INVERTER: INtegrated Variable numbER Tandem rEpeat findeR

Adrianto Wirawan[1,2], Chee Keong Kwoh[1], Li Yang Hsu[2],
and Tse Hsien Koh[2]

[1] School of Computer Engineering, Nanyang Technological University,
50 Nanyang Avenue, Singapore 639798, Singapore
{adri0004,asckkwoh}@ntu.edu.sg
[2] Department of Medicine, Yong Loo Lin School of Medicine,
National University of Singapore, 5 Lower Kent Ridge Road, Singapore 119074, Singapore
li_yang_hsu@nuhs.edu.sg, koh.tse.hsien@sgh.com.sg

Abstract. A tandem repeat in DNA is a sequence of two or more contiguous, approximate copies of a pattern of nucleotides. Tandem repeats occur in the genomes of both eukaryotic and prokaryotic organisms. They are important in numerous fields including disease diagnosis, mapping studies, human identity testing (DNA fingerprinting), sequence homology and population studies. Although tandem repeats have been used by biologists for many years, there are few tools available for performing an exhaustive search for all tandem repeats in a given sequence. In this paper, we present INVERTER, a *de novo* tandem repeat finder without the need to specify either the pattern or a particular pattern size, integrated with a data visualization tool. INVERTER is implemented in Java and has a built-in user-friendly Graphical User Interface. A standalone version of the program can be downloaded from http://bmserver.sce.ntu.edu.sg/INVERTER. Comparison search result of INVERTER with an existing software tool is presented. The use of INVERTER will assist biologists in discovering new ways of understanding both the structure and function of DNA and protein.

Keywords: Variable Number Tandem Repeat, exact match search, non-exact match search, data visualization.

1 Introduction

Most genomes have a high content of repetitive DNA. Fifty percent of the human genome, for example, consists of repeated sequences[1]. A tandem repeat (TR) in DNA is a sequence of two or more contiguous, approximate copies of a pattern of nucleotides. Variable Number Tandem Repeat (VNTR) is a location in a genome where a short nucleotide sequence is organized as a tandem repeat.

VNTRs appear in biological sequences with a wide variety and occur in the genomes of both eukaryotic and prokaryotic organisms. They are found in both coding and non-coding regions of DNA. Expansions of repeats found in the protein-coding portions of genes can affect the function of the gene by causing synthesis of malfunctioning

J.H. Chan, Y.-S. Ong, and S.-B. Cho (Eds.): CSBio 2010, CCIS 115, pp. 151–164, 2010.

proteins. Repeats in non-coding regions have been shown to affect biological processes by affecting gene expression, transcription and translation.

VNTRs are essential in genetics and biology research. They are important as genetic markers[2] as well as responsible for over 30 inherited diseases in humans. Expansions of simple DNA repeats have been linked to hereditary disorders in humans, including Fragile X Syndrome[3], Myotonic Ddystrophy[4], Huntington's Disease[5], spinal and bulbar muscular atrophy[6], various Spinocerebellar Ataxias and Friedreich's Ataxia[7]. These diseases are sometimes called the *repeat expansion diseases* since they are caused by long and highly polymorphic VNTRs [8, 9]. Tetra- or pentanucleotide VNTRs in the human genome are the genetic markers used in DNA forensics[10]. Since the number of adjacent repeated units varies from individual to individual, the copy number of a tandem repeat can be used to identify an individual, and relations such as parent or grandparent. VNTRs are also used in population studies[11], conservation biology[12] as well as multiple sequence alignments[13].

Although VNTRs have been used by biologists for many years, there are few tools available for performing an exhaustive search for all VNTRs in a given sequence. Popular existing software tools for finding VNTRs in a sequence include: Tandem Repeats Finder (TRF)[14], mreps[15], ATRHunter[16], STRING[17], and T-REKS[18].

One of the difficulties involved in locating VNTRs is in the precision of finding a tandem repeat given its loose definition. Exact repeats, i.e. repeats that do not allow any errors, are clearly defined. Once we introduce errors, such as insertions and deletions of single or multiple bases, we have to define what constitutes a tandem repeat. Each of the tools for locating VNTRs relies on certain assumptions and definitions. Thus, the output of the different tools differs, each offering different insights into the presence of repeated sequences. Furthermore, none of the above software tools provide data visualization of the resulting VNTRs to facilitate users to correlate annotations, observe visual patterns, and view useful statistics of the data.

In this paper, we present INVERTER, a *de novo* tandem repeat finder without the need to specify either the pattern or a particular pattern size, integrated with a data visualization tool. INVERTER is aimed to identify both exact match and non-exact match VNTRs and provide data visualization which would allow users to correlate annotations, observe visual patterns, and view useful statistics of the data. INVERTER is implemented in Java and has a built-in user-friendly Graphical User Interface. The use of INVERTER will assist biologists in discovering new ways of understanding both the structure and function of DNA and protein.

The remainder of the paper is organized as follows. Section 2 explains the problem definition and implementation details of INVERTER. Experimental works on 6 completed projects of *Acinetobacter baumanii* genomes using the INVERTER are presented in Section 3. Section 4 concludes the paper.

2 Method

2.1 Problem Definition

There are two principal families of VNTRs: microsatellites and minisatellites. Microsatellites are short tandemly repeated DNA sequences of 1-6 base pairs in

Table 1. Commonly employed terms for VNTRs

Biological definition	Mathematical/Computational description	Features	Example
Perfect	Exact match	100% identical copies	$(A)_n$, $(ATC)_n$
Imperfect	Approximate-Hamming Distance	Substitutions (=mismatches)	$(AC)_nAT(AC)_m$
Interrupted*	Approximate Edit-Distance	substitutions, insertions, deletions (=interruptions)	$(ACG)_nT(ACG)_m$, $(AT)_nCGAG(AT)_m$
Compound/ complex	'Fuzzy'	multiple motifs, periods, substitutions	$(ACG)_nT(TC)_m$

* Interrupted repeats are often included in imperfect repeats

length[19]. Minisatellites, on the other hand, consists of moderately 10-100 base pairs of DNA sequences[20].

It is well known that sequences are subject to many kinds of modifications, such as point mutations, i.e. substitutions, insertions and deletions (indels for short), and expansions, i.e. exact tandem replications of some tracts. Table 1 shows the commonly employed terms for VNTRs based on their types of repeats[21].

Given a sequence S of length l, $S = \{s_1, s_2, \dots s_l\}$. S contains exact match VNTR r of length k at position p if r can be partitioned into consecutive n sub-sequences of pattern q, as defined in the following equation

$$S = wrw',$$

where $r = \{s_p,\dots,s_{p+k}\} = (q)_n$, $n > 1$, $w = \{s_1,\dots,s_{p-1}\}$ and $w' = \{s_{p+k+1},\dots,s_l\}$.

INVERTER is aimed to identify both exact match and non-exact match VNTRs and provide data visualization which would allow users to correlate annotations, observe visual patterns, and view useful statistics of the data. More specifically, INVERTER is more optimized towards minisatellite VNTRs, although it can be used to identify and visualize microsatellite VNTRS. The reason is that due to the size of microsatellite VNTRs, visual patterns and statistical result provided may not be as significant as in minisatellite VNTRs.

2.2 Implementation

INVERTER is designed to be portable and therefore it is implemented in Java due to its platform-independent characteristics. Java supports four popular Operating Systems, i.e. Windows, Linux, Solaris and MacOS.

INVERTER also has a built-in GUI to allow user to set parameters needed for the identification and visualization of VNTRs in DNA sequences. A standalone version can be downloaded from http://bmserver.sce.ntu.edu.sg/INVERTER. Fig. 1 illustrates the workflow diagram of INVERTER. Currently, there are three features that are available to users, i.e. identification of exact match VNTRs and non-exact match VNTRs in DNA sequences as well as visualization of the resulting tandem repeat data.

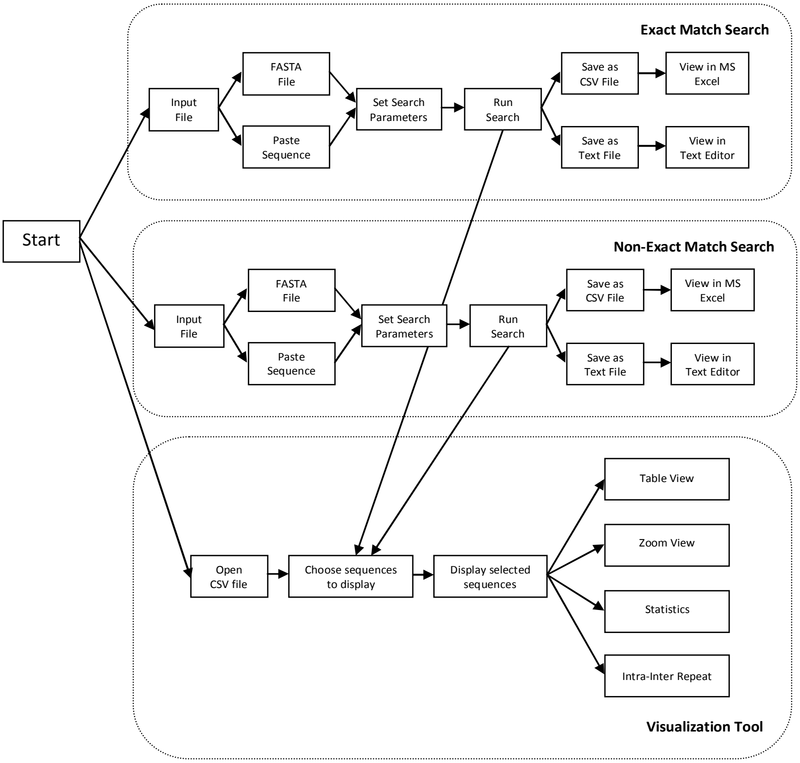
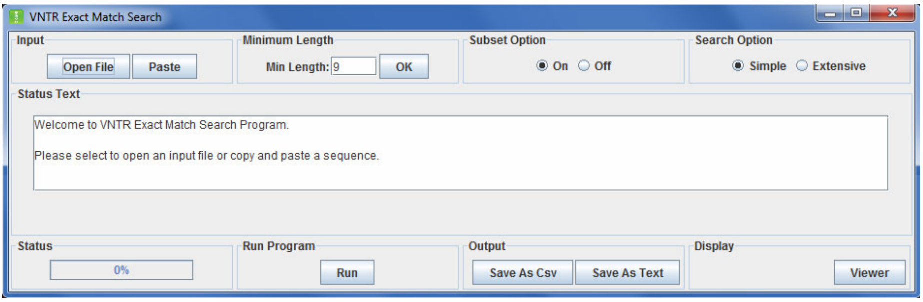


Fig. 1. Workflow diagram of INVERTER

2.2.1 Identification of Exact Match Tandem Repeats

The first feature allows users to identify exact match VNTRs in the input nucleotide sequences. Users can open an input file containing one or more sequences in FASTA format or copy and paste a sequence directly in FASTA format to INVERTER. Several parameters and options of the program can be defined by the users, as seen in Fig. 2.



Min Length: 9

Subset Option: ● On ○ Off

Search Option: ● Simple ○ Extensive

Welcome to VNTR Exact Match Search Program.

Please select to open an input file or copy and paste a sequence.

Status: 0%

Fig. 2. VNTR Exact Match Search

Minimum length l_{min} defines the minimum length of pattern to be reported in the result in terms of base pairs. The default value l_{min} is 9 bps and is chosen based on the analysis of known minisatellites of biological importance. Setting the value of l_{min} to a smaller value will generate more VNTR results but also increase the probability of inclusion of irrelevant repeats (noise). Given a sequence S of length l, the theoretical maximum of l_{max} is $l/2$. However, although it is possible to increase the maximum length value, it is not feasible to verify it using wet lab experiment due to the limitation of current gel-based technology verification technique, e.g. pulsed-field gel electrophoresis (PFGE). Therefore, the maximum length of searched pattern l_{max} is currently limited to 200.

The subset option indicates whether subset VNTRs should be included in the search result. The *on* option will include subset VNTRs in the result, while *off* will exclude them. The default option is *on*. Given a VNTR r of length k, $r = \{r_1, r_2, \dots, r_k\}$, a subset VNTR $z = \{z_1, z_2, \dots, z_k\}$ of length k_z containing m consecutive pattern q_z can be defined as follows:

$$r = vzv',$$

where $z = (q_z)_m$, $1 < m < n$, $1 \le k_z \le k$, $v = \{r_1,\dots,r_{c-1}\}$, $v' = \{r_{c+kz+1},\dots,r_k\}$ and $1 < c \le k_z$. For example, a VNTR r = ACGTACGTACGT contains subset VNTRs CGTACGTA, GTACGTAC and TACGTACG.

The search option indicates whether search should be a *simple* or *extensive search*. The *simple search* denotes that each VNTR is searched only in its origin sequence. On the other hand, *extensive search* searches for the occurrence of each VNTR in every sequence in the input file and is suitable to find inter-relationships of VNTRs among the input sequences. The default option is *simple search*.

Given a set of input sequences S and input parameters l_{min} and l_{max}, the exact match search will generate a list of VNTR V, which is then filtered for the occurence of VNTRs consisting of only N-nucleotides. N-nucleotide stands for an unknown nucleotide in some databases. It can be either of the four nucleotides (A, C, G or T). Therefore, it is logical to remove the N-VNTRs in order to reduce redundancy and improve the efficiency of the result. Subsequently, INVERTER checks for the subset and extensive search flag and then executes the necessary processing accordingly. The pseudocode of the exact match search is illustrated in Fig. 3.

After the search has finished, users can opt to use the visualization tool to view the result or save the result in a comma-separated value (csv) or text format. The saved search result contains the exact match VNTRs with their respective relevant data needed for the visualization tool, i.e. their origin sequence, the length of the origin sequence, the pattern, the initial position of the VNTR in the sequence, the length of the VNTR and the number of count.

2.2.2 Identification of Non-exact Match Tandem Repeats

The second feature of INVERTER allows users to identify non-exact match VNTRs. For this purpose, we implement a heuristic algorithm based on the unsupervised classification K-means algorithm[22]. Other VNTR finder tools using K-means algorithm include T-REKS[18].

```
Start
Get input data and relevant input parameters $l_{min}$ and $l_{max}$;
While there are still unprocessed input sequence in |S|
     For $i$:0 to $l_{max}$
          Search for potential VNTR candidate with length >
            $l_{min}$ using sliding window;
          If a VNTR is found
              Process VNTR data;
              Add VNTR to V;
          End If
     End For
End While
Filter for N-VNTRs;
If Subset Flag is TRUE
     Do subset processing
End If
If Extensive Search Flag is TRUE
     For $i$:1 to $|V|$
          For $j$:1 to $|S|$
              If the origin of current VNTR is $S_j$
                   Get the data from
              Else
                   Search the VNTR in other sequences
                    in the dataset;
          End For
     End For
End If
End
```

Fig. 3. Exact Match Search Pseudocode

We use short strings (SS) to detect for the presence of VNTRs and use the lengths between identical neighbouring SSs as datapoints of the K-means algorithm. Hence, all datapoints are partitioned into *k* clusters for user-defined *k*. For each partition a centroid is defined. *K* initial centroids are selected from the dataset. Distances between each datapoint and the centroids are then calculated to assign the datapoint to the cluster which has the nearest centroid. This procedure repeated iteratively until convergence is met. Statistically, the longer is the sequence the higher is the number of occurrences of a given SS. The increase of the occurrences will amplify a background noise and decrease the quality of detection at the clustering steps. Therefore, we split the input sequences to smaller chunks of length 1500 or less to avoid the reduction of detection quality in long sequences and then concatenate them after the search. This strategy is also used in T-REKS. For our implementation, we choose the length of SS l_{SS} to be 4 and *k* to be 20. These values are obtained empirically.

The candidate VNTR lengths is determined first by selecting the most frequent length *mfl* within each cluster generated. This step is applied to each type of SS found. If a cluster has several most frequent lengths which occur the same number of times, the shortest length is chosen.

Not all *mfl*s may correspond to the VNTR lengths, because a given short string may occur more than one time within a repeat. Hence, we filter the *mfl*s. First, we consider only SSs which are separated by lengths that are equal or close to the *mfl*s. The threshold of closeness of the length to the *mfl*s is proportional to the length, so it takes into account the variability of the lengths in biological tandem repeats. Second, we scan the sequence and do not consider a downstream SS of the neighbouring SSs except for those which length correspond to one chosen *mfl*s. The lengths are then re-calculated and the scanning is repeated for each of the *mfl*s and leads to *k* new sets of re-calculated lengths.

K-means algorithm is simultaneous applied to all *mfl*s of all type of SS which will provide *k* most frequent *mfl*s that can be considered as candidate VNTRs. The level of sequence similarity between the putative repeats of each run is evaluated using Multiple Sequence Alignment (MSA) center-star approach. Based on the obtained MSA of the repeats constituting the runs, we obtain a consensus sequence and subsequently use it as a reference for similarity calculation. Given an alignment made by m repeats of length l. Hamming distance d_i between the consensus sequence and a repeat r_i with $1 \leq i \leq m$ are calculated. A similarity coefficient for the whole alignment as $sim = (m*l - \sum \mathrm{Di})/m*l$ where $0 \leq sim \leq 1$. The pseudocode of the non-exact match search is illustrated in Fig. 4.

```
Start
Get input data and relevant input parameter sim;
While there are still unprocessed input sequence in |S|
      If $l_{Si}$ >= 1500
            Split $S_i$;
            For all the split sequences
                  For all SS
                        Apply k-Means and assign datapoints;
                        Filter mfls;
                        MSA string alignment;
                        Do similarity calculation;
                        Bridge the split sequences;
                  End For
                  If candidate fulfills criteria
                        Add VNTR to list;
                  End If;
            End For
      Else
            For all SS
                  Apply k-Means and assign datapoints;
                  Filter mfls;
                  MSA string alignment;
                  Do similarity calculation;
            End For
            If candidate fulfills criteria
                  Add VNTR to list;
            End If
      End If
End While
```

Fig. 4. Non-Exact Match Search Pseudocode

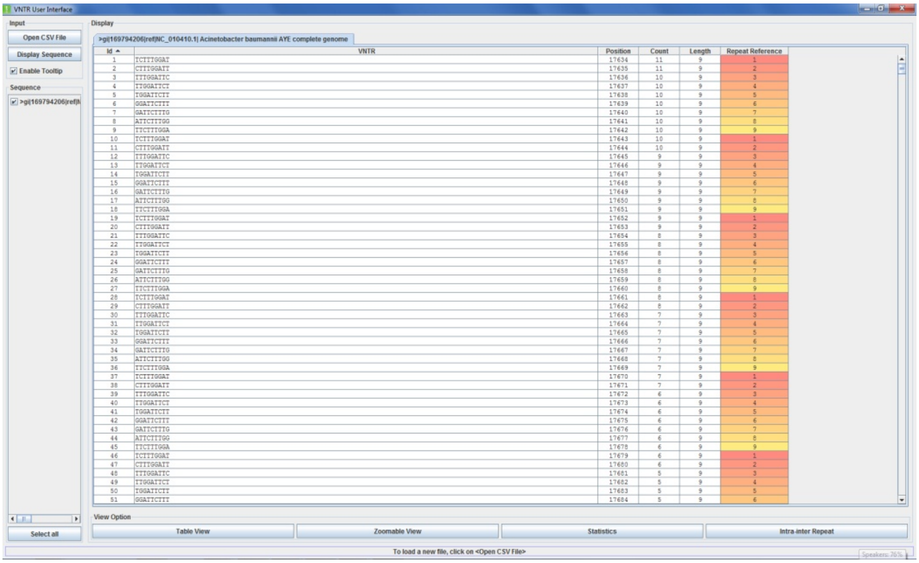
2.2.3 Visualization Tool

To our knowledge, INVERTER is the first tandem repeat discovery tool with an integrated visualization tool. The visualization tool is aimed to provide data visualization of the resulting VNTRs, which allow users to correlate annotations, observe visual patterns, and view useful statistics of the data, which are not available in other tools.

The availability of this novel tool is aimed to help biologists visualize VNTRs in raw genomic data as well as overall view of the entire data in form of useful statistics, hence facilitating new ways of understanding both the structure and function of DNA and protein. An example of this is frequency analysis of DNA sequences, in which some of tandem repeat patterns have been associated with known genetic factors e.g. a 3bp repeats has been found to be characteristic of exonic regions[23, 24] and a 10-11bp repeats has been found to be characteristic of DNA prone to supercoiling[25].

Users can directly use the visualization tool on the latest search result or open a csv file containing the result of a previous search. The visualization tool partitions the VNTRs based on their respective origin sequences in tab form. Users can display all the sequences or choose selectively which ones they are interested for comparison purposes. The visualization tool includes four features, i.e. table view, zoom view, statistics and intra-inter repeat.

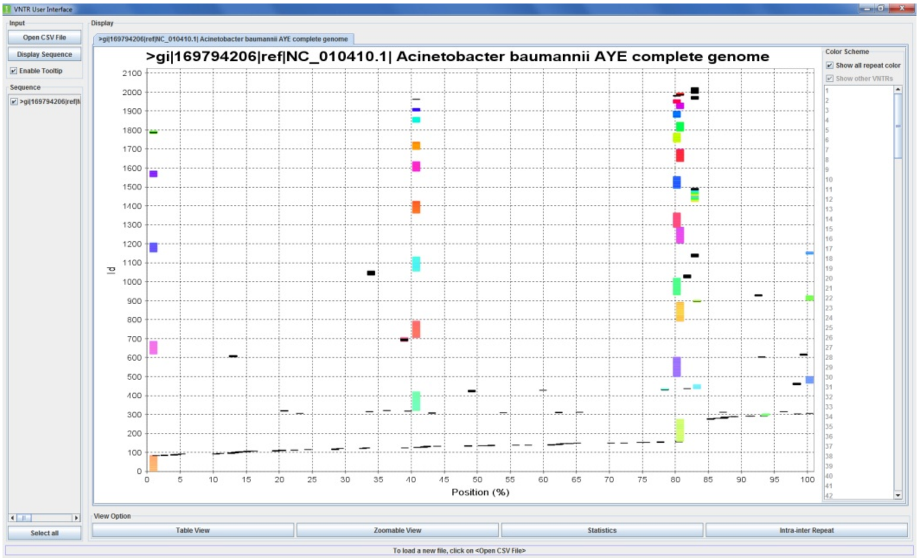
The zoom view provides an overall view of the positions VNTRs in the sequence. The x axis shows the relative position while the y axis shows the id of the VNTRs, respectively. Users can zoom in and zoom out of a particular area in the sequence to get a more detailed visualization of the VNTRs in the sequence. VNTRs marked with repeat reference are color coded in accordance to the table view while VNTRs that are not marked with repeat reference are given grayscale colors, ranging from light gray to black. For the latter, the higher the number of count of a particular VNTR, the lighter the color associated to it. Tooltip showing the VNTR sequence and the starting position of the VNTR will appear if users mouseover a particular VNTR. Furthermore, users can choose to display only selected VNTRs marked with repeat reference, with or without the non-repeat reference VNTRs for isolation purposes. In addition, users can also save the visualization as a PNG image file or print it directly from the user interface. Fig. 5 shows an example of the zoom view of the *Acinetobacter baumannii* AYE complete genome.



Id ▲	VNTR	Position	Count	Length	Repeat Reference
1	TCTTTGGAT	17634	11	9	1
2	CTTTGGATT	17635	11	9	2
3	TTTGGATTC	17636	10	9	3
4	TTGGATTCT	17637	10	9	4
5	TGGATTCTT	17638	10	9	5
6	GGATTCTTT	17639	10	9	6
7	GATTCTTTG	17640	10	9	7
8	ATTCTTTGG	17641	10	9	8
9	TTCTTTGGA	17642	10	9	9
10	TCTTTGGAT	17643	10	9	1
11	CTTTGGATT	17644	10	9	2
12	TTTGGATTC	17645	9	9	3
13	TTGGATTCT	17646	9	9	4
14	TGGATTCTT	17647	9	9	5
15	GGATTCTTT	17648	9	9	6
16	GATTCTTTG	17649	9	9	7
17	ATTCTTTGG	17650	9	9	8
18	TTCTTTGGA	17651	9	9	9
19	TCTTTGGAT	17652	9	9	1
20	CTTTGGATT	17653	9	9	2
21	TTTGGATTC	17654	8	9	3
22	TTGGATTCT	17655	8	9	4
23	TGGATTCTT	17656	8	9	5
24	GGATTCTTT	17657	8	9	6
25	GATTCTTTG	17658	8	9	7
26	ATTCTTTGG	17659	8	9	8
27	TTCTTTGGA	17660	8	9	9
28	TCTTTGGAT	17661	8	9	1
29	CTTTGGATT	17662	8	9	2
30	TTTGGATTC	17663	7	9	3
31	TTGGATTCT	17664	7	9	4
32	TGGATTCTT	17665	7	9	5
33	GGATTCTTT	17666	7	9	6
34	GATTCTTTG	17667	7	9	7
35	ATTCTTTGG	17668	7	9	8
36	TTCTTTGGA	17669	7	9	9
37	TCTTTGGAT	17670	7	9	1
38	CTTTGGATT	17671	7	9	2
39	TTTGGATTC	17672	6	9	3
40	TTGGATTCT	17673	6	9	4
41	TGGATTCTT	17674	6	9	5
42	GGATTCTTT	17675	6	9	6
43	GATTCTTTG	17676	6	9	7
44	ATTCTTTGG	17677	6	9	8
45	TTCTTTGGA	17678	6	9	9
46	TCTTTGGAT	17679	6	9	1
47	CTTTGGATT	17680	6	9	2
48	TTTGGATTC	17681	5	9	3
49	TTGGATTCT	17682	5	9	4
50	TGGATTCTT	17683	5	9	5
51	GGATTCTTT	17684	5	9	6

Fig. 5. Table view of the *Acinetobacter baumannii* AYE complete genome

Table view displays the search result in spreadsheet form. It shows the identification number (id), the VNTR, the initial position of the VNTR in the sequence, the length of the VNTR, the number of count and whether the VNTR is a repeat reference. If a VNTR is marked with repeat reference, it indicates that the VNTR appears in either another position in the origin sequence or in another sequence in the one of selected sequences. The repeat reference cell is color coded, up to 1024 unique colors and the value inside the cell indicates the reference number that particular VNTR. The spreadsheet can be sorted according a particular column (the default is id). In addition, the columns can be expanded and interchanged to allow greater flexibility for users. Example of the table view of the *Acinetobacter baumannii* AYE complete genome is illustrated in Fig. 6.

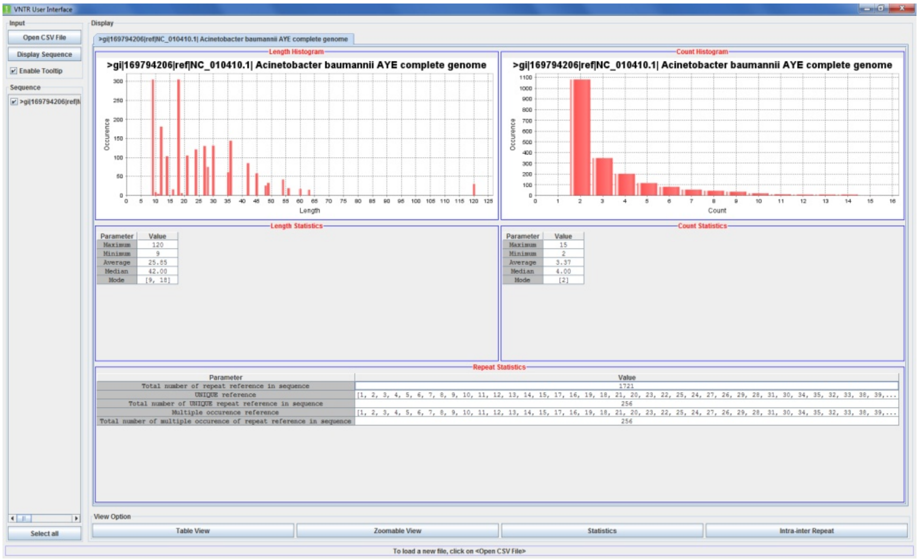


To load a new file, click on <Open CSV File>

Fig. 6. Zoom view of the *Acinetobacter baumannii* AYE complete genome

The main idea of the statistics feature is for users to be able to make a quick decision on whether or not the VNTRs in the sequence are worth pursuing further. This feature will be very useful especially when users are dealing with numerous sequences, as it summarizes how VNTR is distributed over the sequences. The statistics include the maximum value, minimum value, average, median, mode and histogram of the length and count of the VNTRs in the search result, respectively. In addition, repeat reference statistics are also included, namely the total number of repeat reference in the sequence and their respective references, total number of *unique* repeat reference in the sequence and their respective references as well as the total number of multiple occurrence of repeat reference in the sequence. Example of the statistics of the *Acinetobacter baumannii* AYE complete genome is shown in Fig. 7.

Intra-inter repeat shows the intra-relationship and inter-relationship of VNTRs marked with repeat reference. Intra-relationship means that the VNTR appears in



Length Statistics

Parameter	Value
Maximum	120
Minimum	9
Average	25.85
Median	42.00
Mode	[9, 18]

Count Statistics

Parameter	Value
Maximum	15
Minimum	2
Average	3.37
Median	4.00
Mode	[2]

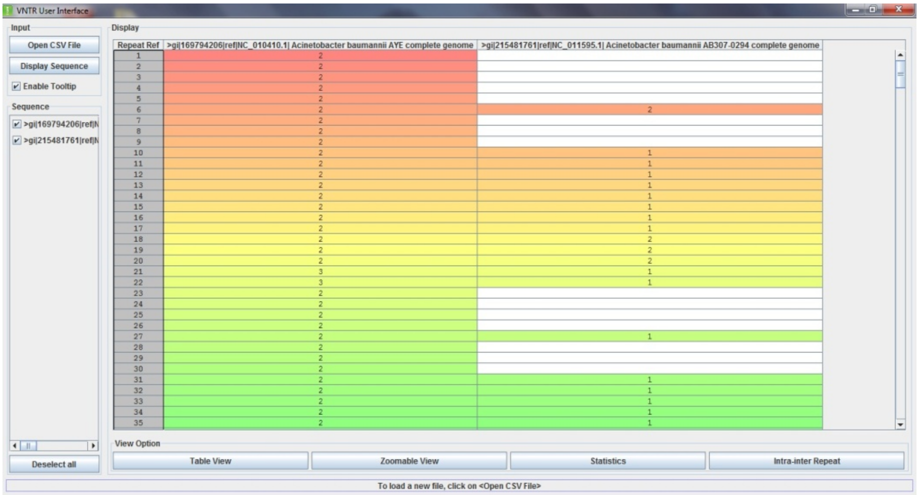
Repeat Statistics

Parameter	Value
Total number of repeat reference in sequence	1721
UNIQUE reference	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 17, 16, 19, 18, 21, 20, 23, 22, 25, 24, 27, 26, 29, 28, 31, 30, 34, 35, 32, 33, 38, 39,...
Total number of UNIQUE repeat reference in sequence	256
Multiple occurence reference	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 17, 16, 19, 18, 21, 20, 23, 22, 25, 24, 27, 26, 29, 28, 31, 30, 34, 35, 32, 33, 38, 39,...
Total number of multiple occurence of repeat reference in sequence	256

To load a new file, click on <Open CSV File>

Fig. 7. Statistics of the *Acinetobacter baumannii* AYE complete genome

another position in the origin sequence while inter-relationship means that the VNTR appears in another sequence. The cell is color coded in accordance to the table view and the value inside the cell of row *r* and column *c* indicates the number of occurrence of that a VNTR that is marked with a repeat reference *r* in sequence *c*. Fig. 8 illustrates an example of relationship of the *Acinetobacter baumannii* AYE and *Acinetobacter baumannii* AB307-0294 complete genomes.



Repeat Ref	>gi|169794206|ref|NC_010410.1| Acinetobacter baumannii AYE complete genome	>gi|215481761|ref|NC_011595.1| Acinetobacter baumannii AB307-0294 complete genome
1	2	
2	2	
3	2	
4	2	
5	2	
6	2	2
7	2	
8	2	
9	2	
10	2	1
11	2	1
12	2	1
13	2	1
14	2	1
15	2	1
16	2	1
17	2	1
18	2	2
19	2	2
20	2	2
21	3	1
22	3	1
23	2	
24	2	
25	2	
26	2	
27	2	1
28	2	
29	2	
30	2	
31	2	1
32	2	1
33	2	1
34	2	1
35	2	1

To load a new file, click on <Open CSV File>

Fig. 8. Intra-inter repeat relationship of the *Acinetobacter baumannii* AYE and *Acinetobacter baumannii* AB307-0294 complete genomes

3 Result

Acinetobacter baumannii is a major nosocomial pathogen with many clones resistant to most available antibiotics. It affects primarily immunocompromised patients, often in intensive care and burns units[26]. The reported antibiotic resistance includes the *carbapenems*, which have been the antibiotics of choice against this organism. Infections can therefore be difficult to treat, and are associated with increased morbidity and mortality[27, 28].

We have examined 6 completed genome projects of *Acinetobacter baumannii* obtained from the NCBI website (http://www.ncbi.nlm.nih.gov/sites/entrez?db=genomeprj&cmd=Retrieve&list_uids=21111,30993,17827,17477,28921,13001), i.e. AB0057, AB307-0294, ACICU, ATCC17978, AYE and SDF. The parameter values of l_{min} and l_{max} chosen for the experiment are 9 and 200, respectively. Our experiment is benchmarked on an Intel Core i5-540M 2.40 GHz CPU, 4 GB RAM running Windows 7 64 bit. The runtime is measured in seconds. The result of the experiment is shown in Table 2.

Table 2. Exact match result on 6 complete *Acinetobacter baumanii* genomes

A. baumanii **genome**	**RefSeq**	**Length (Mbps)**	**Subset option off**		**Subset option on**	
			#VNTRs found	**Runtime (s)**	**#VNTRs found**	**Runtime (s)**
AB0057	NC_011586	4.050513	570	42	1164	43
AB307-0294	NC_011595	3.760981	821	38	2720	39
ACICU	NC_010611	3.904116	508	39	1201	40
ATCC17978	NC_009085	3.976747	394	40	838	41
AYE	NC_010410	3.936291	787	39	2023	39
SDF	NC_010400	3.421954	1524	35	4600	35

We compare INVERTER result with T-REKS[18]. The similarity parameter P_{sim} of T-REKS is set to 1.0 to ensure that it search only for exact match VNTRs and we set the type to DNA sequence.

In general, all of the VNTRs reported by T-REKS are found by INVERTER as well. Furthermore, INVERTER found several more VNTRs that are not reported in T-REKS. However, there are a total of 4 VNTRs that are reported in the T-REKS but are not included in the INVERTER result with subset option off. By modifying the subset option parameter to on, these 4 VNTRs are included in the INVERTER result. The reason is because INVERTER found a more significant VNTR compared to the T-REKS result, and therefore it relegates that particular VNTR as a subset VNTR. In all 4 cases, the more significant VNTR has a higher count value compare to the subset counterpart. The difference of INVERTER result with T-REKS is highlighted in Table 3.

Table 4 shows non-exact match result of INVERTER on the 6 completed genome projects of *Acinetobacter baumannii*. The similarity threshold value *sim* chosen for the experiment is 0.85. The runtime is measured in seconds. As a comparison, T-REKS needs 17 min to analyze a medium size genome of *Drosophila melanogaster* using a Pentium 4 3.0 GHz and 2 GB of RAM[18]. In some genomes, INVERTER found less non-exact match compared to the exact match. One possible reason for this

Table 3. Difference of INVERTER result with T-REKS

A.baumanii genome	T-REKS	INVERTER (Subset option off)
AB307-0294	ATTCTTTGG	CTTTGGATT
ATCC17978	GGATTCTTT	TTCTTTGGA
AYE	GGATTCTTT	TTCTTTGGA
SDF	GACAGCGATTCGGATTCTGACTCA	TGACTCAGACAGCGATTCGGATTC

Table 4. Non-exact match result on 6 complete *Acinetobacter baumanii* genomes

A. baumanii genome	RefSeq	Length (Mbps)	#VNTRs (non-overlapping)	Runtime (s)
AB0057	NC_011586	4.050513	450	802
AB307-0294	NC_011595	3.760981	260	743
ACICU	NC_010611	3.904116	355	771
ATCC17978	NC_009085	3.976747	318	786
AYE	NC_010410	3.936291	416	779
SDF	NC_010400	3.421954	372	677

anomaly is that the non-exact match result excludes overlapping VNTRs of different tandem repeats that can be detected in the same region. Another possibility is that DNA sequences of length 1500 may contain more than the determined maximum number of clusters *k* different lengths of VNTRs (20 in this case) and that the splitting step may lead to the failure to detect some VNTRs. Our future work includes the optimization of parameters for the non-exact matches as well as to compare the time complexity with other tools.

4 Conclusions

In this paper, we present INVERTER, a *de novo* exact match tandem repeat finder which main advantage is that there is no need to specify either a pattern or a particular pattern size, integrated with a data visualization tool and a built-in user-friendly Graphical User Interface. INVERTER is designed to be portable; hence it is written in Java. It is therefore usable without problems on any CPUs with Windows, Linux, Solaris and MacOS operating systems. A standalone version of the program can be downloaded from http://bmserver.sce.ntu.edu.sg/INVERTER. Three features are currently available to users, i.e. identification of exact match VNTRs and non-exact match VNTRs in DNA sequences as well as visualization of the resulting tandem repeat data.

INVERTER is aimed to identify both exact match and non-exact match VNTRs and provide data visualization which would allow users to correlate annotations, observe visual patterns, and view useful statistics of the data. More specifically, INVERTER is more optimized towards minisatellite VNTRs. The visualization tool is aimed to provide data visualization of the resulting VNTRs, which would allow users to correlate annotations, observe visual patterns, and view useful statistics of the data. The visualization tool includes four features, i.e. table view, zoom view, statistics and intra-inter repeat.

Acknowledgments. The work described in this paper was supported by the National Medical Research Council - Exploratory / Developmental Grant NMRC EDG-08nov025.

References

1. Collins, F.S., Morgan, M., Patrinos, A.: The Human Genome Project: Lessons from large-scale biology. Science 300, 286–290 (2003)
2. Kannan, S.K., Myers, E.W.: An algorithm for locating nonoverlapping regions of maximum alignment score. SIAM Journal on Computing 25, 648–662 (1996)
3. Verkerk, A.J.M.H., Pieretti, M., Sutcliffe, J.S., Fu, Y.H., Kuhl, D.P.A., Pizzuti, A., Reiner, O., Richards, S., Victoria, M.F., Zhang, F., Eussen, B.E., Van Ommen, G.J.B., Blonden, L.A.J., Riggins, G.J., Chastain, J.L., Kunst, C.B., Galjaard, H., Caskey, C.T., Nelson, D.L., Oostra, B.A., Warren, S.T.: Identification of a gene (FMR-1) containing a CGG repeat coincident with a breakpoint cluster region exhibiting length variation in fragile X syndrome. Cell 65, 905–914 (1991)
4. Fu, Y.H., Pizzuti, A., Fenwick Jr, R.G., King, J., Rajnarayan, S., Dunne, P.W., Dubel, J., Nasser, G.A., Ashizawa, T., De Jong, P., Wieringa, B., Korneluk, R., Perryman, M.B., Epstein, H.F., Caskey, C.T.: An unstable triplet repeat in a gene related to myotonic muscular dystrophy. Science 255, 1256–1258 (1992)
5. MacDonald, M.E., Ambrose, C.M., Duyao, M.P., Myers, R.H., Lin, C., Srinidhi, L., Barnes, G., Taylor, S.A., James, M., Groot, N., MacFarlane, H., Jenkins, B., Anderson, M.A., Wexler, N.S., Gusella, J.F., Bates, G.P., Baxendale, S., Hummerich, H., Kirby, S.: A novel gene containing a trinucleotide repeat that is expanded and unstable on Huntington's disease chromosomes. Cell 72, 971–983 (1993)
6. La Spada, A.R., Wilson, E.M., Lubahn, D.B., Harding, A.E., Fischbeck, K.H.: Androgen receptor gene mutations in X-linked spinal and bulbar muscular atrophy. Nature 352, 77–79 (1991)
7. Campuzano, V., Montermini, L., Molto, M.D., Pianese, L., Cossee, M., Cavalcanti, F., Monros, E., Rodius, F., Duclos, F., Monticelli, A., Zara, F., Canizares, J., Koutnikova, H., Bidichandani, S.I., Gellera, C., Brice, A., Trouillas, P., De Michele, G., Filla, A., De Frutos, R., Palau, F., Patel, P.I., Di Donato, S., Mandel, J.L., Cocozza, S., Koenig, M., Pandolfo, M.: Friedreich's ataxia: Autosomal recessive disease caused by an intronic GAA triplet repeat expansion. Science 271, 1423–1427 (1996)
8. Mirkin, S.M.: DNA structures, repeat expansions and human hereditary disorders. Current Opinion in Structural Biology 16, 351–358 (2006)
9. Usdin, K.: The biological effects of simple tandem repeats: Lessons from the repeat expansion diseases. Genome Research 18, 1011–1019 (2008)
10. Jeffreys, A.J.: DNA typing: Approaches and applications. Journal of the Forensic Science Society 33, 204–211 (1993)
11. Bruford, M.W., Wayne, R.K.: Microsatellites and their application to population genetic studies. Current Opinion in Genetics and Development 3, 939–943 (1993)
12. Spong, G., Hellborg, L.: A near-extinction event in lynx: Do microsatellite data tell the tale? Conservation Ecology 6 (2002)
13. Benson, G.: Sequence alignment with tandem duplication. In: Conference Sequence alignment with tandem duplication, pp. 27–36 (1997)
14. Benson, G.: Tandem repeats finder: A program to analyze DNA sequences. Nucleic Acids Research 27, 573–580 (1999)

15. Kolpakov, R., Bana, G., Kucherov, G.: mreps: Efficient and flexible detection of tandem repeats in DNA. Nucleic Acids Research 31, 3672–3678 (2003)
16. Wexler, Y., Yakhini, Z., Kashi, Y., Geiger, D.: Finding approximate tandem repeats in genomic sequences. Journal of Computational Biology 12, 928–942 (2005)
17. Parisi, V., De Fonzo, V., Aluffi-Pentini, F.: STRING: Finding tandem repeats in DNA sequences. Bioinformatics 19, 1733–1738 (2003)
18. Jorda, J., Kajava, A.V.: T-REKS: Identification of Tandem REpeats in sequences with a K-meanS based algorithm. Bioinformatics 25, 2632–2638 (2009)
19. Turnpenny, P., Ellard, S.: Emery's Elements of Medical Genetics. Elsevier, London (2005)
20. Jeffreys, A.J., Wilson, V., Thein, S.L.: Hypervariable 'minisatellite' regions in human DNA. Nature 314, 67–73 (1985)
21. Merkel, A., Gemmell, N.: Detecting short tandem repeats from genome data: Opening the software black box. Briefings in Bioinformatics 9, 355–366 (2008)
22. MacQueen, J.B.: Some Methods for Classification and Analysis of MultiVariate Observations. In: Cam, L.M.L., Neyman, J. (eds.) Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297. University of California Press, Berkeley (1967)
23. Sanchez, J., Lopez-Villasenor, I.: A simple model to explain three-base periodicity in coding DNA. FEBS Lett. 580, 6413–6422 (2006)
24. Lopez-Villasenor, I., Jose, M., Sanchez, J.: Three-base periodicity patterns and self-similarity in whole bacterial chromosomes. Biochem. Biophys. Res. Commun. 325, 467–478 (2004)
25. Schieg, P., Herzel, H.: Periodicities of 10-11 bp as indicators of the supercoiled state of genomic DNA. J. Mol. Biol. 343, 891–901 (2004)
26. Turton, J.F., Matos, J., Kaufmann, M.E., Pitt, T.L.: Variable number tandem repeat loci providing discrimination within widespread genotypes of acinetobacter baumannii. European Journal of Clinical Microbiology and Infectious Diseases 28, 499–507 (2009)
27. Wareham, D.W., Bean, D.C., Khanna, P., Hennessy, E.M., Krahe, D., Ely, A., Millar, M.: Bloodstream infection due to Acinetobacter spp: Epidemiology, risk factors and impact of multi-drug resistance. European Journal of Clinical Microbiology and Infectious Diseases 27, 607–612 (2008)
28. Dijkshoorn, L., Nemec, A., Seifert, H.: An increasing threat in hospitals: Multidrug-resistant Acinetobacter baumannii. Nature Reviews Microbiology 5, 939–951 (2007)