ssahaSNP – A Polymorphism Detection Tool by Genomic Alignment

James C. Mullikin[1], Adam Spargo[2], Nikolai Ivanov[2], Mario Caccamo[2] and Zemin Ning[2*]

[1]Genome Technology Branch/NHGRI, NIH, 5625 Fishers Lane, Bethesda, MD 20852, USA
[2]The Wellcome Trust Sanger Institute, Wellcome Trust Genome Campus,
Hixton, Cambridge CB10 1SA UK

*Corresponding author
E-mail address: zn1@sanger.ac.uk

Abstract

Background: Fast and accurate detection of genomic polymorphism is increasingly under demand, fuelled by many large genome projects. This paper presents a software package which can detect homozygous SNPs and insertion/deletion events on a eukaryotic genome scale from millions of shotgun reads. Matching seeds of a few kmer words are found to locate the position of the read on the genome. Full sequence alignment is then performed to detect base variations. Quality values of both variation bases and neighbouring bases are checked to exclude possible sequence base errors. To increase the accuracy of detection in some cases, it requires that the same mutation event is mapped by two or more shotgun reads.

Results: To demonstrate the accuracy of the system, we present variation results compared with finished sequences. Flexibility of the tool in processing different data types is shown by comparing two different strains of *Streptococcus suis,* one sequenced by traditional ABI machines, and the other by 454 technology. The effect of read coverage on SNP/indel accuracy has also been examined. To illustrate the scalability of the package in handling large eukaryotic genomes, we present results from the zebrafish sequencing project at the Sanger Institute with 20 million WGS reads against the draft WGS assembly as well as 1.6 million flow sorting reads against human chromosome 6.

Conclusions: The ssahaSNP package is ideal for large genome projects, particularly in cases where read coverage is not high and draft genome assembly is not in a good shape.

Background

Identification of genetic differences among individuals or species has applications in many fields, such as clinical diagnostics of cancer related diseases, evolutionary history (phylogeny) of species, and even heterozygosity analysis in genome sequencing. Fast and accurate detection of genomic polymorphism is increasingly under demand, particularly now that the human and mouse genome reference sequences are finished or near completion. In the past few years, a number of systems have been developed with different methods of mining genomic polymorphisms, such as Gap4 [1], POLYBASES [2], POLYPHRED [3], PTA [4], TGICL [5], autoSNP [6], miraEST [7], and SeqDoC [8]. Some of them provide visual comparison of sequence traces in local BAC regions, while some systems assemble ESTs first and then detect SNPs in the resulting alignments. For large scale SNP mining projects, recent studies were reported using reduced representation shotgun (RRS) [9], and whole genome alignment by placing a randomly shotgun read to the genome [10-11]. However, the efforts coordinated by The SNP Consortium (TSC) were mainly focused on single nucleotide polymorphisms (SNP). There is a need for systematic studies on point, local and structured polymorphisms. A study on insertions/deletions (indels) was recently reported [12], where BL2Seq, one of the BLAST family programs, was used as the alignment tool. Apart from read alignment, the whole pipeline system developed by Devine and colleagues requires significant efforts in data processing, such as repeat masking, quality value assignment/tracking, and further processing for indels with a length >16 bps. Within the biomedical communities, mutation detection tools with good accuracy and multiple functions are in high demand. In this paper, we outline a package ssahaSNP which quickly detects both SNPs and indels without any sequence repeat masking.

In SNP/indel discovery using genomic alignment, the most reliable method would be multiple sequence alignment of all the traces, compared to the reference sequence in a local region. The trace DNA can be from a single source or mixed populations. If read coverage is low, say <4x, this method will not work very well, not to mention intensive computation for large projects. In sequence assembly, where the source DNA and reference are normally the same, consensus is generated from multiple read alignment. In this case, if we align every read to the consensus sequence, multiple read alignment can be reconstructed from individual alignments as aligned positions of each base for each read are based on a common reference (consensus). When homology between source and reference is relatively high, we may use the same idea for SNP/indel detection. SNP/indel candidates are identified from individual read alignment against the reference. After all the alignment is done, candidates which share the same locus are assembled together to confirm a valid SNP/indel, as shown in Figure 1. At a high level of read coverage, SNPs/indels are mapped a number of times with high accuracy, while polymorphism detection can still be carried out with a reduced level of accuracy at low read coverage.

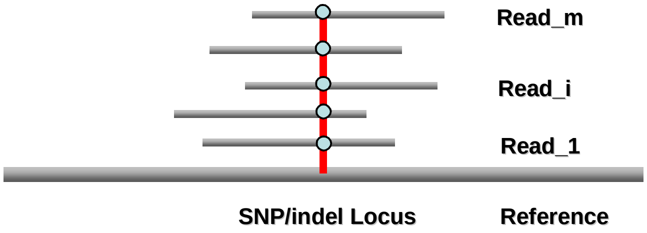


Figure 1. SNP/indel re-assembly after individual variations detected from read alignment. The SNP/indel indicated here is mapped by m reads.

Implementation

SSAHA2 and ssahaSNP

SSAHA2 [13] is a sequence alignment package developed at The Wellcome Trust Sanger Institute, which combines SSAHA [14] with phrap/cross_match [15]. SSAHA achieves its fast search speed by converting sequence information into a "hash table" data structure, which can then be searched very rapidly for matches. A few exactly matched kmer words, the matching seeds, are detected from the database by the SSAHA algorithm. When the location information of matching seeds is obtained, we then cut off both query and subject sequences and pass the two sequences to cross_match for full alignment. A given edge length is added to both query and subject to extend the alignment length, shown in Figure 2. In terms of software implementation, alignment functions from cross_match have been imbedded into the SSAHA system.

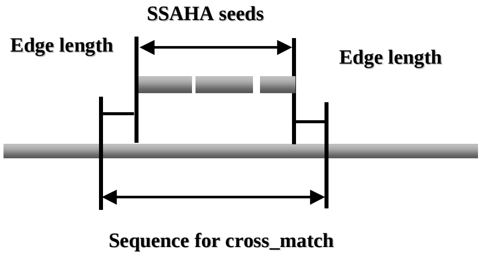


Figure 2. Extra sequences with a given edge length are used to extend the length of alignment.

As a fast tool capable of efficient processing of large data sets, ssahaSNP was used in the SNP detection by the international SNP consortium [10]. In the early version of ssahaSNP, there was an alignment module in the package. However, the alignment quality was not good enough to handle middle sized indels, say > 10 bps. Also SNP calling was carried out for every piece of alignment on the genome and it then relied on the parsing code to exclude those SNP candidates which are mapped multiple times. In the new version of ssahaSNP, we use SSAHA2 as the alignment tool to place genomic reads on finished or draft assembly sequences. Highly repetitive elements are filtered out by ignoring kmer words with high occurrence. We place less repetitive or non-repetitive reads uniquely on the reference genome sequence and find the best alignment according to the pair-wise alignment score if there are multiple seeded regions. If a read has more than one piece of alignment and the same best alignment score is shared by more than one hit , this read is regarded as repetitive and is therefore ignored by ssahaSNP from further data processing.

Memory usage and speed

Detailed descriptions of our hashing algorithm can be found in [14]. For each base in the query sequence, we use one character to store DNA base pairs and one character for quality scores. For the subject sequence, one character is used to store the DNA sequence of every base pair. In the hash table, we store two elements as integers - one for sequence index; one for offset of the kmer words on the sequence. To run the system, the memory required is

$$M = 2 * N_q + 8 * N_s / k + N_s + 4 * 2^{2k} + M_{cross\_match} \qquad (1)$$

where k = kmer word size (mostly 12 by default);

N_q = number of query bases;

N_s = number of subject bases;

$M_{\text{cross\_macth}}$ = memory assigned for running cross_match (200 MB).

To search reads against the human genome NCBI35 (~3.43Gbps), the minimum memory requirement for this task would be ~6.0 GBs of RAM memory in the machine. In cases where there are memory restrictions, the analysis can still be done by splitting the subject sequence in smaller chunks. But this requires an extra phase to find the best alignment in the genome, rather than in each subject file.

In the new version of ssahaSNP, the alignment quality has been significantly improved. Applications are extended from SNP calling only in near exactly matched sequences to indel detections where match identity can be as low as 80%. Due to the cross_match implementation, however, the speed of the new system is much slower than the old version, which can process reads at a rate of 180 per second against the human genome[14]. Using a SGI Altix 3000 (1.5 GHz, Itanium 2), the new ssahaSNP's speed is about 10 reads per second against the human genome. Another difference is in repeat handling which also slows down the system. In the old version, if any kmer words occur more than 7 times, they are ignored in further data processing. This effectively filters out almost all the repetitive sequences. In the new version, this threshold number is increased to 50000 by default. To reduce detection errors while increasing search sensitivity, we have introduced a number of new features, such as best alignment selection, read pair constrains and location of mutation event mapping by multiple reads.

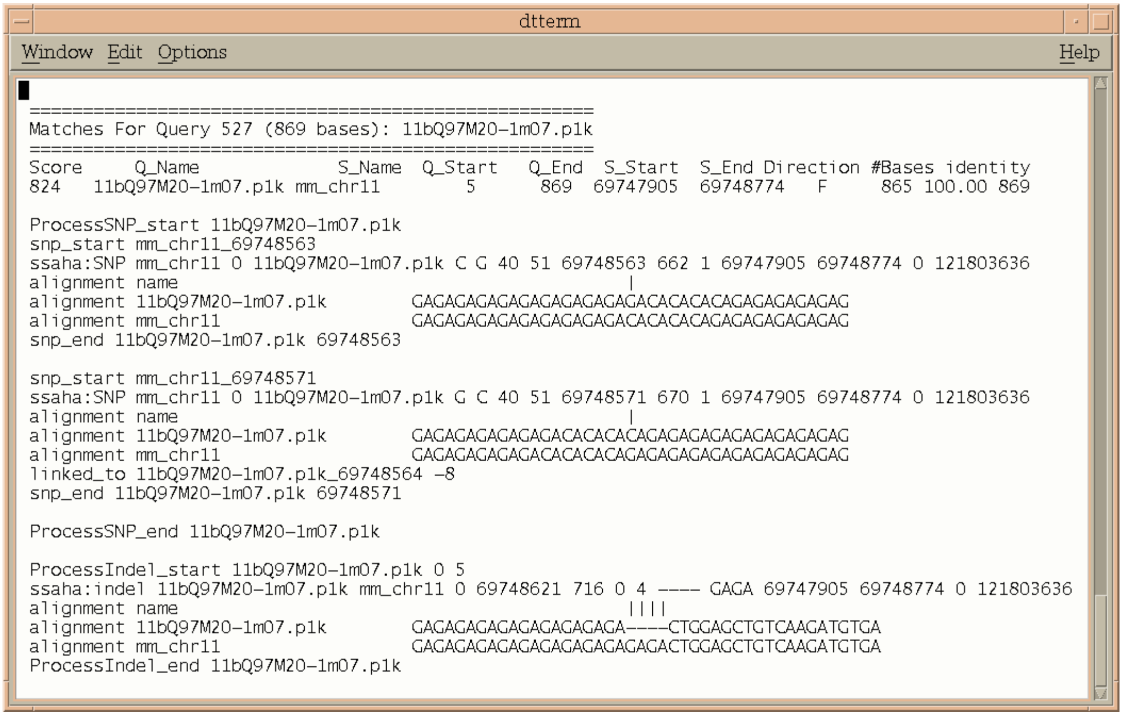
Neighborhood quality standard (NQS)

From the best alignment, SNP candidates are screened, taking into account the quality value (Q value) of the base with the variation as well as the quality values in the neighbouring bases, using neighborhood quality standard (NQS) [9-11]. The standard can be described as three attached conditions: (a) The quality value of the SNP base should be >= 23; (b) The Q value for the 5 bases on either side the SNP should be >=15; (c) Only one mismatch is allowed in the flanking ten bases. As an international standard, NQS has been widely used, particularly in the human SNP project coordinated by The SNP Consortium (TCS). However, this standard has a number of limitations. Even in the comparison of two individual human beings, where the match identity is very high and SNPs are very sparse, NQS consequently excludes most of the doublets and all the triplets. While searching SNPs between two less homologous genomes or two different strains, a significant portion of SNPs will be missed. Very recently, sequencing machines produced by 454 Life Sciences [16][17] offer a rapid way to sequence an entire bacterial genome in one or a few runs. The massively parallel system developed by Rothberg and colleagues [16] is capable of sequencing 25 million bases in a four-hour run – about 100 times faster than the current Sanger sequencing platform. The 454 reads have an average length of ~100 bps and the quality value assigned to each base is not directly correlated to its neighbors if the two base pairs are different. Unlike traditional ABI reads, low quality bases in the 454 reads are not normally located at the two ends of a read. Instead, they are distributed equally across the read. Therefore it is not possible to remove those low quality bases by quality trimming. Also quite often, base pairs in the 454 reads with a very low Q value or even with a zero score are still correct bases, partly because 454 software tools are still in the early stages under development. Under these circumstances, direct use of NQS would not work well for this new type of reads. In ssahaSNP, we have introduced a number of flag options such as “-454 1”, “-NQS 0” and “-quality 20” for selection. The effect of these flags on 454 reads will be discussed in the next section.

For insertions/deletions, there is no widely accepted quality standard. In ssahaSNP, we still use NQS for single base deletions to the reference sequence. For other indels with a length greater than one base, we don’t check quality values. To ensure the indels are detected with high confidence, a conservative method is adopted; we only report the cases in which exactly the same indel is mapped by two or more shotgun reads, or indels with a length >= 10 bps are mapped by single reads.

SNP/indel parser

For most genome projects, the read coverage is normally more than one, i.e. the genome is covered by the shotgun reads more than once. When read coverage is larger than 1.0 X, we need to make sure that the detected SNPs/indels are counted once. The ssahaSNP output file contains the location, sequence variation, and length information of the SNPs or indels found in the best alignment. To visualize the variations, query and subject are aligned in a way that 20 bases are on the left side and 20 bases on right side while SNP/indel bases are in the middle. The format of output files produced by ssahaSNP is shown in Figure 3, while information for each SNP is followed by "ssaha:SNP", each indel is followed by "ssaha:indel". Users are encouraged to develop their own parsers to process the ssahaSNP output file for their specific applications. Together with the release of the main code, we also provide two programs, parse_SNP and parse_indel.



```
=====================================================
Matches For Query 527 (869 bases): 11bQ97M20-1m07.p1k
=====================================================
Score      Q_Name                S_Name  Q_Start   Q_End  S_Start  S_End Direction #Bases identity
824   11bQ97M20-1m07.p1k mm_chr11         5      869  69747905  69748774    F      865 100.00 869

ProcessSNP_start 11bQ97M20-1m07.p1k
snp_start mm_chr11_69748563
ssaha:SNP mm_chr11 0 11bQ97M20-1m07.p1k C G 40 51 69748563 662 1 69747905 69748774 0 121803636
alignment name                                              |
alignment 11bQ97M20-1m07.p1k            GAGAGAGAGAGAGAGAGAGAGACACACACAGAGAGAGAGAG
alignment mm_chr11                      GAGAGAGAGAGAGAGAGAGACACACACACAGAGAGAGAGAG
snp_end 11bQ97M20-1m07.p1k 69748563

snp_start mm_chr11_69748571
ssaha:SNP mm_chr11 0 11bQ97M20-1m07.p1k G C 40 51 69748571 670 1 69747905 69748774 0 121803636
alignment name                                              |
alignment 11bQ97M20-1m07.p1k            GAGAGAGAGAGACACACACAGAGAGAGAGAGAGAGAGAGAG
alignment mm_chr11                      GAGAGAGAGAGACACACACACAGAGAGAGAGAGAGAGAGAG
linked_to 11bQ97M20-1m07.p1k_69748564 -8
snp_end 11bQ97M20-1m07.p1k 69748571

ProcessSNP_end 11bQ97M20-1m07.p1k

ProcessIndel_start 11bQ97M20-1m07.p1k 0 5
ssaha:indel 11bQ97M20-1m07.p1k mm_chr11 0 69748621 716 0 4 ---- GAGA 69747905 69748774 0 121803636
alignment name                                              ||||
alignment 11bQ97M20-1m07.p1k            GAGAGAGAGAGAGAGAGAGA----CTGGAGCTGTCAAGATGTGA
alignment mm_chr11                      GAGAGAGAGAGAGAGAGAGAGAGACTGGAGCTGTCAAGATGTGA
ProcessIndel_end 11bQ97M20-1m07.p1k
```

Figure 3 Format of ssahaSNP output file, while both query and subject sequences are aligned to show the variation.

The parsing code first examines the number of SNPs or indels for each read. If the number is higher than the threshold value (defined by "-copy" option and by default, 30 set for SNP and 5 for indel), this read will not be used for further data processing. For SNP/indel discovery on a genome scale, repetitive elements in the genome are a major problem. Wrongly placed reads can results in many false positive SNPs/indels. Sequence alignment alone cannot solve near-exact repeats. In the parsing code, we use read pair information to exclude those reads which are repetitive and have been placed in a wrong location. For the two pair end reads, we check if the insert size and alignment direction are in line with the pair condition, i.e. the calculated insert size has to be less than the given insert size and alignment direction has to be one forward and one reverse complement. By default, the given insert size is set to 180000, but users can set the value by using the flag "-insert" based on their own data. Figure 4 shows the calculation of insert size, under the circumstances that the two paired reads are mapped to the same contig or to different contigs. The parsing code ignores all the mis-paired reads and therefore, SNPs/indels in those reads are not included in the final results. The read pair formats accepted by the current code are (p, q), (x, y), (x, z), (y, z) and (b, g). For instance, aaa.p1k and aaa.q1k, or aaa.b1 and aaa.g1 will be treated as read pairs. SNPs/indels mapped by unpaired reads are treated as those mapped by paired reads. The users can also switch off the read pair flag by the use of "-pair 0", under which every read will be processed. The formats of output files for SNPs and indels are shown in Figure 5 and Figure 6 respectively. For each SNP or indel reported, location, mapping number, variation base(s) and mapping reads are all listed for further analysis.

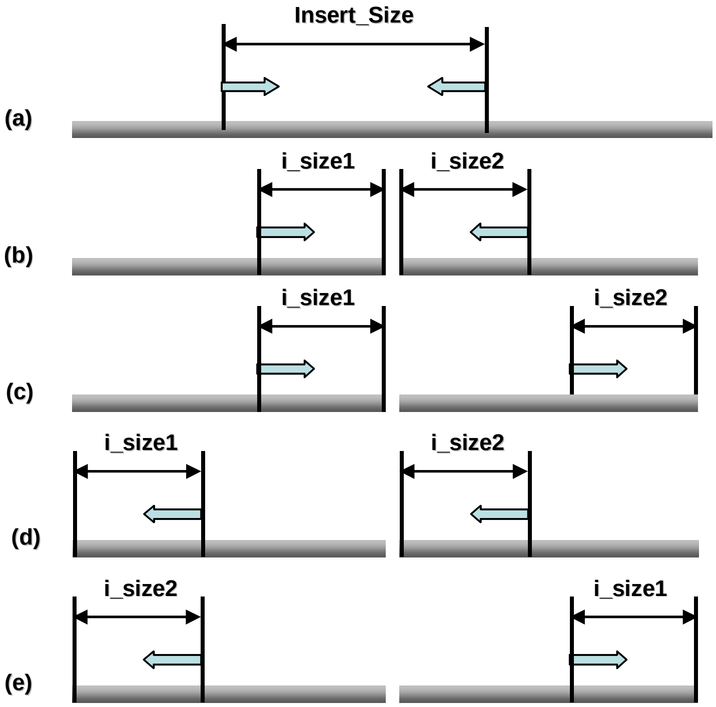
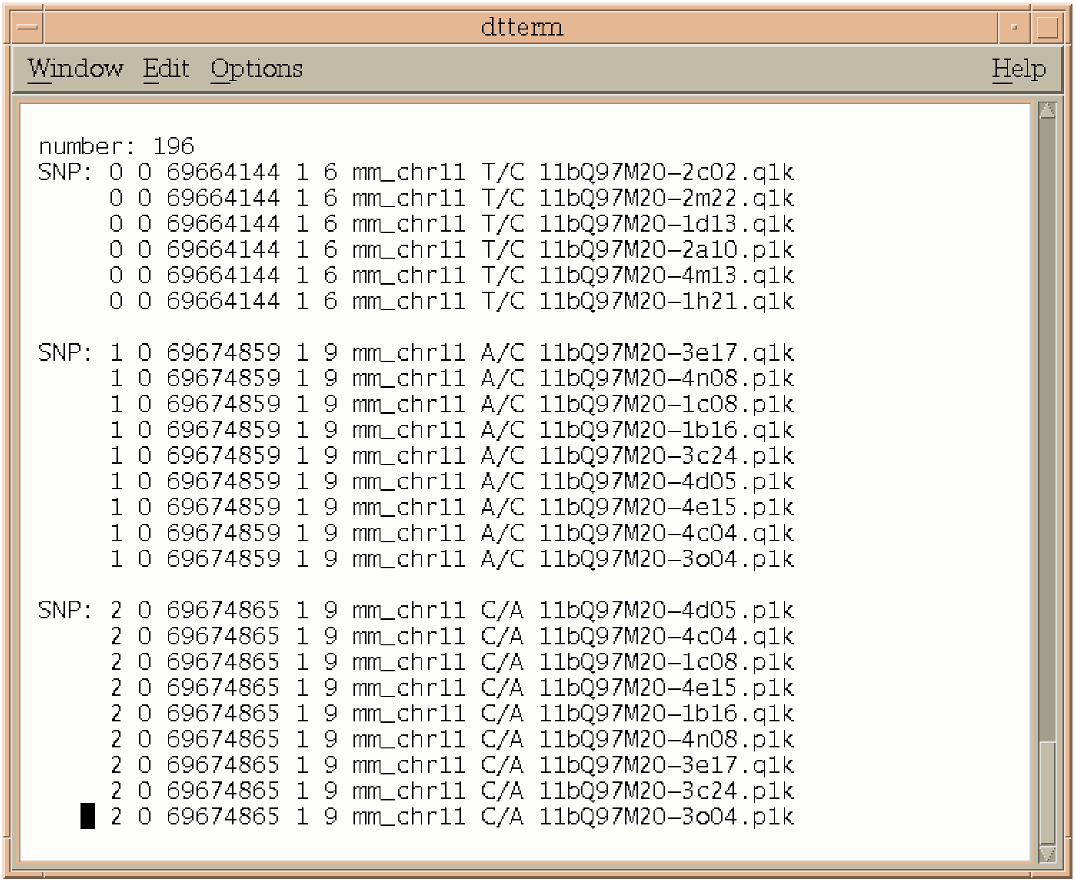


Figure 4. Insert size calculation of two reads under various cases: (1) two paired reads on the same contig (a); (2) two paired reads on different contigs, where insert_size = i_size1 + i_size2 (b, c, d and e).

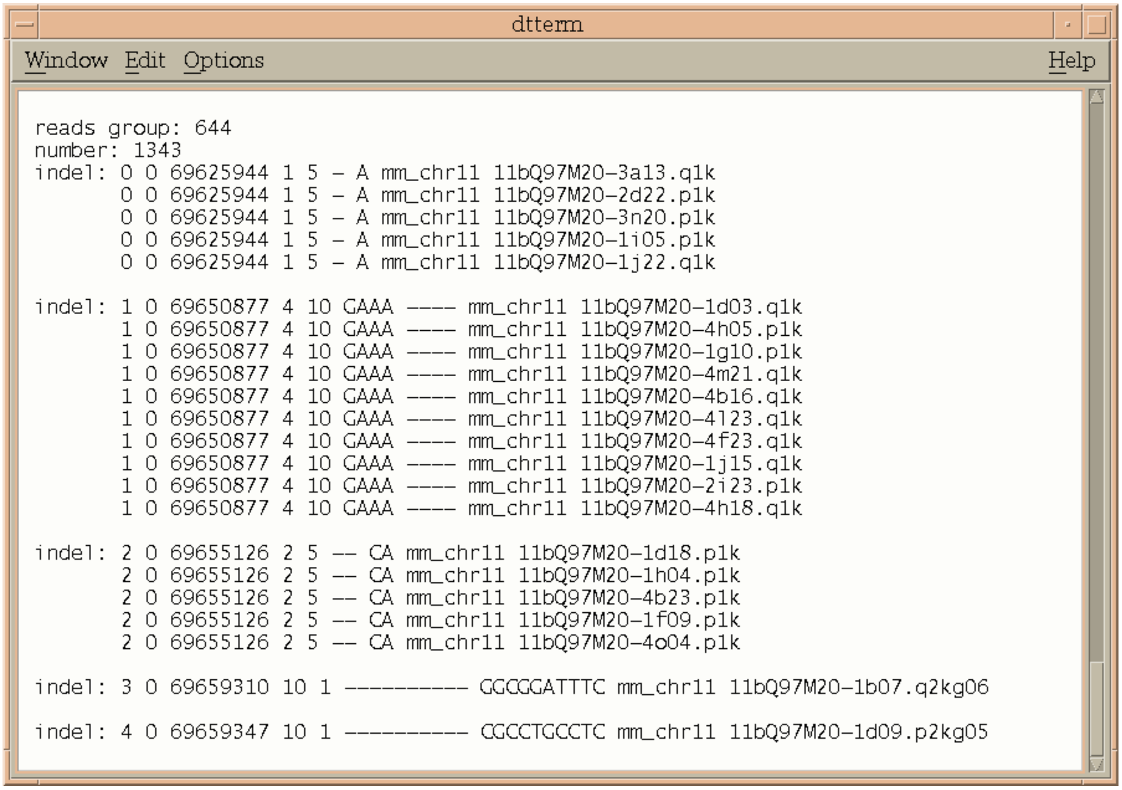


```
number: 196
SNP: 0 0 69664144 1 6 mm_chr11 T/C 11bQ97M20-2c02.q1k
     0 0 69664144 1 6 mm_chr11 T/C 11bQ97M20-2m22.q1k
     0 0 69664144 1 6 mm_chr11 T/C 11bQ97M20-1d13.q1k
     0 0 69664144 1 6 mm_chr11 T/C 11bQ97M20-2a10.p1k
     0 0 69664144 1 6 mm_chr11 T/C 11bQ97M20-4m13.q1k
     0 0 69664144 1 6 mm_chr11 T/C 11bQ97M20-1h21.q1k

SNP: 1 0 69674859 1 9 mm_chr11 A/C 11bQ97M20-3e17.q1k
     1 0 69674859 1 9 mm_chr11 A/C 11bQ97M20-4n08.p1k
     1 0 69674859 1 9 mm_chr11 A/C 11bQ97M20-1c08.p1k
     1 0 69674859 1 9 mm_chr11 A/C 11bQ97M20-1b16.q1k
     1 0 69674859 1 9 mm_chr11 A/C 11bQ97M20-3c24.p1k
     1 0 69674859 1 9 mm_chr11 A/C 11bQ97M20-4d05.p1k
     1 0 69674859 1 9 mm_chr11 A/C 11bQ97M20-4e15.p1k
     1 0 69674859 1 9 mm_chr11 A/C 11bQ97M20-4c04.q1k
     1 0 69674859 1 9 mm_chr11 A/C 11bQ97M20-3o04.p1k

SNP: 2 0 69674865 1 9 mm_chr11 C/A 11bQ97M20-4d05.p1k
     2 0 69674865 1 9 mm_chr11 C/A 11bQ97M20-4c04.q1k
     2 0 69674865 1 9 mm_chr11 C/A 11bQ97M20-1c08.p1k
     2 0 69674865 1 9 mm_chr11 C/A 11bQ97M20-4e15.p1k
     2 0 69674865 1 9 mm_chr11 C/A 11bQ97M20-1b16.q1k
     2 0 69674865 1 9 mm_chr11 C/A 11bQ97M20-4n08.p1k
     2 0 69674865 1 9 mm_chr11 C/A 11bQ97M20-3e17.q1k
     2 0 69674865 1 9 mm_chr11 C/A 11bQ97M20-3c24.p1k
     2 0 69674865 1 9 mm_chr11 C/A 11bQ97M20-3o04.p1k
```

Figure 5 Format of parse_SNP output file.



```
reads group: 644
number: 1343
indel: 0 0 69625944 1 5 - A mm_chr11 11bQ97M20-3a13.q1k
       0 0 69625944 1 5 - A mm_chr11 11bQ97M20-2d22.p1k
       0 0 69625944 1 5 - A mm_chr11 11bQ97M20-3n20.p1k
       0 0 69625944 1 5 - A mm_chr11 11bQ97M20-1i05.p1k
       0 0 69625944 1 5 - A mm_chr11 11bQ97M20-1j22.q1k

indel: 1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-1d03.q1k
       1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-4h05.p1k
       1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-1g10.p1k
       1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-4m21.q1k
       1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-4b16.q1k
       1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-4l23.q1k
       1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-4f23.q1k
       1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-1j15.q1k
       1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-2i23.p1k
       1 0 69650877 4 10 GAAA ---- mm_chr11 11bQ97M20-4h18.q1k

indel: 2 0 69655126 2 5 -- CA mm_chr11 11bQ97M20-1d18.p1k
       2 0 69655126 2 5 -- CA mm_chr11 11bQ97M20-1h04.p1k
       2 0 69655126 2 5 -- CA mm_chr11 11bQ97M20-4b23.p1k
       2 0 69655126 2 5 -- CA mm_chr11 11bQ97M20-1f09.p1k
       2 0 69655126 2 5 -- CA mm_chr11 11bQ97M20-4o04.p1k

indel: 3 0 69659310 10 1 ---------- GGCGGATTTC mm_chr11 11bQ97M20-1b07.q2kg06

indel: 4 0 69659347 10 1 ---------- CGCCTGCCTC mm_chr11 11bQ97M20-1d09.p2kg05
```

Figure 6 Format of parse_indel output file.

Results and discussions

Data validation:

Accuracy is crucial for any mutation detection tools. The likelihood of true polymorphism for the detected candidates of SNP/indel should be high enough and false positive rate should be kept at a minimum level. High accuracy, however, can be achieved by setting a higher level of quality standard, which leads to a smaller number of SNPs/indels and this consequently increases the costs of the project. It was reported by The International SNP Map Working Group that the accuracy for SNP candidates can be as high as 95% using NQS, while examining 24 random examples, using the old version of ssahaSNP [14]. In this paper, we have reduced threshold quality value for the mutation locus as well as for the neighboring bases. Most importantly, quality score profiles in the 454 reads are significantly different from traditional ABI reads. We therefore need to have a detailed examination on how this will affect SNP calling as well as indel detections.

The first test dataset is on some clone data from the NOD (Non-Obese Diabetic) mouse relevant to type 1 diabetes against C57B6/J mouse reference genome sequence NCBI_m34 (http://www.sanger.ac.uk/Projects/M_musculus-NOD/). The regions of the NOD mouse were sequenced to about 10x shotgun coverage and the clone contigs were then finished for further analysis and manual annotation. A few clones were selected to assess the accuracy of mutation detection. We first align the finished clone contigs against the reference sequence. For a given region or a contig, numbers of SNPs or indels as well as locations can be calculated from the alignment. The results may be refereed as reference dataset. We then use ssahaSNP package to process NOD mouse clone reads against the reference sequence to call SNPs and indels. By comparing the ssahaSNP dataset with the reference dataset, information such as true positives and false positives can therefore be obtained. Table 1 shows the true positive and false positive rates at various levels of read coverage for SNP detection as well as for indels. It is seen that even under 2x coverage, all the indels were captured by the detection system. For SNP calling at 4x and 6x, there was one SNP which was not found in the reference dataset. It should be noted that those SNPs detected by ssahaSNP but not in the reference dataset might not necessarily be the errors. It seemed that there was a degree of heterozygosity in the reads and base variations mapped by two different reads could be heterozygous SNPs as the mapping number used was 2. Table 2 shows the SNP and indel detection for clone contig bQ276O13. Again, we had a few false positive indels by comparing to the reference dataset, which could be heterozygous indels as well.

Table 1: NOD mouse clone bQ276O13 against NCBI_M34 chr03 – 4 SNPs and 8 indels

Coverage	True Positive	False Positive (SNP)	True Positive	False Positive (indel)
2x	1 (25.0 %)	0 (0.0 %)	8 (100.0 %)	0 (0.0 %)
4x	4 (100.0 %)	1 (25.0 %)	8 (100.0 %)	0 (0.0 %)
6x	4 (100.0 %)	1 (25.0 %)	8 (100.0 %)	0 (0.0 %)
8x	3 (75.0 %)	0 (0.0 %)	8 (100.0 %)	0 (0.0 %)
10x	4 (100.0 %)	0 (0.0 %)	8 (100.0 %)	0 (0.0 %)

Table 2: NOD mouse clone bQ97M20 against NCBI_M34 chr11 – 17 SNPs* and 33 indels#

Coverage	True Positive	False Positive (SNP)	True Positive	False Positive (indel)
2x	11 (64.71 %)	0 (0.0 %)	19 (57.58 %)	2 (6.06 %)
4x	16 (94.12 %)	0 (0.0 %)	30 (90.91 %)	7 (21.21 %)
6x	16 (94.12 %)	0 (0.0 %)	30 (90.91 %)	2 (6.06 %)
8x	17 (100.0 %)	0 (0.0 %)	32 (96.97 %)	4 (12.12 %)
10x	17 (100.0 %)	0 (0.0 %)	32 (96.97 %)	4 (12.12 %)

*Looking at the alignment, there were 23 SNPs in the reference dataset. However, there are 6 SNPs at the end of contig which were believed to be false positive, or finishing errors by manual examination in the local region. They were excluded from the reference dataset.

#One indel in the reference dataset with 27 bps was believed to be a finishing error and excluded from the reference dataset by manual examination. Two indels in the ssahaSNP dataset were single mapping indels with 10 bps. The false positive rate of indels would be dropped significantly if they were excluded from calculation.

The second test dataset is *Streptococcus suis* with a genome size of ~2.0 Mbps. One strain P1/7 had been sequenced and finished by the Wellcome Trust Sanger Institute (http://www.sanger.ac.uk/Projects/S_suis/). The other strain *S. suis* 89-1591 was sequenced by the JGI (http://genome.jgi-psf.org/draft_microbes/strsu/strsu.home.html). We downloaded the draft assembly from the JGI website and WGS reads from Ensembl trace repository (ftp://ftp.ensembl.org/pub/traces/streptococcus_suis_89_1591/). Using the same method as for the mouse data, we aligned the draft assembly *S. suis* 89-1591 against the Sanger finished sequence to get the reference dataset of SNPs and indels. All the JGI reads were searched against the Sanger finished sequence to get the ssahaSNP dataset. Table 3 shows the effect of read coverage on mutation detection in a region with 571 SNPs and 12 indels. This region is the longest alignment between the two strains with a length of 25135 bps. At a coverage of 4x, for example, the SNP true positive rate can be as high as 97%, while the false positive rate is less than 1.0%. For most genomes, the sequence coverage from contigs at this read coverage is normally 60-70%. This means that if an assembly produced from 4x reads is used to detect mutations, the true positive rate will be 60-70%, much lower than the rate of ssahaSNP. When read coverage is further reduced to 2x, the results from ssahaSNP should be even better as it would be very difficult to build up an assembly at this coverage.

Recently, in order to test the newly acquired 454 sequencing machine, the P1/7 strain was re-sequenced using 454 technology. We used a single machine run's data of 270,741 short reads, estimated to cover the genome 13 times, to test our code. The results with two regions are shown in Table 4, where effect of read coverage is also illustrated. In region_1, the true positive rate is not very high, this is because there are a few areas with high SNP density where 454 reads cannot be placed on the genome. Using a 100 bps window to scan region_1, the maximum number of base variation within the window is 27. There are a few windows in other areas with more than 20 base variations. As 454 reads are about 100 bps on average, SNPs at high density areas will be missed. Alternatively, we selected another region, region_2 with 294 SNPs and 7 indels. The length of this region is 8782 bps and the average SNP density is even higher than that of region_1. However, the maximum number of base variation within a 100 base window is 13, much lower than that in region_1. The true positive rate, shown in Table 4, is higher than that for region_1 at every level of read coverage. As expected, short reads have been aligned to those areas with relatively high SNP density. One distinct feature of 454 reads is the high error rate of insertion and deletion, mostly in areas where single bases are constantly repeated. The indel error rate is estimated at about 3.3% [16][17], i.e, there are 3-4 indels in each 454 read, when aligned to the finished sequence. As a result, the false positive rate for indels is very high. In region_2, there are 7 indels, of which 4 are single base indels. However, using ssahaSNP with "-NQS 1" flag, we detected 94 single base indels and if we don't check quality values for the neighboring bases ("-NQS 0"), the number of single base indels found was 249. It also should be noted that indels with a relatively long length, say >15 bps, are most likely to be missed as the read length would be too short to form an alignment. From these tests, we may conclude that ssahaSNP is not suitable for indel detection using 454 reads.

Table 3: JGI ABI reads against Sanger finished sequence – region_1: 571 SNPs and 12 indels

Coverage	True Positive	False Positive (SNP)	True Positive	False Positive (indel)
2x	514 (90.0%)	4 (0.70%)	9 (75%)	0 (0.0%)
4x	553 (96.85%)	5 (0.88%)	12 (100%)	0 (0.0%)
6x	556 (97.37%)	0 (0.0%)	12 (100%)	0 (0.0%)
8x	564 (98.77%)	1 (0.18%)	12 (100%)	0 (0.0%)
10x	569 (99.65%)	2 (0.35%)	12(100%)	0 (0.0%)

Table 4: 454 reads against JGI draft assembly – region_1 (571 SNPs) and region_2 (294 SNPs)

Coverage	True Positive	False Positive (region_1)	True Positive	False Positive (region_2)
2x	145 (25.39 %)	5 (0.88 %)	149 (50.68 %)	0 (0.0 %)
4x	315 (55.17 %)	10 (1.75 %)	215 (73.13 %)	11 (3.74 %)
6x	299 (52.36 %)	2 (0.35 %)	261 (88.78 %)	18 (6.12 %)
8x	360 (63.04 %)	2 (0.35 %)	270 (91.84 %)	14 (4.76 %)
10x	397 (69.53 %)	3 (0.53 %)	289 (98.30 %)	16 (5.44 %)
13x	435 (76.18 %)	5 (0.88 %)	293 (99.66 %)	16 (5.44 %)

Variation Detections for Larger Eukaryotic Genomes

A whole genome shotgun (WGS) assembly is generated as part of the zebrafish genome project at the Sanger Institute. In the initial phase of WGS reads production, the DNA samples were extracted from more than 1000 five-day-old embryos (http://www.sanger.ac.uk/project/D_rerio). Multiple DNA sources lead to a very high polymorphism level in the dataset and consequently leave tremendous technical challenges in sequence assembling. On the other hand, this polymorphic dataset also offers opportunities for genomic variation studies.

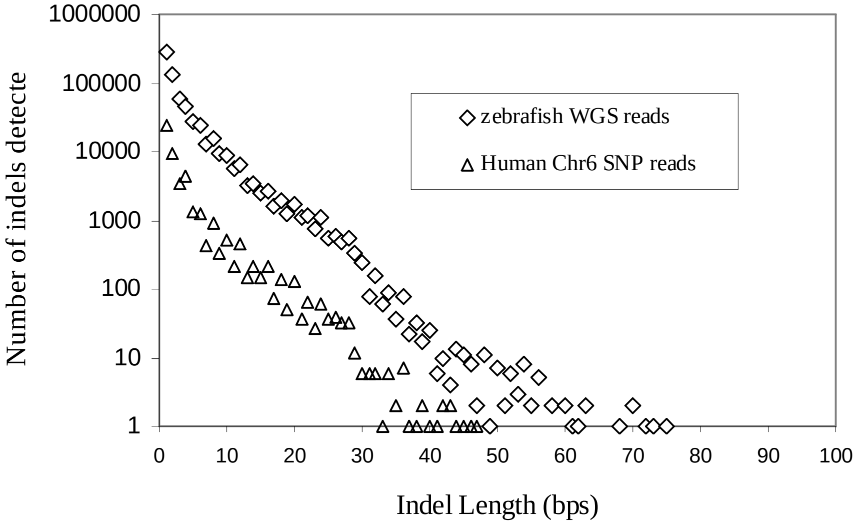


Figure 7. Distribution of the indel length.

We used the ssahaSNP package to detect indels from 20 million WGS reads against the draft WGS assembly as well as the finished clone contigs of 800 Mbps. The total number of detected indels is 663,660. Given the genome size of 1.65 GB, this indicates that the average indel density is at about one indel every 2.48 kilobases. Distribution of indel length over the number of detected indels is shown in Figure 7, where the data of Human Chromosome 6 is superimposed for comparison. In the human dataset, we processed 1.61 millions reads whose DNA samples were from three cell lines. With a total number of 47,692 detected indels, this gives the average indel density at about one indel per 3.58 kilobases. It is seen from the figure that for both datasets the majority of the indels are short, with a length N50 = 2 (half of the indels are less or equal to 2 bps).

In the zebrafish dataset, the shotgun coverage is about 7x and for human Chr6 the coverage is about 6x. Even with multiple haplotypes, it is likely the same indels would be mapped more than two times by the shotgun reads. Figure 8 shows variations of indel number against indel mapping frequency. For the zebrafish dataset, there is a long tail in the figure that extends beyond 100. This is because the WGS assembly is only a draft and many repetitive or long duplicated regions are still not represented in the assembly. For the reads belonging to these gap regions, the correct location does not exist, thus the code maps reads to similar regions to levels that are much higher than the average shotgun coverage. For the finished Chr6 sequence, the situation is much better, with only a very small percentage mapping more often than expected. Human SNPs detected by ssahaSNP can be found in http://www.glovar.org/.

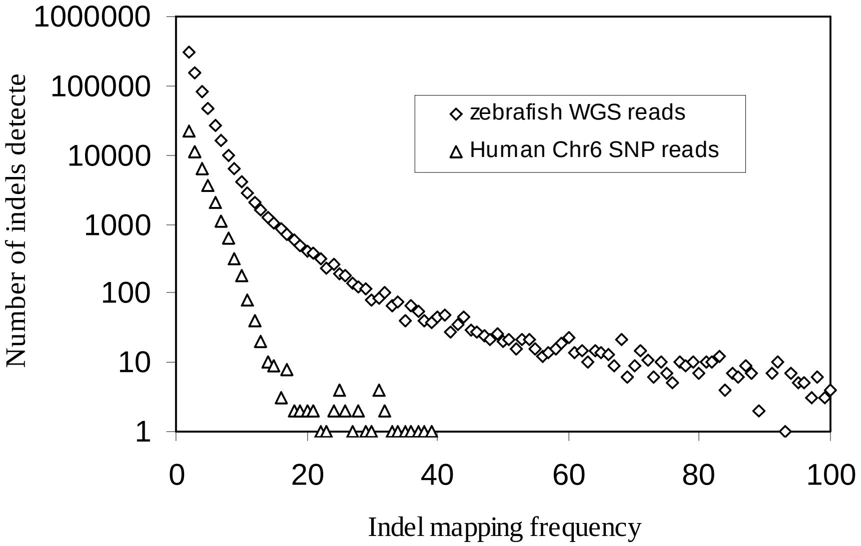


Figure 8. Distribution of indel mapping frequency.

Conclusions

Whole genome alignment provides a fast and effective way for mutation detection. To validate the accuracy of the method, we have carried out tests for various data types. The tool works very well for the case in which there is a high level of homology between the query source DNA and the reference genome, like NOD mouse reads against C57B6/J reference sequence. Good results have also been found when two strains of pathogen sequences are searched against each other. False positive rate is found to be very high for indel detection using 454 reads, especially for single base indels. However, results of SNP detection for 454 reads are acceptable at relative high read coverage. We recommend the package for use in large genome projects, particularly in cases where read coverage is not high and draft assembly is not in a good shape.

Availability and requirements

Project name:	ssahaSNP;
Project home page:	http://www.sanger.ac.uk/Software/analysis/SSAHA2/
Operational systems:	Platform independent;
Programming language:	C and Perl;

Other requirements: None;

Licence: No licence for binary codes;

Any restrictions to use by non-academics: binary code only.

List of abbreviations

SSAHA Sequence Search and Alignment by the Hash Algorithm, SNP single nucleotide polymorphism, indel insertion and deletion, WGS whole genome shotgun, NQS Neighborhood quality standard, ABI Applied Biosystems Incorporation, NOD Non-Obese Diabetic.

Authors' Contributions

JCM initiated the project and implemented the SNP algorithm into the first version of the code. AS and NI imbedded cross_match functions into the SSAHA system for sequence alignment, while AS also rewrote ssahaSNP based on a common SSAHA platform for various applications. MC is responsible for the maintenance of the zebrafish variation data. ZN implemented the indel algorithm and wrote the parsing codes. He was also responsible for SNP/indel validation tests and supervised the project. All the authors read and approved the final manuscript.

Acknowledgement

The authors would like to thank Professor Phil Green, University of Washington, Seattle, who has kindly agreed for the phrap/cross_match package to be used for sequence alignment in the SSAHA system. The project is funded by the Wellcome Trust.

References

1. Bonfield JK, Rada C and Staden R: **Automated detection of point mutations using fluorescent sequence trace subtraction**. *Nucleic Acids Res* 1998, **26**:3404-3409.

2. Marth GT, Korf I, Yandell MD, Yeh RT, Gu Z, Zakeri H, et al. : **A general approach to single-nucleotide polymorphsim Discovery**. *Nat Genet* 1999, **23**: 452-456.

3. Nickerson DA, Taylor SL, and Rieder MJ: **Identifying single nucleotide polymorphisms (SNPs) in human candidate genes**. In *Research abstracts from the DOE human genome program Contractor-Grantee Workshop VIII*. Feb. 27 to Mar. 2, 2000. Santa Fe, NM.

4. Paracel: **PTA: Paracel transcript assembler user manual**. Paracel, Inc., 2002, Pasadena, CA.

5. Pertea G, Huang X, Liang F, Antonescu V, Sultana R, Karamycheva S, Lee Y, White J, Cheung F, Parvizi B, et al.: **TIGR gene indices clustering tools (TGICL): A software system for fast clustering of large EST datasets**. *Bioinformatics 2003*, **19:** 651–652.

6. Barker G, Batley J, O'Sullivan H, Edwards KJ and Edwards D: **Redundancy based detection of sequence polymorphisms in expressed sequence tag data using autoSNP**. *Bioinformatics* 2003, **19**: 421–422.

7. Chevreux B, Pfisterer T, Drescher B, Driesel AJ, Müller WEG, Wetter T and Suhai S**: Using the miraEST Assembler for Reliable and Automated mRNA Transcript Assembly and SNP Detection in Sequenced ESTs.** *Genome Res* 2004**, 14**:1147-1159**.**

8. Crowe, M: **SeqDoC: rapid SNP and Mutation detection by direct comparison of DNA sequence chromatograms.** *Bioinformatics* 2005, **6**:133.

9. Altshuler D, Pollara VJ, Cowles CR, Van Etten WJ, Badwin J, Linton L and Lander ES: **An SNP map of the human genome generated by reduced representation shotgun sequencing.** *Nature* 2000, **407**:513-516;

10. The International SNP Map Working Group: **A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms**. *Nature* 2001, **409**:928-933.

11. Mullikin JC, Hunt SE, Cole CG, Mortimore BJ, et al .: **An SNP map of human chromosome 22**. *Nature* 2000, **407**:516-520.

12. Bennett EA, Coleman LE, Tsui C, Pittard WS and Devine SE **: Natural genetic variation caused by transposable elements in humans.** *Genetics* 2004, **168**:933-951.

13. http://www.sanger.ac.uk/Software/analysis/SSAHA2/

14. Ning Z, Cox AJ and Mullikin JC: **SSAHA: A Fast Search Method for Large DNA Databases.** *Genome Res* 2001, **11**:1725-1729.

15. http://www.phrap.com/

16. Margulies M, Egholm M, Altman W et al. : **Genome sequencing in microfabricated high-density picolitre reactors.** *Nature* 2005, **437**:376-380;

17. Rogers YH and Venter JC: **Massively parallel sequencing**. *Nature* **437**:326-327.