Supplementary material

- TandemTools results on the simulated datasets (DXZ1)
- TandemTools results on the simulated datasets (D6Z1)
- Analyzing ETRs in the GAGE locus at the human X chromosome
- TandemTools results on cenX assemblies
- TandemTools results on cen8 assembly
- Discordance test
- Unit-based statistic
- Alternative technologies for ETR assembly quality assessment

TandemTools results on the simulated datasets (DXZ1)

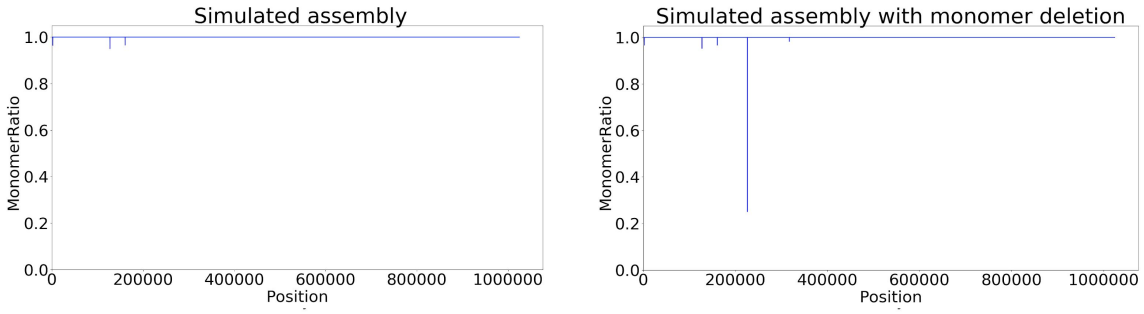

Figure S1. *MonomerRatio* for *simulated* and *simulated*$_{del\_monomer}$ assemblies. Even though *Ratio(CenMonomer)* is defined for each monomer (rather than for each nucleotide) in the centromere, we show nucleotide coordinates over the centromere (X-axis) for consistency with other metrics. The sharp drop in *Ratio(CenMonomer)* in the *simulated*$_{del\_monomer}$ assembly corresponds to the position of the monomer deletion.

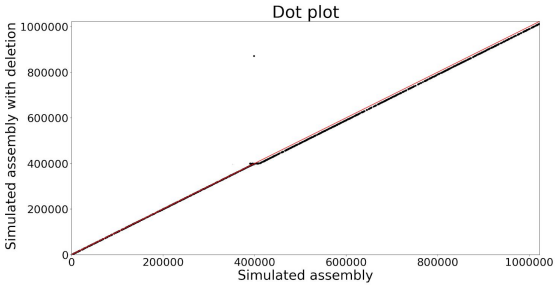

Figure S2. The dot plot illustrates the discrepancy between *simulated* and *simulated*$_{del}$ assemblies at the deletion breakpoint (400 kbp).

TandemTools results on the simulated datasets (D6Z1)

The preliminary analysis of human centromere 6 using ONT dataset generated by the T2T consortium (Miga et al., 2019) reveals that, unlike centromere X, centromere 6 has two main HORs: the canonical HOR sequence (D6Z1) consisting of 18 monomers and D6Z1 HOR with a deletion of 3rd, 4th, and 5th monomers. So we simulated 900 copies of the D6Z1, 30% of which harbored the deletion. All copies were randomly mutated with a 1% divergence rate from the consensus sequence (substitutions only). We concatenated these copies to obtain an ETR of length ~2.7 Mb. Afterward, we simulated 1,200 reads from this ETR using NanoSim (Yang et al., 2017) trained on the ONT dataset generated by the T2T consortium. We further introduced several artificial errors: a 5 kbp deletion at 300 kbp, a 20 kbp deletion at 1,700 kbp, and ~1% of the sequence length random single-nucleotide substitutions. We refer to the resulting sequence as *cen6*.

TandemMapper results. The comparison with minimap2 confirmed that TandemMapper better handles deletions and produces fewer inaccurate alignments (Table S1).

	correctly mapped reads	incorrectly mapped reads	# alignments extended through the deletion breakpoints	running time (s)	memory footprint (GB)
TandemMapper (solid *k*-mers)	**98.7% (1,527)**	**0.0% (0)**	**0**	541	5.6
minimap2	97.4% (1,507)	1.7% (27)	42	529	5.7
Winnowmap	97.4% (1,506)	1.8% (28)	42	**45**	**1.7**

Table S1. Benchmarking of TandemMapper, minimap2, and Winnowmap on the simulated cen6 sequence. Minimap2 and Winnowmap were run using recommended parameters for mapping ONT reads (`-cx map-ont`). The best value for each column is indicated in bold. A read is considered correctly mapped if its starting position is within 100 bp from the read simulated position calculated for the longest read alignment (an alignment is elongated to both ends of a read). Only reads longer than 5 kbp with alignments longer than 3 kbp were considered. The total number of such reads in the read set is 1,547. Although minimap2 and Winnowmap mapped in total 7 reads more than TandemMapper (1,534 vs 1,527), 5 out of these 7 reads came from the regions of the deletions and 2 reads were mapped incorrectly. The benchmarking was done on a server with Intel Xeon X7560 2.27 GHz CPUs using 16 threads.

Indel-based metrics. Figure S3 illustrates that discrepancies in the *breakpointRatio(Kmer)* and *breakpointRatio^{+}(Kmer)* values reveal both the 5 kbp and 20 kbp deletions.

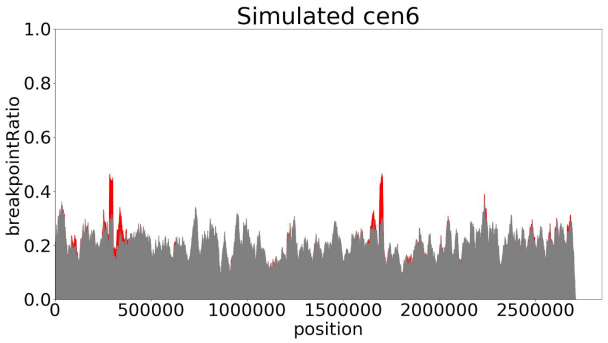

Figure S3. Breakpoint metric for cen6. The peaks at the breakpoint plot correspond to the positions of the deletions (300 kbp and 1,700 kbp). The red and gray plots are based on the *breakpointRatio(Kmer)* and *breakpointRatio*$^{+}$*(Kmer)* values, respectively.

***k*-mer-based metric.** The simulated cen6 sequence has a high number (17%) of spurious *k*-mers uniformly distributed along the sequence that is expected due to introduced artificial substitutions (Figure S4).

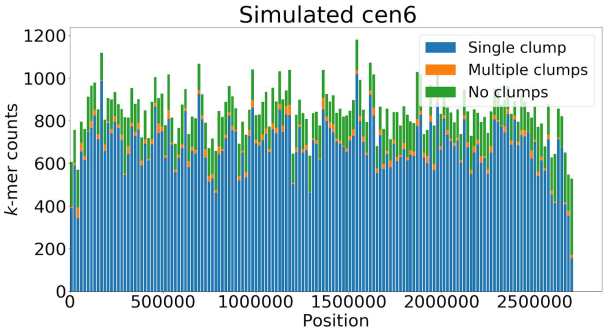

Figure S4. Distribution of different types of unique solid *k*-mers in the simulated cen6 sequence. Each bar shows the number of different types of *k*-mers in a bin of length 5 kbp.

Analyzing ETRs in the GAGE locus at the human X chromosome

We applied TandemTools to analyze the ETR in the GAGE gene cluster at the human chromosome X (Scanlan et al., 2004). This ETR is formed by a repeat unit of length of 9,556 bp (Killen et al., 2014).

We analyzed the GAGE locus in two versions of the T2T consortium assembly of the X chromosome (Miga et al., 2019): v0.6 version (before error correction) and v0.7 (after error correction). Figure S5 illustrates a sharp coverage drop and the corresponding peak in the breakpoint metric plot at ~410 kbp in the v0.6 assembly. Further analysis revealed a 1.7 kbp deletion at this position that was corrected in the v0.7 assembly as described in Miga et al., 2019. Both the coverage and breakpoint metric plots for the v0.7 assembly suggest that this assembly has no structural errors.

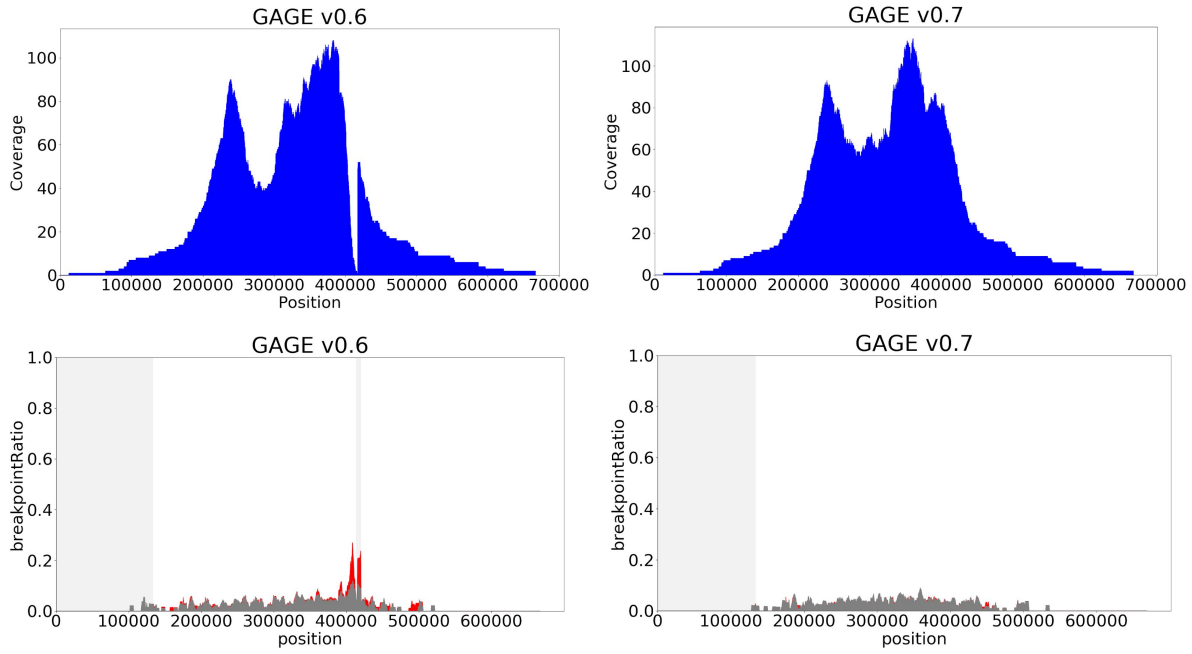

Figure S5. Coverage (top) and breakpoint (bottom) metrics for the v0.6 (left) and v.07 (right) assemblies of the GAGE locus. The GAGE locus spans from 210 kbp to 420 kbp. For the breakpoint metric, the red and gray plots are based on the *breakpointRatio(Kmer)* and the *breakpointRatio*$^{+}$*(Kmer)* values, respectively.

TandemTools results on cenX assemblies

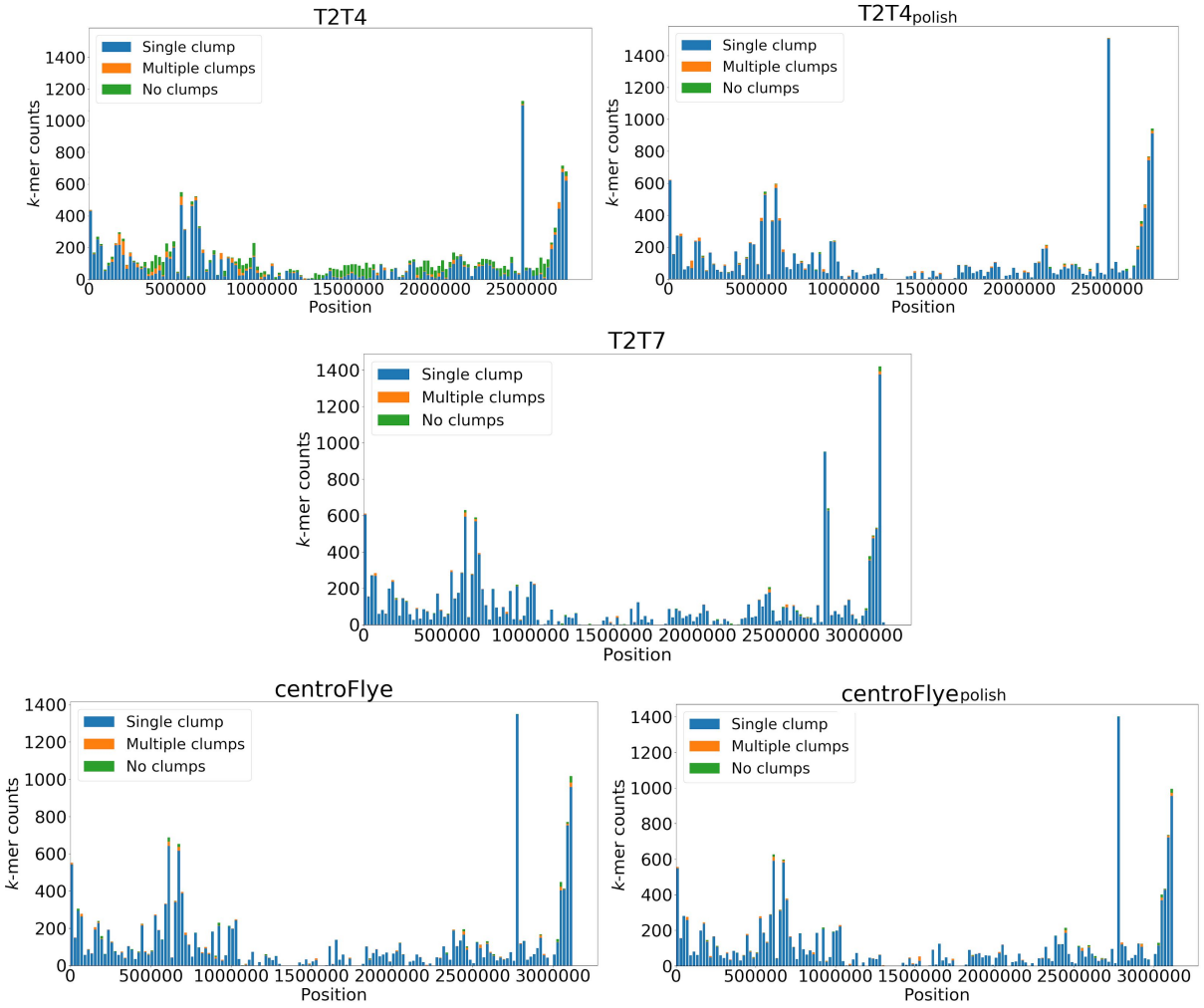

Figure S6. Distribution of different types of unique solid *k*-mers along the T2T4, T2T4$_{polish}$, T2T7, centroFlye, and centroFlye$_{polish}$ assemblies. Each bar shows the number of different types of *k*-mers in a bin of length 20 kbp. The plot for the T2T4 assembly shows that many unique solid *k*-mers in the assembly are spurious due to limited polishing.

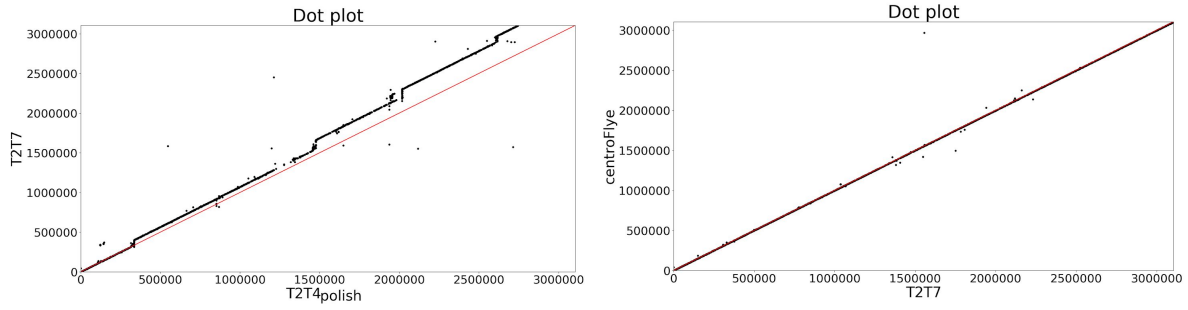

Figure S7. Dot plots for the T2T7 versus T2T4$_{polish}$ and T2T7 versus centroFlye assemblies.

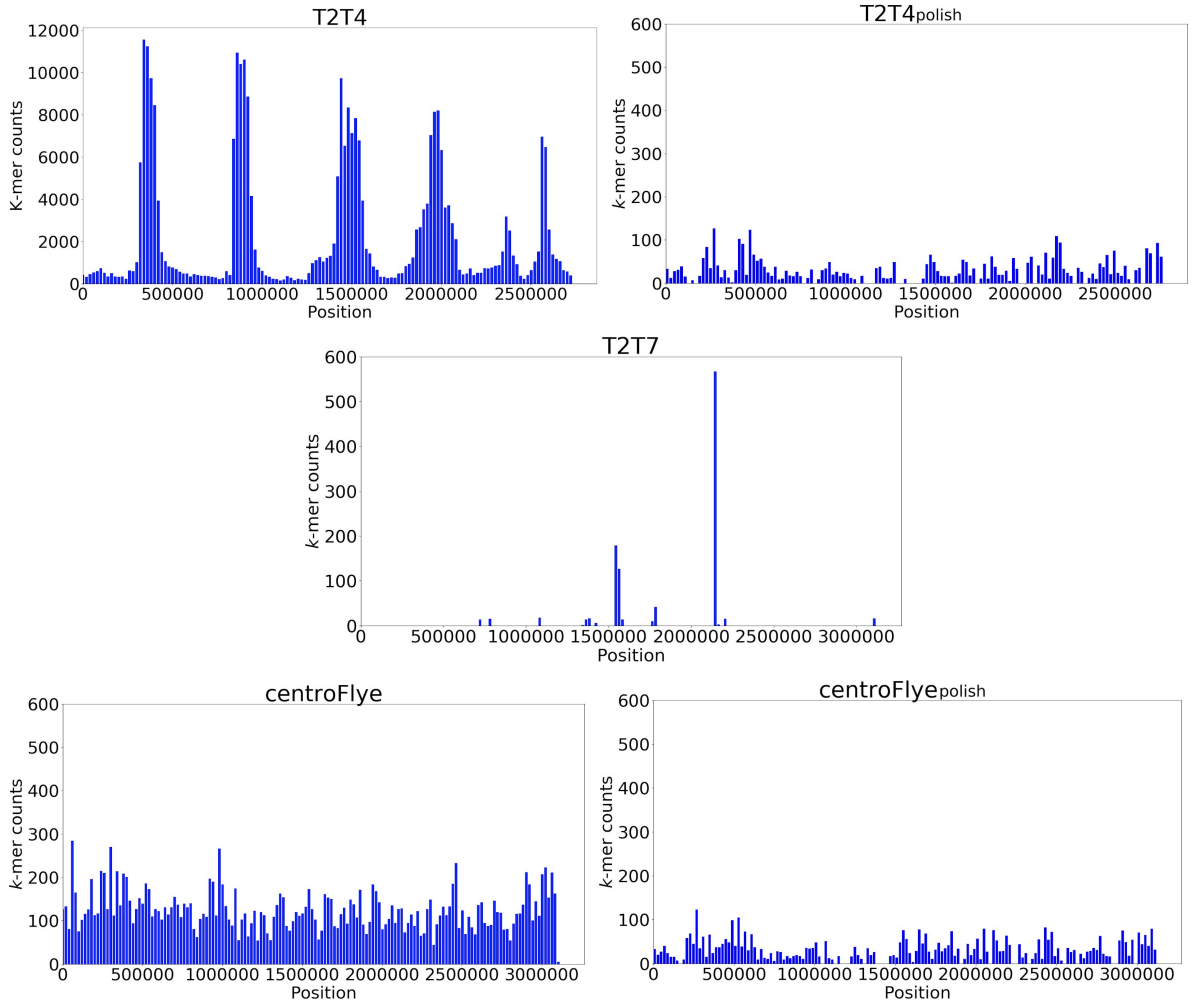

Figure S8. Distribution of *k*-mers absent in PacBio HiFi read set but present in the T2T4, T2T4$_{polish}$, T2T7, centroFlye, and centroFlye$_{polish}$ assemblies. Each bar shows the number of *k*-mers in a bin of length 20 kbp that are present in an assembly but missing in HiFi reads. The numbers of *k*-mers that do not occur in the HiFi read set are 223,579 (T2T4), 1,711 (T2T4$_{polish}$), 842 (T2T7), 7,867 (centroFlye), and 1,635 (centroFlye$_{polish}$).

TandemTools results on cen8 assembly

We ran TandemTools on the assembly of the human centromere 8 (cen8) generated by HiCanu (Nurk et al., 2020). Unlike the centromere X, centromere 8 has a more complex structure (Ge et al., 1992). The higher order array (D8Z2) comprises 3 predominant HOR units – 3.9, 2.5 and 1.9 kbp lengths – that occur in clusters within the array. While cenX was assembled from ultra-long error-prone ONT reads, cen8 was assembled by HiCanu from shorter but accurate HiFi reads. Since ultra-long reads often provide more reliable mappings in ETRs, we analyzed cen8 assembly using ONT reads.

Since HiFi reads are highly accurate, the HiCanu assembly achieves an excellent base-level accuracy without any polishing (Figure S9). However, because HiFi reads are shorter than ONT reads, their assemblies may collapse some regions in ETRs. The TandemQUAST breakpoint metric revealed an indel at 450 kbp (Figure S10). Comparison with the cen8 assembly generated by the T2T consortium from ONT reads provided further support for 14 kbp long deletion in the HiCanu assembly at this position (https://github.com/nanopore-wgs-consortium/CHM13).

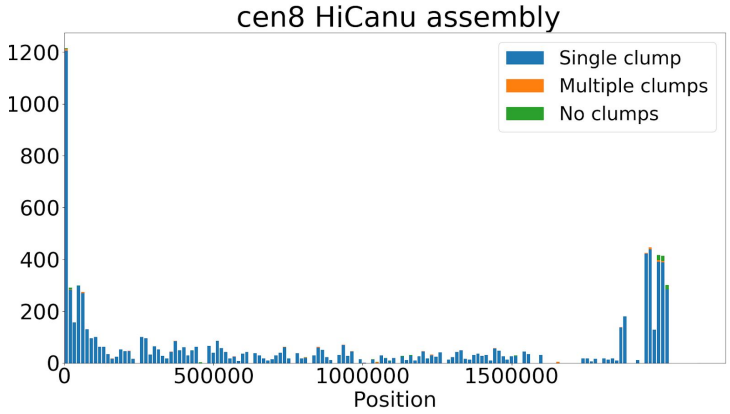

Figure S9. Distribution of different types of unique solid *k*-mers along the cen8 assembly. Each bar shows the number of different types of *k*-mers in a bin of length 20 kbp. 99% of solid k-mers form a single clump in the assembly.

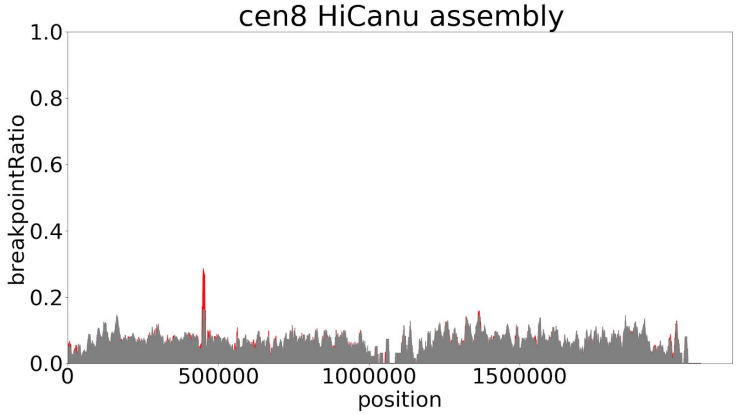

Figure S10. The breakpoint (bottom) metric plot for the HiCanu assembly of cen8. For the breakpoint metric, the red and gray plots are based on the *breakpointRatio(Kmer)* and the *breakpointRatio*$^{+}$*(Kmer)* values, respectively.

Discordance test

Bzikadze and Pevzner, 2019 introduced the *discordance test* for comparing two assemblies. TandemQUAST uses a slightly modified version of this test. We say that *k*-mer is shared between an assembly and a read aligned to this assembly if it occurs in both the assembly and the read approximately at the same position in their alignment. Given a set of *k*-mers *Kmers*, we define *sharedKmers*(*R, A*) as the number of *k*-mers from *Kmers* that are shared between read *R* and assembly *A*. The larger the size of *sharedKmers*(*R, A*) is, the more the assembly is "supported" by the read with respect to a given set of *k*-mers. Given a read set *Reads*, we calculate *sharedKmers(Reads, A)* as the sum of *sharedKmers(R, A)* over all reads in *Reads*.

To compare two assemblies *A'* and *A''*, we define *Kmers* as the set of shared solid unique *k*-mers between them. The discordance between these assemblies is computed as *discordance(A', A'') = sharedKmers(Reads, A') - sharedKmers(Reads, A'')*. We consider a read *R* as *discordant* with respect to assemblies and a set of *k*-mers *Kmers* if |*discordance(A', A'')*| exceeds *k*. A discordant read is classified as *voting* for *A'* (*A''*) if *discordance(A', A'')* is positive (negative).

Figure S11 shows a cluster of discordant reads voting for *simulated* over $simulated_{del}$ assembly at the deletion breakpoint and no reads voting for $simulated_{del}$ assembly.

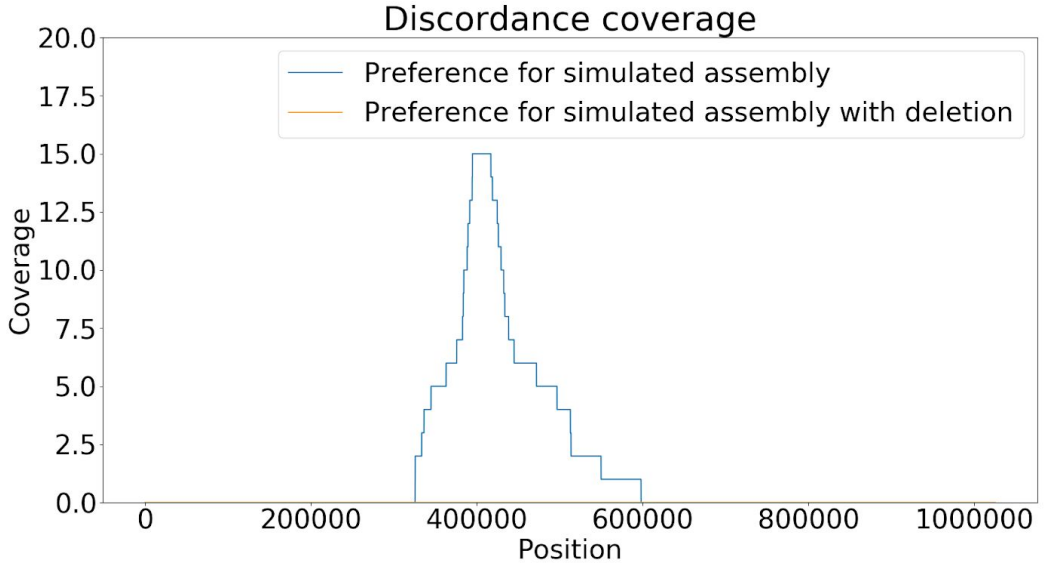

Figure S11. Coverage of *simulated* and $simulated_{del}$ assemblies by discordant reads.

Unit-based statistic

If an assembly is represented as an array of monomers, TandemQUAST splits this array into repeated *units* (a sequence of monomers, for example, a series of twelve monomers forming a HOR on cenX can be represented as $m_1m_2...m_{12}$. To automatically derive a unit's decomposition into monomers, TandemQUAST uses the StringDecomposer tool (Dvorkina et al., 2020) to translate the assembly from the nucleotide to the monomer alphabet (the alphabet size is the number of distinct monomers). Afterward, it considers all *t*-mers in the monomer alphabet (the default value *t*=3) and constructs a weighted de Bruijn graph on these *t*-mers, where the weight of an edge is defined as the number of occurrences of the corresponding *t*-mer in the assembly.

TandemQUAST starts traversing the graph from a vertex *v* incident to the maximum weight edge (with ties broken arbitrarily) and then iteratively selects an edge of maximal weight until it forms a cycle of at least 4 edges. At each step, we check that the string corresponding to the selected path is presented in the monomer sequence. Finally, the string corresponding to the selected cycle is added to a set of *units*. If the unit is already presented in the set we increase its count by 1. The weight of each traversed edge is decreased by 1. Edges of weight 0 are removed from the graph. The procedure is repeated until the graph contains no cycles. We consider a unit as *standard* if (i) it is not a substring of any other unit; (ii) it appears at least *MaxCount*/2 where *MaxCount* is the number of occurrences of the most frequent unit.

TandemQUAST reports the assembly length in units, the number of distinct units, the coordinates and monomer sequence of each unit in the assembly, and the unit frequency in the assembly and the read set.

Analysis of the $simulated_{del\_monomer}$ assembly demonstrated that it has 495 units, 494 of them are standard 12-monomers $m_1...m_{12}$ units, and, as expected, one unit has non-standard sequence $m_1m_2m_3m_7...m_{12}$.

Analysis of the simulated cen6 sequence correctly identified that it has 900 units in total, where 730 units have monomer sequence $m_1...m_{18}$ and 270 units have monomer sequence $m_1m_2m_6...m_{18}$.

Table S2 lists the distinct HOR units and their distribution in the cenX assemblies and the reads. The centroFlye and T2T7 assemblies share the same set of 1,508 units, including 37 non-standard units. The centroFlye and $centroFlye_{polish}$, as well as T2T4 and $T2T4_{polish}$ assemblies also have the same set of units. The T2T4 assembly has a smaller length than the centroFlye and T2T7 (~2.7 Mbp vs ~3.1 Mbp), so the total number of units is lower, although the set of non-standard units is the same. All non-standard units are supported by reads.

	T2T4	T2T7	centroFlye	Reads
$m_1...m_{12}$	1,298	1,471	1,471	25,654
$m_1...m_{10}m_6...m_{12}$	8	8	8	376
$m_1...m_6m_9...m_{12}$	8	8	8	328
$m_1...m_9m_5...m_{12}$	4	4	4	159
$m_1...m_5m_7...m_{12}$	5	5	5	164
$m_1...m_5m_8...m_{12}$	3	3	3	255
$m_1...m_{10}$	1	1	1	226
$m_1...m_5$	1	1	1	204
$m_1...m_4m_{10}...m_{12}$	1	1	1	154
$m_2...m_{12}$	1	1	1	164
$m_3...m_{12}$	1	1	1	231
$m_6...m_{12}$	1	1	1	106
$m_1...m_7$	1	1	1	122

Table S2. Distribution of distinct units in the T2T4, T2T7, and centroFlye assemblies and the read set. The first and the last units in the assembly are not listed in the table. The first unit in T2T4 and T2T7 assemblies is $m_4...m_{12}$, and in the centroFlye assembly is $m_6...m_{12}$. The last unit in all assemblies is $m_1...m_{10}$. The first unit in centroFlye assembly differs from those in T2T4 and T2T7 assemblies because of the choice of start sites and differences in the consensus HOR sequence.

Alternative technologies for ETR assembly quality assessment

CLR PacBio reads probably add little to centromere assemblies since they are shorter than ONT reads and have similar error rates. Although they are better suited for polishing than ONT reads, difficulties with mapping shorter error-prone reads to repetitive centromeres may offset this advantage.

Optical mapping data was used by the T2T Consortium only for quality assessment (Miga et al., 2019). Even though incorporating optical mapping data into TandemTools remains an open problem, we hypothesize that the quality assessment metrics based on other data types, such as HiFi PacBio read, will be more beneficial.

Hi-C data. Mapping of short Hi-C reads to ETRs presents a complex challenge that, to the best of our knowledge, remains unaddressed. Even though Hi-C data may be useful for quality assessment of ETR assemblies (especially for analysis of diploid assemblies) it is non-trivial to incorporate such data into TandemTools.

10X Genomics data may potentially be useful but it is also non-trivial to incorporate this data type in TandemTools. We note that an even simpler problem of developing a 10X-based tool for analyzing the quality of general assemblies remains unsolved.

References:

1. Ge, Y. *et al.* (1992) Sequence, higher order repeat structure, and long-range organization of alpha satellite DNA specific to human chromosome 8. *Genomics*, **13**, 585-593.
2. Killen,M.W. *et al.* (2011) Configuration and rearrangement of the human GAGE gene clusters. *Am. J. Transl. Res.*, **3**, 234–242.
3. Scanlan,M.J. *et al.* (2004) The cancer/testis genes: review, standardization, and commentary. *Cancer Immun.*, **4**, 1.