BMC Bioinformatics

SOFTWARE

Open Access

Pytrf: a python package for finding tandem repeats from genomic sequences

Lianming Du[1], Dalin Sun[1], Jiahao Chen[1], Xinyi Zhou[1], Kelei Zhao[1], Qianglin Zeng[1] and Nan Yang[2*]

*Correspondence:
yangnan0204@126.com

[1] Antibiotics Research and Re-Evaluation Key Laboratory of Sichuan Province, Institute for Advanced Study, Chengdu University, Chengdu 610106, China
[2] Key Laboratory of Qinghai-Tibetan Plateau Animal Genetic Resource Reservation and Utilization, Sichuan Province and Ministry of Education, Southwest Minzu University, Chengdu 610225, China

Abstract

Background: Tandem repeats (TRs) are major sources of genetic variation and important genetic markers. Their expansions are not only involved in gene expression regulation but also associated with many nervous system diseases and cancers. However, there is a lack of an efficient tandem repeat identification tool for seamless integration with larger bioinformatics programs developed with the popular Python language.

Results: We introduce pytrf, a Python package for identification of both exact and approximate TRs from genomic sequences. It allows seamless embedding into other programs developed by Python or using in Python interactive environment and Jupyter notebooks. It also provides command line tools for assisting users to find tandem repeats from FASTA/Q files. Compared to other tools, the pytrf shows the highest performance in aspect of running time with comparable peak memory usage.

Conclusions: Pytrf provides simple interfaces and command line tools to facilitate identification of tandem repeats from genomic sequences. Pytrf can easily be installed from PyPI (https://pypi.org/project/pytrf) and the source code is freely available at https://github.com/lmdu/pytrf.

Keywords: Tandem repeats, Genome analysis, Microsatellite, Minisatellite, Genetic marker

Background

Tandem repeats (TRs) are genomic sequences in which a pattern of nucleotides is tandemly repeated multiple times in succession. They are ubiquitously dispersed in nearly all genomes of eukaryotic and prokaryotic organisms [1], even prevalent in organellar DNA [2]. The tandem repeats are broadly divided into microsatellites (also known as short tandem repeats or simple sequence repeats), minisatellites and satellite DNA according to the motif size [3]. There is no consensus among researchers regarding the minimum repeats and motif size to constitute a microsatellite or minisatellite. Many studies generally use 1–6 bp as motif size to identify microsatellites from genomic sequences [4–6]. Repeats with motif size ranging from 7 to 100 bp and over 100 bp are usually considered as minisatellites and satellite DNA, respectively [7]. Due to the variability in the number of repeats between individuals in a population, microsatellites and

minisatellites are sometimes referred to collectively as variable number tandem repeats (VNTRs) [8].

TRs are one of the largest sources of genetic variation and polymorphism which play diverse roles in genome structure, function, and evolution as well as influence various human phenotypes [9, 10]. They can tune gene expression at both transcriptional and translational levels [11] and have recently been proved to regulate gene splicing [9]. For example, STRs can bind to transcription factor DNA-binding domains to modulate their affinities and apparent on-rates leading to gene expression alterations [12]. In addition to length variation, the motif composition variation of TRs can also affect the gene expression [13]. TR expansions are associated with many nervous system diseases [14] and are implicated in complex disorders such as autism spectrum disorders [15, 16] and cancers [17]. Due to their high polymorphism rates, TRs have been widely used as useful genetic markers in population genetics, genetic mapping, forensics and phylogenetics [18, 19].

With the rapid evolution of next-generation sequencing technologies, the genome sequences of numerous species have been uncovered. The vast growth of genomic data enables us to decode not only coding genes but also other functional non-coding elements. In the past, many algorithms and tools have been developed to find TRs on a genome-wide scale [3, 20]. In recent years, Python has emerged as the most commonly used language for deciphering genomic data due to its simple syntax and widespread adoption [21, 22]. However, there is a lack of an efficient tandem repeat identification tool that can leverage the Python ecosystem for seamless integration into larger programs. Here, we present a novel Python package named pytrf for fast finding tandem repeats from genomic sequences. Pytrf has been designed to search for both exact and approximate tandem repeats. It has the capability to find generic tandem repeats without motif length limitation. Although the pytrf is written in C language, it is compiled as Python package for allowing seamless embedding into other programs or using in Jupyter notebooks. Moreover, pytrf also provides command line tools for researchers to facilitate detection of tandem repeat sequences from genomes.

Implementation

Algorithms

Pytrf adopts a fast search algorithm described in our previous study [23] to find exact tandem repeat sequences. We have optimized the algorithm and developed it as a Python package using C language. There are two steps involved in finding approximate tandem repeats. First, the pytrf finds an exact tandem repeat sequence with custom minimum repeats and length as a seed using sliding window approach (Fig. 1A). Then, the seed will be extended to both left and right through aligning flanking sequences to motif sequence (Fig. 1B, C). The alignment similarity is measured by using edit distance which allows substitutions and indels. We employ the wraparound dynamic programming algorithm (DPA) [24] instead of the previously used classic DPA to calculate alignment edit distance. Let S be the extending sequence with maximum length of n, M denotes the motif sequence with length of m, D represent a two-dimensional matrix used to hold the edit distance values, $D[i, j]$ means the edit distance value in row i and column j, where $0 \leq i \leq n$ and $0 \leq j \leq m$. Initially, $D[i, 0] = i$ and $D[0, j] = j$. The

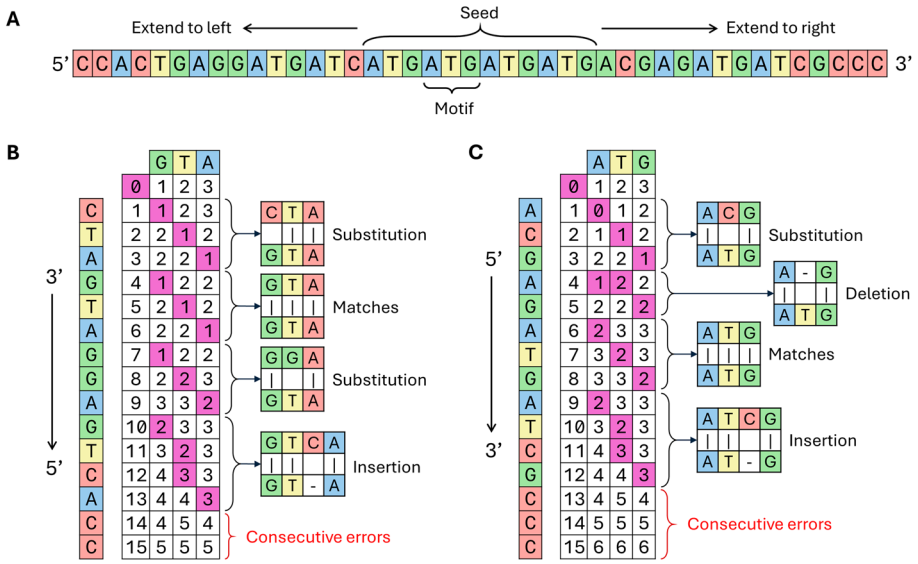

Fig. 1 The overview of pytrf algorithm. **A** the found seed repeat sequence and motif sequence before extending to both left and right. **B** extending repeat sequence to left by aligning sequence to the reversed motif sequence using wraparound dynamic programming to calculate edit distance. **C** extending repeat sequence to right

remaining cells are filled row by row in two phases. In the first phase, where $1 \leq i \leq n$ and $1 \leq j \leq m$, the edit distance values are computed using following formula:

$$D[i,j] = min \begin{cases} \begin{cases} D[i-1,0] + cost \\ D[i-1,m] + cost \\ D[i,0] + 1 \\ D[i-1,1] + 1 \end{cases} & if \quad j = 1 \\ \begin{cases} D[i-1,j-1] + cost \\ D[i,j-1] + 1 \\ D[i-1,j] + 1 \end{cases} & if \quad j > 1 \end{cases}$$

$$cost = \begin{cases} 0 & if \quad S[i] = M[j] \\ 1 & if \quad S[i] \neq M[j] \end{cases}$$

In the second phase, where $1 \leq i \leq n$ and $1 \leq j < m$, the edit distance values are updated using following formula:

$$D[i,j] = min \begin{cases} D[i,j] \\ \begin{cases} D[i,m] + 1 & if \quad j = 1 \\ D[i,j-1] + 1 & if \quad j > 1 \end{cases} \end{cases}$$

After filling the current row i, the minimum value of row i will be found and compared with the minimum value of the previous row $i-1$. If the minimum value of the current row is greater than that of the previous row, it means that an edit operation has occurred, which is counted as an alignment error. When the number of successive edit operations exceeds the user specified value (default: 3), the extension will be terminated immediately, and the row number i will return to the last row without edit

operations. Finally, an optimal alignment path is generated by backtracking the edit distance matrix from the rightmost minimum value of row *i*. Meanwhile, the number of matches, substitutions, deletions, and insertions are calculated separately. The extending alignment identity can be estimated using the following equation:

$$Identity = \frac{matches}{matches + substitutions + deletions + insertions}$$

Performance assessment

We have evaluated the performance of pytrf for finding microsatellites and minisatellites by monitoring the running time and peak memory usage. The pytrf command line tools applied pyfastx [25] to parse FASTA and FASTQ sequences. We have compared the pytrf with other ten tools including MISA [26], SciRoKoCo [27], PERF [28], RPTRF [29], GMATA [30], Phobos, MREPS [31], Kmer-SSR [32], SSRIT [33], Tantan [34], TRF [35], ULTRA [36] (See Table S1, Additional file 1). The genome sequences of human, dog, chicken, zebrafish and fruit fly were downloaded from UCSC genome database with total bases varying from 0.14 GB to 3.12 GB (See Table S2, Additional file 1). The running time and peak memory usage for each tool were measured by using linux built-in 'time' command located in/usr/bin folder with −f '%e %M' option on Ubuntu 20.04 with 32 GB memory. The detailed running commands and parameter settings can be found in Additional file 1. We used custom Python script to convert the results of each tool to bed formatted file. Intervene [37] was employed to analyze intersection between these bed files.

Results

Package overview

The Pytrf contains three modules including STRfinder, GTRFinder and ATRFinder which need DNA sequence and sequence name as required input parameters (Fig. 2). STRFinder is specially designed for efficient identification of microsatellites. GTRFinder allows users to find exact tandem repeats without limitation of motif size. We can obtain exact tandem repeat (ETR) object by iterating over STRFinder and GTRFinder. The ETR object represents a tandem repeat found in sequence and provides many properties to assist feature extraction. While ATRFinder can produce approximate tandem repeat (ATR) object without motif size limitation. In addition to the same attributes as ETR, ATR also enables to obtain seed information and alignment information. The pytrf also provides corresponding command line tool for each module to help users to find tandem repeats from FASTA or FASTQ files.

Performance comparison

We have performed comparative analysis between pytrf and several other tools for finding microsatellites, minisatellites and approximate tandem repeats. Among the compared tools, MREPS failed to process some genomes which have non-ATGC base codes and many unknown bases in a window, and RPTRF also failed to process large genomes due to insufficient memory. For microsatellite detection, all the tested tools exhibit comparable performance in processing smaller genomes, whereas the performance

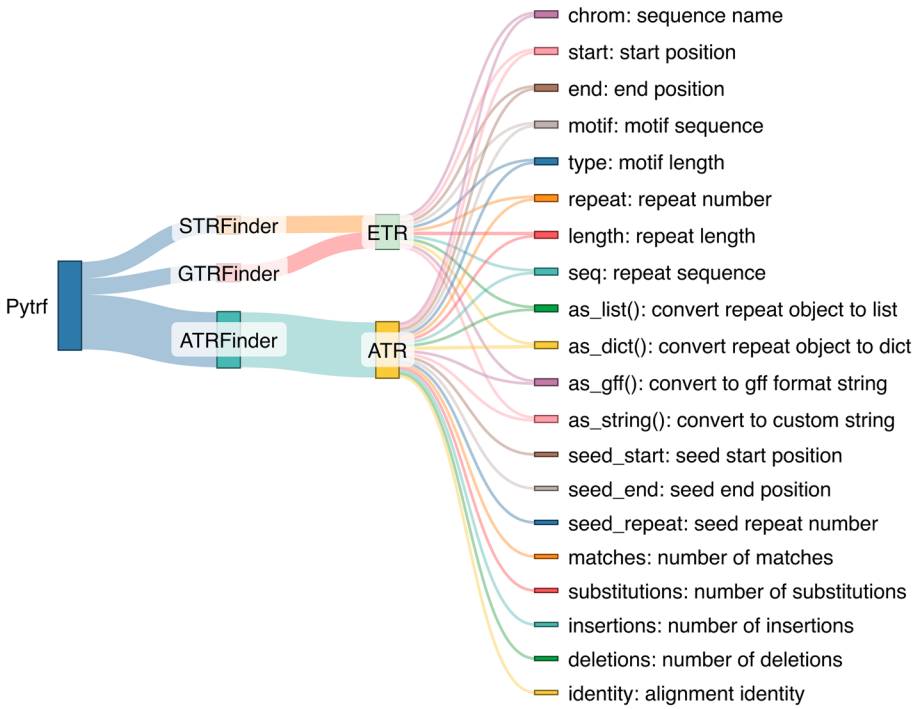

Fig. 2 The class structures of pytrf and their properties and methods

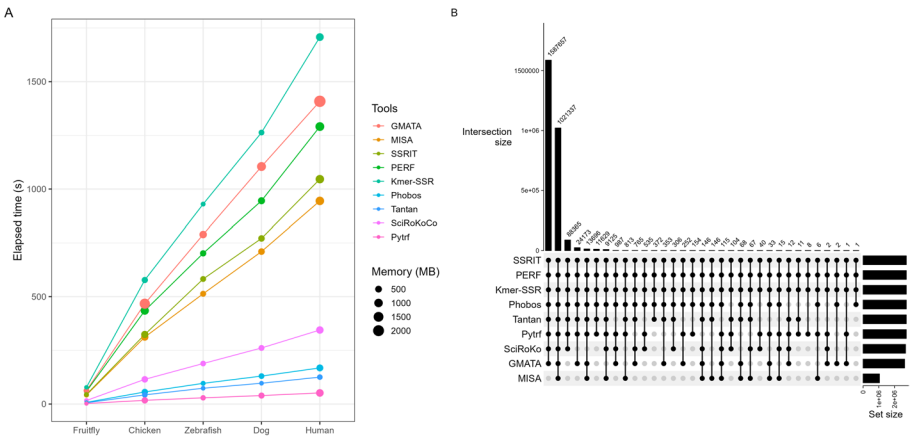

Fig. 3 The performance comparison for finding perfect microsatellites. **A** comparison of elapsed time and peak memory usage. **B** upset plot for intersections of search results

differences grow dramatically with the increase of genome size (Fig. 3A). The pytrf shows the highest performance in the aspect of elapsed time and its peak memory usage is comparable to SciRoKoCo and Phobos. The intersection analysis of search results indicates that the pytrf finds most tandem repeats identified by other tools and even discovers more tandem repeats than GMATA, MISA and SciRoKoCo (Fig. 3B). Similarly, the pytrf consumes the least time to search for minisatellites with motif length of 7–100 bp (Fig. 4A). The search results of pytrf covered almost all minisatellites that identified by other tools (Fig. 4B). For ATRs identification, the pytrf demonstrates comparable

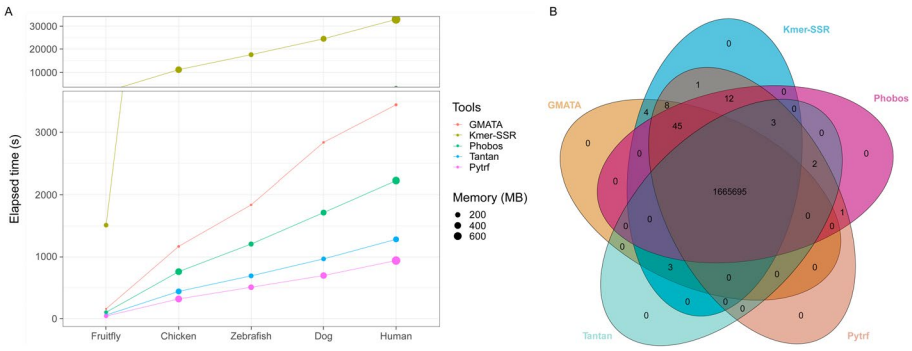

Fig. 4 The performance comparison for finding perfect minisatellites with motif size of 7–100 bp. **A** comparison of elapsed time and peak memory usage. **B** venn plot for showing number of intersections between different tools

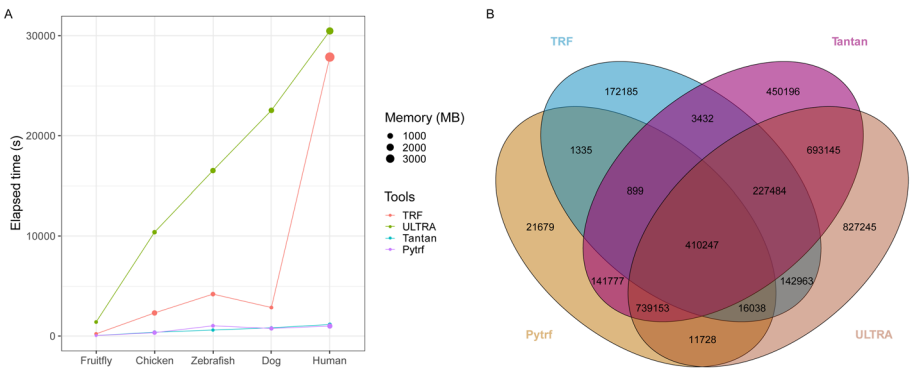

Fig. 5 The performance comparison for finding approximate tandem repeats with motif size of 1–100 bp. **A** comparison of elapsed time and peak memory usage. **B** venn plot for showing number of intersections between different tools

performance in both aspects of elapsed time and memory usage to Tantan and is much faster than TRF and ULTRA (Fig. 5A). Additionally, the pytrf finds the least specific ATRs and most of its detected ATRs can be covered by other tools (Fig. 5B). Although pytrf is very fast for finding ATRs, it can't find ATR without an exact seed that has at least two copies of a motif and is thus more suitable for finding tandem repeats with shorter motif length.

Conclusions

Pytrf is an artful lightweight tool that yields the best performance for finding both exact and approximate tandem repeats from DNA sequences. It provides simple interfaces for assisting integration with external larger programs to identify tandem repeats. Furthermore, it also offers efficient command line tools for facilitating detection of tandem repeats from genomes.

Availability and requirements

Project name: pytrf

Project home page: https://github.com/lmdu/pytrf

Operating system(s): Platform independent

Programming language: Python, C
Other requirements: pyfastx
License: MIT.
Any restrictions to use by non-academics: None.

Abbreviations

TRs	Tandem repeats
VNTRs	Variable number tandem repeats
ATR	Approximate tandem repeat
ETR	Exact tandem repeat

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12859-025-06168-3.

Additional file 1.

Acknowledgements
Not applicable.

Author contributions
LD and NY conceived the project. LD implemented the software and was a major contributor in writing the manuscript. DS, JC, XZ and QZ performed the benchmarking. KZ revised the manuscript. NY supervised the project. All authors read and approved the final manuscript.

Funding
This work was supported by the National Natural Science Foundation of China [No. 32200525].

Availability of data and materials
The download URLs of tools and genomes used to estimate the performance of pytrf can be found in supplementary files. The detailed commands and parameters for performance assessment can also be found in supplementary files. All scripts used for performance benchmarking can be found at https://github.com/lmdu/pytrf.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
No Conflict of interest is declared.

Received: 9 December 2024 Accepted: 15 May 2025
Published online: 04 June 2025

References

1. Du L, Liu Q, Zhao K, et al. PSMD: An extensive database for pan-species microsatellite investigation and marker development. Mol Ecol Resour. 2020;20(1):283–91.
2. Mokhtar MM, Atia MAM. SSRome: an integrated database and pipelines for exploring microsatellites in all organisms. Nucleic Acids Res. 2019;47(D1):D244–52.
3. Lim KG, Kwoh CK, Hsu LY, Wirawan A. Review of tandem repeat search tools: a systematic approach to evaluating algorithmic performance. Brief Bioinform. 2013;14(1):67–81.
4. Huang J, Li YZ, Du LM, et al. Genome-wide survey and analysis of microsatellites in giant panda (Ailuropoda melanoleuca), with a focus on the applications of a novel microsatellite marker system. BMC Genomics. 2015;16(1):61.
5. Lei Y, Zhou Y, Price M, Song Z. Genome-wide characterization of microsatellite DNA in fishes: survey and analysis of their abundance and frequency in genome-specific regions. BMC Genomics. 2021;22(1):421.
6. Song X, Yang T, Zhang X, et al. Comparison of the microsatellite distribution patterns in the genomes of euarchontoglires at the taxonomic level. Front Genet. 2021;12: 622724.
7. Wang S, Lorenzen MD, Beeman RW, Brown SJ. Analysis of repetitive DNA distribution patterns in the Tribolium castaneum genome. Genome Biol. 2008;9(3):R61.
8. Ichikawa K, Kawahara R, Asano T, Morishita S. A landscape of complex tandem repeats within individual human genomes. Nat Commun. 2023;14(1):5530.

9. Hamanaka K, Yamauchi D, Koshimizu E, et al. Genome-wide identification of tandem repeats associated with splicing variation across 49 tissues in humans. Genome Res. 2023;33(3):435–47.
10. Jam H, Li Y, DeVito R, et al. A deep population reference panel of tandem repeat variation. Nat Commun. 2023;14(1):6711.
11. Wright SE, Todd PK. Native functions of short tandem repeats. Elife. 2023;12: e84043.
12. Horton CA, Alexandari AM, Hayes MGB, et al. Short tandem repeats bind transcription factors to tune eukaryotic gene expression. Science. 2023;381(6664):eqdd1250.
13. Lu TY, Smaruj PN, Fudenberg G, et al. The motif composition of variable number tandem repeats impacts gene expression. Genome Res. 2023;33(4):511–24.
14. Hannan AJ. Tandem repeats mediating genetic plasticity in health and disease. Nat Rev Genet. 2018;19(5):286–98.
15. Trost B, Engchuan W, Nguyen CM, et al. Genome-wide detection of tandem DNA repeats that are expanded in autism. Nature. 2020;586(7827):80–6.
16. Mitra I, Huang B, Mousavi N, et al. Patterns of de novo tandem repeat mutations and their role in autism. Nature. 2021;589(7841):246–50.
17. Erwin GS, Gürsoy G, Al-Abri R, et al. Recurrent repeat expansions in human cancer genomes. Nature. 2023;613(7942):96–102.
18. Wyner N, Barash M, McNevin D. Forensic autosomal short tandem repeats and their potential association with phenotype. Front Genet. 2020;11:884.
19. Xie N. Building a catalogue of short tandem repeats in diverse populations. Nat Rev Genet. 2024;25(7):457.
20. Das G, Ghosh I. Benchmarking tools for DNA repeat identification in diverse genomes. bioRxiv. 2021. https://doi.org/10.1101/2021.09.10.459798.
21. Shajii A, Numanagić I, Leighton AT, Greenyer H, Amarasinghe S, Berger B. A Python-based programming language for high-performance computational genomics. Nat Biotechnol. 2021;39(9):1062–4.
22. Berger B, Yu YW. Navigating bottlenecks and trade-offs in genomic data analysis. Nat Rev Genet. 2023;24(4):235–50.
23. Du L, Zhang C, Liu Q, Zhang X, Yue B. Krait: an ultrafast tool for genome-wide survey of microsatellites and primer design. Bioinformatics. 2018;34(4):681–3.
24. Fischetti VA, Landau GM, Sellers PH, et al. Identifying periodic occurrences of a template with applications to protein structure. Inform Process Lett. 1993;45:11–8.
25. Du L, Liu Q, Fan Z, et al. Pyfastx: a robust Python package for fast random access to sequences from plain and gzipped FASTA/Q files. Briefings Bioinform. 2021;22(4):bbaa368.
26. Beier S, Thiel T, Münch T, Scholz U, et al. MISA-web: a web server for microsatellite prediction. Bioinformatics. 2017;33(16):2583–5.
27. Kofler R, Schlötterer C, Lelley T. SciRoKo: a new tool for whole genome microsatellite search and investigation. Bioinformatics. 2007;23(13):1683–5.
28. Avvaru AK, Sowpati DT, Mishra RK. PERF: an exhaustive algorithm for ultra-fast and efficient identification of microsatellites from large DNA sequences. Bioinformatics. 2018;34(6):943–8.
29. Behboudi R, Nouri-Baygi M, Naghibzadeh M. RPTRF: a rapid perfect tandem repeat finder tool for DNA sequences. Biosystems. 2023;226: 104869.
30. Wang X, Wang L. GMATA: an integrated software package for genome-scale SSR mining, marker development and viewing. Front Plant Sci. 2016;7:1350.
31. Kolpakov R, Bana G, Kucherov G. MREPS: efficient and flexible detection of tandem repeats in DNA. Nucleic Acids Res. 2003;31(13):3672–8.
32. Pickett BD, Miller JB, Ridge PG. Kmer-SSR: a fast and exhaustive SSR search algorithm. Bioinformatics. 2017;33(24):3922–8.
33. Temnykh S, DeClerck G, Lukashova A, et al. Computational and experimental analysis of microsatellites in rice (*Oryza sativa* L.): frequency, length variation, transposon associations, and genetic marker potential. Genome Res. 2001;11(8):1441–52.
34. Frith MC. A new repeat-masking method enables specific detection of homologous sequences. Nucleic Acids Res. 2011;39(4):e23.
35. Benson G. Tandem repeats finder: a program to analyze DNA sequences. Nucleic Acids Res. 1999;27(2):573–80.
36. Olson DR, Wheeler TJ. ULTRA-effective labeling of tandem repeats in genomic sequence. Bioinform Adv. 2024;4(1):vbae149.
37. Khan A, Mathelier A. Intervene: a tool for intersection and visualization of multiple gene or genomic region sets. BMC Bioinform. 2017;18(1):287.

Publisher's Note