads per region. Reads are length limited sequences. Regions in the DNA have to be covered by several reads depending on the technology and the use case, to enable better differentiation between sequencing errors and variants.

By mapping the reads of the fastq file against a reference genome of choice, the BAM/SAM format (binary alignment map, sequence alignment map) can be created. For long read mapping we used minimap2 (Li, 2018) and for short read mapping BWA MEM (Li & Durbin, 2009, 2010), which are both very established and widely used tools. Mapped reads have, given to their connection to a reference genome, coordinates which enable identification of genes and other known structures.

Assembly on the other hand is only the calculation of the most likely base arrangement given the quality and coverage of each base. As there is no alignment to a known structure as the reference genome, it's more difficult to determine the definite position of genes and such. However the reference genome itself is an assembly or is the alignment of many assemblies from genomes across the world.

Important: A fasta file can contain assembled sequences (assembly) or sequencing reads (as the reads from fastq and BAM). It is always important to know when a tool requires a fasta file which format is wanted.

"Simulated approach" in the Pipeline:

In our simulated approach, we start with a reference genome. After modifications induced by MonSTR (Manipulation on STRs), our simulation tool, we still have an assembly fasta.

The STR Callers usually operate soon after the sequencing step, therefore they use the file formats from this previous step like fastq, BAM or sequencing read fasta and will not operate on assembled genomes, as we currently have. So another simulation is needed. One that simulates a realistic number of reads in defined lengths to make long or short read sequences in fastq/fasta formats. Those reads should look, if they would get reassembled again, like the input. For these sequencing read simulations a few factors are vital. They influence not only read length and number of reads, but influence sequence gaps or overlaps - the coverage. Published equation for coverage in Equation 01 and transformation of formula to simulation needs in Equation 02.

- Number of Reads	***N***
- Coverage	***C***
- Average Read Length	***L***
- Length of the genome	***G***

- for **short reads** usually standard: Paired End - resembled by a factor **2**. (*Paired-End vs. Single-Read Sequencing*, n.d.)

$$Coverage\ =\ \frac{Number\ of\ Reads*Average\ Read\ Length}{Genome\ Length}\ or\ for\ Short\ Reads:\ C\ =\ \frac{(N*L)*2}{G}$$

***Equation01**: Calculation of coverage (Casey, n.d.)*

The coverage is the number a user wants to influence. A given factor is the genome length, and indirectly given by the sequencing technology that is supposed to be simulated is the Average Read Length. The factor that is to be assessed is the total Number of Reads.

$$Number\ of\ Reads\ =\ \frac{Coverage * Genome\ Length}{Average\ Read\ Length} of\ for\ Short\ Reads:\ N\ =\ \frac{C * G}{2 * L}$$

***Equation02:** Calculation of Number of reads*

For short read sequencing simulation Mason (Holtgrewe, 2010) was used and for long read simulations SURVIVOR (Sedlazeck et al., 2017) was used. Masons short reads are in fastq format and can be directly mapped using BWA MEM. The mapped BAM file with short read sequencing data is input format to all short read STR Callers.

SURVIVOR modelles long reads into a fasta format. This can be mapped using minimap2 or some long STR Callers use it directly. After mapping, the long read BAM files are valid input files for most other long read STR Callers. However, here is another exception to the rule as one STR calling tool processes fastq files. So another conversion might be necessary from BAM to fastq utilizing bedtools (Quinlan & Hall, 2010). In all BAM File-preparations sorting, indexing (Danecek et al., 2021) and applying reads groups (Poplin et al., n.d.; Van der Auwera

& O’Connor, 2020) is advisable. How the region files play a role in STR calling and evaluation will be further discussed in the later sections Testing of STR Callers and Evaluation.

Real data approach:

The HG002 approach is a lot simpler. The HG002 genome is in fastq-format, it is available from the different sequencers. Every now and then new versions are getting published, if they could improve analysis of the genome. For example, the newest HG002 ONT version was only released in August this year. It’s error rate is lower than anything before from ONT, with less than 2% error rate over 8% previous error rate. Each of the three genomes had to be mapped to the reference genome Grch38 that we used as reference throughout the study. Further file preparations and conversions had to be executed if necessary for a variant caller. This procedure was described before in the “simulated approach”. Details about STR Calling considering the region files and about the evaluation will be discussed, as mentioned before, in later chapters.

2.3. STR Simulation: MonSTR (Manipulation ON STRs)

Manipulation of a Referencefile in order to simulate STR!!

The simulator takes a haploid file as reference(.fasta) and a region file (.bed) containing information about known STR-regions as input. All of the supplied regions can be modified in

- expansion (% of regions that will randomly be positive or negative expanded [0.00-1.00]),
- mutation (% chance for a base to be substituted [0.00-1.00]),
- number of indels (X times less likely than chance for mutation to insert or delete a base [0.00-1.00]). Further can the simulation file (.fasta) be created as
- haploid [h] or
- diploid [d]. If diploid is chosen,
- the percentage of regions that should get homozygous can be set [0.00-1.00].

The simulator works on assembled genomes, as well as on only one or more assembled chromosomes, likewise if the bed-file contains such entrances. Anything else could run error free, but will not manipulate anything, as manipulations only occur in the known regions.

Important:

Input Bed File, need to be sorted in region-start and region-end (f.e. bedtools sort -i myfile.bed > myfile.sorted.bed)

Current Version expects fasta and bedfiles with Chromosome-names without "chr" (“1” instead of “chr1”). There is a folder in the github with other versions that might contain readers dealing with "chr" naming.

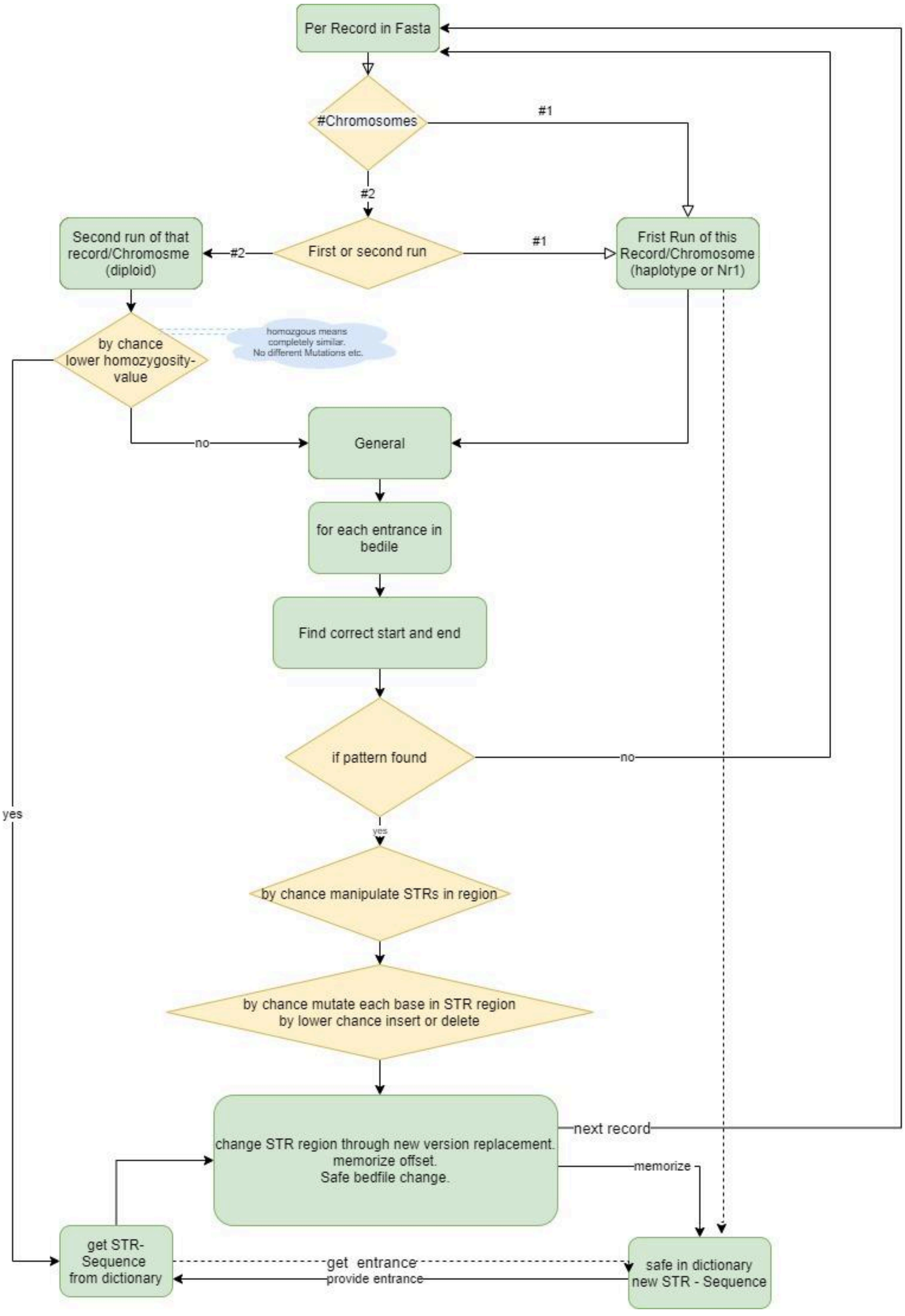
Bed File should not have a header
Further Input Bed File should have the columns:
Chromosome Nr; Region Start; Region End; Motif Length; Motif
for example :

22	20348371	20348390	4	TTTA
22	20353556	20353575	4	AAAC
22	20354654	20354669	4	ATTT
22	20374713	20374727	3	TTG

Important: Simulator currently set on GangSTR Bed File settings
Currently the main function has a calculation “startposition-1” when reading and “startposition+1” when working with GangSTR bedfiles.
line nr 248 #-1 , when not working with gangstr-bedfiles.
line nr 248 and 360 -1 , when working with gangstr-bedfiles.
line nr 269 and 360 # +1, when not working with gangstr-bedfiles.
!It is important to be aware of the meaning of the start-position in the bed file one uses, and adapt the code if necessary!

These expositions and further more detailed ones are available on the GitHub of the MonSTR Tool. See https://github.com/DamarisLa/STRsimulator, which is currently not public, but access can be assigned (s1810458011@students.fh-hagenberg.at)

2.3.1. Implementation MonSTR (Manipulation ON STRs)


Per Record in Fasta
#Chromosomes
#1
#2
Second run of that record/Chromosme (diploid)
#2
First or second run
#1
Frist Run of this Record/Chromosome (haplotype or Nr1)
by chance lower homozygosity-value
homozgous means completely similar. No different Mutations etc.
no
General
for each entrance in bedile
Find correct start and end
if pattern found
no
yes
yes
by chance manipulate STRs in region
by chance mutate each base in STR region by lower chance insert or delete
change STR region through new version replacement. memorize offset. Safe bedfile change.
next record
memorize
get STR-Sequence from dictionary
get entrance
provide entrance
safe in dictionary new STR - Sequence

***Picture 03**: Flowchart of MonSTR*

In Picture 03 the MonSTR implementation is represented. The MonSTR opens a fasta file record-wise. Depending if the setting was chosen to be haploid or diploid, either only the main function gets run once for that chromosome or twice. In the diploid version, run one is the same as the haploid run, the second run contains another condition that decides randomly, in a range according to the settings, if the script for copying the firsts chromosomes content (homozygous) or the main tree is re-opened (heterozygous).

In the main branch the bed file entrances of the matching chromosome are computed linearly: First the start- and end position assigned by the bed file plus the current offset have to be confirmed as the correct start and end of the pattern. Here the start position is getting checked. If the start position is not where expected, stepwise one step to the left and right of that startbase is checked about whether the following few bases are equal to the known motif. If not the next the positions one further away in both directions are getting checked, and so on. If the start is found and has at least two repeats of the motif following, the length gets investigated, to determine the correct end.

For sorted GangSTR-Region Files this applies to 99%. If the bed file is sorted and not overlapping this check is nearly unnecessary, however if needed this search is as successful, as that the next region is again completely corrected and not an issue. In all test runs, an earlier error had no influence on offset and following regions because of this start-end-confirmation process.

When Start and End are approved, the pattern gets again, according to the settings by chance manipulated: first in STR expansion size, randomly in increase or decrease (currently by plus fife repeats and or minus maximum the whole read length of known pattern). Next the mutation follows. The mutation occurs as well by the given chance, however a base gets substituted correlating to the theory of transition and transversion. A base transition is far more likely (around 10x more likely) which is resembled in our mutation code. Inserts and deletions are on the other hand x times more unlikely, here given by the assigned parameter. If a base has to get inserted according to calculation the inserted base is picked completely randomly.

Final step is

- to write the changes into a bed file which will be important for the evaluation and
- of course the generation of the new manipulated assembly genome to a fasta.

The exact history of the development of the tools is explained in the README.MD in the OldVersion File in Github.

The Folder BedFileParser contains parsers that reformat Bed Files into Tool-specific Files needed by several STR-Tools compared.

These parsers are as well a necessary part of the pipeline.

2.3.2. Testing

The testing of the simulator occured by debugging several regions directly. Later when implementation was complete, a second program was written, that reads the coordinates of the modified bedfiles (after simulation) and then opens the sequences from those regions in modified genome fasta. When the pattern described was not similar to the sequence in the modified fasta-file a counter was raised. As only a zero count implementation was good enough, the MonSTR had to be adjusted a few times. Critical regions had to be printed and analyzed. Several minor errors could be found and solved. Finally the biggest issue the implementation showed was the necessity of a sorted bed file and one without overlapping regions. When changing a region the offset to the original coordinate is calculated and considered for the later positions. Due to the design to go through to the coordinates from low to high, it is given that if a position was changed and an offset added, the next position has to occur later in the coordinate system. Therefore the second region cannot be overlapping or even have an earlier start coordinate than the previously changed regions. Otherwise it is difficult till impossible to find the second region and causes an invisible error, mostly in the form of no modification at all, or the drift to a single repeat pattern nearby.

Most errors disappear if the bed file is sorted. The issue with the overlapping region was not further investigated and solved as we had to limit our testing anyway to an easily evaluable amount. The gangstr bed file mentioned counts 832380 repetitive regions. The randomly chosen 20 regions per chromosome easily avoided overlaps. Overlaps are anyway impossible to find both in the same genome, they only result from the polymorphic nature looking at more than one organism.

Most debugging occured on Chromosome 22, which is the shortest Chromosome, with a bed file containing only around 6000 regions. MonSTR takes for Chromosome and this bed file a few minutes on a local installation and around 9 minute in the cluster. As the entire genome is significantly larger, the runtime popped up as another problematic factor. Running the full unshorted bed file on the full genome can take up to two days (no matter if locally or in the cluster).

Even though the implementation is linear to quadratic, the bottleneck was finally narrowed down to the fasta file reader and writer. Future improvements would be a swap of that module or more importantly the threading of the process, so that several chromosomes can be run parallel.

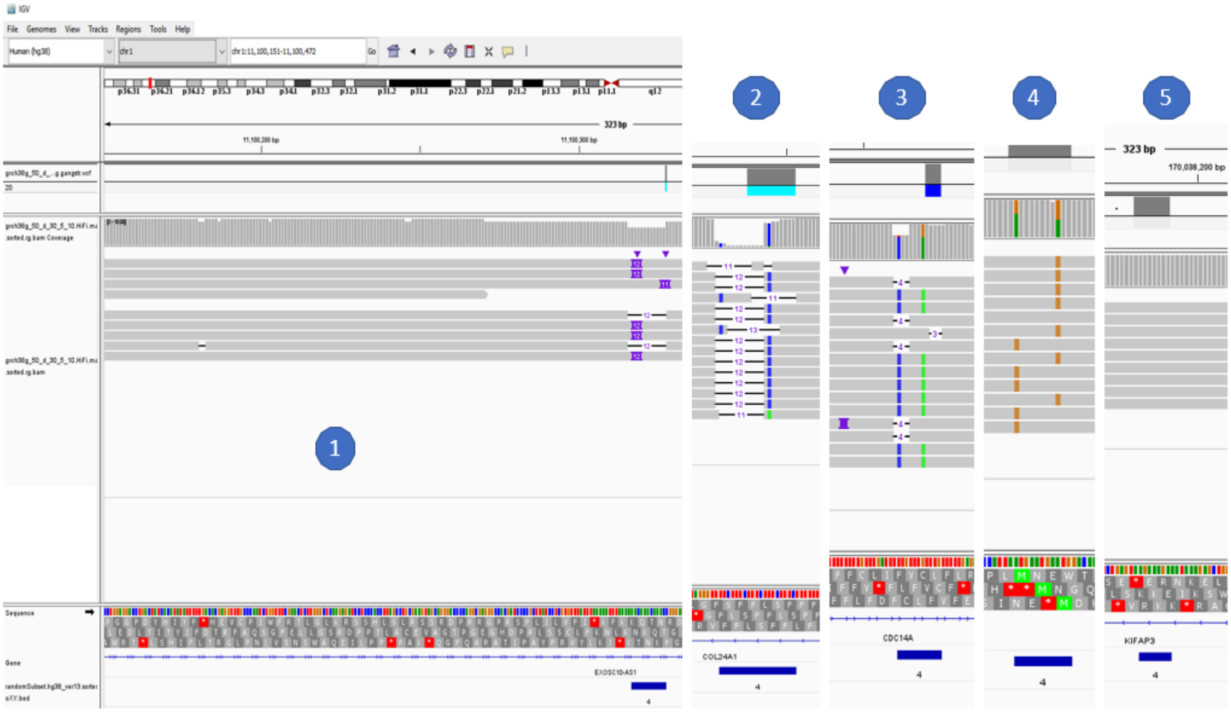
For our use case with 440 regions MonSTR works fast enough, with only a few minutes.

After all immediate detectable issues were found, further analysis required real scenarios and had to be run in the cluster.

2.3.3. Analysing

The simulator was run with different settings in the cluster, following a sequence read simulation with Mason into short reads. The data was mapped, prepared and ran on GangSTR, one of STRCalling tools.

As the data was no patient data a download of the results was helpful to investigate the bed file changes directly. IGV Browser (Robinson et al., 2011) enables opening and visualizing bed-, vcf-, indexed BAM files as well as the reference genome at the same time (see Picture 04). All files show their sequences aligned to each other. It's possible to click through all the bed file events and observe these regions in all other files. The bam file shows in the IGV Browser inserts (purple) and deletions (black lines). Hence correlating to the setting of MonSTR, the bed file regions should show now for example (see runtable) around 50% of repeat regions with an insert or deletion of larger nature (STR), in 30% of the regions these changes should only be visible in around half of the reads (homozygous) and in the other 70% the insertions and deletions should cover all or nearly all of the reads (heterozygous). The MonSTR simulations were quite successful. All possible changes could be found in the generated chromosomes.


IGV
File Genomes View Tracks Regions Tools Help
Human (hg38)
chr1
chr1:11,100,151-11,100,472
Go
p36.31 p36.21 p36.12 p35.3 p34.3 p34.1 p32.3 p32.1 p31.2 p31.1 p22.3 p22.1 p21.2 p13.3 p13.1 p11.1 q12
323 bp
11,100,200 bp
11,100,300 bp
grch38g_50_d_...g.gangstr.vcf
grch38g_50_d_30_5_10.HiFi.m
.sorted.rg.bam Coverage
grch38g_50_d_30_5_10.HiFi.m
.sorted.rg.bam
Sequence
Gene
randomSubset.hg38_ver13.sorted.XY.bed
EXOSC10-AS1
COL24A1
CDC14A
KIFAP3
170,038,200 bp
1
2
3
4
5

Picture 04: *IGV Browser Example on Test Run 50% STR Chance, diploid, 30% homozygous, 5%Mutation, 10x less Indels than Mutations. (1) IGV Browser (from top to bottom): Upper rus are for settings. Reference Genome can be loaded, Reference Chromosome chosen. To see STRs zoom in would be necessary. The next line shows where abouts in the chromosome one is, underneath that coordinates that give orientation. Then after the first big line the loaded Files start. Top file here is the VCF File from GangSTR run. The second file is the bam-file of the run. Third in the lower part of the picture are the reference genome, and below that the Bed File just as fat blue stripes. Clicking on the Bed File it is activated. By pressing "ctrl +F" is navigating to the next region to the right, "ctrl+B" navigation to the next region to the left. This means all files move on to the next region. In the bam-file one can see the exact observed change, in the VCF grey if nothing is found, light blue if homozygous, dark blue if heterozygous. In (1) there are 2 reads with deletions of the STR region, and 6 reads with an STR insert. As well, between the top 3 and lower 5 is a sequence gap. This region could have a heterozygous insert/delete of a STRs, the VCF (GangSTR) observes the change as homozygous. (2) In all reads of the BAM File show a deletion of 12 bases (so a 3 base-motif*4 or a 4 base-motif*3), VCF detected that as well successfully as a homozygous STR modification. (3) This is a tricky one. There are two substitutions on one allele, on the other one is a STR deletion. VCF registered it but apparently as a heterozygous STR. To the left is one purple position, which is an insert of two bases "TC" (that can be found by clicking on it), this appears only on one read, which leads to the conclusion that it must be an sequencing error. (4) Here are two heterozygous SNPs, and no STR change, VCF observed no change. (5) no change at all (should happen in 50% of the cases relating to the STR changes.*

2.4. Testing of STR Callers

2.4.1. Concept

To analyse the different STR Callers we chose three different pathways:

- **Simulated data**:
  - Reference Genome Grch 38 (.fasta) has to be manipulated.
  - Region File (.bed) supplies coordinates of STRs in the Reference Genome. The region file used for all simulations contained 440 regions. These regions were 20 randomly selected regions per Chromosome. Chromosomes X and Y were excluded, they are more difficult to detect.

    This 440 position bed file was the input region file, for the simulation, and for all STR Callers that require a region file.

    The same region file served later as a template for a region file that got blindfolded on six of the 20 regions. Total number of regions: 308 of 440 regions. This blind folded file was only used as an input bed file if required by the STR Callers in additional runs, with the purpose is to figure out if a caller is capable of de novo detection.
  - Further parameters were defined (see Table 02, and see Implementation) to set what kind of modification with which likeliness have to be executed.

    All simulation runs are only getting in around 50% of the regions manipulated.

    Manipulation occurs with a decrease of maximum the STR length, or an increase of maximum 5 additional repeats.
  - All changes compared to the old regions will be saved in a new Region File. This file is only required in the final analysis.
  - The modified genome, with those known changes, will now be available as an artificial “probe”.

***Table 02**: Defined adjustments to the genome: The bedfile was limited to 440 chosen region entrances.*

Template Genome - Bedfile	Chance of STR change (in %)	Diploid (yes/no)	Percent Homozygous (in %)	Chance for substitution (in %)	Indels: in Times rarer than substitution
GrCh 38 - GangStr hg38_ver13*	50	yes	30	0	0
GrCh 38 - GangStr hg38_ver13*	50	yes	30	1	10
GrCh 38 - GangStr hg38_ver13*	50	yes	30	5	10

*The 440 regions consist of 20 randomly selected regions per Chromosome.Chromosomes X and Y were excluded.

- **Real data GIAB HG002**:

For testing real data behavior, the GIAB HG002 was converted into tool input formats and run together with the full bed file hg38_ver13.bed file of GangSTR, if possible. If a tool was not acceptable for that file, then it was substituted by the hg38_ver13 bed file with only 440 positions. Which happened parently in one case. Some tools do not require any bedfiles.

- **Forensic data**

This is an additional task that will be added to the pipeline after the internship. The complexity of the STRs are presented in Table03. “Bracketed Repeat Region” shows which motifs times how many repeats follow each other. In this example are 29 Repeats total (see Locus/Allele), that are between identifying flanking regions. The locus is in Chromsome21. Of D21S11 exist 98 different versions, with repeat variation between 24.2 and 39 repeats in total. Every motif in the repeat-sequence can vary and additionally every single nucleotide change makes a difference, even if it is in the flanking region. One single nucleotide variation is enough to identify two sequences as unequal. The original list of those highly polymorphic loci contains as well the value of their frequencies in different ethnicities across the world and their total frequency. A “simulated” modification is therefore not simple and requires more time than this internship offered and will therefore continue later on.

Table 03: *Example of forensic STR.*

Locus / Allele	D21S11 29
Bracketed Repeat Region	[TCTA]4 [TCTG]6 [TCTA]3 ta [TCTA]3 tca [TCTA]2 tccata [TCTA]11
5’ Flank	AAATATGTGAGTCAATTCCCCAAGTGAATTGCCT
Repeat Sequence	TCTATCTATCTATCTA TCTGTCTGTCTGTCTGTCTGTCTG TCTATCTATCTA TA TCTATCTATCTA TCA TCTATCTA TCCATA TCTATCTATCTATCTATCTATCTATCTATCTATCTATCTATCTA

3’ Flank	TCGTCTATCTAT

2.4.2. STR Callers

Table 04 should give a quick overview of the STR Callers investigated and with that probably as well a large amount of all available STR Calling Tools published, due to the fact that this variant is so particular. All tools are developed for and capable of the detection of STRs. Some showed limitations in file formats, sequencing technologies, developing purpose and other reasons which lead to an exclusion of the tools from further analysis. However they will be part of the paper, just to give an overview about all available tools. Even though we only focussed tools that are able to detect STRs genome-wide, does not mean the other tools might not exactly be what a reader is looking for.

Table 04: *STR Callers with Information about name (and year of publication), Input and Output Formats, their Methodology, if they find Novel STRs (TRUE) or not (FALSE)(including not mentioned (FALSE) => maybe they do), and other Information that might be helpful.*
Red marked STR Callers got excluded due to different reasons.

Tool Name	Input Format	Output Format	Methodology	Finds novel STRs	Good to know
Short Read Tools	**Input Format**	**Output Format**	**Methodology**	**TRUE/ FALSE**	**Good to know**
GangSTR (2019)	bam, ref, bed,	vcf	maximum likelihood => TR lengths. end-to-end method =>sequence alignments	FALSE	Repeats longer than read length processable => extremely long TRs (thousands of bp) will be noisy estimates
HipSTR (2017)	bam, bed, fasta	str-vcf	parametric model, stutter noise profile, hidden markov model => realigning the STRs	TRUE	only TRs smaller than read length
STRetch (2018)	fastq.gz, bed, bam	tsv	Using a decoy-chromosome: mapping with bwa-mem & samtools. count the number of reads mapping to str decoy chromosome bedtools. using paired information. Median coverage calculation calculated with goleft covmed from bedtools => later to normalize counts.	TRUE	analysis running with a bed file reduces chance of de novo detection. => run without bed file == de novo Not 2 STRs at the same position Motives till 6 bp. Only expanded reads development needed for paired end reads genome wide

ExpansionHunter (2019)	bam/CRAM, reference, variant-catalog	JSON, VCF	sequence graphs: graph-based model for realigning, binary alignment/map file, regular expression syntax.	FALSE	variant-catalog (Json file specific for variants to genotype) difficult format for variant catalog
RepeatSeq (2013)	bam, fasta, region file	vcf, repeat seq	first mapping, the sorting, locally realigning using GATK IndelRealigner tool, genotyping using a fully bayesian approach, considering ref length of repeat, the repeat unit size, and the average base quality of the mapped read. usually diploid genotyping model by default.	FALSE	
STRling (2019)	reference, sample.cram/bam	bed tools bin-file	kmer counting to recover mis-mapped STR reads soft-clipped reads to precisely discover the position of the STR expansion in the reference genome	TRUE	STR indexer: bed file creation from the reference -g option can detect expansion from short read sequencing data even if not in reference
Long Read Tools	**Inputformat**	**Outputformat**	**Methodology**	**TRUE/ FALSE**	**Good to know**
TideHunter (2019)	fasta (.gz) / fastq (.gz)	fasta or tabular	fast seed-and-chain algorithm (noisy long-read alignment) consensus calling by multiple seq alignment of detected repeat units	TRUE	PacBio & ONT up to 20% error rate. does not have any limitation of the maximal repeat pattern size tandem-period 2-42
TandemGenotypes myfrith (2019)	refGene.txt, microsat.txt, alignments.maf	tg.txt; plots (extra module)	ranked by priority score(repeats)	FALSE	PacBio & ONT two allele option, and two allele merge option human chimpanzee alignments possible required region files focusses more on in-gene STR detection

NanoSatellite (2019)	fasta, coordinates of TR, motif of TR, bam, index.gz, fast5	(VNTR)	tandem repeat consensus sequence, reference squiggles, delineation, segmentation, tandem repeat unit clustering+tandem repeat sequences composition / squiggle-based tandem repeat length	FALSE	Note on Git: This tool was built to solve issues with Nanopore sequencing data quality, which are nowadays largely solved by better base calling. Additionally, this tool was not updated to keep up with changes in the data formats created by ONT, and may not work with your current input data. To investigate tandem repeats in your data we currently recommend STRique, tandem-genotypes, and TRiCoLOR.
TRiCoLOR (2020)	ref.fasta, bam, bed	vcf	POA (partial order alignment)=> compute low-error consensus seqs fast regex-based approximate string-matching algorithm: for repeat motif and multiplicity of TRs	TRUE	PacBio & ONT needs phased bam files, which requires previous runs of other variant callers. => time consuming extra loop
PacmonSTR (2014)	bam, bed (6 columns), ref, (padding for bed file)	bedfile	reference-based probabilistic approach to identity the TR region, estimate the number of TR elements in long DNS reads 3-step modified smith-waterman approach, expected number of TR-elements with Hidden-Markov Model based method, and then a clustering performed by gaussian mixture models using akaike information criteria and coverage expectations.	FALSE	PacBio & ONT (officially just PacBio) bed file must contain 6 columns
NCRF (2019)	fasta, motif(s)	ncrf- file	aligner (smith-waterman) mit affine gap penalties	FALSE	PacBio & ONT implementation exception:

			dynamic programming matrix		sequences not longer 500kbp and motifs not longer 200bp. needs a script that calls one or a few motifs at a time, with this small countable numbers of motif per run, the tool is very slow when it comes to detection of even 440 regions. It's work and time intensive and maybe not the right method for research use cases.
STRique (2019)	index (fast5 index),	fast5 output with masked reads	str crips-cas-based enrichment strategy for nanopore sequencing combined with an algorithm for raw signal analysis. HMM repeat count mechanism.	FALSE	ONT
STRaglr (2021)	bam	tsv	1. extracting insertions composed of TRs 2. genotyping identified expanded loci.	TRUE	PacBio & ONT

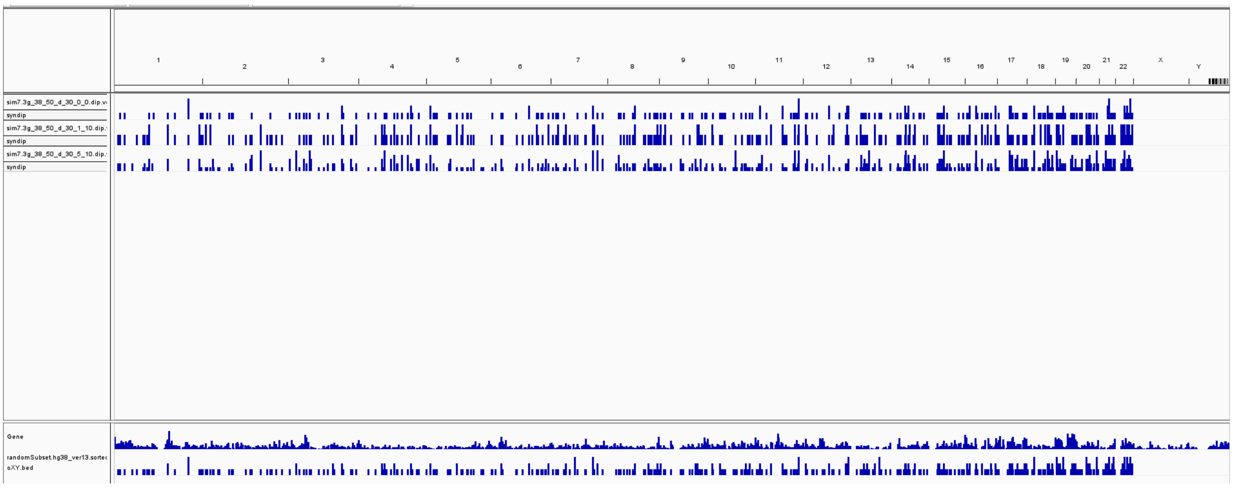
2.5. Evaluation

2.5.1. Concept

The evaluation of all tools is quite tricky as several tools provide different outputs. However most of them print their results into a VCF File, the others in one or the other fasta or tabular format. The VCF evaluation can be summed up. The other versions might need a converter into VCF or a parser for manual evaluation. As the forensic runs only occur after the internship, as well the final evaluation for the publication proceeds later on.

However the VCF evaluation was already addressed with following steps and included into the pipeline:

- The fasta files that were created in the MonSTR, and used as a genome for all simulated test runs, contain a diploid chromosome set.
- This set had to be splitted into two haploid files. The chromosomes represent maternal and paternal DNA.
- Dipcall (Li et al., 2018) is a tool that calls variants observable in one of the haplotype genomes (heterozygous) or both (homozygous) compared to the reference genome. It takes as input files the reference, the paternal and the maternal DNA. In this way, changes that MonSTR induced to the reference genome, which are only changes in known STR regions, will be detected by the Dipcall individually. All called variants are summed up in a VCF file (see Picture 05). The higher a blue line the higher the coverage. Therefore a middle line is heterozygous, a full line homozygous.


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
X
Y
sim7.3g_38_50_d_30_0_0.dip.v
syndip
sim7.3g_38_50_d_30_1_10.dip.
syndip
sim7.3g_38_50_d_30_5_10.dip.
syndip
Gene
randomSubset.hg38_ver13.sortec
oXY.bed

***Picture 05**: Dipcall events compared to input bed file (440 regions) from before MonSTR to after MonSTR and DipCall the output formats VCF. The top 3 tracks are the dipcall VCF. The lowest one is the bed file and in between the reference genome. All regions seem to be in line. As only with a 50% chance was manipulated per haploid set, not all regions can be found by Dipcall as there is no change the variant caller can detect. It seems however a successful transfer from fasta into VCF. The top track hints at an error in the file generation and could ask for a repetition of all runs. In a pipeline this is not an issue, further error analysis is necessary first.*

Final step is an evaluation as the benchmark and the truth dataset are in the same format. Hap.py (Krusche et al., 2019) or RTG (Cleary et al., n.d.) are tools that compare different VCF. Hap.py for instance takes the truth VCF and the VCF from the Tools and could as well accept additionally a confidence bed file and generates a summary CSV with metrics like type of

variant, total truth, total query, recall metric, precision metric, frac_NA metric, TiTV_ration of Truth and Query, het_hom_ratio of total truth and query.
Which of those two tools will be utilized is a further step that has to be tested, and then added to the pipeline. This will be the coming up steps.

3. Discussion and Conclusion

3.1. STR Calling

The STR simulation with MonSTR was for the requirements of our project very successfully. The test runs of so many different tools caused a lot of attention, but provided already interesting insights. Important steps for a pipeline are successfully elaborated and ready for the last missing parts. Reflecting about the finished parts of the project already opens ideas for internal improvements as well as insights into missing approaches. A lot of the STR Callers are quite specific to pattern length, pattern complexity, position in the DNA, amount of motifs detectable at the same time. Therefore a future development of an STR Calling Tool for research, with a full-genome, de novo approach could as well be a goal.

Future Improvements

- full automatisation of evaluation pipeline including read simulation to also automatically measure runtime and memory consumption of STR callers.
- Simulation and evaluation of PCR stutter for STR changes
- Incorporation of more realistic error and mutation models especially with respect to the length.

3.2. General

In a benchmark study a lot of factors come together. Unexpected events can occur at all times, when including so many different datasets and tools. To be still up to date at the end of the process, it's important to stay posted on new publications. In July and August new HG002 datasets and a new STR Caller still lined up to be part of the pipeline.

The bio comp world is a quick moving one. This calls for flexibility and adaptability. A big part of one's time is spent, by reading into new developments, discussing workflows with experienced colleagues, installing and updating tools, writing of observations and tidying up the server environment.

As developers from the whole world share their code, one's documentation of implementations and examinations have to be understandable to members of the bio comp community across the world. Nearly everybody uses similar installation processes and still it is sometimes surprisingly difficult to understand and copy that process. However, a lot of the tools are supported by their developers and communication for questions and wishes via help desks are most of the time possible.

High modularity is very helpful and available. Nearly every step included different tools and modules from developers across the world. We implemented numerous tools and developed ourselves an STR Caller and a few parsers to change every now and then one format into another format. In total around fife parsers were necessary alone to translate the bed files into the formats required by the tools. To work with pipelines, and modular tools, enables one to replace parts. This leads to the flexibility needed that was mentioned before. Another successfactor in this internship became the shell language which is very powerful. A lot of one-line commands allow quick file transformations, which help to save laborious implementation steps. Utilizing them requires training which yet pays out in most of the times.

This project and the internship really gave a great impression into the bio comp world. The research group of Fritz is in regular contact with so many other researchers. Even though there is a little eternity of different topics to improve and develop, due to conferences, workshops, meetings and the before names git repositories with their help desks, the communication paths seem to be short enough to find answers and solutions.

I am thankful to have seen so much.

4. Bibliography

About the BCM-HGSC. (2016, March 4). https://www.hgsc.bcm.edu/about-bcm-hgsc

Casey, R. (n.d.). *What is sequencing coverage?* Retrieved August 24, 2021, from https://thesequencingcenter.com/knowledge-base/coverage/

Characterization of pairwise and multiple sequence alignment errors. (2009). *Gene*, *441*(1-2), 141–147.

Chiu, R., Rajan-Babu, I.-S., Friedman, J. M., & Birol, I. (2021). Straglr: discovering and genotyping tandem repeat expansions using whole genome long-read sequences. *Genome Biology*, *22*(1), 224.

Cleary, J. G., Braithwaite, R., Gaastra, K., Hilbush, B. S., Inglis, S., Irvine, S. A., Jackson, A., Littin, R., Rathod, M., Ware, D., Zook, J. M., Trigg, L., & De La Vega, F. M. (n.d.). *Comparing Variant Call Files for Performance Benchmarking of Next-Generation Sequencing Variant Calling Pipelines*. https://doi.org/10.1101/023754

Danecek, P., Bonfield, J. K., Liddle, J., Marshall, J., Ohan, V., Pollard, M. O., Whitwham, A., Keane, T., McCarthy, S. A., Davies, R. M., & Li, H. (2021). Twelve years of SAMtools and BCFtools. *GigaScience*, *10*(2). https://doi.org/10.1093/gigascience/giab008

Dashnow, H., Lek, M., Phipson, B., Halman, A., Sadedin, S., Lonsdale, A., Davis, M., Lamont, P., Clayton, J. S., Laing, N. G., MacArthur, D. G., & Oshlack, A. (2018). STRetch: detecting and discovering pathogenic short tandem repeat expansions. *Genome Biology*, *19*(1), 121.

Depienne, C., & Mandel, J.-L. (2021). 30 years of repeat expansion disorders: What have we learned and what are the remaining challenges? *American Journal of Human Genetics*, *108*(5), 764–785.

De Roeck, A., De Coster, W., Bossaerts, L., Cacace, R., De Pooter, T., Van Dongen, J., D'Hert, S., De Rijk, P., Strazisar, M., Van Broeckhoven, C., & Sleegers, K. (2019). NanoSatellite: accurate characterization of expanded tandem repeat length and sequence through whole genome long-read sequencing on PromethION. *Genome Biology*, *20*(1), 239.

Dohm, J. C., Peters, P., Stralis-Pavese, N., & Himmelbauer, H. (2020). Benchmarking of long-read correction methods. *NAR Genomics and Bioinformatics*, *2*(2), lqaa037.

Dolzhenko, E., Deshpande, V., Schlesinger, F., Krusche, P., Petrovski, R., Chen, S., Emig-Agius, D., Gross, A., Narzisi, G., Bowman, B., Scheffler, K., van Vugt, J. J. F. A., French, C., Sanchis-Juan, A., Ibáñez, K., Tucci, A., Lajoie, B. R., Veldink, J. H., Raymond, F. L., … Eberle, M. A. (2019). ExpansionHunter: a sequence-graph-based tool to analyze variation in short tandem repeat regions. *Bioinformatics* , *35*(22), 4754–4756.

Fan, H., & Chu, J.-Y. (2007). A brief review of short tandem repeat mutation. *Genomics, Proteomics & Bioinformatics*, *5*(1), 7–14.

Frequently Asked Questions. (n.d.). Retrieved August 23, 2021, from https://www.ncbi.nlm.nih.gov/grc/help/faq/

Fungtammasan, A., Ananda, G., Hile, S. E., Su, M. S.-W., Sun, C., Harris, R., Medvedev, P., Eckert, K., & Makova, K. D. (2015). Accurate typing of short tandem repeats from genome-wide sequencing data and its applications. *Genome Research*, *25*(5), 736.

Gao, Y., Liu, B., Wang, Y., & Xing, Y. (2019). TideHunter: efficient and sensitive tandem repeat detection from noisy long-reads using seed-and-chain. *Bioinformatics* , *35*(14), i200–i207.

Genome in a Bottle. (n.d.). Retrieved August 23, 2021, from https://www.nist.gov/programs-projects/genome-bottle

Gettings, K. B., Borsuk, L. A., Steffen, C. R., Kiesler, K. M., & Vallone, P. M. (2018). Sequence-based U.S. population data for 27 autosomal STR loci. *Forensic Science International. Genetics*, *37*, 106–115.

Giesselmann, P., Brändl, B., Raimondeau, E., Bowen, R., Rohrandt, C., Tandon, R., Kretzmer, H., Assum, G., Galonska, C., Siebert, R., Ammerpohl, O., Heron, A., Schneider, S. A., Ladewig, J., Koch, P., Schuldt, B. M., Graham, J. E., Meissner, A., & Müller, F.-J. (2019). Analysis of short tandem repeat expansions and their methylation state with nanopore sequencing. *Nature Biotechnology*, *37*(12), 1478–1481.

Harris, R. S., Cechova, M., & Makova, K. D. (2019). Noise-cancelling repeat finder: uncovering tandem repeats in error-prone long-read sequencing data. *Bioinformatics* , *35*(22), 4809–4811.

Highnam, G., Franck, C., Martin, A., Stephens, C., Puthige, A., & Mittelman, D. (2013). Accurate human microsatellite genotypes from high-throughput resequencing data using informed error profiles. *Nucleic Acids Research*, *41*(1), e32.

History. (n.d.). Retrieved August 17, 2021, from https://www.bcm.edu/about-us/mission-vision-values/history

History Of Innovative Medical Research. (2019, May 1). https://www.tmc.edu/about-tmc/history/

Holtgrewe, M. (2010). Mason – A Read Simulator for Second Generation Sequencing Data. *Technical Report FU Berlin*. http://publications.imp.fu-berlin.de/962/2/mason201009.pdf

Koboldt, D. C. (2020). Best practices for variant calling in clinical sequencing. *Genome Medicine*, *12*(1), 91.

Krusche, P., Trigg, L., Boutros, P. C., Mason, C. E., De La Vega, F. M., Moore, B. L., Gonzalez-Porta, M., Eberle, M. A., Tezak, Z., Lababidi, S., Truty, R., Asimenos, G., Funke, B., Fleharty, M., Chapman, B. A., Salit, M., Zook, J. M., & Global Alliance for Genomics and Health Benchmarking Team. (2019). Author Correction: Best practices for benchmarking germline small-variant calls in human genomes. *Nature Biotechnology*, *37*(5), 567.

Li, H. (2018). Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* , *34*(18), 3094–3100.

Li, H., Bloom, J. M., Farjoun, Y., Fleharty, M., Gauthier, L., Neale, B., & MacArthur, D. (2018). A synthetic-diploid benchmark for accurate variant-calling evaluation. *Nature Methods*, *15*(8), 595–597.

Li, H., & Durbin, R. (2009). Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* , *25*(14), 1754–1760.

Li, H., & Durbin, R. (2010). Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics* , *26*(5), 589–595.

Logsdon, G. A., Vollger, M. R., & Eichler, E. E. (2020). Long-read human genome sequencing and its applications. *Nature Reviews. Genetics*, *21*(10), 597–614.

Luo, R., Wong, C.-L., Wong, Y.-S., Tang, C.-I., Liu, C.-M., Leung, C.-M., & Lam, T.-W. (n.d.). *Clair: Exploring the limit of using a deep neural network on pileup data for germline variant calling*. https://doi.org/10.1101/865782

Mitsuhashi, S., Frith, M. C., Mizuguchi, T., Miyatake, S., Toyota, T., Adachi, H., Oma, Y., Kino, Y., Mitsuhashi, H., & Matsumoto, N. (2019). Tandem-genotypes: robust detection of tandem repeat expansions from long DNA reads. *Genome Biology*, *20*(1), 58.

Mousavi, N., Shleizer-Burko, S., Yanicky, R., & Gymrek, M. (2019). Profiling the genome-wide landscape of tandem repeat expansions. *Nucleic Acids Research*, *47*(15), e90–e90.

Paired-End vs. Single-Read Sequencing. (n.d.). Retrieved August 24, 2021, from https://emea.illumina.com/science/technology/next-generation-sequencing/plan-experiments/paired-end-vs-single-read.html

Pflanzer, L. R., & Lee, S. (2018, April 3). *Our DNA is 99.9% the same as the person next to us — and we're surprisingly similar to a lot of other living things*. Business Insider. https://www.businessinsider.com/comparing-genetic-similarity-between-humans-and-other-things-2016-5

Pightling, A. W., Petronella, N., & Pagotto, F. (2014). Choice of Reference Sequence and Assembler for Alignment of Listeria monocytogenes Short-Read Sequence Data Greatly Influences Rates of Error in SNP Analyses. *PloS One*, *9*(8). https://doi.org/10.1371/journal.pone.0104579

Poplin, R., Ruano-Rubio, V., DePristo, M. A., Fennell, T. J., Carneiro, M. O., Van der Auwera, G. A., Kling, D. E., Gauthier, L. D., Levy-Moonshine, A., Roazen, D., Shakir, K., Thibault, J., Chandran, S., Whelan, C., Lek, M., Gabriel, S., Daly, M. J., Neale, B., MacArthur, D. G., & Banks, E. (n.d.). *Scaling accurate genetic variant discovery to tens of thousands of samples*. https://doi.org/10.1101/201178

Quinlan, A. R., & Hall, I. M. (2010). BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* , *26*(6), 841–842.

quinlan-lab. (n.d.). *GitHub - quinlan-lab/STRling: Detect novel (and reference) STR expansions from short-read data*. Retrieved August 24, 2021, from https://github.com/quinlan-lab/STRling

Rajan-Babu, I.-S., Peng, J. J., Chiu, R., IMAGINE Study, CAUSES Study, Li, C., Mohajeri, A., Dolzhenko, E., Eberle, M. A., Birol, I., & Friedman, J. M. (2021). Genome-wide sequencing as a first-tier screening test for short tandem repeat expansions. *Genome Medicine*, *13*(1), 126.

Raz, O., Biezuner, T., Spiro, A., Amir, S., Milo, L., Titelman, A., Onn, A., Chapal-Ilani, N., Tao, L., Marx, T., Feige, U., & Shapiro, E. (2019). Short tandem repeat stutter model inferred from direct measurement of in vitro stutter noise. *Nucleic Acids Research*, *47*(5), 2436–2445.

Riman, S., Iyer, H., Borsuk, L. A., & Vallone, P. M. (2019). Understanding the behavior of stutter through the sequencing of STR alleles. In *Forensic Science International: Genetics Supplement Series* (Vol. 7, Issue 1, pp. 115–116). https://doi.org/10.1016/j.fsigss.2019.09.045

Robinson, J. T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E. S., Getz, G., & Mesirov, J. P. (2011). Integrative genomics viewer. In *Nature Biotechnology* (Vol. 29, Issue 1, pp. 24–26). https://doi.org/10.1038/nbt.1754

Sedlazeck, F. J., Dhroso, A., Bodian, D. L., Paschall, J., Hermes, F., & Zook, J. M. (2017). Tools for annotation and comparison of structural variation. *F1000Research*, *6*, 1795.

Shin, S. C., Ahn, D. H., Kim, S. J., Lee, H., Oh, T.-J., Lee, J. E., & Park, H. (2013). Advantages of Single-Molecule Real-Time Sequencing in High-GC Content Genomes. *PloS One*, *8*(7), e68824.

Stutter analysis of complex STR MPS data. (2018). *Forensic Science International. Genetics*, *35*, 107–112.

Table Browser. (n.d.). Retrieved August 23, 2021, from https://genome.ucsc.edu/cgi-bin/hgTables

Ummat, A., & Bashir, A. (2014). Resolving complex tandem repeats with long reads. *Bioinformatics* , *30*(24), 3491–3498.

Van der Auwera, G. A., & O'Connor, B. D. (2020). *Genomics in the Cloud: Using Docker, GATK, and WDL in Terra*. O'Reilly Media.

Website. (n.d.). https://www.pacb.com/blog/the-evolution-of-dna-sequencing-tools/attachment/short-reads-and-hifi-reads-genome-assembly-comparison/

What does the fact that we share 95 percent of our genes with the chimpanzee mean? And how was this number derived? (n.d.). Retrieved August 30, 2021, from https://www.scientificamerican.com/article/what-does-the-fact-that-w/

Willems, T., Zielinski, D., Yuan, J., Gordon, A., Gymrek, M., & Erlich, Y. (2017). Genome-wide profiling of heritable and de novo STR variations. *Nature Methods*, *14*(6), 590–592.