

RESEARCH

Open Access



Comprehensive evaluation and guidance of structural variation detection tools in chicken whole genome sequence data

Cheng Ma^{1,2†}, Xian Shi^{1,3†}, Xuzhen Li^{4,5†}, Ya-Ping Zhang^{1,3,6,7*} and Min-Sheng Peng^{1,3,7*}

Abstract

Background Structural variations (SVs) are widespread across genome and have a great impact on evolution, disease, and phenotypic diversity. Despite the development of numerous bioinformatic tools, commonly referred to as SV callers, tailored for detecting SVs using whole genome sequence (WGS) data and employing diverse algorithms, their performance necessitates rigorous evaluation with real data and validated SVs. Moreover, a considerable proportion of these tools have been primarily designed and optimized using human genome data. Consequently, their applicability and performance in Avian species, characterized by smaller genomes and distinct genomic architectures, remain inadequately assessed.

Results We performed a comprehensive assessment of the performance of ten widely used SV callers using population-level real genomic data with the validated five common types of SVs. The performance of SV callers varies with the types and sizes of SVs. As compared with other tools, GRIDSS, Lumpy, Wham, and Manta present better detection accuracy. Pindel can detect more small SVs than others. CNVnator and CNVkit can detect more medium and large copy number variations. Given the poor consistency among different SV callers, the combination calling strategy is not recommended. All tools show poor ability in the detection of insertions (especially with size > 150 bp). At least 50x read depth is required to detect more than 80% of the SVs for most tools.

Conclusions This study highlights the importance and necessity of using real sequencing data, rather than simulated data only, with validated SVs for SV caller evaluation. Some practical guidance and suggestions are provided for SV detection in future researches.

Keywords Structural variation, Copy number variation, Population genomic sequence, WGS, Read depth, SV caller

[†]Cheng Ma, Xian Shi and Xuzhen Li contributed equally to this work.

*Correspondence:

Ya-Ping Zhang

zhangyp@mail.kiz.ac.cn

Min-Sheng Peng

pengminsheng@mail.kiz.ac.cn

¹Key Laboratory of Genetic Evolution & Animal Models and Yunnan Key Laboratory of Molecular Biology of Domestic Animals, Kunming Institute of Zoology, Chinese Academy of Sciences, Kunming 650223, China

²Department of Medical Biochemistry and Microbiology, Uppsala University, BMC, Uppsala SE-75123, Sweden

³University of Chinese Academy of Sciences, Beijing 100049, China

⁴State Key Laboratory for Conservation and Utilization of Bio-Resources in Yunnan, Yunnan Agricultural University, Kunming 650201, China

⁵College of Biological Big Data, Yunnan Agriculture University, Kunming 650201, China

⁶State Key Laboratory for Conservation and Utilization of Bio-Resources in Yunnan, Yunnan University, Kunming 650091, China

⁷KIZ-CUHK Joint Laboratory of Bioresources and Molecular Research in Common Diseases, Kunming Institute of Zoology, Chinese Academy of Sciences, Kunming 650223, China



Background

Structural variations (SVs) are an important source of genomic mutations, which are diverse in type and size, ranging anywhere from 50 bp to well over megabases (Mb) of sequence, affecting more size of the genome than single nucleotide polymorphisms (SNPs) and short insertions and deletions (INDELs) (≤ 50 bp) [1–4]. Generally, according to different types of variation, SVs can be classified into (a) deletion (DEL); (b) insertion (INS); (c) duplication (DUP), including tandem DUP and interspersed DUP; (d) inversion (INV); and (e) translocation (BND) [3, 5]. DEL and DUP are also known as Copy Number Variation (CNV). Numerous studies have shown that SVs have a great impact on human evolution [6, 7] and diseases [8–11], and SVs also play important roles in domestication [12, 13] and artificial breeding processes, which shape the phenotypic diversity of domestic animals and plants [14–17]. For example, in domestic chickens, a 17.7 kb DEL and a 4.2 kb *EAV-HP* INS cause feathered legs [18] and blue eggshell phenotype [19, 20], respectively (Figure S1A–B). CNV of the *EDN3* gene together with a complex SV rearrangement are responsible for dermal hyperpigmentation [21] (Figure S1C–E).

Identifying SV is essential for genomic interpretation and functional verification. With the rapid accumulation of WGS data, a large number of SV detection tools (i.e., SV callers) have been developed [5, 22–24] using the following four methods alone or in combination [25, 26]: (a) Read Pair (RP) based tools use the insert size information of pair-end reads to characterize the discordant alignment features; (b) Read Depth (RD) based tools refer to the read depth information to detect CNVs; (c) Split Read (SR) based tools utilize the split features to identify the breakpoints of SVs; and (d) Assembly (AS) based approaches detect SVs by assembling the sequencing reads into contigs. Previous studies have shown that the detection ability and accuracy were different between simulated and real data for various types and sizes of

SVs [22, 23]. To date, no single SV caller can detect all types of SVs across a wide range of sizes [24, 27]. Studies have applied multiple SV callers and adopted overlapping results [28–31] to reduce false positive rates. Although SV callers have been widely used, their applicability and performance have not yet been evaluated based on population-level WGS data with validated SVs (Table S1).

In this study, we investigated ten popular SV callers [30, 32–40] based on different detection strategies (Table 1). We conducted a comprehensive evaluation of their performance and applicability with real WGS data for populations. Especially, the validated five common types of SVs (Figure S1) were contained. Our results showed that very high heterogeneity exists among different SV callers, and no one tool is qualified for the detection of all types and wide size ranges of SVs. The real dataset and validated SVs are essential for the evaluation of the performance of various existing and underdeveloped SV callers.

Results

Population genomic SVs called by ten tools

We applied the ten SV callers (Table 1) to the WGS data for five chicken populations consisting of 48 samples (Table S2). The SV callers present powers as well as limits in the detection of five types of SVs as described in their original references (Table 1). DEL can be detected by all tools; nine of ten tools can detect DUP, while BreakDancer [32] has no ability to detect DUP. MetaSV [30], Wham [40], GRIDSS [36], and Pindel [39] show similar applicability but are unable to detect BNDs. Lumpy [37] and Manta [38] can detect BNDs but fail to identify INSs and INV, respectively. BreakDancer [32] and Delly [35] present heterogeneity among samples in the detection of INS, INV, and BND (Table 1; Fig. 1 and Figure S2). For all tested tools, higher consistency was observed in detecting DEL than the other types of SVs (Figure S2). Pindel [39] detected the most (on average, 0.81 million SVs per genome), while CNVkit [33] detected the least (on

Table 1 The general information and the average detected SV number of ten selected SV detection tools

Tools	Language	Method(s) ^a	Types of SVs and average number ^b				
			DEL	DUP	INS	INV	BND
BreakDancer [32]	C++	RP	1,154	NA ^c	96	217	632
CNVkit [33]	Python	RD	62	48	NA	NA	NA
CNVnator [34]	C++	RD	1,166	77	NA	NA	NA
Delly [35]	C++	RP+SR	363	22	23	14	6
GRIDSS [36]	Java	RP+SR+AS	26,092	200	19,339	169	NA
Lumpy [37]	C	RP+SR+RD	3,849	352	NA	36	342
Manta [38]	C++	RP+SR+AS	4,103	216	2,481	NA	486
MetaSV [30]	Python	Merged	3,073	200	404	45	NA
Pindel [39]	C++	SR	325,769	110,071	374,645	2,833	NA
Wham [40]	C++	RP+SR	2,757	618	822	511	NA

a. RP: read pair; RD: read depth; SR: split read; AS: assembly; Merged: different methods were merged depending on the input files. b. the average number of SVs detected in 48 WGS datasets. c. "NA" represents the targeted type of SV that cannot be detected by the corresponding tools

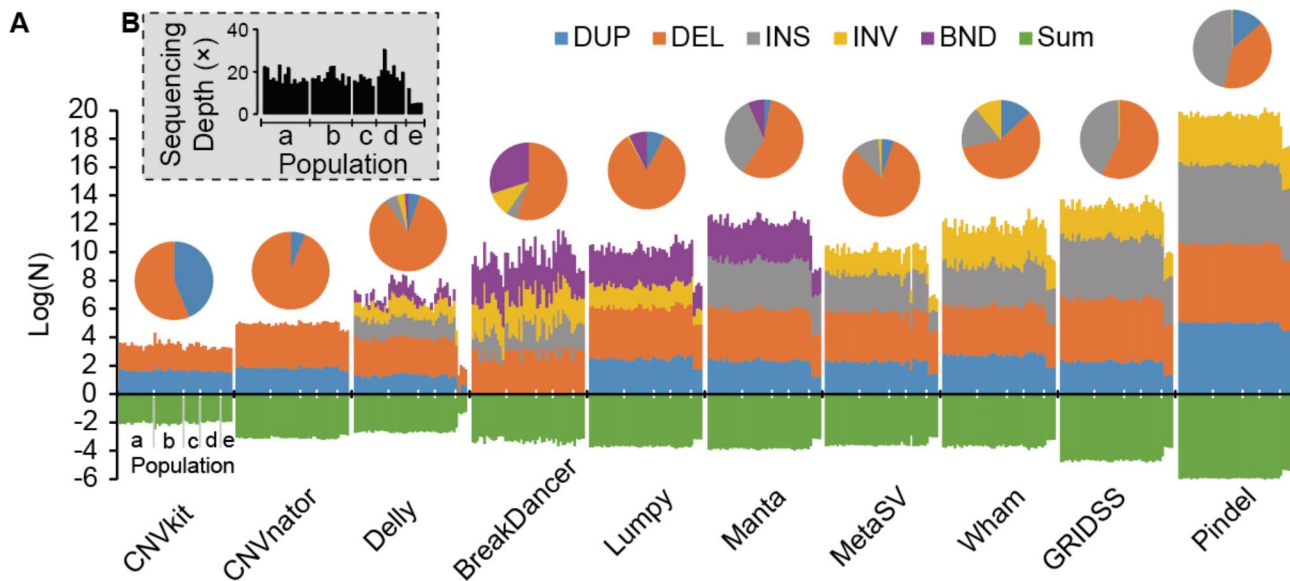


Fig. 1 The calling results of ten tools to detect different types of SVs in different populations. **(A)** The total number and proportion of different types of SVs. The numbers were log₁₀ converted, and the values below the zero line represent the total number of all detected SVs. Different colors represent different types of SVs. The pie charts show the proportion of different types of SVs. For each tool, five populations (a-e) with different read depths were measured. **(B)** The read depth for each individual in the five populations

average, 110 SVs per genome) number of all SVs (Table 1; Fig. 1). In detail, Delly [35] called the least amount of DUP, INS, INV, and BND, and CNVkit [33] detected the least number of DEL (on average of 62 DELs per sample). Pindel [39] detected the most DEL, DUP, INS, and INV (Table 1; Fig. 1 and Figure S2). The genomes with low read depth ($\sim 5\times$ for population e) present fewer SVs than other genomes with moderate or high read depths (other populations, average $> 15\times$). For the proportions of each type of SV, except for Pindel [39] (DEL accounts for 40.06% of all SVs), DEL accounts for the highest proportion (ranging from 55.19% in BreakDancer [32] to 93.82% in CNVnator [34]) of all called SVs (Table 1; Fig. 1).

Overlapping SVs called by different tools

To evaluate the detection consistency between various tools, we applied the tools for one genome (sample33) randomly selected from the WGS dataset (Fig. 2). For INS and DUP, there was no overlap among all detectable tools (Fig. 2A-B). For INV and DEL, only a small proportion of SVs were detected by all detectable tools (Fig. 2C-D). The overlap region of INV detected by all INV detectable software is 3,719 bp, only accounting for 0.51% (the highest proportion among all INV callers) in Lumpy [37] (Fig. 2C). In addition, the overlap region of DEL detected by all ten tools is 4,132 bp, accounting for 0.02% (the highest proportion among all DEL callers) in MetaSV [30] (Fig. 2D). Most of the SVs are specific for each tool. Pindel [39] specific INS, DUP, and DEL are the largest in total length, and the Wham [40] specific INV is the largest (Fig. 2). Moreover, as most of the studies are

more focused on SVs greater than 50 bp, and the detection ability and accuracy based on WGS data in detecting SVs larger than 1 Mb is insufficient [3, 5], we further compared the SVs in size larger than 50 bp but less than 1 Mb using chromosome 5 of the selected sample (Figure S3-S6). Consistent with the above analyses, the results showed that most of the SVs were tool specific, and only a small proportion of SVs were detected by more than one method (Figure S3-S6).

Size distribution of SVs detected by various software

Our results show that most SV callers can cover a wide range of sizes, but some tools are limited for SVs in certain size ranges (Fig. 3, Figure S7 and S8). The SVs called by CNVkit [33] and CNVnator [34] are larger than 1 Kb. Most of the SVs called by Pindel [39] and GRIDSS [36] are small in size ($SV < 1$ Kb). Most of the DELs and DUPs called by Delly [35], Lumpy [37], Manta [38], MetaSV [30], and Wham [40] range in size from 50 bp to 0.1 Mb (Fig. 3, Figure S7 and S8). For INV, large ($SV > 0.1$ Mb) INV accounted for a higher proportion of the total detected INVs, especially in Wham [40], MetaSV [30], and Delly [35] (Fig. 3 and Figure S7E). In addition, none of these tested tools has good ability and representativeness in detecting INS in medium and large sizes. Most INVs called by Pindel [39], GRIDSS [36], Delly [35], and Wham [40] are shorter than 50 bp, and the INVs detected by Manta [38] and BreakDancer [32] are also small in size ($S, 50 \text{ bp} < \text{INS} \leq 1 \text{ Kb}$) (Fig. 3 and Figure S7D).

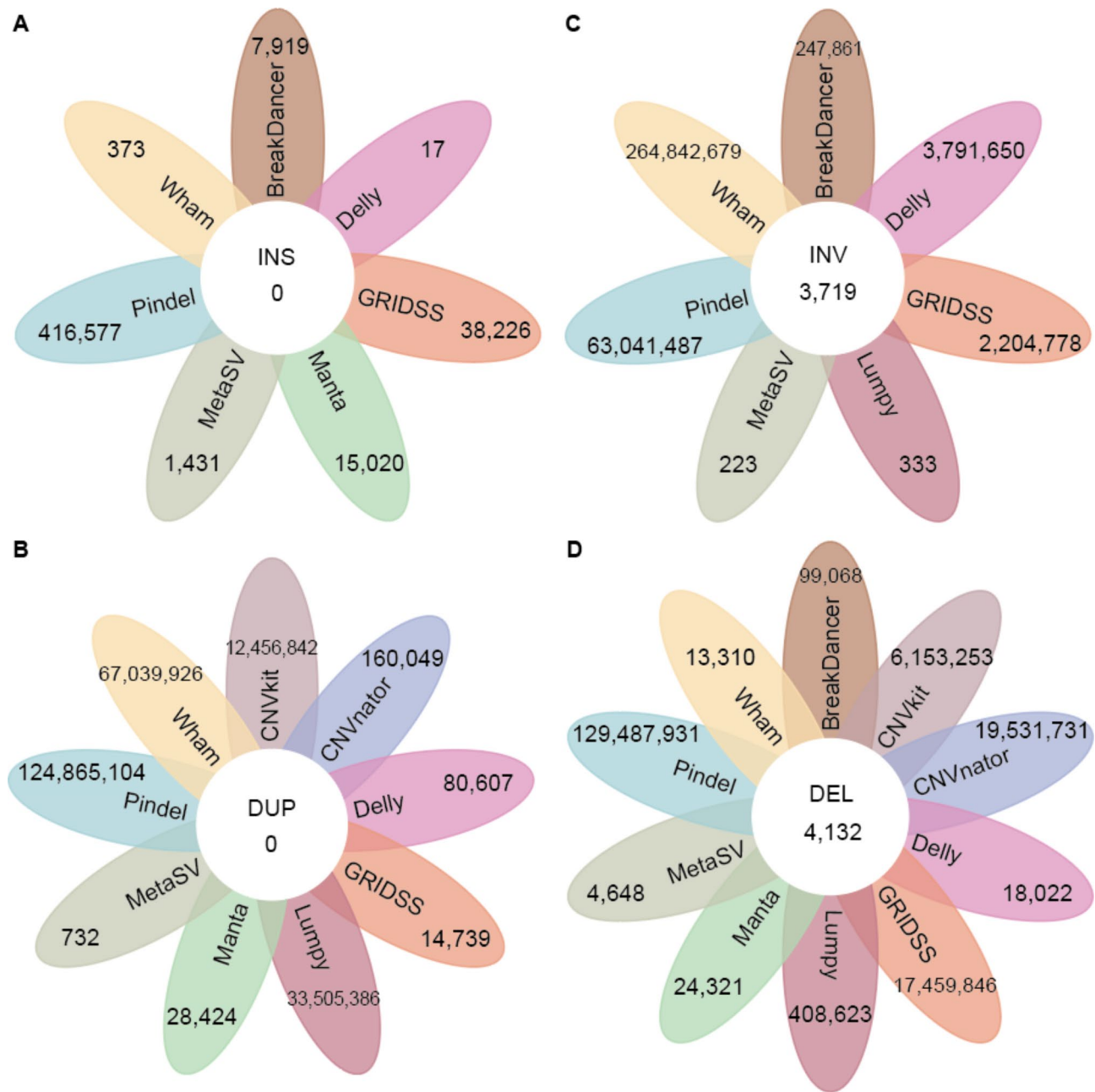


Fig. 2 The consistency evaluation of different tools for (A) INS, (B) DUP, (C) INV, and (D) DEL. Petal figures show the total size distribution of different SVs detected by various tools, figures on the petals are the total size of SVs detected only by the target method, and figures on the core are the size share detected by all methods

Detection capability and accuracy of SV callers

In terms of using WGS data with validated SVs, we evaluated the detection capability (see Methods) for ten SV callers. For the detection rate (Fig. 4A), GRIDSS [36] has the highest detection rate (100%) in all five tested types of SVs. It is the only tool that can detect target INS, although the detection accuracy is poor (Figure S9). Wham [40], CNVnator [34], Lumpy [37], and Pindel [39] have higher detection rates in interspersed DUP+INV (Figure S1D), tandem DUP (Figure S1C), and DEL (Figure S1A). Delly

[35], BreakDancer [32], and MetaSV [30] have higher detection rates in tandem DUP (Figure S1C). Manta [38] performed well at detecting interspersed DUP+INV (Figure S1D) and DEL (Figure S1A). CNVkit [33] shows poor detection rates for all targeted SVs. The positive rates were consistent with the detection rates (Fig. 4B and Figure S9). Except for CNVnator [34], CNVkit [33], and BreakDancer [32], the false positive rates of all other tools were zero in all five targeted SVs (Fig. 4C and Figure S9). The false positive rate for DEL (Figure S1A) detected

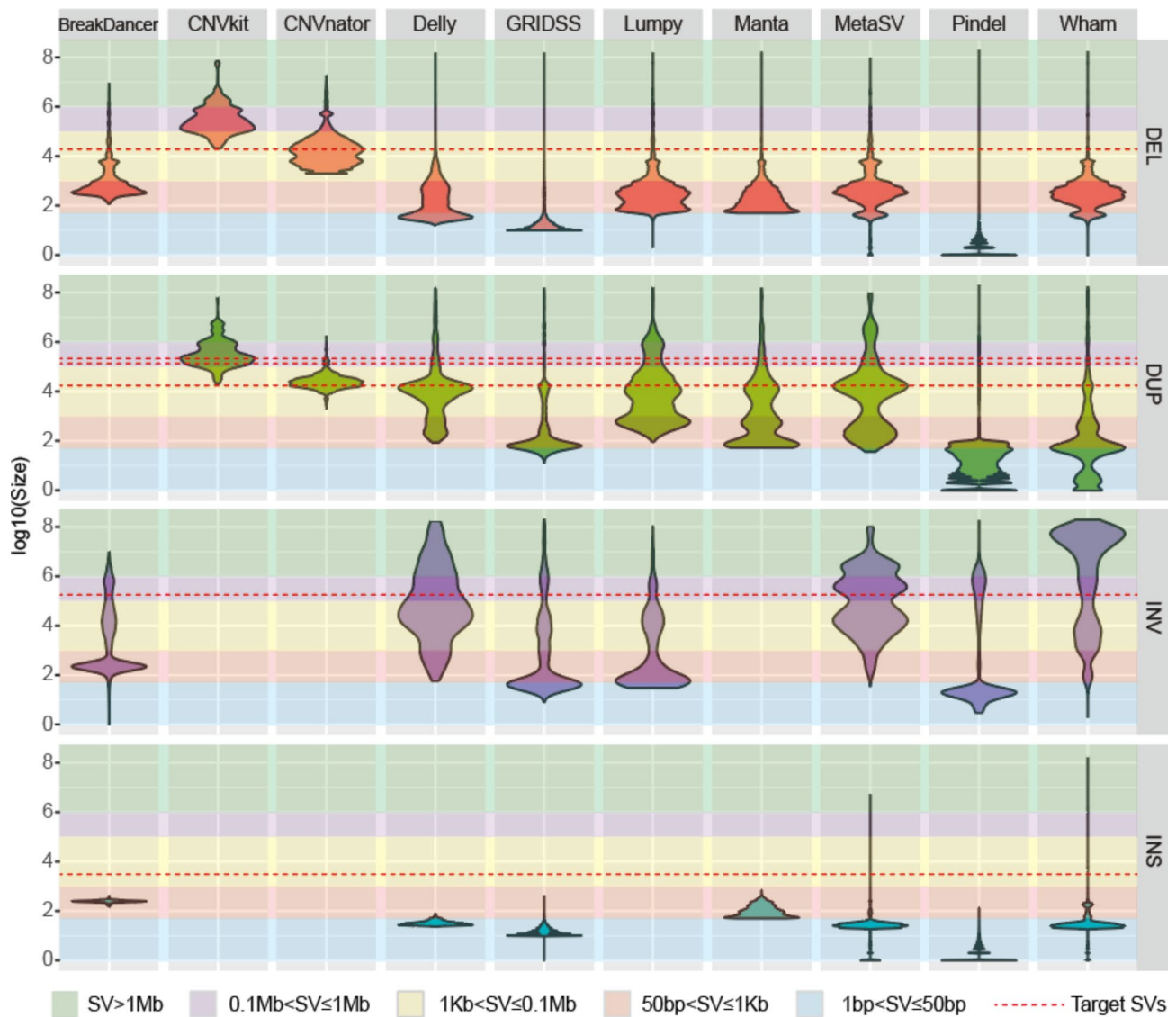


Fig. 3 The size distribution of different types of SVs detected by various tools. The size (length of SV) was log10 converted. Different background colors represent different categories of size. The red dashed line represents the size of the validated SV evaluated in this study (Figure S1)

by CNVnator [34] (1.4%) was higher than that of the others. CNVkit [33] detected interspersed DUP+INV (Figure S1D) with an approximately 1.2% false positive rate (Fig. 4C and Figure S9). In summary, GRIDSS [36] is the only tool that can comprehensively detect all five targeted types of SVs (Figure S1) with the highest positive rate and lowest false positive rate.

Impact of read depth

We evaluated the data saturation for SV calling based on WGS data with read depths ranging from 1× to 100×. For all tested tools, except for CNVkit [33], the detection rate improved with increasing average read depth (Fig. 5A and Figure S10). Our results show that at least 50× average read depth is needed for most of the tools to detect

more than 80% of the total SVs (Fig. 5A and Figure S10). For Lumpy [37], MetaSV [30], and Wham [40], more than 70× read depth is needed. For CNVnator [34] and GRIDSS [36], the detection rate improved sharply with increasing read depth, and 20× and 30× read depths were sufficient for 80% of the SV calling, respectively (Fig. 5A and Figure S10). To detect SVs with larger sizes, an increase in read depth is necessary (Figure S11-S22). Low read depth data are not sufficient for some types of SV calling for some tools. For example, when the read depth is lower than 20×, Delly [35] cannot detect INV (Figure S12), and MetaSV [30] and Wham [40] cannot detect INS (Figure S13 and Figure S14). When the read depth is lower than 10×, Delly [35] cannot detect DUP and INS

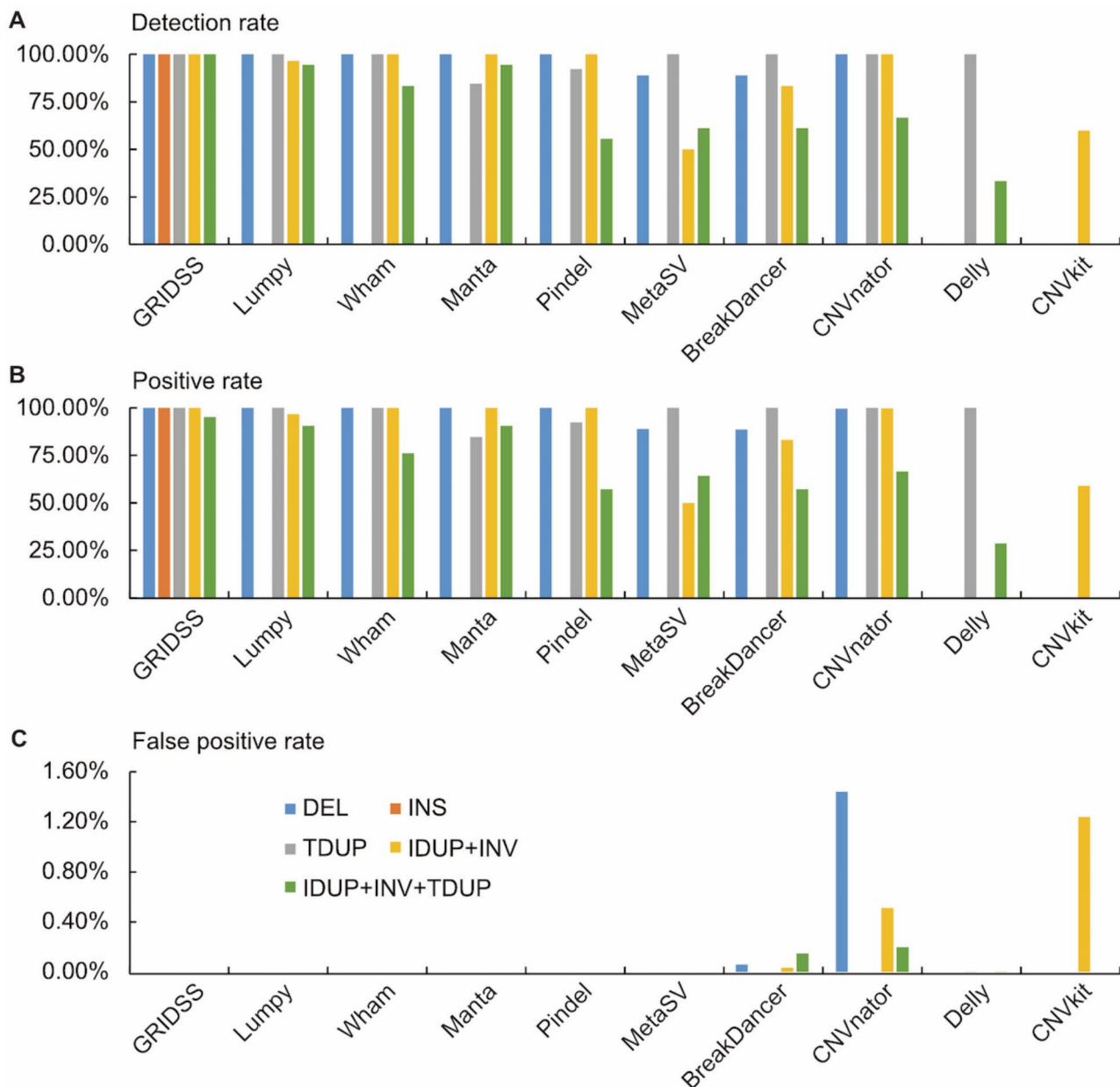


Fig. 4 The detection ability of SV detection tools. **(A)** Detection rate of various tools. **(B)** The positive rate of SV detected by various tools. **(C)** The false positive of detected SV

(Figure S12), and Lumpy [37] cannot detect INV (Figure S15).

Runtime and memory consumption

We calculated the CPU running time (Fig. 5B and Figure S23) and maximum memory (RAM) consumption (Fig. 5C and Figure S24) for the ten SV callers. Compared with other tools, Pindel [39] and Delly [35] are more time-consuming (Fig. 5B), but most of the calling can be done within one day. The running time of Pindel [39] is highly affected by the change in data size (Fig. 5B and Figure S23). However, the running times of CNVKit [33]

(Figure S23B) and CNVnator [34] (Figure S23C) increase slowly with increasing data size. When the data size is less than 20 Gb, the time consumption of Pindel [39] is less than that of Delly [35]. However, the runtime increased sharply with the data size increases in Pindel [39], and approximately two days were needed to finish SV calling when the data size reached 60 Gb (Fig. 5B). Fortunately, the other tools were fast, all of the tested data can be finished within 10 h, and some software (Wham [40], Lumpy [37], BreakDancer [32], CNVkit [33] and CNVnator [34]) can finish the detection in less than one hour (Fig. 5B and Figure S23). Since MetaSV [30] requires the

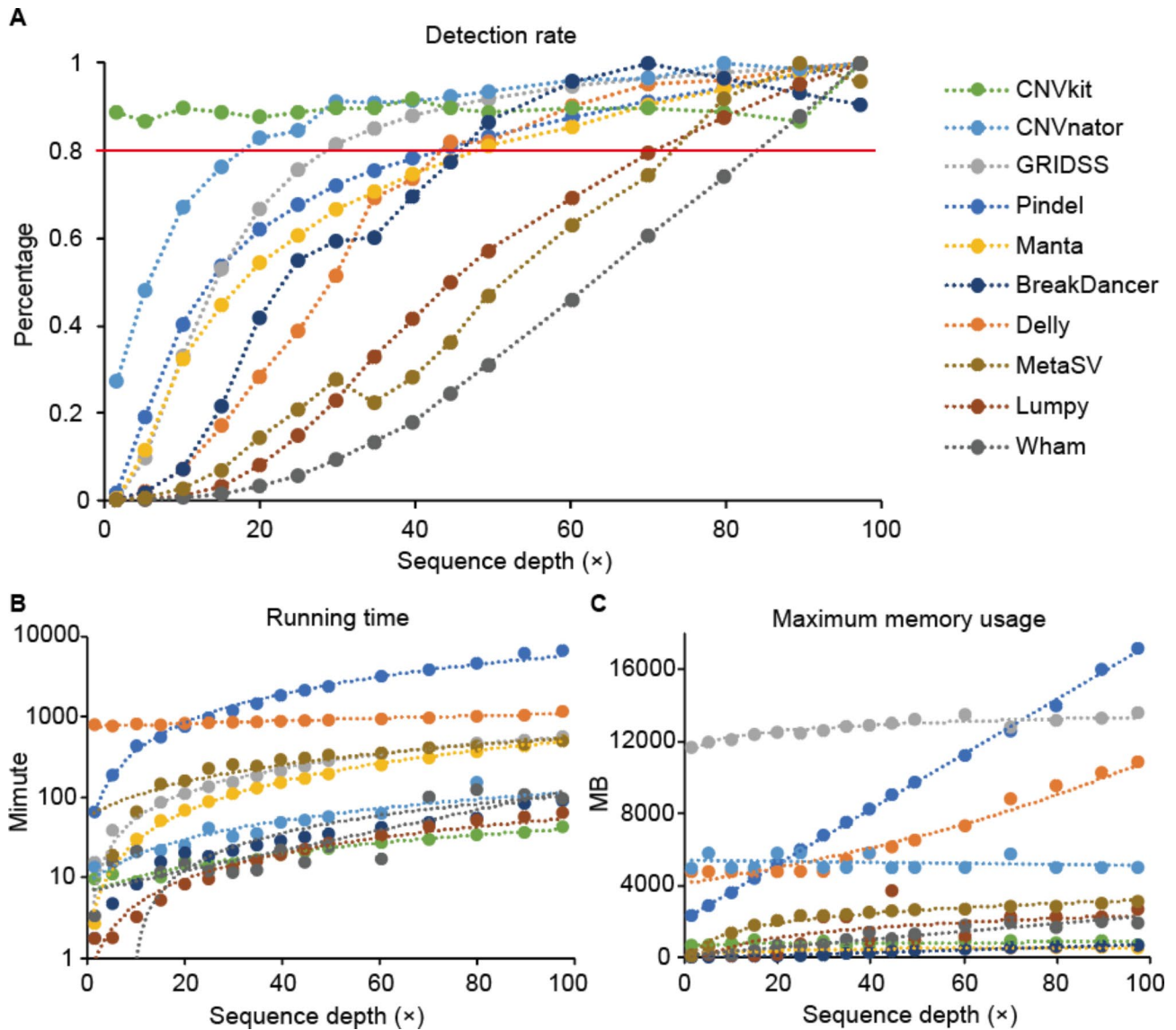


Fig. 5 The impacts of read depth on tool performance and detection results. (A-C) The influence of data quality on the detection rate (A), running time (B), and maximum memory usage (C) of various tools

results of several other programs as input, its calculation times are not included in the comparison. Among all the tested tools, the running time increases with increasing data size, and the correlation coefficients (R^2) are all higher than 0.7 (Figure S23). In addition, all tested tools were RAM consumption friendly, with a maximum RAM consumption of less than 20 GB (Fig. 5C and Figure S24). Except for CNVnator [34], the memory consumption of all tested software increases with increasing data size (Figure S24). For Manta [38] and BreakDancer [32], the maximum RAM consumption is less than 1 GB, and for Delly [35], Pindel [39] and GRIDSS [36], more than 10 GB RAM is needed for SV calling (Figure S24).

Discussion

Selecting tools for their best uses

According to our comprehensive assessments, we showed that the performance of SV callers varies with the types and sizes of SVs. Selecting SV callers should rely on what types of SVs are of interest and what sizes of SVs are expected. In general, GRIDSS [36], Lumpy [37], Wham [40], Manta [38] and Pindel [39] are candidate tools with high detection accuracy. For BND detection, Lumpy [37] or Manta [38] are appropriate. For all other types of SVs, GRIDSS [36] or Wham [40] are recommended. Pindel [39] performs well for short (≤ 50 bp) SVs. To cover a wide range of sizes, GRIDSS [36] and Wham [40] are appropriate tools.

Caveats about the intersection strategy with multiple tools

The strategy of taking the intersection or union of SVs identified by multiple tools [8, 10, 15, 30, 41–44] has been applied in various studies. Herein, we argue that this simplified strategy deserves more attention. Our results show few SVs being shared by multiple SV callers. Using different combinations of SV callers can produce distinct results (Fig. 2 and Figure S3–S6). For example, GRIDSS [36] shares more INS with Pindel [39] but less with BreakDancer [32]. Pindel [39] shares more DUP with Wham [40] and MetaSV [30] (Figure S3). For most cases in practice, the combination of no more than three methods may generate some consensus results (Fig. 2 and Figure S3–S6).

High read depth is necessary for most tools.

Previous studies proposed that at least 30× read depth, that is, the benchmark adopted in population genomic investigations [43, 45–47], could afford SV analyses [48–50]. However, our results show that 30× read depth is still insufficient for all seven of ten SV callers to detect approximately 80% of the SVs, except for GRIDSS [36], CNVnator [34] and CNVkit [33] (Fig. 5 and Figure S10). For Wham [40], Lumpy [37], and MetaSV [30], a 30× read depth cannot support the detection of 30% or even less of the total SVs. Generally, 50× read depth is needed for representative (80%) SV calling for most tools.

INS detection needs improvement

Our results show that all the tested SV callers have no ability to accurately detect INSSs, especially medium- or large-size INSSs. This can be ascribed to the short-read length of WGS (~150 bp) and discarding unmapped reads during mapping to the reference genome. However, INS is common in the genome; for example, more than half of INSSs are >1 kb in the human genome [2]. With the rapid accumulation of WGS data, the development of new strategies or algorithms for INS detection covering wide size ranges is urgently needed. Long-read sequencing technology, telomere-to-telomere and/or pangenome-based SV detection have potential [13, 15, 16, 31, 51, 52]. The graph-based breakpoint connection assembly strategy is compatible with WGS data [53, 54]. For these under developing SV callers, further evaluation studies based on more real datasets and more types and sizes of validated SVs are also needed.

Limitation and future research directions

While this study offers valuable insights into genomic SV detection in chickens, it also serves as a reference for SV detection in other poultry (*Aves*) species. Despite providing a comprehensive evaluation of commonly used SV callers, future endeavors must address certain limitations to establish a more universally applicable SV detection framework and gold standard. This entails augmenting

the dataset with additional validated SVs, broadening the scope to encompass diverse populations, and continuously evaluating newly updated methodologies and software tools, with a particular focus on machine learning or deep learning-based approaches [55, 56]. Such efforts will not only enhance the accuracy and reliability of SV detection but also pave the way for a deeper understanding of genetic variation and its implications in poultry populations.

Materials and methods

Tools selection

According to the citation and the recommendation or suggestion of Kosugi et al. [23] and Cameron et al. [22], ten widely used tools (Table 1) were selected to cover four mainstream algorithms (RD, RP, SR and AS).

Targeted SVs

To evaluate the detection accuracy of the ten tools, five reported or validated SVs were measured, including three simple SVs and two complex combinations of these simple SV types (Figure S1). The DEL type was selected from [18], which is a 17.7 kb deletion associated with the feathered legs in domestic chickens. In addition, an ~4.2 kb *EAV-HP* INS in the 5' flanking region of *SLCO1B3* causes blue eggshell phenotype in chickens [19, 20]. A complex genomic rearrangement involved in interspersed DUP and INV was reported to cause dermal hyperpigmentation in chickens [21]. In addition, in another of our unpublished studies, a new tandem DUP and its combination with the previously reported SV [21] were also associated with the hyperpigmentation phenotype in chickens. Based on these identified SVs, we further compared and evaluated the detection capability of various methods with real WGS data containing validated SVs (Figure S1).

Sample collection and sequencing

Samples from five breeds of chicken with target SVs were newly collected (Table S2), for a total of 48 samples. DNA was extracted from the blood using phenol-chloroform method and sequenced. As the direct ancestor of domestic chicken [57], the *Gallus gallus spadiceus* subspecies of Red junglefowl was selected as the control group ($n=20$) for the CNVkit [33] and Delly [35] case-control based algorithms. In addition, to evaluate the influence of the data quality (average read depth) on the detection results and tool performance, an additional high-quality (100×) WGS data (sample49) was added to the above datasets, and various qualities of data (read depth range from 0.1× to 98×) were generated using BamDeal (v.0.25, <https://github.com/BGI-shenzhen/BamDeal>) based on these high-quality data.

Sample collection and sequencing

The index sequences and low-quality reads with (a) more than 50% bases with $Q \leq 5$ and (b) more than 10% “N” content were removed as previous work [58]. Low quality reads and adapters were removed using Trimmomatic (v0.39) [59]. Using the BWA (version 0.7.17) [60] mem option with the default parameters, the clean reads of each sample were mapped to the chicken reference genome GRCg6a (Ensembl release 96). The mapped reads were sorted, the PCR duplications were marked with Picard tools (version 2.18.6) (<https://broadinstitute.github.io/picard/>), and the final bam file of each sample was used for next SV calling.

SV calling

For each sample, ten selected tools were used to perform SV calling, and for each tool, the data processing process was consistent, the bam file was used as input, and the vcf file was generated as the final output results. For MetaSV [30], which merges SV calling results from other multiple tools, four tools were used as sources for integration, including Pindel [39], Manta [38], Lumpy [37] and Wham [40]. The detailed usage and parameter settings for each software are provided in the Supplemental Methods in the Additional file.

Comparison of the detected SV number and size

According to the types of SV detected by different methods, the number of different SVs were counted. In addition, the lengths of different types of SV detected by various methods were counted, and the values were *log* transformed for visualization. To evaluate the detection ability of various methods in detecting SVs at different sizes, we classified the length of SVs into five levels: (a) very small ($1 \text{ bp} \leq \text{length} \leq 50 \text{ bp}$), (b) small ($50 \text{ bp} < \text{length} \leq 1 \text{ Kb}$), (c) medium ($1 \text{ Kb} < \text{length} \leq 100 \text{ Kb}$), (d) large ($100 \text{ Kb} < \text{length} \leq 1 \text{ Mb}$) and (e) very large ($\text{length} > 1 \text{ Mb}$). Because only four tools used in this study can detect BNDs (translocation), this type of SV was only included in the SV number evaluation process. First, to measure whether different breeds of data sources will make a difference in the results, we compared the size distribution of all four types of SV detected by the ten used tools based on five different breeds. After determining that the results would not be affected by different breed data, we counted the total number and length distribution of all detected SVs of all samples.

Evaluation of the SV detection capability

For each method, the SV detection capability was measured in (a) detectable rate, (b) positive rate and (c) false positive rate at the individual level, taking the average of each breed, and for individuals with multiple target SV segments, the average value was also taken. Detectable

rate, which indicates the proportion of detected target SVs to the total target SVs. It is measured as:

$$\text{Detectable rate} = \frac{\text{The number of target SVs detected}}{\text{Total number of target SVs}}$$

In addition, the positive rate was measured as the proportion of the true SV fragment length of the detected target SV to the true length of the target SV, which is expressed as follows:

$$\text{Positive rate} = \frac{\text{The length of true part of a target SV detected}}{\text{The length of a target SV}}$$

In the same way, the false positive rate was measured as the proportion of false SV fragment length of detected target SVs to the true length of target SVs, which is expressed as:

$$\text{False positive rate} = \frac{\text{The length of false part of a target SV detected}}{\text{The length of a target SV}}$$

Comparison of the breakpoint detection accuracy

Alignment to the chicken reference genome (GRCg6a), the breakpoint was defined as the position where an SV occurred. Based on the physical position of the genome, each breakpoint has two incisions, one on the left (L) and the other on the right (R). For each specific SV type, we examined whether the breakpoint detected by various methods was shifted, and we counted the length of the shift and performed *log10* conversion for the visualization plot. To visually demonstrate the accuracy of various methods in detecting SV breakpoints, compared with the real SV segment length, if the shift causes the length of the detected SV segment to become longer, we marked it as positive (+). Otherwise, if the shift causes the length of the detected SV segment to become shorter, we mark it as negative (-). The greater the absolute value, the farther away from the true breakpoint, and the more inaccurate the result was. If any method failed to detect an SV type, the breakpoint shift size was set to $-(\text{target SV length})/2$.

Evaluation of the detection consistency among methods.

One genomic dataset (sample33, Table S2) was randomly selected to measure the consistency among four types (DEL, DUP, INS and INV) of SV detected by all ten tested tools. We counted the overlap size in base pairs (bp) among all target SV detectable tools. At the whole-genome level, we compared the intersections of each

target SV detectable software and their specific SVs using the petal figure. Because there are so many sites involved in structural variation that it is difficult to display the whole genome dataset with the UpsetR [61] package, we chose chromosome 5 as an example to show the detailed intersection between any two or more tools. UpsSetR [61] and Venn [62] packages were used to plot the results.

Evaluation of the influence of data size on detection results

To evaluate the impact of read depth on SV calling and to provide guidance for future SV-related data preparation, we analyzed a wide range of datasets with varying read depths (from 1× to 100×) generated from a high read depth genome (sample49, Table S2). Different read depths were simulated using this high-quality data. The total number and size distribution of all SVs for different quality data were counted. The results were plotted using R script.

Evaluation of the runtime and maximum memory consumption

To measure the performance of each software for different size data. The CPU runtime and maximum memory (RAM) consumption of each size of data called by all tested software were recorded. All the tools were running on a Linux system (version 3.10.0-862.el7.x86_64, Red Hat 4.8.5–28) with the configuration of Intel(R) Xeon(R) Gold 6132 CPU @ 2.60 GHz. For each tool, we use exponential, linear, logarithmic and power methods to fit the data and then select the one with the largest correlation coefficient (R^2) as the best fitting formula. For MetaSV [30], since it requires the detection results of other methods as input files, we only counted the running time of using this software and did not compare it with other methods.

Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1186/s12864-024-10875-1>.

Supplementary Material 1
Supplementary Material 2
Supplementary Material 3
Supplementary Material 4

Author contributions

M.-S.P. and Y.-P.Z. supervised and supported this project. C.M. conceived this study. C.M., X.S. and X.L. performed the data analyses. C.M. and M.-S.P. prepared the manuscript. Y.-P.Z. revised the manuscript. All authors read and approved the final manuscript.

Funding

This work was supported by grants from the Second Tibetan Plateau Scientific Expedition and Research Program (STEP) (2019QZKK050), the NSFC (31771405), the Yunnan Provincial Science and Technology Department Grant (202001AU070099), the Spring City Plan: The High-level Talent Promotion

and Training Project of Kunming (2022SCP001), and the Animal Branch of the Germplasm Bank of Wild Species, Chinese Academy of Sciences (the Large Research Infrastructure Funding). M.-S.P. is supported by the Yunnan Revitalization Talent Support Program.

Data and code availability

All of the data used in this study have been deposited in the Sequence Read Archive (SRA) of NCBI under the accession PRJNA943525 (<https://www.ncbi.nlm.nih.gov/search/all/?term=PRJNA943525>), and a data copy was also deposited in Genome Sequence Archive (GSA) at the National Genomics Data Center under the data accession number CRA009347 (<https://ngdc.cncb.ac.cn/gsa/search?searchTerm=CRA009347>). The script and all the parameters setting are available in Supplementary Materials and Methods. The analyses output data were also deposited in GSA under the project accession number PRJCA015393 (<https://ngdc.cncb.ac.cn/bioproject/browse/PRJCA015393>).

Declarations

Ethics approval and consent to participate

This study was approved by the Animal Care and Ethics Committee of Kunming Institute of Zoology, Chinese Academy of Sciences (KIZ-LA-PF16-01-V4.0). Sampling procedures complied with the guidelines of animal use protocols approved by the Animal Care and Ethics Committee of Kunming Institute of Zoology, Chinese Academy of Science, China (Approval ID: IACUC-OE-2021-12-004).

Consent for publication

Not applicable.

Competing interests

The authors declare no competing interests.

Received: 6 November 2023 / Accepted: 8 October 2024

Published online: 16 October 2024

References

- Abel HJ, Larson DE, Regier AA, Chiang C, Das I, Kanchi KL, Layer RM, Neale BM, Salerno WJ, Reeves C, et al. Mapping and characterization of structural variation in 17,795 human genomes. *Nature*. 2020;583(7814):83–9.
- Sudmant PH, Rausch T, Gardner EJ, Handsaker RE, Abyzov A, Huddleston J, Zhang Y, Ye K, Jun G, Fritz MH, et al. An integrated map of structural variation in 2,504 human genomes. *Nature*. 2015;526(7571):75–81.
- Ho SS, Urban AE, Mills RE. Structural variation in the sequencing era. *Nat Rev Genet*. 2020;21(3):171–89.
- Collins RL, Brand H, Redin CE, Hanscom C, Antolik C, Stone MR, Glessner JT, Mason T, Pregno G, Dorrani N, et al. Defining the diverse spectrum of inversions, complex structural variation, and chromothripsis in the morbid human genome. *Genome Biol*. 2017;18(1):36.
- Spielmann M, Lupianez DG, Mundlos S. Structural variation in the 3D genome. *Nat Rev Genet*. 2018;19(7):453–67.
- Perry GH, Yang F, Marques-Bonet T, Murphy C, Fitzgerald T, Lee AS, Hyland C, Stone AC, Hurler ME, Tyler-Smith C, et al. Copy number variation and evolution in humans and chimpanzees. *Genome Res*. 2008;18(11):1698–710.
- Jiang Z, Tang H, Ventura M, Cardone MF, Marques-Bonet T, She X, Pevzner PA, Eichler EE. Ancestral reconstruction of segmental duplications reveals punctuated cores of human genome evolution. *Nat Genet*. 2007;39(11):1361–8.
- Li Y, Roberts ND, Wala JA, Shapira O, Schumacher SE, Kumar K, Khurana E, Waszak S, Korbel JO, Haber JE, et al. Patterns of somatic structural variation in human cancer genomes. *Nature*. 2020;578(7793):112–21.
- Quigley DA, Dang HX, Zhao SG, Lloyd P, Aggarwal R, Alumkal JJ, Foye A, Kothari V, Perry MD, Bailey AM, et al. Genomic Hallmarks and Structural Variation in metastatic prostate Cancer. *Cell*. 2018;174(3):758–e769759.
- Hadi K, Yao X, Behr JM, Deshpande A, Xanthopoulos C, Tian H, Kudman S, Rosiene J, Darmofal M, DeRose J, et al. Distinct classes of Complex Structural Variation uncovered across thousands of Cancer Genome Graphs. *Cell*. 2020;183(1):197–e210132.
- Collins RL, Brand H, Karczewski KJ, Zhao X, Alföldi J, Francioli LC, Khera AV, Lowther C, Gauthier LD, Wang H, et al. A structural variation reference for medical and population genetics. *Nature*. 2020;581(7809):444–51.

12. Wang GD, Shao XJ, Bai B, Wang J, Wang X, Cao X, Liu YH, Wang X, Yin TT, Zhang SJ, et al. Structural variation during dog domestication: insights from gray wolf and dhole genomes. *Natl Sci Rev*. 2019;6(1):110–22.
13. Yu H, Lin T, Meng X, Du H, Zhang J, Liu G, Chen M, Jing Y, Kou L, Li X, et al. A route to de novo domestication of wild allotetraploid rice. *Cell*. 2021;184(5):1156–e11701114.
14. Clop A, Vidal O, Amills M. Copy number variation in the genomes of domestic animals. *Anim Genet*. 2012;43(5):503–17.
15. Alonge M, Wang X, Benoit M, Soyk S, Pereira L, Zhang L, Suresh H, Ramakrishnan S, Maumus F, Ciren D, et al. Major impacts of widespread structural variation on Gene expression and crop improvement in Tomato. *Cell*. 2020;182(1):145–e161123.
16. Wang K, Hu H, Tian Y, Li J, Scheben A, Zhang C, Li Y, Wu J, Yang L, Fan X, et al. The Chicken Pan-genome reveals Gene Content Variation and a promoter region deletion in IGF2BP1 affecting body size. *Mol Biol Evol*. 2021;38(11):5066–81.
17. Huang Y, Huang W, Meng Z, Braz GT, Li Y, Wang K, Wang H, Lai J, Jiang J, Dong Z, et al. Megabase-scale presence-absence variation with *Tripsacum* origin was under selection during maize domestication and adaptation. *Genome Biol*. 2021;22(1):237.
18. Li J, Lee M, Davis BW, Lamichhane S, Dorshorst BJ, Siegel PB, Andersson L. Mutations Upstream of the TBX5 and PITX1 Transcription Factor Genes Are Associated with feathered legs in the Domestic Chicken. *Mol Biol Evol*. 2020;37(9):2477–86.
19. Wang Z, Qu L, Yao J, Yang X, Li G, Zhang Y, Li J, Wang X, Bai J, Xu G, et al. An EAV-HP insertion in 5' flanking region of SLC01B3 causes blue eggshell in the chicken. *PLoS Genet*. 2013;9(1):e1003183.
20. Wragg D, Mwacharo JM, Alcalde JA, Wang C, Han JL, Gongora J, Gourichon D, Tixier-Boichard M, Hanotte O. Endogenous retrovirus EAV-HP linked to blue egg phenotype in Mapuche fowl. *PLoS ONE*. 2013;8(8):e71393.
21. Dorshorst B, Molin AM, Rubin CJ, Johansson AM, Strömstedt L, Pham MH, Chen CF, Hallböök F, Ashwell C, Andersson L. A complex genomic rearrangement involving the endothelin 3 locus causes dermal hyperpigmentation in the chicken. *PLoS Genet*. 2011;7(12):e1002412.
22. Cameron DL, Di Stefano L, Papenfuss AT. Comprehensive evaluation and characterisation of short read general-purpose structural variant calling software. *Nat Commun*. 2019;10(1):3240.
23. Kosugi S, Momozawa Y, Liu X, Terao C, Kubo M, Kamatani Y. Comprehensive evaluation of structural variation detection algorithms for whole genome sequencing. *Genome Biol*. 2019;20(1):117.
24. Mahmoud M, Gobet N, Cruz-Dávalos DI, Mounier N, Dessimoz C, Sedlazeck FJ. Structural variant calling: the long and the short of it. *Genome Biol*. 2019;20(1):246.
25. Tattini L, D'Aurizio R, Magi A. Detection of genomic structural variants from next-generation sequencing data. *Front Bioeng Biotechnol*. 2015;3:92.
26. Escaramis G, Docampo E, Rabionet R. A decade of structural variants: description, history and methods to detect structural variation. *Brief Funct Genomics*. 2015;14(5):305–14.
27. Guan P, Sung WK. Structural variation detection using next-generation sequencing data: a comparative technical review. *Methods*. 2016;102:36–49.
28. van Belzen I, Schönhuth A, Kemmeren P, Hehir-Kwa JY. Structural variant detection in cancer genomes: computational challenges and perspectives for precision oncology. *Npj Precision Oncol*. 2021;5(1):15.
29. Gong T, Hayes VM, Chan EKF. Detection of somatic structural variants from short-read next-generation sequencing data. *Brief Bioinform* 2021, 22(3).
30. Mohiyuddin M, Mu JC, Li J, Bani Asadi N, Gerstein MB, Abyzov A, Wong WH, Lam HY. MetaSV: an accurate and integrative structural-variant caller for next generation sequencing. *Bioinformatics*. 2015;31(16):2741–4.
31. Dubois F, Sidiropoulos N, Weischenfeldt J, Beroukhir R. Structural variations in cancer and the 3D genome. *Nat Rev Cancer*. 2022;22(9):533–46.
32. Chen K, Wallis JW, McLellan MD, Larson DE, Kalicki JM, Pohl CS, McGrath SD, Wendt MC, Zhang Q, Locke DP, et al. BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. *Nat Methods*. 2009;6(9):677–81.
33. Talevich E, Shain AH, Botton T, Bastian BC. CNVkit: genome-wide Copy Number Detection and visualization from targeted DNA sequencing. *PLoS Comput Biol*. 2016;12(4):e1004873.
34. Abyzov A, Urban AE, Snyder M, Gerstein M. CNVnator: an approach to discover, genotype, and characterize typical and atypical CNVs from family and population genome sequencing. *Genome Res*. 2011;21(6):974–84.
35. Rausch T, Zichner T, Schlattl A, Stutz AM, Benes V, Korbel JO. DELLY: structural variant discovery by integrated paired-end and split-read analysis. *Bioinformatics*. 2012;28(18):i333–9.
36. Cameron DL, Schröder J, Penington JS, Do H, Molania R, Dobrovic A, Speed TP, Papenfuss AT. GRIDSS: sensitive and specific genomic rearrangement detection using positional de Bruijn graph assembly. *Genome Res*. 2017;27(12):2050–60.
37. Layer RM, Chiang C, Quinlan AR, Hall IM. LUMPY: a probabilistic framework for structural variant discovery. *Genome Biol*. 2014;15(6):R84.
38. Chen X, Schulz-Trieglaff O, Shaw R, Barnes B, Schlesinger F, Källberg M, Cox AJ, Kruglyak S, Saunders CT. Manta: rapid detection of structural variants and indels for germline and cancer sequencing applications. *Bioinformatics*. 2016;32(8):1220–2.
39. Ye K, Schulz MH, Long Q, Apweiler R, Ning Z. Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics*. 2009;25(21):2865–71.
40. Kronenberg ZN, Osborne EJ, Cone KR, Kennedy BJ, Domyan ET, Shapiro MD, Elde NC, Yandell M. Wham: identifying structural variants of Biological Consequence. *PLoS Comput Biol*. 2015;11(12):e1004572.
41. Wong K, Keane TM, Stalker J, Adams DJ. Enhanced structural variant and breakpoint detection using SVMerge by integration of multiple detection methods and local assembly. *Genome Biol*. 2010;11(12):R128.
42. Becker T, Lee WP, Leone J, Zhu Q, Zhang C, Liu S, Sargent J, Shanker K, Mil-Homens A, Cerveira E, et al. FusorSV: an algorithm for optimally combining data from multiple structural variation detection methods. *Genome Biol*. 2018;19(1):38.
43. Almarri MA, Bergström A, Prado-Martinez J, Yang F, Fu B, Dunham AS, Chen Y, Hurles ME, Tyler-Smith C, Xue Y. Population structure, stratification, and Introgression of Human Structural Variation. *Cell*. 2020;182(1):189–e199115.
44. Ebert P, Audano PA, Zhu Q, Rodriguez-Martin B, Porubsky D, Bonder MJ, Sulovari A, Ebler J, Zhou W, Serra Mari R et al. Haplotype-resolved diverse human genomes and integrated analysis of structural variation. *Science* 2021, 372(6537).
45. Byrka-Bishop M, Evani US, Zhao X, Basile AO, Abel HJ, Regier AA, Corvelo A, Clarke WE, Musunuri R, Nagulapalli K, et al. High-coverage whole-genome sequencing of the expanded 1000 Genomes Project cohort including 602 trios. *Cell*. 2022;185(18):3426–e34403419.
46. Soylev A, Le TM, Amini H, Alkan C, Hormozdiari F. Discovery of tandem and interspersed segmental duplications using high-throughput sequencing. *Bioinformatics*. 2019;35(20):3923–30.
47. Ma C, Khederzadeh S, Adeola AC, Han XM, Xie HB, Zhang YP. Whole genome resequencing reveals an association of ABCC4 variants with preaxial polydactyly in pigs. *BMC Genomics*. 2020;21(1):268.
48. Yoon S, Xuan Z, Makarov Y, Ye K, Sebat J. Sensitive and accurate detection of copy number variants using read depth of coverage. *Genome Res*. 2009;19(9):1586–92.
49. Yang L. A practical guide for structural variation detection in the Human Genome. *Curr Protoc Hum Genet*. 2020;107(1):e103.
50. Sims D, Sudbery I, Ilott NE, Heger A, Ponting CP. Sequencing depth and coverage: key considerations in genomic analyses. *Nat Rev Genet*. 2014;15(2):121–32.
51. Pan-cancer analysis of whole genomes. *Nature*. 2020;578(7793):82–93.
52. Mao Y, Zhang G. A complete, telomere-to-telomere human genome sequence presents new opportunities for evolutionary genomics. *Nat Methods*. 2022;19(6):635–8.
53. Lin J, Yang X, Kusters W, Xu T, Jia Y, Wang S, Zhu Q, Ryan M, Guo L, Zhang C, et al. Mako: a graph-based Pattern Growth Approach to Detect Complex Structural variants. *Genomics Proteom Bioinf*. 2022;20(1):205–18.
54. Yang J, Chaisson MJ. TT-Mars: structural variants assessment based on haplotype-resolved assemblies. *Genome Biol*. 2022;23(1):110.
55. Popic V, Rohlicek C, Cunial F, Hajirasouliha I, Meleshko D, Garimella K, Maheshwari A. Cue: a deep-learning framework for structural variant discovery and genotyping. *Nat Methods*. 2023;20(4):559–68.
56. Linderman MD, Wallace J, van der Heyde A, Wieman E, Brey D, Shi Y, Hansen P, Shamsi Z, Liu J, Gelb BD et al. NPSV-deep: a deep learning method for genotyping structural variants in short read genome sequencing data. *Bioinformatics* 2024, 40(3).
57. Wang MS, Thakur M, Peng MS, Jiang Y, Frantz LAF, Li M, Zhang JJ, Wang S, Peters J, Otecko NO, et al. 863 genomes reveal the origin and domestication of chicken. *Cell Res*. 2020;30(8):693–701.

58. Gu LH, Wu RR, Zheng XL, Fu A, Xing ZY, Chen YY, He ZC, Lu LZ, Qi YT, Chen AH, et al. Genomic insights into local adaptation and phenotypic diversity of Wenchang chickens. *Poult Sci*. 2024;103(3):103376.
59. Bolger AM, Lohse M, Usadel B. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics*. 2014;30(15):2114–20.
60. Li H, Durbin R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*. 2009;25(14):1754–60.
61. Conway JR, Lex A, Gehlenborg N. UpSetR: an R package for the visualization of intersecting sets and their properties. *Bioinformatics*. 2017;33(18):2938–40.
62. Gao CH, Yu G, Cai P. ggVennDiagram: an intuitive, easy-to-Use, and highly customizable R Package to Generate Venn Diagram. *Front Genet*. 2021;12:706907.

Publisher's note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.