Detection of Genomic Inversion from Single End Read

Pankaj Ghimire

B.S., Tribhuvan University, 2009

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

At the

University of Connecticut

2012

APPROVAL PAGE

Masters of Science Thesis

Detection of Genomic Inversion from Single End Read

Presented by

Pankaj Ghimire, B.S.

Co-Major Advisor_____
Ion Mandoiu

Co-Major Advisor_____
Yufeng Wu

Associate Advisor_____
Reda A. Ammar

Associate Advisor_____
Dong-Guk Shin

University of Connecticut
2012

ACKNOWLEDGEMENTS

This work would not have taken this shape, if there were lack of support from special people. I would like to express my gratitude to my advisors, Professor Ion Mandoiu and Professor Yufeng Wu, who helped and guided me in all aspect of this thesis work. Their all time inspiration and guidance were vital to this work. I am also thankful to my associate advisors Professor Dong-Guk Shin and Professor Reda A. Ammar for their precious time and remarks. I am grateful to Sahar Al Seesi and Jin Zhang for their continuous advice throughout the project.

I would like to thank Computer Science and Engineering Department, my lab colleagues, all my friends, and my family for providing unconditional support and impetus to come across this phase.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

ABSTRACT

Structural Variations (SVs) are genomic rearrangements that include both copy-number variants, such as insertion, deletions, duplications and balanced variants like inversion and translocations. These SVs are getting more attentions for research and investigation because of their role on human phenotype, genetic diseases and genomic rearrangements. Evolution of Next-generation Sequencing has provided golden opportunities to investigate these variants and make their wider and clear spectrum in human genome. This investigation includes identification of type of SVs and their breakpoints at base pair level. For their effective identification and breakpoint resolution, many techniques are devised mainly based on paired end read. With relatively low cost and high efficiency different platforms including ION TORRENT, Illumina can generate high throughput Single End reads. In this thesis we provide a novel approach based on Single End reads to detect genomic inversions in human genome. We also compare our approach with existing methods based on paired end reads and show that our approach is competitive in terms of sensitivity and precision at relatively low coverage for detection of breakpoints of genomic inversion.

CHAPTER 1

INTRODUCTION

The successful completion of Human Genome Project opened up a new avenue for the comparative study of human genome by providing '3 billion bases' reference genome. After this, several comparative genomic studies are conducted which have shown that there are large scale of different type of Structural Variations (SV) in human genome ranging from single base to several megabases. These SV may cause the copy-number to be varied with respect to reference called Copy Number Variant like deletions, duplication (tandem duplications and interspersed duplication)and Insertions(novel sequence insertion and mobile element insertion) or may not change copy-number but change the order and orientation of sequences with respect to reference called Copy Number Invariants. This includes Inversion and Trans-location of gene sequence.

Similar to other human genomic alterations, SV can have impact on human phenotype by disrupting the usual DNA. Diseases can be a consequence of this ability to interfere with gene function, protein function, and gene expression. Therefore, identifying the type of SV and finding their precise location of occurrence (breakpoints) is cardinal in genomic research. If there exists problem in resolving breakpoints even with few bases it will be highly ambiguous to make a conclusion whether SVs falls in regulatory region or in overlapping exons which leads to delusion of functional impact of SVs. These SVs can be detected only when DNA sequences are compared with standard sample called reference. Two techniques have been used to identify SVs in the human genome: Technique based on hybridization (array comparative genomic hybridization (aCGH) and

Single Nucleotide Polymorphism array technology) and Technique based on end sequence profiling (ESP), also called paired-end mapping [5].

Hybridization techniques test the relative frequencies of probe DNA segments between two genomes [6]. Although by considering allelic ratios at heterozygous sites, they are able to detect CNVs like insertions and deletions, they can only detect handful of balanced variant like, inversions [7]. Newer techniques and methods are being devised for detection of Structural Variations (CNV and Inversion and trans-location) with emergence of cost effective and high throughput sequencing technologies where two paired reads are generated at an approximately known distance in the donor genome containing SV. Although Sequencing of SV allows us to identify their location of base-pairs and type, finding proper resolution of their breakpoints are still challenging. All the approaches defined and developed to find breakpoints of SV to date basically rely on Pair-End reads. Unfortunately, methods based on Pair-End reads have limitation in breakpoint resolution because of uncertainty in distance between sequenced ends. In this context, we have put forth a novel method for detection of genomic inversions that relies on Single End (SE) reads.

To implement our method, we map SE reads generated from donor genome containing genomic inversion enabling ungapped alignments with reference genome. If a SE read is hovering a junction of inversion in one direction, we get the partial alignment of same read over other corresponding junction of inversion in opposite direction. Alignments of all such reads are processed based on their mapping location, orientation, number of softclipped bases, and number of mapped bases to infer the candidate breakpoints of inversions. The list of candidate breakpoints is filtered in the second phase to remove

false positives and final list of breakpoints are generated. In this thesis we present our pipeline, results analysis based on simulated data and comparison with existing methods that are being used for the inversion detection in the sections 3.3, 3.4 and 3.5

.

CHAPTER 2

BACKGROUND

2.1 Basic Definition

DNA: Deoxyribose nucleic acid or DNA is the most fascinating molecule in the entire world. Its massive amount of base pairs consisting of a varying number of genes (per organism) contains hereditary information that is used in the development and functioning of an entire organism. In fact, it is hard to imagine life or living without DNA being involved. The double helix structure that Watson and Crick discovered in the nineteen fifties holds many more mysteries than any other molecule could ever do; mysteries that are in need of elucidation [8]. This is probably what inspires us every day, in our quest of understanding DNA [1].

Structural Variations: Structural variations used to be defined as all genomic rearrangements that are bigger than one thousand base pairs (>1 kb) [11, 12]. Since our detection techniques have further developed, the current definition can be adjusted to include all variations bigger than 50 base pairs [11]. Structural variations in its broadest sense can even simply be defined as all genomic variations in an organisms genome that are bigger than one base pair [9]. Several different types of mutations fit these two last definitions: deletions, insertions (novel sequence insertions and mobile-element insertions), inversions, duplications (tandem duplications and interspersed duplications), and translocations [9]. The type of rearrangement can be identified by comparing the sequence of someone's DNA sample to the sequence of another DNA sample. Usually, a

reference genome is used in this comparison. However, when trying to identify *de novo* rearrangements, the DNA sequence of the parents is used. *De novo* (or new) rearrangements are structural variations that a child has, but the parents of that child do not have. They are often a result of a rearrangement in the paternal chromosome of the germ cell during meiosis [14].

Structural variations can be divided into several categories. Firstly, they are either recurrent or non-recurrent. Sometimes, rearrangements occur more often in a certain DNA fragment, due to favorable circumstances. They are therefore present in many individuals. These are recurrent structural variations, meaning that they happen more often. Non-recurrent structural variations on the other hand occur on rare spots in the DNA. Sometimes an individual can even seem to be the only one with a certain structural variation at a certain spot. Secondly, structural variations are either intrachromosomal or interchromosomal. Rearrangements in one chromosome are named intrachromosomal, while rearrangements between two chromosomes are called interchromosomal. Finally, structural variations can either occur in somatic cells or in germ cells. A rearrangement in a somatic cell only affects the organism in which the rearrangement has happened in. A mutation in a germ cell on the other hand will only have effect on the offspring [1].

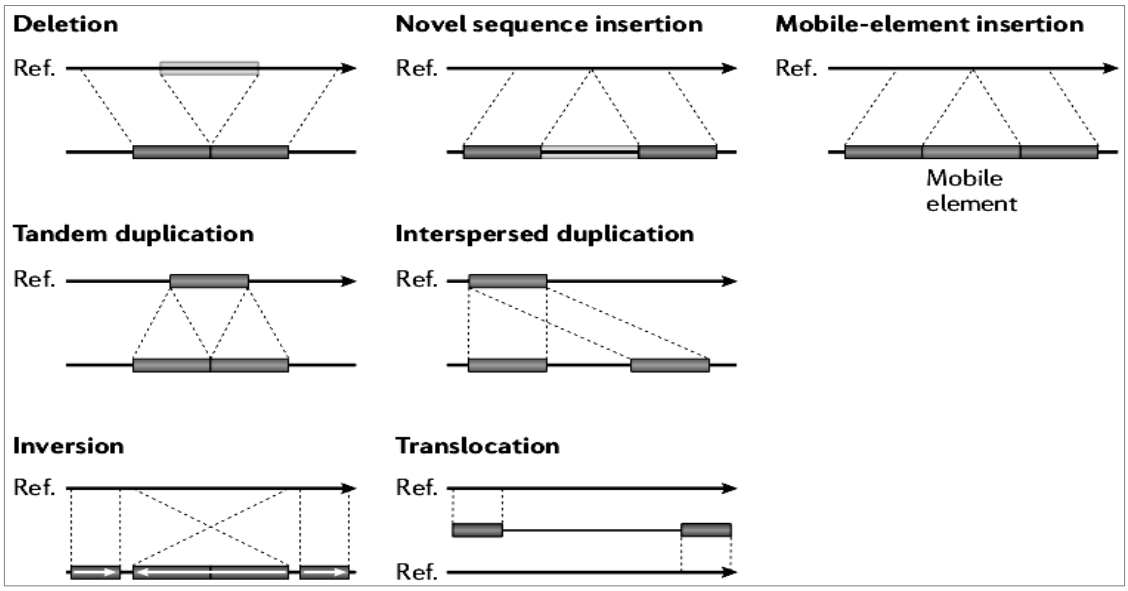

Figure1: figure showing Basic Structural Variations [11]

Deletion*:* Deletion is a type of structural variation which causes loss of bases with respect to reference genome.

Insertion*:* This type variation occurs when there are extra bases in donor genome with respect to reference genome.

Duplication

Segmental duplication or low-copy repeat: A segment of DNA >1 kb in size that occurs in two or more copies per haploid genome, with the different copies sharing >90% sequence identity. They are often variable in copy number and can therefore also be CNVs [15].

***Inversion**:* A segment of DNA that is reversed in orientation with respect to the rest of the chromosome. Pericentric inversions include the centromere, whereas paracentric inversions do not [15].

***Translocation**:* A change in position of a chromosomal segment within a genome that involves no change to the total DNA content. Translocations can be intra- or inter-chromosomal [15].

***Indels**: A*bbreviated combination of **in**sertions and **del**etions. Indels refers to DNA mutations. Indels involving one or two base pairs can have devastating consequences to the gene because translation of the gene is "frameshifted". Indels have a size ranging from 1 base pair upto 50 base pair [15].

Single Nucleotide Polymorphism: A single base substitution of one nucleotide with another observed in the general population at a frequency greater than 1%.

Breakpoints: A breakpoint is the location at either end of structural variations.

2.2 SAM File Format

SAM format is TAB-delimited. Headers are started with @ sign and there are other components in the following order.

1. Query/template/pair Name 2. FLAG (bitwise FLAG) 3. Reference Name 4. Position (1-based left most position) 5. Mapping Quality (In Phred Scale) 6. CIGAR (String) 7.Mate Reference Name (= if same as Reference Name) 8. Mate Position (1-based Position) 9. Insert Size 10. Query sequence 11. Query Quality 12. Variable Optional fields

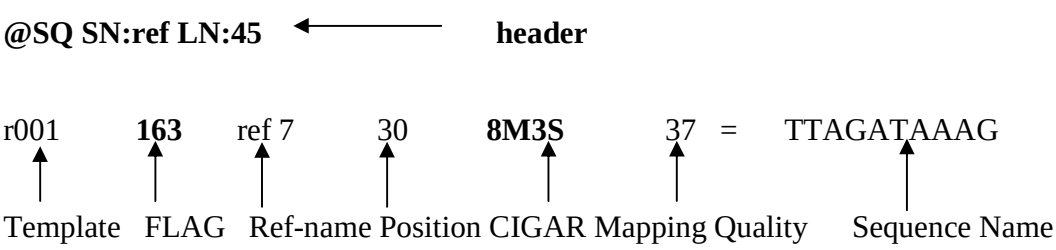

Figure 2: figure showing SAM Sequence

Each bit in flag is defined as

FLAG	***Description***
0x1	templates having multiple segments in sequencing
0x2	each segment properly aligned according to the aligner
0x4	segments unmapped
0x8	next segments in the template unmapped
0x10	SEQ being reverse complemented

0x20	SEQ of the next segment in the template being reversed
0x40	the first segment in the template
0x80	the last segment in the template
0x100	secondary alignments
0x200	not passing quality controls
0x400	PCR or optical duplicate

CIGAR String represents the following CIGAR Operations

Op	***Description***
M	alignment match (can be a sequence match or mismatch)
I	insertion to the reference
D	deletion from the reference
N	skipped region from the reference
S	soft clipping (clipped sequences present in SEQ)
H	hard clipping (clipped sequences NOT present in SEQ)
P	padding (silent deletion from padded reference)
=	sequence match
X	sequence mismatch

2.3 Structural Variation in Human Genome

Through different scientific studies, it has found that about all human being from around world has 99.9% of identical DNA sequence. Thus it is only the small fraction of genome that constitutes genetic variation between individuals and responsible for phenotypic variation and disease susceptibility [21, 22]. Before the breakthrough of sequencing technology, only the rare change in quantity and structure of chromosome were observed in comparison study of genetic variation which included aneuplodies, rearrangement, heteromorphism and fragile sites. These changes were large (~3 Mb or more) enough to be observed using microscope and thus named as microscopic structural variants. With the advancement of molecular biology along with sequencing technology, new variations such as SNPs, and small (<1kb) insertions, deletions and duplications were observed. After the completion of primary sequence of human genome more tools and techniques were developed that started characterizing human genetic compositions at nucleotide level. Peculiarly, genome-scanning array technologies and comparative DNA-sequence analyses revealed large number of genomic variations that are smaller than microscopic level and larger than those detected by conventional sequence analysis. Those variations are defined as submicroscopic structural variations [21]. Hundreds of submicroscopic copy-number variants (CNVs) and inversions have been described in the human genome with help of those technologies. Figures below shows the number of CNVs and Inversions found and their size distributions

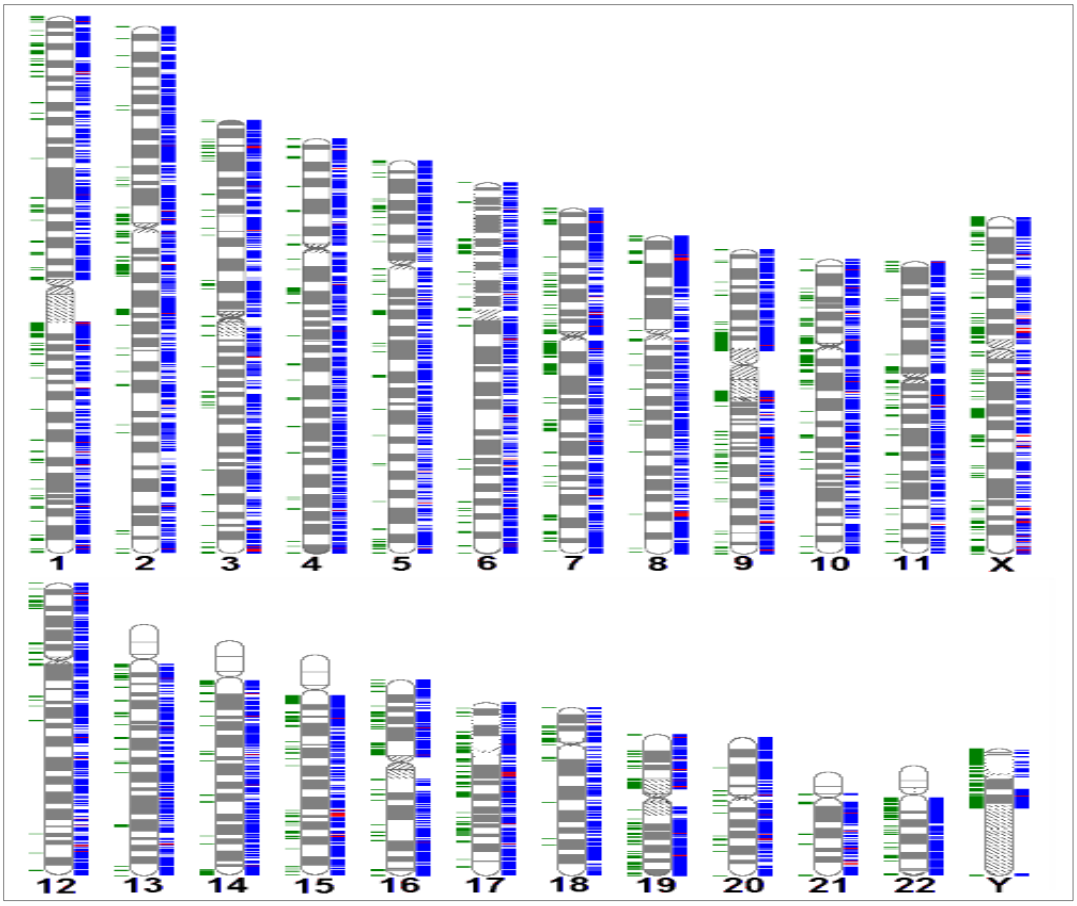

Figure 3: Blue bars indicate reported CNVs; Red bars indicate reported inversion breakpoints; Green bars to the left indicate segmental duplications [21].

STAT	Merged Level	Sample Level
CNVs	21801	610834
Inversions	892	1734

Table T1: Table showing the CNVs and inversions [21].

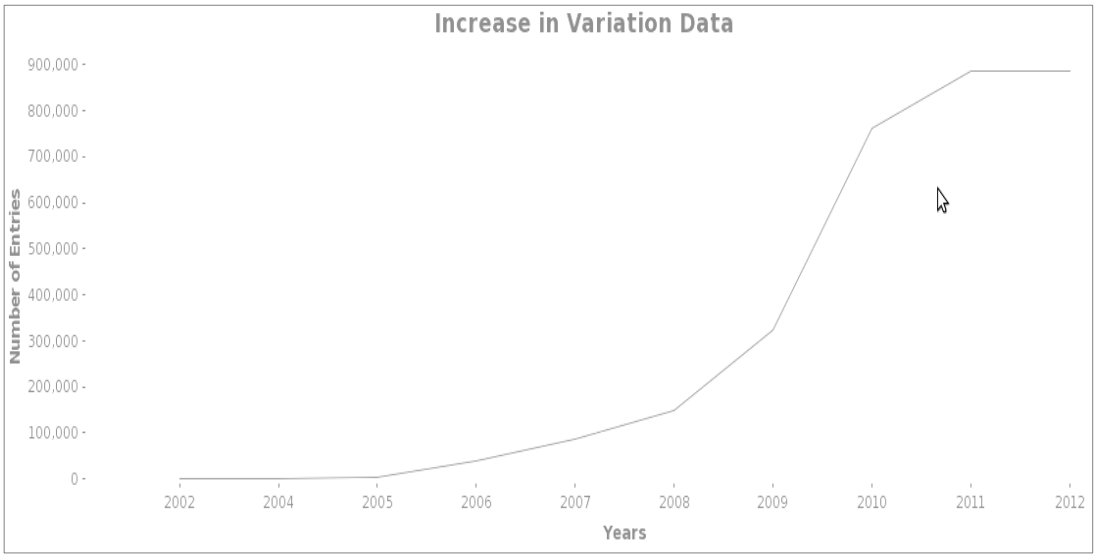

Figure 4: Graph showing the increase in published CNV and InDel data that have been added to the database since the start in 2002 [21]

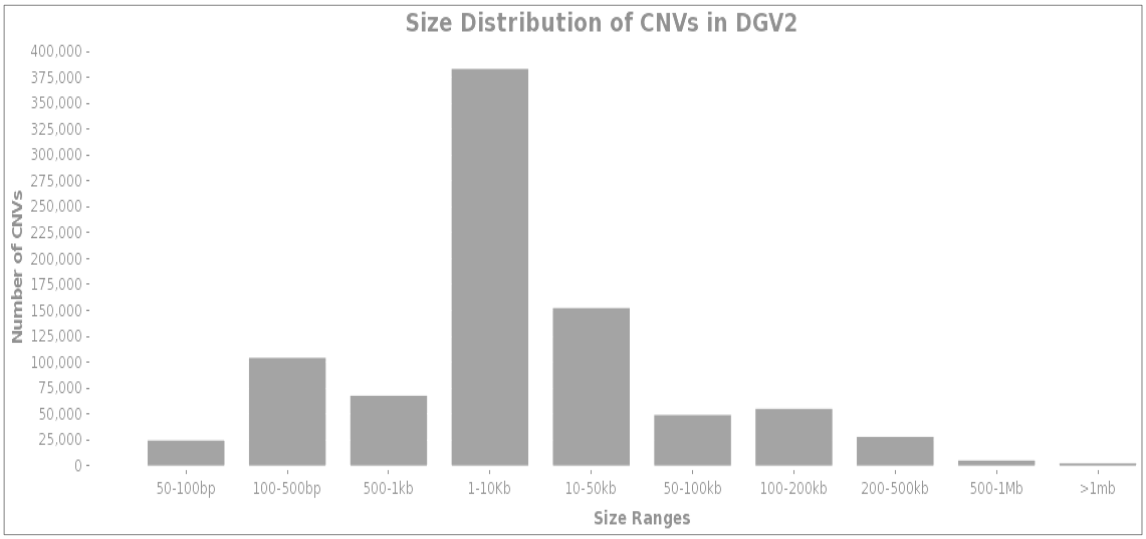

Figure 5: Figure showing graph displays the size distribution of CNVs in the database [21].

Similar to other CNVs, It has long been possible to detect inversions of large chromosomal regions in karyotype level in G-band karyotypes. But, this technique is confined to identification of variants that are several megabases in size, and even significantly larger inversions may not be detected if the inverted segment leads to slight difference in the banding pattern. From the very beginning of chromosomal study, inversions are always variants of interest but they were not identified for clinical significance [16]. Inversions are the most common human constitutional karyotype make inversions astonishing as genomic rearrangements is their role in recent primate evolution. Nine cytogenetically visible pericentric inversions were found while comparing the human and chimpanzee genomes [25] and many submicroscopic inverted sequences [26]. The majority of the nine visible inversions occurred along the chimpanzee lineage, but inversions on chromosomes 1 and 18 are specific to the human lineage. This finding implies that inversions are important genomic rearrangement that occurs quite frequently in primate chromosomal evolution. Thus identification of a large number of inversions between closely related species, and signatures of selection associated with these, will shed light on the role of genomic inversion in speciation [27].

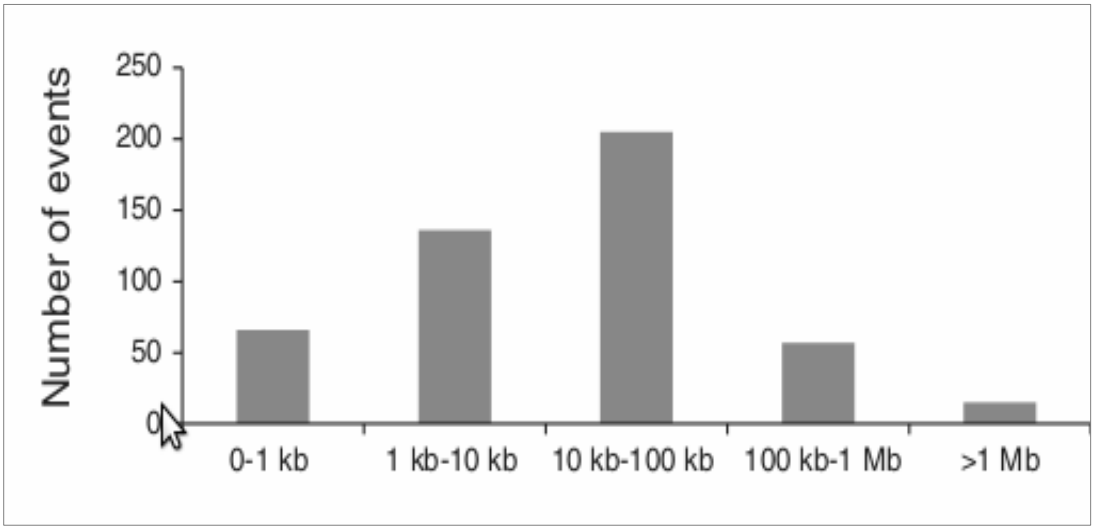

Figure 6: Figure shows that the majority of inversions reported to date are in the 10 to 100 kb size bin [20].

2.4 Role of Structural Variations

Previously SNP (Single Nucleotide Polymorphism) was considered to be the most significant for the variation of genome but later it was found that there exists a structural variation which causes variation in thousands of base-pairs. These types of variant can cover millions of bases of DNA, containing entire genes and their corresponding regulatory regions [21, 26, 28, 29]. Although structural variants in some genomic regions have no distinct and direct phenotypic consequence [21, 26, 28, 29], those in others may influence gene dosage causing genetic diseases. Structural variations can come into play either alone or in combination with other genetic or environmental factors to influence genetic variation and gene functionality [30]. The extents of effects of structural variations on phenotype depend on a combination of the location and the type of structural variation. The location is probably the determining factor in defining the consequence of structural variation. Since a mutation in so-called ‘junk DNA’ might not

even have any consequences [1]. Firstly, structural variations can occur in the regulatory sequence of a gene. Although these regulatory sequences are in non-coding region of DNA they can influence the gene expression. Thus gene expression could change if the promoter sequence of a certain gene changes. A deletion or inversion of (a part of) the regulatory sequence can cause a decrease in gene expression. Insertions can also decrease gene expression when they occur in the promoter. But, if a promoter of an active gene is coincidentally inserted right in front of a relatively inactive gene, an insertion can cause an increase in gene expression [1]. A deletion in the downstream regulatory sequence of *TNFAIP3* is associated with systemic lupus erythematosus [31].

Another instance of a change in phenotype due to a rearrangement in the non-coding DNA sequence is in the non-coding functional RNA, among others: micro-RNA (miRNA). Micro-RNAs are thought to control the activity of approximately 30 percent of all proteins [32]. When a structural variation changes a miRNA, the activity of a protein could change as well. Therefore it is no surprise that micro RNAs have been shown to play important roles in different diseases, such as cancer and immune diseases [32]. A deletion of the miRNA *Dgcr8* in mice results in defects in the synaptic transmission of the pre-frontal cortex, which could give insights in the pathology of human schizophrenia [33].

Structural variations can also occur in genes, even though there is selective constraint against this in germ cells .The effects of these mutations in coding DNA are more likely than of non-coding DNA and can have worse consequences. Seventeen percent of all rearrangements for example directly alter gene function [10]. The amount of genes affected by a variation clearly increases with an increase in size of the variation. This is

especially true for mutations smaller than ten thousand base pairs. Approximately 125 genes are affected by a ten thousand base pair rearrangement [10]. Genes can be affected by structural variations in different ways. Firstly, the gene dosage can be altered. When a person has a third 21st chromosome, he or she will suffer from Down syndrome. Secondly, a gene could be disrupted, by for instance an insertion. This would result in a disrupted non-functional protein. Thirdly, genes that are fused together by a rearrangement can form a new functional protein [9]. An example of this is the BCR-ABL fusion gene that is caused by a translocation and that is found in leukemia patients [32, 33]. A fourth mechanism is the alteration of gene expression due to structural variations. Gene expression can for instance be increased when a gene with low transcription activity will translocate to another promoter of a gene with high transcription activity. A final mechanism is the unmasking of recessive mutations [9]. Rearrangements related to SV can either occur in a germ cell or in a somatic cell; the consequences are totally different. A mutation during meiosis of a germ cell can cause a congenital (and eventually hereditary) disease, while a somatic mutation can contribute to a tumor. SV are thus associated with many different diseases. These range from aniridia to susceptibility to HIV infection to genomic disorders such as the Williams-Beuren syndrome [1, 2, 3, 4].

Structural variations not only have negative effects, but they also seem to have a function. Many deletions for instance (including the deletion of entire genes) have been found to be distributed in the whole genome. Structural variations can thus possibly also play a significant part in genome evolution [34].This might be the cause for the existence of population based differences in structural variations. The UGT2B17 gene for example is

associated with ethnic differences in risk of prostate cancer [9, 12]. Moreover, different populations have different skin colors, eye colors and hair colors which are also contributed by SVs [1].

2.5 Discovery of Structural Variations

Since SVs are important genomic arrangements that have several consequences in phenotype, gene functionality and diseases, their proper discovery is very important in genomic research. Discovery of variations incorporates the processes of variant detection, validation and characterization at the sequence level [11]. In this thesis we explain current methods for discovery of SVs, including experimental approaches using microarrays, single-molecule analysis and sequencing-based computational approaches.

2.5.1 Hybridization based Array Approach

Microarrays based techniques is considered as the first breakthrough in CNV discovery and genotyping. Under this technology two approach are most prevalence: first, array comparative genomic hybridization (array CGH) and second, SNP microarrays. Although both of these techniques are based on inferring copy number gains or losses compared to a reference sample or population they do differ in the details and application of the molecular assays [11]. Even though, they are able to detect structural variations like insertion, deletion significantly, detection of genomic inversions is only handful [6].

Array CGH platforms are based on the technique of comparative hybridization of two labeled samples test and reference to a set of hybridization targets either formed by long oligonucleotides or, historically, bacterial artificial chromosome (BAC) clones. The signal to noise ratio of test to sample is calculated, normalized and presented in log2 scale. This ratio is then used as a proxy for copy number. An increase in log2 ratio indicates the gain in copy number in test with respect to reference, while a decrease in log2 ratio indicates the loss in copy number. An important consideration is the effect of the reference sample on the copy-number profile. For example, when only one sample is

examined, a loss in the reference sample is indistinguishable from a gain in the test sample. To address this issue, a well-characterized reference is vital to make final conclusion of array CGH data [11, 35]. Since early studies of germ line CNVs were based on BAC arrays or low-resolution oligonucleotide platforms, CNVs typically greater than 100 kb were detected [11, 21, 28, 36]. Although initial phase of studies uncovered the high number of CNVs in healthy individuals, corresponding breakpoints of these variations were not sufficiently well-defined to allow accurate assessment of the proportion of the genome altered or its gene content. As this result was overestimation of extend to copy number polymorphism due to large-insert BAC clones[11,36], it was refined by using oligonucleotide microarrays or sequence-based studies of the same DNA samples[11, 37-40].Now a days , Roche NimbleGen and Agilent Technologies are the top provider of whole-genome array CGH platforms which routinely produce arrays with up to 2.1 million (2.1M) and 1M long oligonucleotides (50–75-mers), respectively, per microarray. By setting the requirement of 3-10 consecutive probes's signal to detect CNV, CGH and SNP can detect several hundred CNV in a genome. Due to easy availability of custom, high probe density arrays array CGH platform is replacing traditional karyotyping analysis in clinical diagnostics to find copy-number alterations [11].

Similar to array CGH, **SNP microarray platforms** are also based on hybridization. But they have some key differences to array CGH platforms. First difference is, in SNP array the hybridization is performed on a single sample per microarray, and log-transformed ratios are generated by clustering the intensities measured at each probe across many samples [41, 42, 43]. Second, SNP platforms take advantage of probe designs that are

specific to single-nucleotide differences between DNA sequences, either by single-base-extension methods which is implemented in Illumina platform or differential hybridization implemented in Affymetrix [41, 42, 43]. Moreover, SNP array platform also uses SNP allele-specific probes to increase CNV sensitivity, distinguish alleles and identify regions of uniparental disomy through the calculation of a metric termed B allele frequency (BAF) [11] .Although early SNP array had poor coverage over CNV regions recent arrays (such as the Affymetrix 6.0 SNP and Illumina 1M platforms) have excellent performance because of better SNP selection criteria for complex genomic regions and non-polymorphic copynumber probes(which are examined for log ratios but not BAF) [41,42,46].They are becoming popular than array CGH platform and replacing them gradually in the large-scale discovery of CNVs in a broad variety of populations [11, 22,29,41,42,45,47].

SNP array platform also has disadvantage over array CGH, as SNP microarrays tend to offer lower signal-to-noise ratio per probe than array CGH platforms. This disadvantage become more significant in comparisons of array CGH and SNP platforms in terms of detection of CNVs by a purely ratio-based approach [21, 28, 44]. To validate results and improve confidence of CNV detection some studies combine array CGH and SNP platforms [41, 45, 46].

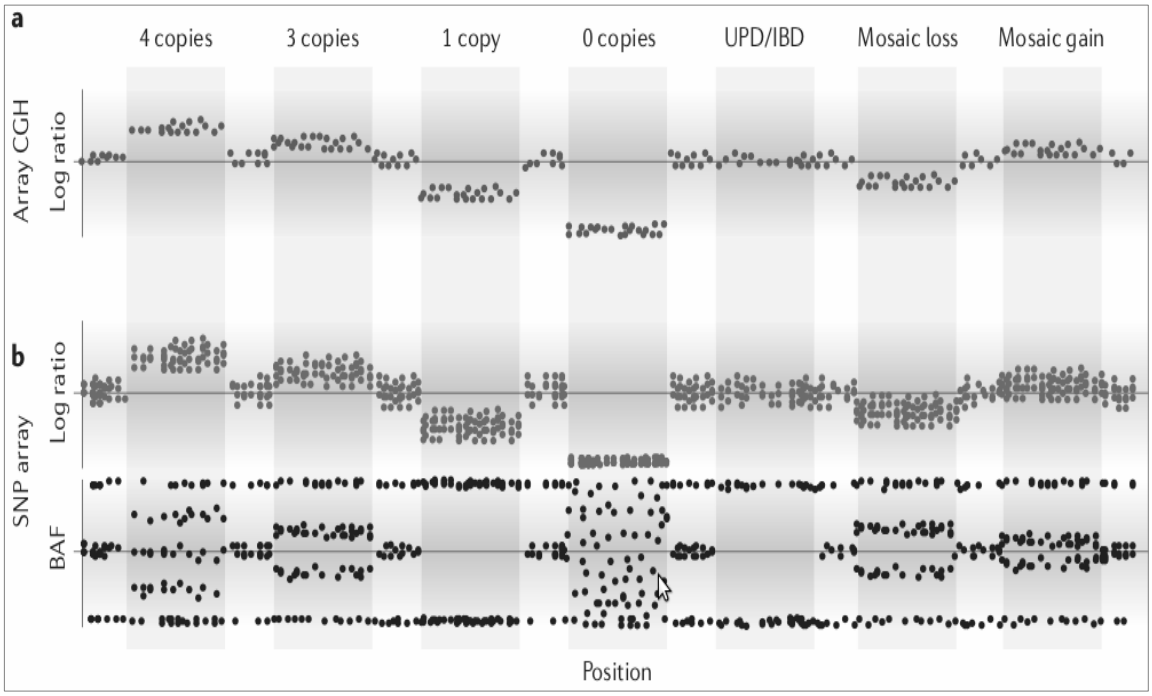

Figure 7: Figure showing log ratio of copy number for array CGH, SNP array platforms and BAF for SNP array platform [11].

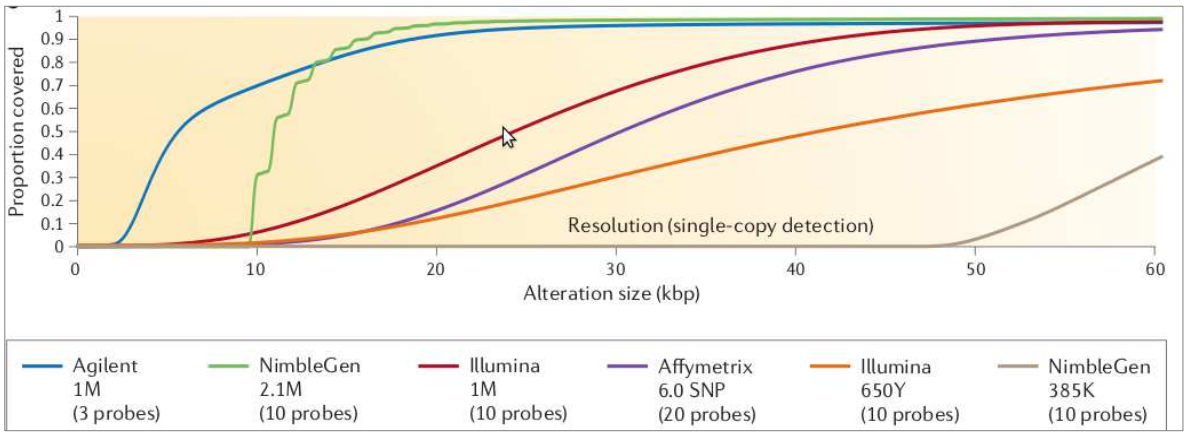

Figure 8: Figure shows the probe coverage of several major array platforms as determined by ResCalc [48].

Method	**Translocation**	**Inversion**	**LCV(>50kb)**	**CNVindel (1-50kb)**	**Small sequence variants(<1kb)**
Karyotyping	Yes(>3Mb)	Yes (>3Mb)	Yes(>3Mb)	No	No
Clone-based array-CGH	No	No	Yes(>50kb)	No	No
Oligonucleotide-based array-CGH	No	No	Yes	Yes	Yes(SNPs)
Sequence-assembly Comparison	Yes	Yes	Yes	Yes	Yes
Clone paired-end sequencing(fosmid)	Yes	Yes (breakpoints)	Yes(>8kb deletions)	Yes (>8kb of deletions); (<40kb of insertions)	No

Table T2: Table showing methods for detecting structural variation in human genome [11].

Pathogenic studies require thousands of individuals and controls to assess the different diseases. Thus it will be easier in terms of cost and throughput to use Microarrys for such studies. Using array data, we can conduct genome wide studies to detect and genotype different structural variants. For example, 2,493 Illumina SNP profiles were used to retrieve a detail picture of large CNVs in the 0.5–1% frequency range [49]. It will also help in future to study larger populations and investigate human diseases [11].

Although, array data are being extensively used to identify structural variations, there have limitation in detection of larger size CNVs, balance variants like inversions and in breakpoint resolution at single base pair level. The size and breakpoint resolution of any prediction is correlated with the density of the probes on the array, which is limited by either the density of the array itself (in aCGH) or by density of known SNP loci (for SNP array) [6]. Another important limitation of array technique is to use it in repeat-rich and duplicated regions. Since Array CGH and SNP platforms are based on the assumption that each location to be diploid in the reference genome, which is not true in case of duplicated sequence. Since CNVs have a strong positive correlation with segmental duplications and many breakpoints lie in duplicated regions, we need other additional technology to find the accurate boundaries and copy numbers of these events [38, 49, 50, 51].

2.5.2 Single-molecule Analysis

Single-molecule Analysis is an important way to visualize and understand the location and structure of larger variants at single-molecule level. This analysis includes techniques such as fluorescent in situ hybridization (FISH), fiber-FISH and Karyotying. These techniques are effective for identification of common and rare large genome structural

variants. However, their low throughput and low resolution limit their application to a few individuals and to particularly large structural differences (~500 kb to 5 Mb). Different methods are being developed to use large scale stretched DNA fragments for direct visualization to improve resolution and scalability of this approach [11]. Optical mapping is a technique based on a modification of traditional restriction mapping. In this technique restriction digestion is performed on immobilized DNA to identify the fragment sizes and changes in their relative order on the basis of comparison to an in silico digested version of the reference genome sequence [11]. Originally, it was developed to analyze yeast genome but was used for fine-scale structural analysis of human genomes, detection of inversions and trans-locations, as well as copy number alterations, and their breakpoints [11, 37, 53, 54]. Optical Mapping technique has very limited throughput and its entire analysis depends on the reference genome. DNA barcoding methodologies are also being developed as alternative techniques which would be helpful for high-throughput detection of balanced structural differences in cellular level in future [11].

2.5.3 SV Detection Based on Sequencing

DNA sequencing is done to obtain the order of four basic nucleotides in a DNA. This will be helpful to find the SVs in comparative genome study. Different sequencing methods and technologies have evolved in the race of reducing sequencing cost and increasing throughput.

In high-throughput shotgun **Sanger sequencing**, genomic DNA is fragmented, then cloned to a plasmid vector and used to transform *E. coli.* For each sequencing reaction, a single bacterial colony is picked and plasmid DNA isolated. Each cycle sequencing

reaction takes place within a microliter-scale volume, generating a ladder of ddNTP-terminated, dye-labeled products, which are subjected to high-resolution electrophoresis separation within one of 96 or 384 capillaries in one run of a sequencing instrument. As fluorescently labeled fragments of discrete sizes pass a detector, the four-channel emission spectrum is used to generate a sequencing trace [55].

After three decades of continuous improvement, the Sanger biochemistry can be applied to achieve read-lengths of up to ~1,000 bp, and per-base 'raw' accuracies to 99.999%. In the context of high-throughput shotgun genomic sequencing, Sanger sequencing costs on the order of $0.50 per kilobase [55].

The advancement of **Next Generation Sequencing** (NGS) has proved itself as a high throughput and cost-effective sequencing technology. It has provided golden opportunities for effective genomic variant detection. NGS has capability to sequence million of bases simultaneously completing sequencing of full human genome in couple of days with twenty fold less cost than all previous methods [56].

The concept of cyclic-array sequencing can be summarized as the sequencing of a dense array of DNA features by iterative cycles of enzymatic manipulation and imaging-based data collection. The commercial products that are based on this sequencing technology include Roche's 454, Illumina's Genome Analyzer, ABI's SOLiD and the Heliscope from Helicos [55].Along with these technologies there is also a commercial Ion Torrent platform that has semiconductor based detection system [57].

Roche 454 GenomeSequencer

In 2005, 454 Life Sciences lunched GenomeSequencer as a first next-generation system which was based on pyrophosphate detection [61]. It is also called pyrosequencing technology. It employs Emulsion PCR amplification approach to detect sufficient light signal in the sequencing-by-synthesis reaction step. In this sequencing system, DNA fragments are ligated to beads by means of specific adapters. After the completion of PCR amplification cycles, each bead along with its fragment is placed at the top end of an optical fiber that has the other end facing to a sensitive CCD camera. This camera enables the positional detection of emitted light. In the final step, to start the synthesis of complementary strand polymerase enzyme and primer are added to the beads. The incorporation of a base by the polymerase enzyme in the growing chain releases a pyrophosphate group, which can be detected as emitted light. Although 454 sequencing platform has overcome substitution error, it has limitation during base calling of homopolymers DNA segments (of lengths greater than 6).For this reason homopolymers segments are prone to base insertion and deletion errors during base calling. At present, the GS FLX Titanium series allows generation of more than 1,000,000 single reads per run with an average read length of 400 bases [60].

Illumina Genome Analyzer

The Illumina Genome Analyzer also known as Solexa sequencer is the most widely available HTS technology. In this platform, the amplified sequencing features are generated by bridge PCR and after immobilization in the array, all the molecules are sequenced in parallel by means of sequencing by synthesis [60, 62, 63].

During the sequencing process, each nucleotide is recorded through imaging techniques, and is then converted into base calls. The Illumina sequencer is able to sequence reads up to 100 bp (with longer ones expected in the near future) with relatively low error rates. Read-lengths are limited by multiple factors such as incomplete cleavage of fluorescent labels or terminating moieties which cause signal decay and dephasing. In this platform sequencing errors are mainly due to substitution errors, while insertion/deletion errors are much less common. Average raw error-rates are on the order of 1–1.5% [64], but higher accuracy bases with error rates of 0.1% or less can be identified through quality

Metrics associated with each base-call. Illumina Genome Analyzer IIx is able to generate up to 200 million 100 bp paired-end reads per run for a total of 20 Gb of data with a throughput of around 2 Gb per day. The latest MiSeq is said most to be most easiest and accurate benchtop product among Illumina products [60].

ABI's SOLiD

The ABI SOLiD sequencer is another widely used sequencing platform acquired by Applied Biosystems in 2006. The sequencing process used by ABI SOLiD is very similar to the Solexa work flow; however, there are also some differences. First of all, the clonal sequencing features are generated by emulsion PCR, instead of bridge PCR. Second, the SOLiD system uses a di-base sequencing technique in which two nucleotides are read (via sequencing by ligation) simultaneously at every step of the sequencing process, while the Illumina system reads the DNA sequences directly. Although there are 16 possible pairs of di-bases, the SOLiD system uses only four dyes and so sets of four di-bases are all represented by a single color. As the sequencing machine moves along the

read, each base is interrogated twice: first as the right nucleotide of a pair, and then as the left one. In this way, it is possible to derive each subsequent letter if we know the previous one, and if one of the colors in a read is misidentified (e.g. due to a sequencing error), this will change all of the subsequent letters in the translation. Even if this may seem to generate problems in read sequencing, it can be advantageous during the read alignment to a reference genome. The raw 'per-color' error rate is around 2-4% .The latest 5500 W Series Genetic Analysis Systems are able to generate fragment sequencing of up to 75 bp, paired-end sequencing of up to 75 x 35 bp, and mate-paired sequencing of up to 60 x 60 bp [65].

Ion Semiconductor Sequencing

Ion Torrent Systems Inc. (now owned by Life Technologies) developed a system based on using standard sequencing chemistry, but with a novel, semiconductor based detection system. This sequencing platform also uses Emulsion PCR amplification approach for clonal sequencing. This method of sequencing is relied on the detection of hydrogen ions that are released during the polymerization of DNA, as opposed to the optical methods used in other sequencing systems. A microwell containing a template DNA strand to be sequenced is flooded with a single type of nucleotide. If the introduced nucleotide is complementary to the leading template nucleotide it is incorporated into the growing complementary strand. This causes the release of a hydrogen ion that triggers a hypersensitive ion sensor, which indicates that a reaction has occurred. If homopolymer repeats are present in the template sequence multiple nucleotides will be incorporated in a single cycle. This leads to a corresponding number of released hydrogens and a

proportionally higher electronic signal. Although it has relatively low substitution error, it has indels in sequencing reads due to homopolymer detection error [57].

Method	**Single-Molecule real time sequencing (Pacific Bio)**	**Ion Semiconductor (Ion Torrent Sequencing)**	**Pyrosequencing (454)**	**Sequencing by synthesis (Illumina)**	**Sequencing by litigation (Solid Sequencing)**	**Chain Termination (Sanger Sequencing)**
Read Length	2900 bp average	200bp	700bp	50 to 250 bp	50+35 or 50-50 bp	400bp to 900 bp
Accuracy	87%(read length mode), 99% (accuracy mode)	98%	99.9%	98%	99.9%	99.9%
Reads per run	35-75 thousand	Up to 5 million	1 million	Up to 3 million	1.2 to 1.4 billion	N/A
Time per run	30 mins to 2 hours	2 hours	24 hours	1 to 10 days depending upon sequencer	1 to 2 weeks	20 mins to 3 hours
Cost per 1M b (in US $)	$2	$1	$10	$0.05 to $0.15	$0.13	$2400
Advantages	Longest read length. Fast. Detects 4mC, 5mC, 6mA	Less expensive equipment , Fast	Long read size , Fast	High sequence yield	Low cost per bases	Long individual reads useful for many applications
Disadvantages	Low yield at high accuracy. Equipment can be expensive	Homopolymers errors	Runs are expensive. Homopolymers errors	Equipment can be very expensive	Slower than other methods	More expensive and impractical for larger sequencing project

Table T3: Comparison of next-generation sequencing methods [58, 59].

Paired-End/Mate Pair Reads

Sequencing technologies can generate pair of reads (i.e. two reads at approximately known distance, known as insert size) by sequencing both sides of DNA segments. To generate Mate Pair reads, first genomic DNA is fragmented and size-selected inserts are circularized and linked by means of an internal adapter. Second, this circularized and linked fragment is then randomly sheared, and segments containing adapter are purified. In third and final step, mate pairs are generated by sequencing around the adapter. In contrast, Paired-End Reads are generated by fragmentation of genomic DNA into short segments, followed by sequencing of both ends of the segments. Paired-end reads provide tighter insert-size distributions, and thus higher resolution, whereas mate pairs give the advantage of larger insert sizes .In computational approaches they do not have any significant differences though wet lab approaches to generate them are different. Thus here we only mention paired-end reads [6].

Techniques based on Paired-End Reads

Before the breakthrough of Next-generation Sequencing, relatively low coverage and expensive Sanger sequencing techniques are used to generate long pair end reads. But after the introduction of Next-generation sequencing platforms like Roche's 454, Illumina's Genome Analyzer, ABI's SOLiD and Ion Semiconductor Sequencer, both single end and paired end reads are generated in terms of billions with in short time period with relatively low cost. To extensively utilize these high throughput data different

strategies are developed. We mention here four general types of strategies, all of which focus on mapping sequence reads to reference genome and subsequently finding the discordant signatures or patterns that are indicators of different type of SVs.

Read Pair: Assessing the insert size of read-pair and abnormal orientation of read pairs in which the mapping span and/or orientation of the read pairs are inconsistent with the reference genome, one can observe different SVs. Read pairs mapping larger distance than defined insert size define deletions, those mapped with smaller distance are indicative of insertions, and orientation inconsistencies can indicate inversions and specific class of tendum duplication [11]. Different SV detection tools including PEMer, VariationHunter , MoDIL, BreakDancer and SVDetect are based on this approach but they do differ on the variant of signatures they detect and on the clustering procedures.

Read-depth: All the SV signatures cannot be detected by above mention approach. This approach is based on a random (typically Poisson or modified Poisson) distribution in mapping depth and investigate the divergence from this distribution to find out duplications and deletions in the sequenced sample. The basic idea of this approach is that duplicated regions will show significantly higher read depth and deletions will show reduced read depth when compared to diploid regions [11]. Different tools including RDXplorer and CNVnator are based on this approach.

Split-read: This approach can detect deletions as well as small insertions with single-base-pair resolution.This approach were first applied to longer Sanger sequencing reads. This technique is used to define the breakpoint of a structural variant based on a ‘split’

sequence-read signature. If the split reads are mapped such that they are mapped far from each other than those reads indicates a deletion or in the reference indicates an insertion; if the split reads are mapped in reverse orientation that indicates the inversion [11]. Some example of tools based on this technique are PRISM , Pindel ,and SVseq.

Sequence Assembly: Generating assembly of the short reads and mapping them to the reference also help us to find SVs. There are assembly algorithms based on deburjion graph methods that generate the contigs from short reads. Mapping this contigs with reference gives us the clue to detect Svs. Some de novo assembly algorithms based on next-generation whole-genome shotgun (NG-WGS) data include EULER-USR, ABySS, SOAPdenovo and ALLPATHS-LG [11].

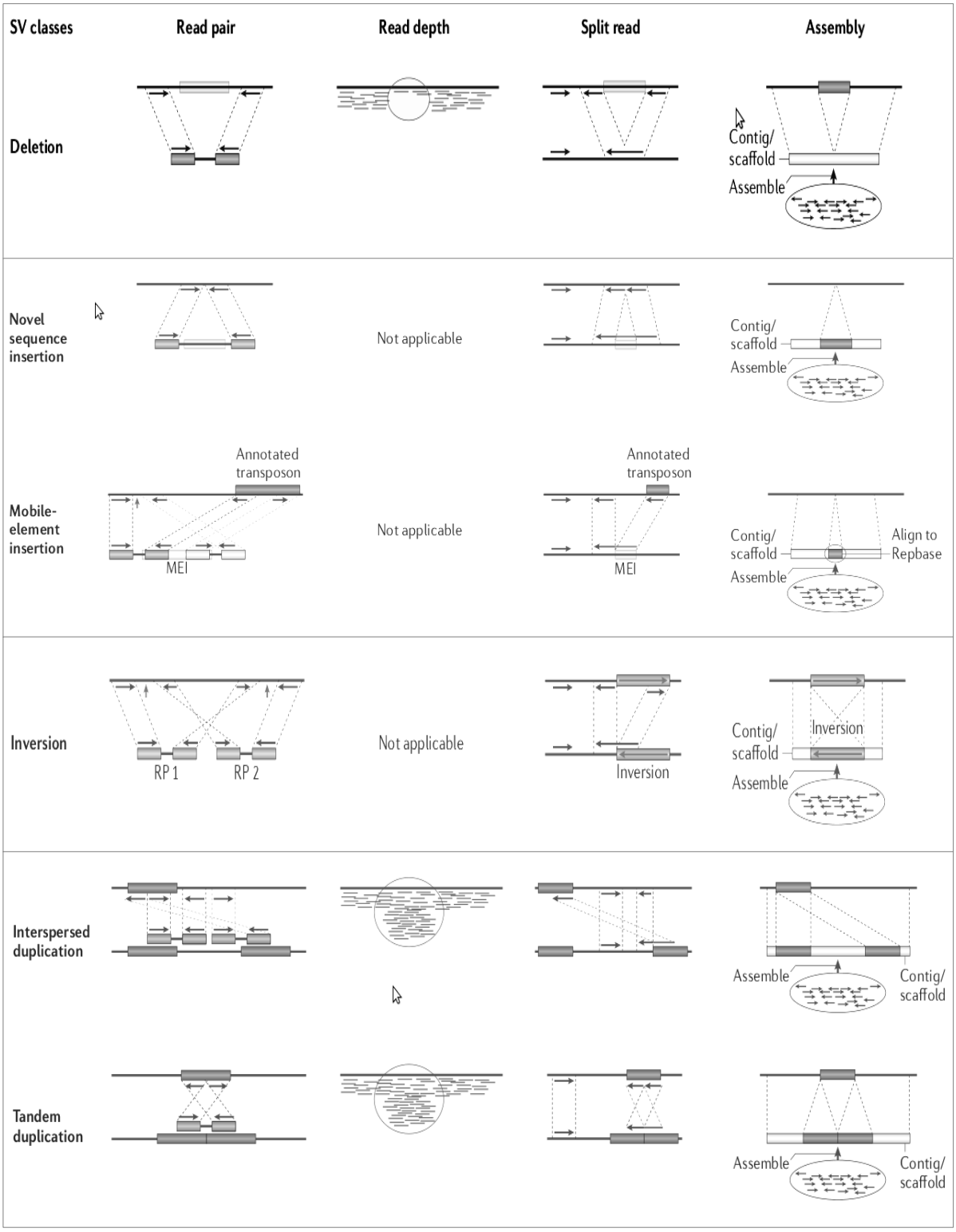

Figure 9: Figure showing different SV signatures and detection strategies based on Paired-end reads [11].

Advantages

With help of NGS technology high throughput paired-end read sequences are being generated at low cost and small time frame. Techniques based on paired-end reads have made easier to detect different varieties of SVs and to present clear spectrum of genomic variations in the genome. Large number of reads provides the easy comparison for copy number of donor genome and reference genomes and gives us opportunity to find novel structural variations.

Limitation

Each of four above mentioned approaches based on paired-end reads has limitations depending on variant type, size and the properties of the underlying sequence at the SV locus.Read Depth method is applicable to detect SVs based on absolute copy-number; the breakpoint resolution is very weak. Read-pair approaches are powerful, but resolving ambiguous mapping assignments in repetitive regions is challenging and accurate prediction of SV breakpoints depends on very tight fragment size distributions, which can make library construction difficult and costly [6]. Similarly, split-read algorithms can be devised to detect a wide range of SV classes with exact breakpoint resolution; however, split read is currently reliable only in the unique regions of the genome. Sequence assembly promises to be the most versatile method by facilitating pair-wise genome comparisons; however, it has been shown to be heavily biased against repeats and duplications causing to collapse assembly over such regions [66, 67].

CHAPTER 3

METHODS

Methods based on paired-end reads need very tight fragment size distributions and high coverage for accurate SVs detection which can make library construction difficult and costly [11]. Also the short paired-end reads have been more challenging to map accurately and uniquely against reference genome than relatively longer reads. In this context we are presenting Inversion detection pipeline based on Single End Reads to show that our approach is applicable in the relatively low coverage and perform well to detect inversion variants. We have divided our pipeline in two phases. In the first phase candidate breakpoint pairs are inferred and in the second phase false positives are filtered to find true inversion breakpoints.

3.1 Read Mapping

The preliminary step of our pipeline is read mapping. Single End reads generated from donor genome are aligned to reference genome using a mapper suitable for mapping single end reads. To make pipeline efficient, alignment process is divided into two phases. During first phase, we do full length alignment of whole reads against the reference genome. These results in SAM file containing alignment detail of whole reads in reference genome. This step is supposed to map the all reads at unambiguous positions in the reference genome except those reads which are hovering the region of inversion. In the second phase, SAM file obtained from first phase is processed to extract the unmapped reads. These unmapped reads have the alignments with their location field set to 0 in the SAM file, which are extracted and changed to fastq/fasta file format by picking read header, sequence and base quality using custom bash script. These unmapped reads are supposed to contain reads of our interest.

All extracted unmapped reads are performed ungapped alignments against reference genome enabling the softclipping using Smith-Waterman algorithm incorporated in the mapper to get all best alignments. From the second phase of mapping we can obtain the alignments of those reads which are covering the junction of inversions with a CIGAR of mapped and softclipped bases. The SAM file obtained after the second phase is sorted based on read header if it is not sorted. This sorting makes sure that we will get all alignments of individual read consecutively.

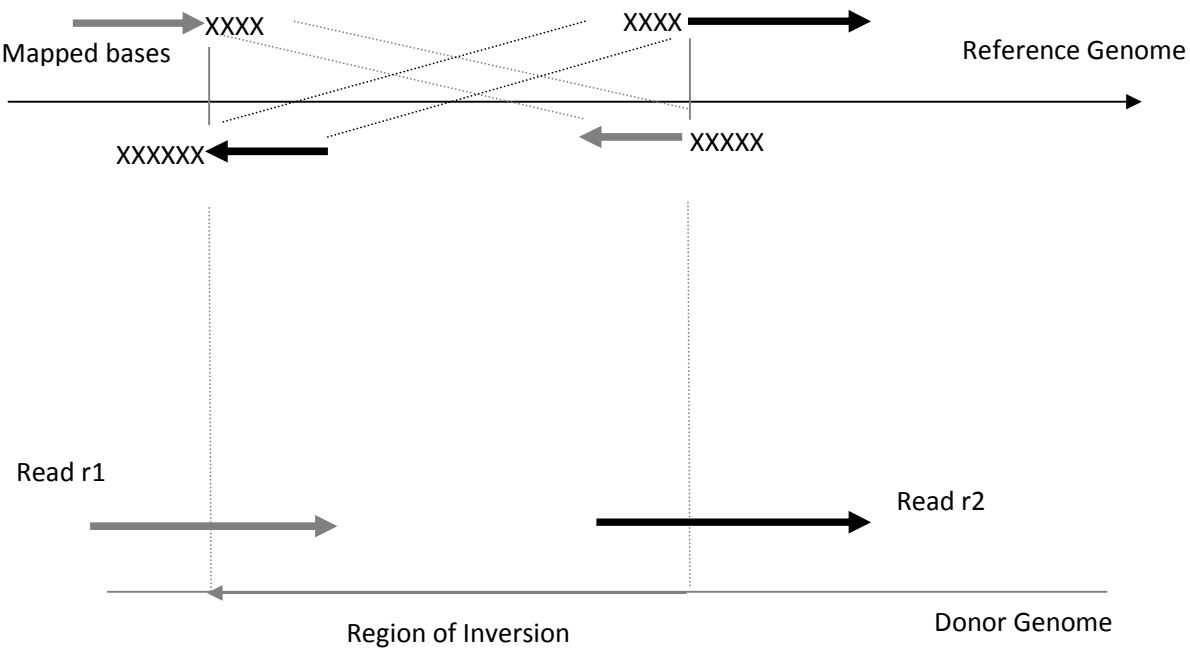

Figure 10: Alignments of Read r1 and r2 (over the junction of inversion) after aligning against reference genome in the second phase. These reads r1 and r2 are the reads of our interest.

3.2 Processing SAM and Generating Candidate Breakpoints

After completing two phase mapping of the single end reads, obtained sorted SAM file is considered as main input in our pipeline. We process SAM file by scanning from the first line of the SAM file. SAM file first contains header section which starts with '@ ' and contains information such as contig name and length and are located above first alignment in the SAM file. Thus, our program ignores the line that starts with '@'.After scanning header section of the SAM file, it scans the each alignment, to check whether the alignment is mapped or not. To check this, our program checks the location field of alignments in SAM format. If location field is set to 0, this indicates the read is unmapped thus program skips that alignment and start scanning next alignments of another read. We know the fact that a single end read can have multiple alignments at multiple locations of different chromosomes/contigs in reference genome, and all such alignments are depicted in SAM file with same header name. Our program implements HashMap data structure to store the chromosome name and corresponding alignments of a read in that chromosome. First, we store alignments of a read belonging to a particular chromosome in array-list. This array list is then inserted into a HashMap as a value with chromosome name as a key. We repeat this for all alignments of a read. More formally, we hash all the alignments of a single end read based on chromosome name, which is at third position in SAM format of an alignment. After hashing all alignments of a read we start processing the hash map. For each chromosome (key) in a hash map we iterate all the alignments in the corresponding list to find those alignment pair which are first: aligned opposite to each other, second: total mapped bases are at least 90% of read length, third: softclipped bases are more than 10. These three constraints are the most essential for the inference of genomic inversion and its breakpoints from the alignments.

To deal with first constraint, we decode the flag field present in the SAM format of alignment. This flag bit is converted into binary bit and checked if 0X10 flag bit is set or not. If 0X10 bit is set then alignment has reverse direction otherwise alignment has forward direction. To deal with second and third constraints our program parse CIGAR field of SAM format alignment. Using regular expression, our program separates mapped and softclipped bases from CIGAR, which are subsequently used to find the total mapped bases of two alignments and their softclipped bases length. If the pair of alignments in the list fulfill these three constraints we infer the pair of candidate breakpoints of inversion from them, store them in a list. Finally, we clear HashMap to start hashing alignments of another reads. Position field of SAM file is the co-ordinate of first mapped base pair. Position field and CIGAR field give us the co-ordinate of breakpoints. To infer the breakpoint pairs we check type of softclipping from CIGAR string of the alignment. If CIGAR string has softclipped bases on the left side, the breakpoint of a inversion is given by the location field of the alignment. If CIGAR string has softclipped bases on right side, the breakpoint of inversion is given by sum of location and mapped bases. Our definition of breakpoints is the position of first and last base pair of inversion, subtraction of 1 from right breakpoint is done to get location of last base pair in the inversion for the inferred location. Based on genomic coordinates breakpoints are assigned either to list of left breakpoint or right breakpoint and inserted to another HashMap with chromosome name as key and left and right breakpoints as value. After SAM file scanning is completed, this HashMap is processed to find the supporting read counts which are indicated by duplicate entry in the HashMap. This supporting read count is the important

parameter which helps to create the more precise candidate breakpoint list. We ignore those breakpoints whose support count is only one; that is underpinned by only one read.

Pseudo Code

```
for each key(chromosome)  retrieve list of alignments L = HashMap(key)
        for i in the list L
                for j =i+1 in the list L
                check following conditions for ith and jth alignments
                a. direciton of alignmets are reverse
                b. total mapped bases >= 90% of READ_LENGTH
                c. softclipped length >10
                if(a AND b AND c)
                        GO TO  STEP 1. and store in
                        HashMap  H <chromosome, bppairlist>
                        set the flag indicating jth alignment is checked.
                else
                        j++;
                end if
                end for
```

```
        i++;

    end for

end for each
```

STEP 1 : Calculate breakpoint positions of those alignments in the following way

```
    pos ←  position of alignment in the reference
    ls ← left softclipped bases
    rs ← right softclipped bases
        if ls>10
            bppos1 ← pos
        end if
        if rs >10
        bppos2 ←  pos + (readlength-rs)
        end if
        if(bppos1>bppos2)then
            bppos1 ←  bppos1 -1
            leftbp ← bppos2
            rightbp ← bppos1
         else
            bppos2 ←  bppos2 -1
            leftbp ← bppos1
            rightbp ←bppos2
         end if
```

```
if (overlapping of base pairs in the alignment,  d  >0)

                    if(alignment giving rightbp has leftsoftclip) then
                            rightbp ← rightbp + d


                    if(alignment giving leftbp has rightsoftclip) then

                            leftbp ←  leftbp - d

                    end if

                    add (leftbp &rightbp ,  bppairlist).

end if


2. sort Hashmap H<chromosome, bppairlist> based on chromosome
 for each key (chromosome) in Hashmap H ,
                    List bppairlist = H(key)
                     sort bppairlist based on leftbp
                    set readsupportcounter=1
                    for i in  bppairlist
                            if(leftbp of bppairlist(i+1)- leftbp of bppairlist(i) <=5  AND
ABSOLUTE(rightbp                                of bppairlist(i+1)- right of bppairlist(i) )
<=5)
                                    readsupportcounter++;


                            else
                                    if readsupportcounter > CONSTRAINT
                                            HashMap bpset <bppairlist(i), counter>
                                            HashMap bhchr<bppairlist(i), chromosome>
                                    else
```

```
                                readsupportcounter =1;
                        end if
                end if
                i++;
        end for
end for each
```

3.3 Filtering and Finalizing Breakpoints

After completion of the first phase, we obtain candidate list of breakpoint pairs whose support count is greater or equal to **CONSTRAINT**. We can set this constraint depending on the coverage of the reads. For higher coverage (>10X) we can set the CONSTRAINT higher (normally >2) and for lower coverage (<5X) we can set it to >=1.In the second phase or final phase, we filter the false positives to increase sensitivity of our pipeline. To do this, first we create local regions based on coordinate and chromosome name of candidate breakpoint pairs. For each such pair first, we retrieve the segment of the reference genome located in between two breakpoints (left breakpoint and right breakpoint) in a particular chromosome. This segment is named as candidate region. Second, we retrieve the segment of reference genome of length equal to read length starting from left breakpoint coordinate – READ LENGTH up to left breakpoint, which is called left region. Third, we also create right region by retrieving the segment of reference genome of length equal to read length starting from right breakpoint coordinate up to right breakpoint coordinate + READLENGTH.

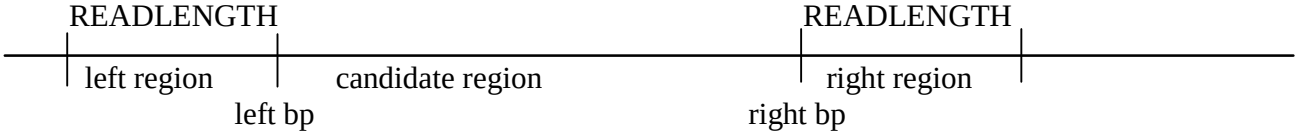

Figure 11: figure showing local regions retrieved from reference genome based on left and right breakpoints and corresponding chromosome

After getting left region, candidate region and right region from breakpoint pair writes derive final local regions and store them in fasta file in the following way.

Pseudo Code

For each breakpoint pairs in **bppairlist**

String candidateregion ← Reference.substring(leftbp, rightbp+1)

String extension 1 ← Reference.substring(leftbp-READLENGTH, leftbp+1)

String extension2 ← Reference.substring(rightbp, rightbp+READLENGTH+1)

//generate the region without Inversion//

localregion ← extension1+candidateregion+extension2

localregion1 ← localregion.substring(0,2*READLENGTH)

localregion2 ← localregion.substring(localregion.LENGTH -2*READLENGHT, localregion.LENGTH)

//generate the region with inversion//

candidateregion ← ReverseComplement(candidateregion)

localregion ← extension1+candidateregion+extension2

localregion3 ← localregion.substring(0,2*READLENGTH)

localregion4 ← localregion.substring(localregion.LENGTH-2*READLENGTH, localregion.LENGTH)

end for each

To write local regions in fasta format , we create a unique header on the basis of name of local region, corresponding left and right breakpoints , name of chromosome and type of local region (region with inversion or without inversion) in the following way concat(>nameoflocalregion/leftbp/rightbp/chromosomename/type of region) .

For example, header for a localregion generated by breakpoints 22234 and 22456 in ChrY with inversion would be >localregion1/22234/22456/ChrY/inv

Then we write the sequence of the region in the next line.

After creating local reference, we index it using suitable aligner and perform full length alignment of all generated single end reads to this local reference.

The output SAM stream after aligning whole reads against local reference file is used to count the number of overlapping alignments over the breakpoints. To count alignments overlapping over breakpoints we use fractional proportion of alignments. For example, if a read has 5 alignments in the local regions we assign 1/5 weight to each of the alignments of that read.

Ideally, *for true breakpoints,* localregion3 and localregion4 (regions with inversion) will have fully mapped alignments' fractional count nearly equal to the read coverage where as localregion1 and localregion2 (region without inversion) will have any fully mapped alignments' fractional count nearly equal to 0. Similarly, for *false breakpoints*, localregion3 and localregion4 (regions with inversion) will have fully mapped alignments' fractional count equal to 0 where as localregion1 and localregion2 (regions without inversion) will have fully mapped alignments' fractional count equal to the read coverage. So setting the following condition will help us filter the false positives.

Condition:

if (fractional alignment count of localregion3 > fractional alignment count localregion1 AND fractional alignment count of localregion4>fractional alignment count of localregion2)

breakpoint pair generating these localregions are true breakpoint pair

else

breakpoint pair generating these localregions are false breakpoint pair

end if

After these filtering steps false positives are reduced significantly and we get final breakpoint pair list with left breakpoint location, its corresponding fractional alignment count, right breakpoint location, its corresponding fractional alignment count chromosome name, **readsupportcounter** for breakpoint pair.

CHAPTER 4

EXPERIMENT AND RESULTS

4.1 Read Simulation and Mapping Statistics

To test our pipeline we have taken hg19 human reference genome and implanted 90 inversions [68] in known positions using Perl script. The number of inversions in different chromosomes and their size distribution are shown in Figure 12 and Figure 13 respectively.

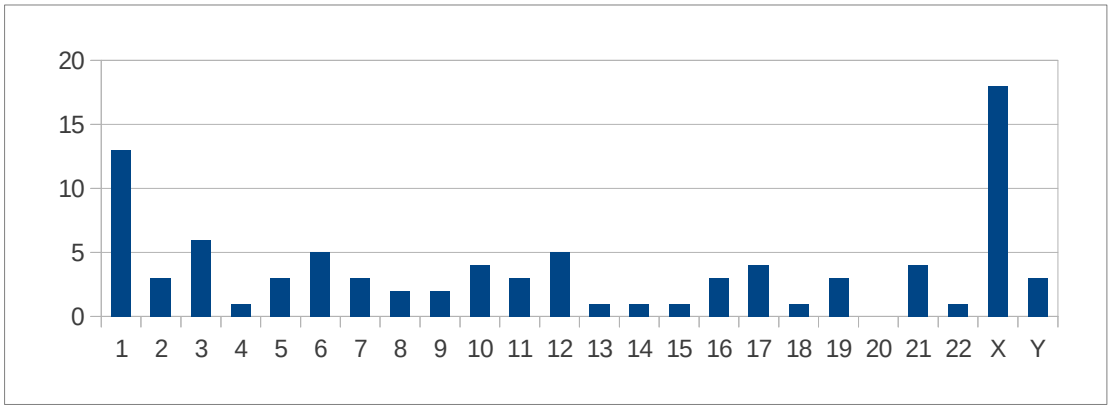

Figure12: Figure showing number of inversions in different chromosomes

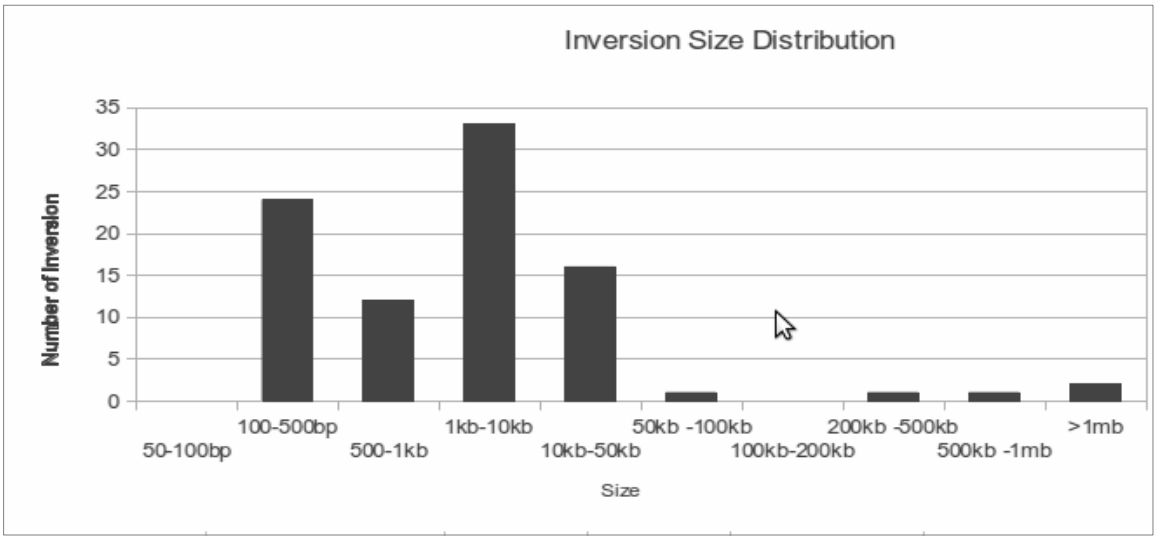

Figure 13: Figure showing size distribution of 90 inversions

Ideal Single End Reads (error free) of different lengths 100bp, 200bp and 400bp are simulated using Wgsim[72] read simulator. To simulate error free reads, parameters like base error rates, standard deviation, rate of mutation, fraction of indels, and probability of indel extension are set to zero. Since wgsim simulator has limitations in total number of reads simulation, we use it repetitively to get total read coverage for each chromosome. First reads with coverage 10x are simulated and later coverages 5X and 2.5X are derived taking half and one fourth of the reads from 10x coverage reads. These reads are mapped using stable version of TMAP 2.3.2 [71].There are simply two steps in mapping with TMAP. In the first step and only once we need to build index of the reference genome against which we are going to map reads. Second step is to map reads using this index. Two phases of mapping processes were executed using TMAP. In the first phase, we use TMAP map1 which is based on BWA [73] short read alignment for full length read alignment disabling softclipping. Unmapped reads from this phase is mapped again

against reference genome using map2 enabling the softclipping. Figure 14 shows the reads mapped by map1 and map2 phase using TMAP.

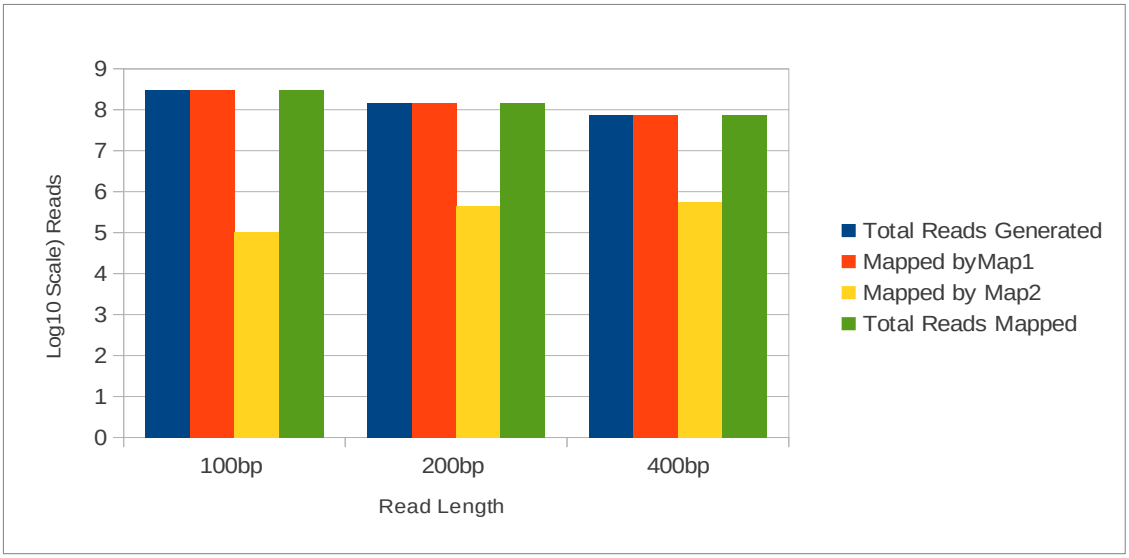

Figure 14: Figure showing the total reads generated, mapped by two phase mapping for 100bp, 200bp and 400bp ideal reads.

To test our program with erroneous data, we again simulated the reads of 100bp, 200bp and 400bp with following error statistics in Wgsim simulator.

Parameters	Value
Base error rate	2%
Rate of mutation	1%
Fraction of indels	15%
Probability of indel extended	0.30

Table T4: Table showing parameters set to simulate erroneous reads.

Similar to ideal reads mapping read with error were also mapped by TMAP using two phase mapping. The detail of erroneous read simulated and mapped by two mapping phases are shown in figure 15 below.

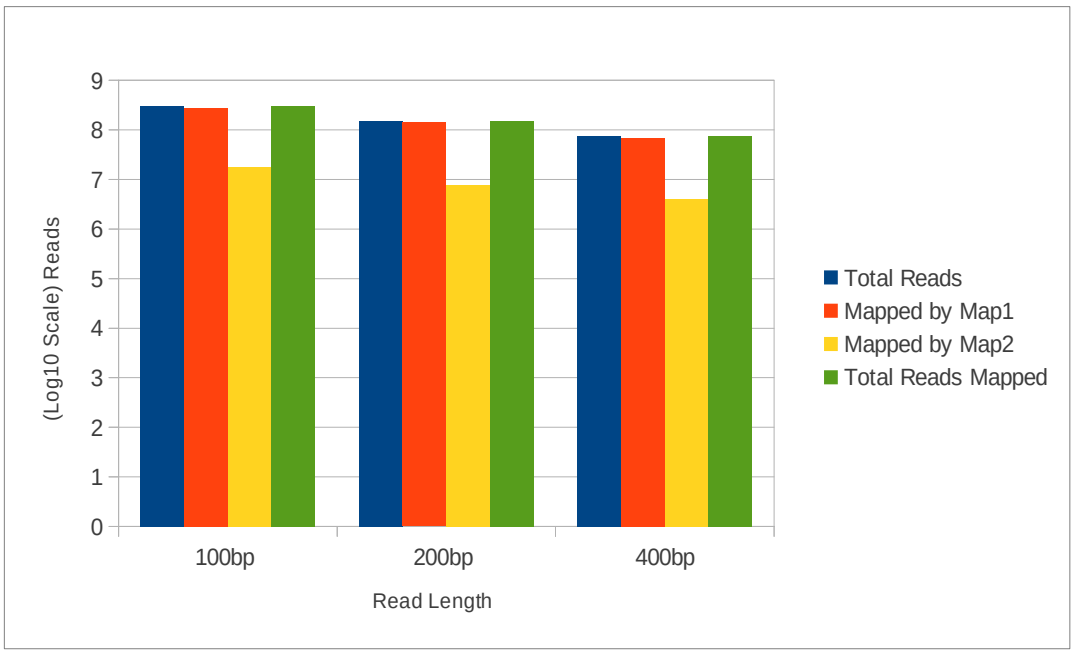

Figure 15: Figure showing the total reads generated mapped by two phases of mapping for erroneous reads of length 100bp 200bp and 400bp.

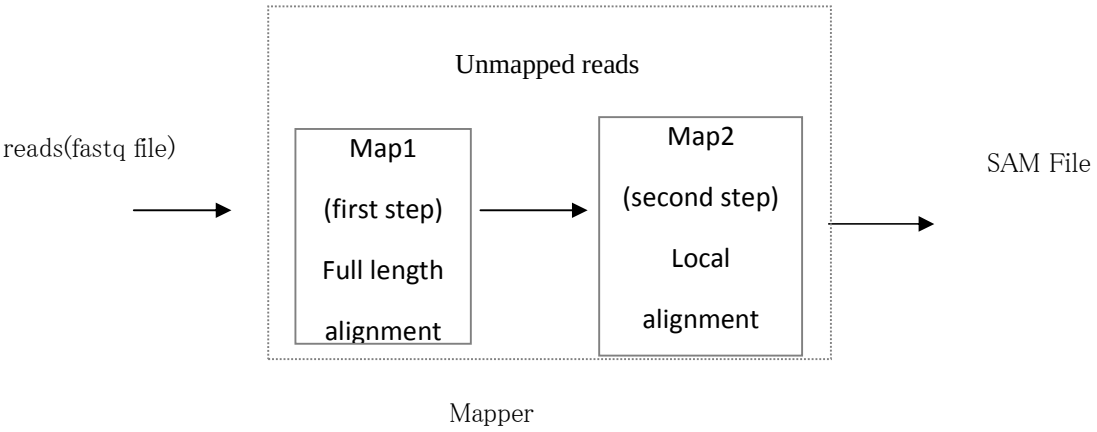

Figure 16: Block diagram of Mapping Process

It is obvious from the figure 14 and 15 that more ideal reads are mapped by map1 phase than erroneous reads. Subsequently, there are more unmapped erroneous reads going to map2 phase than ideal reads. Erroneous reads due to alteration in bases and indels, have higher probability to map to other locations (than the locations from where they were generated) in the reference genome than ideal reads. Due to which more erroneous reads are mapped in map2 phase than ideal reads. But total ideal reads mapped (from both map1 and map2 steps) are higher than erroneous reads.

4.2 Result Analysis

After getting SAM file from mapping of simulated reads using TMAP 2.3.2 in two phases, we feed itto our program for detection of genomic inversion and inference of its breakpoint locations in different chromosomal locations. Beside SAM file, our program takes reference file, whole genome reads and output name. We have set the read support counter constraints to be >=2.First phase of our program finds the candidate breakpoints pair and based on those second phase generates local regions .These local regions are again mapped with whole reads to filter out false positives. After filtering false positives, output is written in a text file which contains, breakpoint pairs, and support read count, chromosome name and fractional alignment counts for each of the breakpoints. Results of both the phases are tabulated on Tables T5, T6, T7 and T8 for ideal simulated reads of different lengths and coverage. Similarly, Table T9 shows the result of our program for different read lengths with errors. To evaluate the performance of our program, we have calculated the statistical parameters like Sensitivity and Positive Predictive Values (PPV). Figures below show the false positives, sensitivity and PPV of different phases from different coverage to explain performance of our program.

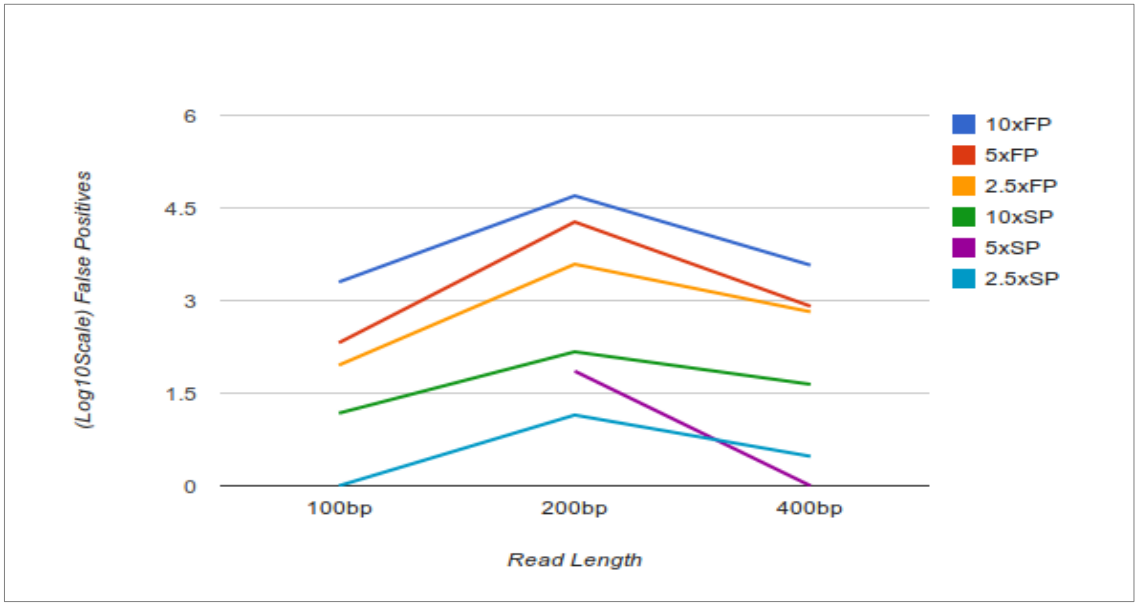

Figure 17: figure showing false positives in first phase and second phase.

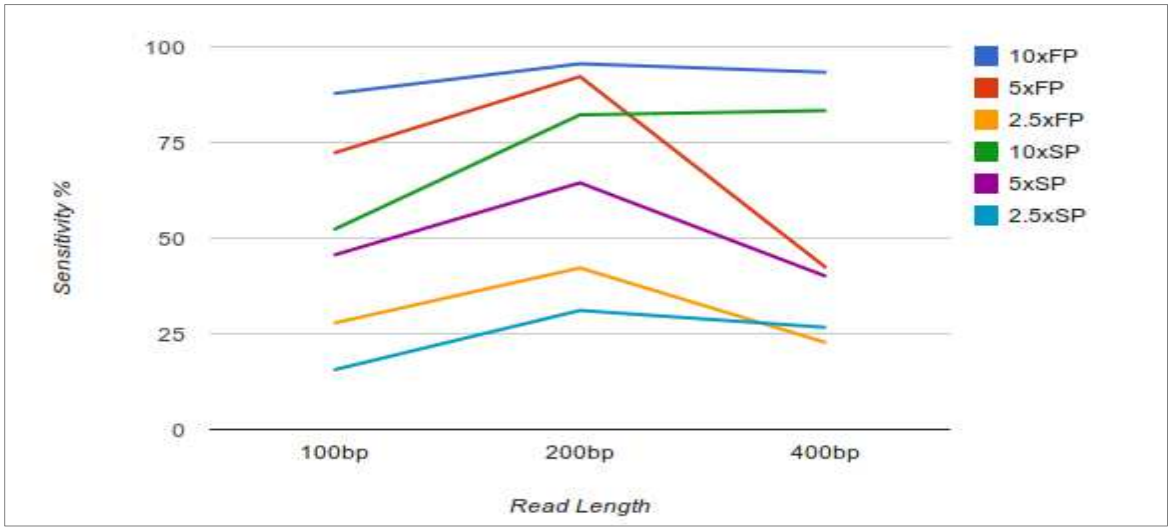

Figure 18: Figure Showing Sensitivity for First Phase and Second Phase for different coverage for different read lengths

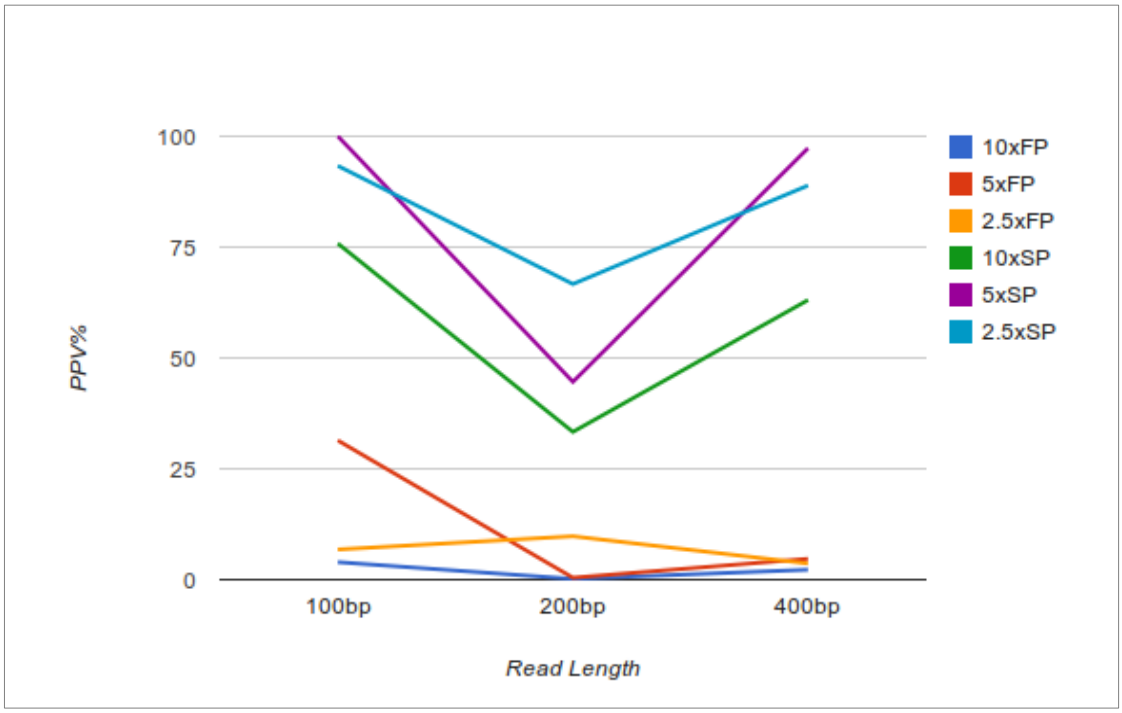

Figure 19: Figure Showing PPV for First Phase and Second Phase for different coverage for different read lengths

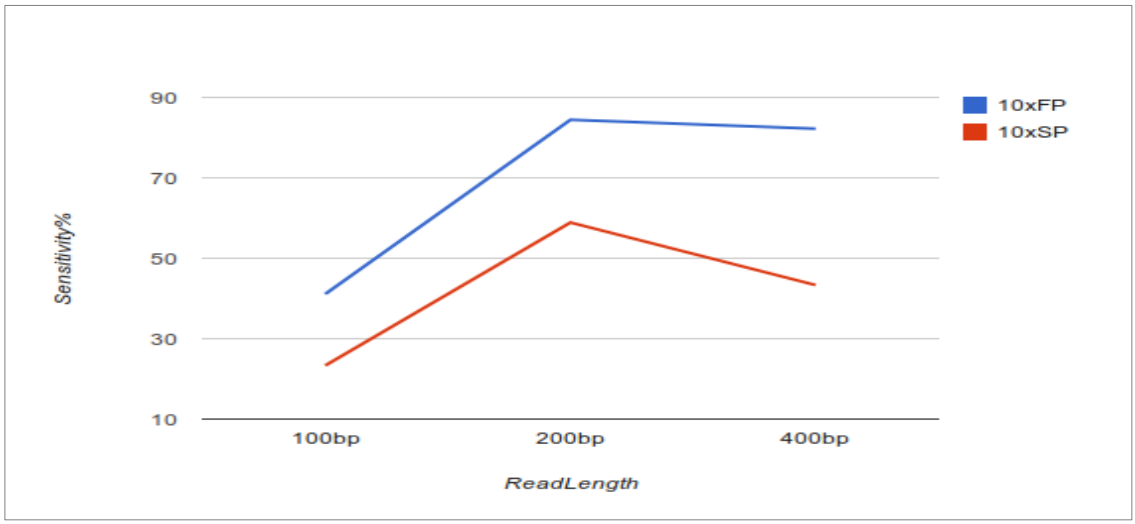

Figure 20: Figure showing Sensitivity for first and Second phase for 10X coverage for erroneous reads

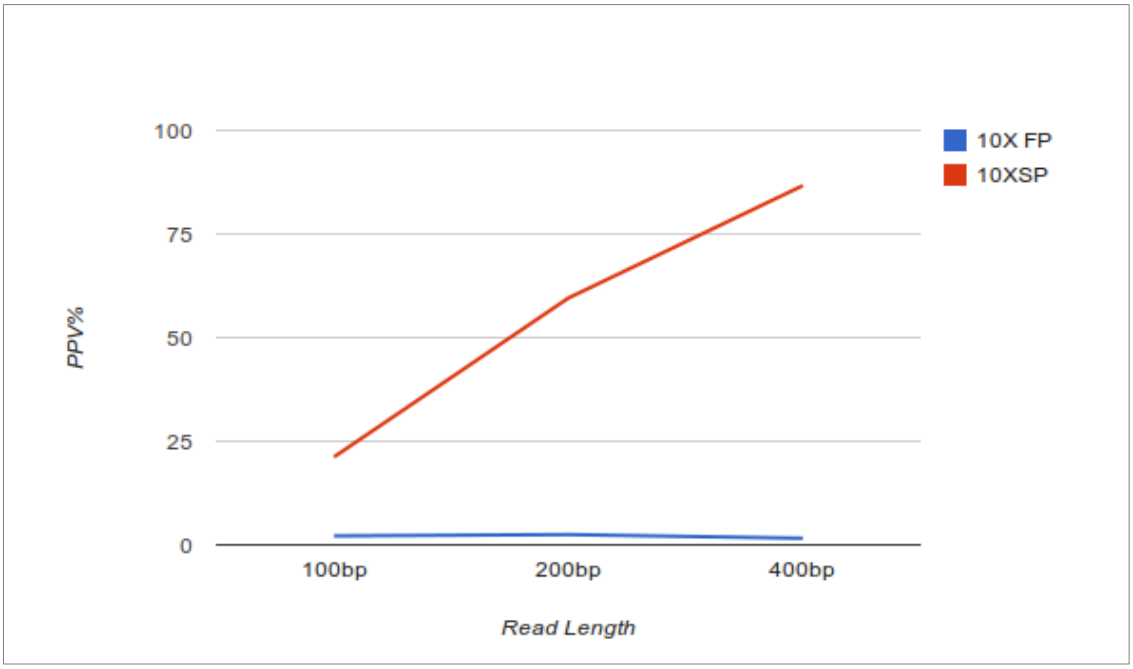

Figure 21: Figure showing PPV for first and second phase for 10X coverage for erroneous reads

From the figure 17, we can observe that for coverage 10X we have high number of false positives in both first and second phase in comparison to lower coverage 5X and 2.5X.With 10X coverage, we get more reads and more alignments which cause to rise the false positives. We can observe that 200bp read length has high number of false positives than 100bp read and 400bp reads. As we increase read length, we also increase the chance to map the read uniquely. Thus for 400bp reads we have lesser false positives. For reads with length 100bp, since these reads are short, they are relatively prone to be mapped to many different locations including the location from where they were generated than 200bp and 400bp reads. Figure 14 shows we only get few 100bp reads unmapped in the first phase in comparison to 200bp and 400bp reads which consequently, reduce the false positives. But in the mean time, we also lose the reads of

our interest in the first mapping phase. In the second phase, false positives are reduced significantly due to the filtering step for all read lengths and coverage. Figure 17 shows the sensitivity of our approach for both the phases for all read lengths. We can clearly observe that in the first phase for all read lengths and coverage, we get higher sensitivity than second phase. But this is also incorporated with higher false positives. This indicate that we also have lower PPV in first phase(shown in figure Figure 19).After filtering and finalizing step in the second phase, false positives are filtered out significantly. Unfortunately, this also filters out the some true positives. Thus after second phase it is obvious (from figure 19) that we have improved PPV in all the phases and for all coverage than first phase but have reduced sensitivity. Thus there exists tradeoff between sensitivity and PPV. In the first phase though sensitivity is satisfactory we have very poor PPV in contrast to second phase where PPV is improved while sensitivity is reduced. Comparatively, we have good PPV and sensitivity for 400bp reads.

Similarly, for reads with error, first phase of mapping outputs more unmapped reads than it was with ideal reads. Thus, in the second phase, we get number alignments and more false positives. Another issue with reads with error is, they are easily mapped to other location of reference genome with competitive mapping quality. Since this error also includes base errors, this force mapper to map in many different location and orientation, we get relatively high number of false positives in comparison to ideal reads. Consequently, we will have reduced sensitivity and PPV in the final phase result in contrast to those of ideal reads.

4.3 Comparison with Existing tools

To compare our method with existing tools we choose SVDetect [69], BreakDancer [70] both of which are based on pair end reads. We simulated error free pair end reads with coverage 10X of length 200bp with insert size 1000, from a '90 inversions implanted' donor genome using wgsim simulator. Those pair end reads are mapped using BWA mapping tool to obtain the final SAM file.

The final SAM file is given input to break dancer pipe line with all the parameters set to default except parameter 's' which is set to 100 (minimum size of region). First configuration file is generated from the bam2cfg.pl program which is then fed into the 'breakdancermax' program.

Similarly to use SVDetect tool, first the SAM file is preprocessed using program 'BAM_preprocessingPairs.pl' .This preprocessing filters out concordant pairs and keep only the discordant pairs. Then 'SVDetect.conf' file is created where, we set all the parameters to default. In the mean time '.length' file is also created to store the contig lengths residing in the reference genome. Different SVDetect commands are run providing 'SVDetect.conf' file as input. The final output, false and true positives detected, Sensitivity and PPV are tabulated in Table T10.

Out of 90 implanted inversions, our approach has found 74 inversions in comparison to BreakDancer 's 58 inversions and SVDetects's 49 inversions. Figure 22 below shows that our approach has found more true breakpoints than other two. False positives are more in our approach than BreakDancer and SV Detect. Since our approach solely relies the alignments of the Single End reads, there is always a decent chance of Single End reads mapped to many different locations in reference genomes other than the true locations

resulting more false positives where as SVDetect and BreakDancer relies on paired-end reads which are separated by predefined insert-size. To detect balanced copy number event like inversions they only consider those reads which have abnormal orientation but approximately correct insert-size. This consideration always helps them to swipe off false positive efficiently in comparison to Single End reads approach of no predefined insert size. Figure 23 shows that our approach has relatively low PPV value than BreakDancer and SVDetect tools the problem with SVDetect is: it does not have capability to resolve the breakpoints at base pair level. It only gives the range of breakpoints by giving starting and ending co-ordinates of each breakpoint. These ranges are also very wide and far (in average) 1000 bp from true breakpoint location. Although BreakDancer tool has provided exact breakpoint co-ordinates they are in average 1000 bp away from the true breakpoints co-ordinates. Our approach has relatively higher precision than BreakDancer and SVDetect. The breakpoints co-ordinates generated by our approach are no more than 5 bp far from true breakpoint co-ordinates.

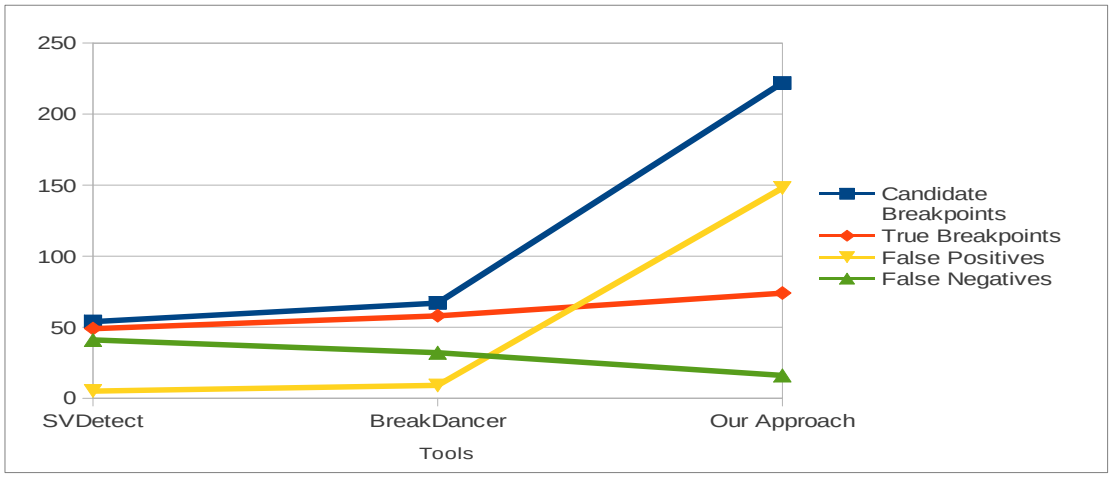

Figure 22: Figure showing the comparison of our tools with other existing tools based different parameters.

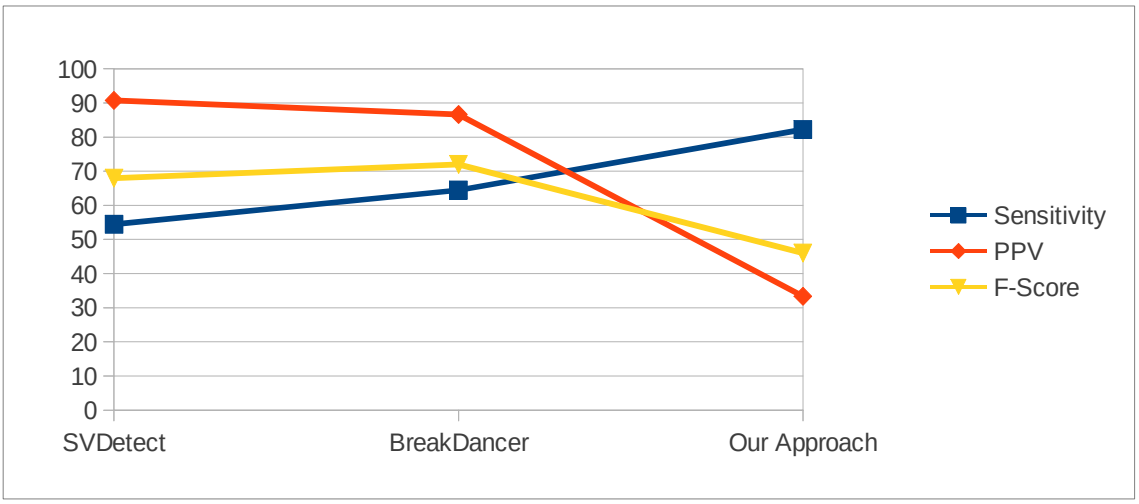

Figure 23: Figure showing the comparison of our approach with other tools based on Sensitivity and PPV and F-Score.

CHAPTER 5

LIMITATION AND FUTURE ENHANCEMENT

It is always challenging to detect, SVs and resolve their breakpoints at base pair level. Although many methods and techniques are devised for the detection of balanced SV like genomic inversion, there always remains limitation in resolving breakpoints due to size of inversion, type of inversion(homozygous or heterozygous) and its complexity, complex and repetitive structure of reference genome, read lengths, sequencing errors, mapping algorithms and mapping accuracy. Although we have overcome the limitation of insert size, by considering Single End Reads our approach also has limitations. As much as we increase the length of single-end reads we also reduce the capability to detect small inversions lesser than read length. Since we use two mapping steps to map our reads against reference genome, for efficiency purpose, we also loose many valuable reads in the first mapping step. This is caused due to repetitive regions (normal repeats and inverted repeats) in the reference genome and accuracy of the mapping algorithm. Additionally in the second mapping step we try to retrieve all possible alignments of a read to make sure that we do not miss important alignments pairs to infer breakpoints. This adds overwhelming number of false positives in first phase of our approach. If we only select best alignments with higher mapping quality we will lose many precious alignment pairs due to tie in mapping score, consequently we lose true breakpoints in the first phase.

Despite of limitations and challenges, there also exists some ways to overcome some of those. For instance, we can extend our capability to detect smaller inversions by considering three pieces of alignments (i.e. alignments have softclipped bases on both

sides of matched bases), although this has possibility to add more false positives. We can use high coverage data (>10X) to find heterozygous inversions, but high coverage reads could yield overwhelming false positives creating problem in filtering steps. Our approach has overhead of running time in the second phase, generating local regions, indexing and mapping whole reads to them which could be reduced by making more stringent constraint in first phase i.e. by using larger read support count for breakpoint pairs but this will likely cause to lose true breakpoints. Fine tuning of different parameters and making them strict could help to reduce this difficulty.

CHAPTER 6

CONCLUSION

In this thesis work we put forth a pipeline to detect genomic inversion in human genome using Single-End reads. We have used simulated platform to verify our approach for different read lengths and variable coverage. With Single End reads generated with relatively low coverage, we are able to detect the breakpoint pair of genomic inversions with relatively good resolution and accuracy .Our pipeline is relatively cost efficient because it discards the need of preparation of insert size library and related biochemical treatments. Moreover, Next Generation Sequencing technology is gradually becoming more cost effective, efficient and capable of ultra high throughput than ever before. These technological achievements can be fully utilized to the mission of achieving broader and clear spectrum of genomic inversions along with other structural variations in the genome in near future and our pipeline will become more relevant in this mission.

Read Length	**coverage**	**Phase**	**Total Candidates obtained**	**Total True Positives**	**True Positives obtained by Program**	**False Positives**	**False Negatives**	**Sensitivity (%)**	**PPV (%)**
100bp	10X	I	1998	90	79	1919	11	87.87	3.95
		II	72	90	47	15	43	52.22	75.81
	5X	I	207	90	65	142	25	72.22	31.40
		II	41	90	41	0	49	45.56	100
	2.5X	I	90	90	25	65	65	27.78	6.81
		II	15	90	14	1	76	15.56	93.33

Table T5 : Output of our approach for ideal reads of 100bp read lengths for coverage 10X,5X and 2.5X for SupportCount >=2

Read Length	**coverage**	**Phase**	**Total Candidates obtained**	**Total True Positives**	**True Positives obtained by Program**	**False Positives**	**False Negatives**	**Sensitivity (%)**	**PPV (%)**
200bp	10X	I	49657	90	86	49571	4	95.56	0.17
		II	222	90	74	148	16	82.22	33.33
	5X	I	18752	90	83	18669	7	92.22	0.44
		II	130	90	58	72	32	64.44	44.62
	2.5X	I	3881	90	38	3843	52	42.22	9.78
		II	42	90	28	14	62	31.11	66.67

Table T6 :Output of our approach for ideal reads of 200bp read lengths for coverage 10X,5X and 2.5X for SupportCount >=2

Read Length	**coverage**	**Phase**	**Total Candidates obtained**	**Total True Positives**	**True Positives obtained by Program**	**False Positives**	**False Negatives**	**Sensitivity (%)**	**PPV (%)**
400bp	10X	I	3743	90	84	3659	6	93.33	2.24
		II	119	90	75	44	15	83.33	63.03
	5X	I	808	90	38	770	52	42.22	4.70
		II	37	90	36	1	54	40.00	97.30
	2.5X	I	657	90	24	633	66	26.67	3.65
		II	27	90	24	3	66	26.67	88.89

Table T7:Output of our approach for ideal reads of 400bp read lengths for coverage 10X,5X and 2.5X for SupportCount >=2

Coverage	Read Length	Phase	Total Candidates obtained	Total True Positives	True Positives obtained by Program	False Positives	False Negatives	Sensitivity (%)	PPV (%)
10X	100bp	I	162	90	70	92	20	77.78	43.21
		II	47	90	42	5	48	46.67	89.36
	200bp	I	5775	90	83	5762	7	92.22	1.44
		II	91	90	56	35	34	62.22	61.54
	400bp	I	768	90	79	689	11	87.78	10.29
		II	84	90	71	13	19	78.89	84.52

TableT8 : Output of our approach for ideal reads of 100bp,200bp and 400bp read lengths for coverage 10X,5X and 2.5X for SupportCount >2

Coverage	Read Length	Phase	Total Candidates obtained	Total True Positives	True Positives obtained by Program	False Positives	False Negatives	Sensitivity (%)	PPV (%)
10x	100bp	I	1707	90	37	1670	53	41.11	2.17
		II	99	90	21	78	69	23.33	21.21
	200bp	I	3057	90	76	2981	14	84.44	2.49
		II	89	90	53	36	37	58.89	59.55
	400bp	I	3775	90	74	3701	16	82.22	1.6
		II	45	90	39	6	51	43.33	86.67

Table T9 : Output of our approach for erroneous reads of 100bp read lengths for coverage 10X,5X and 2.5X for SupportCount >=2

Tools	Candidate Breakpoints	True Pos	True BPs	FalsePos	False Neg	Sensitivity %	PPV %	Distance from True Bps
SVDetect	54	90	49	5	41	54.44	90.70	1000bp
BreakDancer	67	90	58	9	32	64.44	86.57	1000bp
Our Method	222	90	74	148	16	82.22	33.33	5 bp

Table T10: Table showing comparison of our tool with other tools

REFERENCES

1. Jager, M. Structural variations in the human genome

http://igitur-archive.library.uu.nl/student-theses/2011-0817-200603/UUindex.html

2.Gonzalez E., Kulkarni H., Bolivar H., Mangano A., Sanchez R., Catano G., Nibbs R.J., Freedman B.I., Quinones M.P., Bamshad M.J. *et al.* (2005) The influence of CCL3L1 gene-containing segmental duplications on HIV-1/AIDS susceptibility. *Science*, **307**, 1434-1440.

3. Cheng F., Song W., Kang Y., Yu S. and Yuan H. (2011) A 556 kb deletion in the downstream region of the PAX6 gene causes familial aniridia and other eye anomalies in a chinese family. *Mol. Vis.*, **17**, 448-455

4.Osborne L.R., Li M., Pober B., Chitayat D., Bodurtha J., Mandel A., Costa T., Grebe T., Cox S., Tsui L.C. *et al.* (2001) A 1.5 million-base pair inversion polymorphism in families with williams-beuren syndrome. *Nat. Genet.*, **29**, 321-325.

5. S. Sindi, E. Helman , A. Bashir and B. J. Raphael: A geometric approach for classification and comparison of structural variants Vol. 25 ISMB 2009, i222–i230.

6. P. Medvedev, M. Stanciu & M. Brudno : Computational methods for discovering structural variation with next-generation sequencing(2009) S20 VOL.6

7.Sindi, S. & Raphael, B. Identification and frequency estimation of inversion polymorphisms from haplotype data.In Research in Computational Molecular Biology: Proc. RECOMB 2009 vol. 5541 (ed. Batzoglou, S.) 418– 433 (Springer, Berlin, 2009)

8. Watson J.D. and Crick F.H. (1993) Molecular structure of nucleic acids. A structure for deoxyribose nucleic acid. 1953. *JAMA*, **269**, 1966-1967

9.Human Genome Structural Variation Working Group, Eichler E.E., Nickerson D.A., Altshuler D., Bowcock A.M., Brooks L.D., Carter N.P., Church D.M., Felsenfeld A., Guyer M. *et al.* (2007) Completing the map of human genetic variation. *Nature*, **447**, 161-165.

10. Korbel J.O., Urban A.E., Affourtit J.P., Godwin B., Grubert F., Simons J.F., Kim P.M., Palejev D., Carriero N.J., Du L. *et al.* (2007) Paired-end mapping reveals extensive structural variation in the human genome. *Science*, **318**, 420-426

11. Alkan C., Coe B.P. and Eichler E.E. (2011) Genome structural variation discovery and genotyping. *Nat. Rev. Genet.,* **12**, 363-376

12. Campbell C.D., Sampas N., Tsalenko A., Sudmant P.H., Kidd J.M., Malig M., Vu T.H., Vives L., Tsang P., Bruhn L. *et al.* (2011) Population-genetic properties of differentiated human copy-number polymorphisms. *Am. J. Hum. Genet.,* **88**, 317-332.

13.Grossmann V., Hockner M., Karmous-Benailly H., Liang D., Puttinger R., Quadrelli R., Rothlisberger B., Huber A., Wu L., Spreiz A. *et al.* (2010) Parental origin of apparently balanced de novo complex chromosomal rearrangements investigated by microdissection, whole genome amplification, and microsatellite-mediated haplotype analysis. *Clin. Genet.,* **78**, 548-553.

14. Schofield C.M., Hsu R., Barker A.J., Gertz C.C., Blelloch R. and Ullian E.M. (2011) Monoallelic deletion of the microRNA biogenesis gene Dgcr8 produces deficits in the development of excitatory synaptic transmission in the prefrontal cortex. *Neural Dev.*, **6**, 11.

15. Lars Feuk, Andrew R. Carson and Stephen W. Scherer Structural variation in the human genome

16. Homas NS, Bryant V, Maloney V, Cockwell AE, Jacobs PA: Investigation of the origins of human autosomal inversions. Hum Genet 2008, 123:607-616

17. Schmidt S, Claussen U, Liehr T, Weise A: Evolution versus constitution: differences in a chromosomal inversion. Hum Genet 2005, 117:213-219.

18. Hsu LY, Benn PA, Tannenbaum HL, Perlis TE, Carlson AD: Chromosomal polymorphisms of 1, 9, 16, and Y in 4 major ethnic groups: a large prenatal study. Am J Med Genet 1987, 26:95-101.

19. MacDonald IM, Cox DM: Inversion of chromosome 2 (p11p13): frequency and implications for genetic counselling. Hum Genet 1985, 69:281-283

20. Lars Feuk : Inversion variants in the human genome: role in disease and genome architecture Feuk Genome Medicine 2010, 2:11

21.Iafrate AJ, Feuk L, Rivera MN, Listewnik ML, Donahoe PK, Qi Y, Scherer SW, Lee C. Detection of large-scale variation in the human genome. Nat Genet. 2004 Sep; 36(9):949-51.

22. Lars Feuk, Andrew R. Carson and Stephen W. Scherer: Structural variation in the human genomeNature doi: 10.1038/nrg1767

23. Przeworski, M., Hudson, R. R. & Di Rienzo, A.Adjusting the focus on human variation. Trends Genet. 16, 296–302 (2000).

24. Reich, D. E. et al. Human genome sequence variation and the influence of gene history, mutation and recombination. Nature Genet. 32, 135–142 (2002).

25. Yunis JJ, Prakash O: The origin of man: a chromosomal pictorial legacy. Science 1982, 215:1525-1530.

26. Feuk L, Macdonald JR, Tang T, Carson AR, Li M, Rao G, Khaja R, Scherer SW: Discovery of human inversion polymorphisms by comparative analysis of human and chimpanzee DNA sequence assemblies. PLoS Genet 2005, 1:e56.

27. Navarro A, Barton NH: Chromosomal speciation and molecular divergence accelerated evolution in rearranged chromosomes. Science 2003, 300:321-324.

28. Sebat, J. et al. Large-scale copy number polymorphism in the human genome. Science 305, 525–528 (2004).

29. Tuzun, E. et al. Fine-scale structural variation of the human genome. Nature Genet 37, 727–732 (2005).

30. Inoue, K. & Lupski, J. R. Molecular mechanisms for genomic disorders. Annu. Rev. Genomics Hum. Genet. 3, 199–242 (2002)

31.Adrianto I., Wen F., Templeton A., Wiley G., King J.B., Lessard C.J., Bates J.S., Hu Y., Kelly J.A., Kaufman K.M. *et al.* (2011) Association of a functional variant downstream of TNFAIP3 with systemic lupus erythematosus. *Nat. Genet.*, **43**, 253-258.

32. Stephens P.J., McBride D.J., Lin M.L., Varela I., Pleasance E.D., Simpson J.T., Stebbings L.A., Leroy C., Edkins S., Mudie L.J. *et al.* (2009) Complex landscapes of somatic rearrangement in human breast cancer genomes. *Nature,* **462**, 1005-1010

33. Leary R.J., Kinde I., Diehl F., Schmidt K., Clouser C., Duncan C., Antipova A., Lee C., McKernan K., De La Vega F.M. *et al.* (2010) Development of personalized tumor biomarkers using massively parallel sequencing. *Sci. Transl. Med.,* **2**, 20ra14

34. Conrad D.F., Andrews T.D., Carter N.P., Hurles M.E. and Pritchard J.K. (2006) A high-resolution survey of deletion polymorphism in the human genome. *Nat. Genet.,* **38**, 75-81.

35. Park, H. et al. Discovery of common Asian copy number variants using integrated high-resolution array CGH and massively parallel DNA sequencing. Nature Genet. 42, 400–405 (2010)

36. Redon, R. et al. Global variation in copy number in the human genome. Nature 444, 444–454 (2006).

37. Kidd, J. M. et al. Mapping and sequencing of structural variation from eight human genomes. Nature 453, 56–64 (2008).

38. Conrad, D. F. et al Origins and functional impact of copy number variation in the human genome. Nature 464, 704–712 (2010)

39. McCarroll, S. A. et al. Integrated detection and population-genetic analysis of SNPs and copy number variation. Nature Genet. 40, 1166–1174 (2008)

40. Perry, G. H. et al. The fine-scale and complex architecture of human copy-number variation. Am. J. Hum. Genet. 82, 685–695 (2008)

41. Lubs, H. A. A marker X chromosome. Am. J. Hum.Genet. 21, 231–244 (1969).

42. Hinds, D. A. et al. Whole-genome patterns of common DNA variation in three human populations. Science 307, 1072–1079 (2005)

43. The International HapMap Consortium. A haplotype map of the human genome. Nature 437, 1299–1320 (2005).

44. Solinas-Toldo, S. et al. Matrix-based comparative genomic hybridization: biochips to screen for genomic imbalances. Genes Chromosomes Cancer 20, 399–407 (1997).

45. Venter, J. C. et al. The sequence of the human genome. Science 291, 1304–1351 (2001)

46. Dhami, P. et al. Exon array CGH: detection of copynumber changes at the resolution of individual exons in the human genome. Am. J. Hum. Genet. 76, 750–762 (2005)

47. Maegenis, R. E., Donlon, T. A. & Wyandt, H. E. Giemsa-11 staining of chromosome 1: a newlydescribed heteromorphism. Science 202, 64–65 (1978)

48. Coe, B. P. et al. Resolving the resolution of array CGH. Genomics 89, 647–653 (2007)

49. Itsara, A. et al. Population analysis of large copynumber variants and hotspots of human genetic disease. Am. J. Hum. Genet. 84, 148–161 (2009)

50. Locke, D. P. et al. BAC microarray analysis of 15q11–q13 rearrangements and the impact of segmental duplications. J. Med. Genet. 41, 175–182 (2004)

51. Bailey, J. A. et al. Recent segmental duplications in the human genome. Science 297, 1003–1007(2002)

52. Schwartz, D. C. et al. Ordered restriction maps of Saccharomyces cerevisiae chromosomes constructed by optical mapping. Science 262, 110–114 (1993).

53. Teague, B. et al. High-resolution human genome structure by single-molecule analysis. Proc. Natl Acad. Sci. USA 107, 10848–10853 (2010).

54. Antonacci, F. et al. A large and complex structural polymorphism at 16p12.1 underlies microdeletion disease risk. Nature Genet. 42, 745–750 (2010).

55. Jay Shendure & Hanlee Ji Next-generation DNA Sequencing Nature 1,1135-1145 (2008)

56. Mitra, R.D.; Church G.M. In situ localized amplification and contact replication of many individual DNA molecules. Nucleic Acids Res. 1999, 27, e34

57. Rusk N (2011). "Torrents of sequence". *Nat Meth* **8** (1): 44–44.

58. Quail, Michael; Smith, Miriam E; Coupland, Paul; Otto, Thomas D; Harris, Simon R; Connor, Thomas R; Bertoni, Anna; Swerdlow, Harold P; Gu, Yong (1 January 2012). "A tale of three next generation sequencing platforms: comparison of Ion torrent, pacific biosciences and illumina MiSeq sequencers". *BMC Genomics* **13** (1): 341.

59. Liu, Lin; Li, Yinhu; Li, Siliang; Hu, Ni; He, Yimin; Pong, Ray; Lin, Danni; Lu, Lihua; Law, Maggie (1 January 2012). "Comparison of Next-Generation Sequencing Systems". *Journal of Biomedicine and Biotechnology* **2012**: 1–11

60. Alberto M., Matteo B., Alessia G., Francesca G.,Francesca T. and Maria L B. ;

Bioinformatics for Next Generation Sequencing Data. Genes 2010, 1, 294-307

61. Nyren, P.; Lundin, A. Enzymatic method for continuous monitoring of inorganic pyrophosphate synthesis. Anal. Biochem. 1985, 151, 504–509

62. Fedurco, M.; Romieu, A.; Williams, S.; Lawrence, I.; Turcatti, G. BTA, a novel reagent for DNA attachment on glass and efficient generation of solid-phase amplified DNA colonies. Nucleic Acids Res. 2006, 34, e22

63. Adessi, C.; Matton, G.; Ayala, G.; Turcatti, G.; Mermod, J.J.; Mayer, P.; Kawashima, E. Solid phase DNA amplification: characterisation of primer attachment and amplification mechanisms. Nucleic Acids Res. 2000, 28, e87

64. Illumina Home Page. http://www.illumina.com/.

65. ABI SOLiD Home Page http://www.appliedbiosystems.com

66. She, X. et al. Shotgun sequence assembly and recent segmental duplications within the human genome. Nature 431, 927–930 (2004)

67. Alkan, C., Sajjadian, S. & Eichler, E. E. Limitations of next-generation genome sequence assembly. Nature Methods 8, 61–65 (2011).

68.Levy S, Sutton G, Ng PC, Feuk L, Halpern AL, et al. (2007) The Diploid Genome Sequence of an Individual Human. PLoS Biol 5(10): e254. doi:10.1371/journal.pbio.0050254

69.Bruno Z.; Valentina B.; Isabelle JL.; Sophie L.; Patricia Ln; Alain N.; Olivier D.; Emmanuel B. :SVDetect: a tool to identify genomic structural variations from paired-end and mate-pair sequencing data. Bioinformatics 2010 26: 1895-1896

70. Chen, K. et al. (2009) BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. Nat Methods 6, 677-681.

71. Homer, N, and Merriman, B. TMAP: the Torrent Mapping Alignment Program. In Preparation

72. Li H.*:* wgsim - Read simulator for next generation sequencing http://github.com/lh3/wgsim

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