# Supplemental Material for lobSTR: A Novel Pipeline for Short Tandem Repeats Profiling in Personal Genomes

Melissa Gymrek, David Golan, Saharon Rosset &Yaniv Erlich

## Table of Contents

## SUPPLEMENTAL TEXT

## lobSTR Algorithm

*Sensing*

The aim of the sensing step is to find informative reads and characterize their STR sequence. The first task of the algorithm is to detect whether a read contains a repetitive sequence. The algorithm breaks the sequence read into overlapping windows with length of *w* nucleotides and *r* nucleotide overlap between consecutive windows. In practice, we use *w=24* and *r*=12. Then, it measures the sequence entropy of each window, according to:

$$E(S_j) = -\sum_{i \in \Sigma} f_i \log_2 f_i \qquad (1)$$

where *E* is the entropy, $S_j$ is the sequence of the *j*-th window, Σ is the alphabet set, *i* is a symbol in the alphabet, and $f_i$ is the frequency of symbol *i*. A fully random sequence results in the maximal entropy score that equals to $\log_2 |\Sigma|$, whereas a repetitive sequence overuses a few symbols and results in a low entropy score, ideally zero in the case of a perfect homopolymer run.

The entropy score proved extremely powerful in discriminating STR sequences from other genomic sequences (**Supplemental Figure 2A**). We calculated the entropy score of sliding windows of 24bp from all documented human STR sequences of repeat unit length of 2-6bp that span up to 100bp. In parallel we scored one million randomly sampled human genomic sequences of 24bp. Then, we classified the input sequences according to their entropy score. The area under the receiver-operating curve (ROC) was 98.3% when the entropy measurement used the four nucleotides as the input alphabet. We further boosted the classification performance by calculating the entropy using dinucleotide symbols, meaning that “AA” maps to one symbol, “AC” maps to a different symbol, and so forth. In this case, the area under the ROC climbed to 99.4%, which renders it a nearly perfect classifier. Accordingly, lobSTR uses the dinucleotide symbols for the entropy measure, and we empirically found that an entropy threshold of 2.2 bits provides the optimal performance in terms of speed and number of aligned STR reads.

lobSTR uses the pattern of entropy scores to identify informative reads that fully encompass STR regions. These reads display a series of windows with entropy score below-threshold (the STR region) that are flanked by one or more windows with entropy score above-threshold (the non-repetitive regions) (**Supplemental Figure 2B**). The algorithm only retains reads that follow this pattern. Approximately 97% of whole genome sequence reads are excluded by this rapid procedure. This significantly contributes to the algorithm's speed, since only a few simple entropy calculations are required to identify the informative STR reads in massive sequencing datasets.

The next task of the sensing step is to determine the length of the repeat unit. Most STR loci do not contain a perfect series of the same repeat unit (Benson 1999). We took a spectral analysis approach that quickly integrates information over the entire STR region to reliably identify the repeat consensus even in imperfect repeats (Sharma et al. 2004). Starting from the window with the lowest entropy score, consecutive windows scoring below the threshold are merged. The sequence of the merged repetitive region is represented as *M*, an *n* x 4 binary matrix, where *n* is the number of nucleotides in the repetitive region, the *i*-th row of the matrix corresponds to the *i*-th position of the sequence, and each column corresponds to a different nucleotide type (A,C,G,T). For instance, the DNA sequence ACCGT is represented as:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The power spectrum of the STR matrix is calculated by performing a Fast Fourier Transform along the columns of the matrix:

$$S(f) = \sum_{y=1}^{4} \left( \sum_{x=1}^{n} M_{x,y} \cdot e^{-\frac{2\pi i x f}{n}} \right)^2 \quad (3)$$

Where $M_{x,y}$ is the element in the *x*-th row and *y*-th column of *M*.

STRs have a unique fingerprint in the frequency domain (Sharma et al. 2004; Zhou et al. 2009). Similar to repetitive signals in the time domain, the spectral response of STR elements is characterized by harmonics - a strong signal in recurrent frequency bins and a weak signal in other bins (**Supplemental Figure 2C**). The peaks of the

harmonics for a repeat unit of length *k* dwell in the *n*i/k mod n* bins, *i* = {0, ±1, ±2, …}. For instance, in a case of *n*=24, a dinucleotide STR generates a strong signal in bins 0 and ±12. A trinucleotide STR generates a strong signal in bins 0, ±8, and ±16. The algorithm integrates over the normalized energy of the first harmony (i.e. *i=1*) of each possible repeat unit between 2 to 6bp. The consensus repeat unit length is selected according to the highest energy of the corresponding frequency bin (**Supplemental Figure 2D**). Some STR regions may show strong signals in more than one energy bin (i.e., repeats of period 4 show strong energy in both the second and fourth harmonics, and repeats with several insertions or deletions may have more than one strong harmonic). If the second highest harmonic has energy within 30% of the highest, lobSTR will attempt alignment using the second best period choice if alignment using the first choice fails.

Finally, the algorithm determines the actual STR sequence. It uses a rolling hash function to record all possible *k*-mers in the STR region, where *k* is the reported repeat unit length described above. The most frequently occurring *k*-mer is set to be the repeat unit of the STR. The output of the sensing step is (a) the consensus sequence of the STR's repeat unit in the canonical form (see **Supplemental Methods** for the canonical form definition) (b) the sequence read divided into three regions: the STR region, and upstream and downstream flanking regions that correspond to the location of the above threshold windows.

*Alignment*

The aim of the alignment step is to reveal the identity and the repeat length of an STR-containing read. We do not attempt to align the entire sequence read to the genome to avoid time-prohibitive gapped alignment. Instead, lobSTR employs a divide-and-conquer approach. It separately anchors the upstream and downstream flanking regions of STR-containing sequence reads, without mapping the STR region itself. This procedure identifies the genomic location of the STR and reveals the repeat length by measuring the distance between the flanking regions.

A major challenge of the divide-and-conquer approach is to specifically map the short flanking regions to the genome. To circumvent this problem, we restrict the alignment to STR loci with the same repeat sequence that was reported by the sensing step. We built a reference that holds the flanking regions of all the 240,351 STR loci with

repeat unit 2-6bp in the human genome according to the Tandem Repeat Finder table (Benson 1999). The flanking strings are compressed using a Burrows Wheeler Transform (Burrows and Wheeler 1994) (BWT) to allow efficient searching. All flanking strings of STRs with the same repeat structure are organized under the same BWT structure (**Supplemental Methods**). Thus, lobSTR only searches a single BWT data structure that corresponds to the repeat sequence, which typically holds up to a few thousand loci. Then, lobSTR intersects the potential mapping positions of the upstream and downstream regions to identify a single compatible location and excludes multiple mappers. This procedure not only speeds up the alignment, but enables higher rates of unique mapping even when the flanking regions are only a few nucleotides long.

To determine the length of the STR in the read, the algorithm uses the following equation:

$$L = s - (d - u) + L_{ref} \quad (4)$$

Where $L$ is the observed STR length, $s$ is the length of the sequence read, $d$ is the genomic coordinate of the last nucleotide in the downstream region, $u$ is the genomic coordinate of the first nucleotide of the upstream region, and $L_{ref}$ is the length of the STR region in the reference genome.

One important aspect of Eq. 4 is that inaccuracies in the sensing step regarding the exact boundaries of the STR do not affect the reported length of the STR. However, insertions or deletions in the flanking regions might be reported as STR differences, although they are not actual differences in the STR region itself. To mitigate this issue, lobSTR performs local realignment of the entire read once a match is found using the Needleman-Wunsch algorithm (Needleman and Wunsch 1970). Indels that are detected in the flanking regions are not taken into account and removed from Eq. 4, providing accurate length calling of the STR region. In addition, the local realignment is used to produce a CIGAR string with the locations of the indels in the read. Downstream genotype callers can use the output of lobSTR to call SNPs and indels in the STR region and its flanking regions.

The output of the alignment step is the genomic coordinates of the aligned read, the strand, the STR region extracted from the read, the STR motif, the nucleotide length difference compared to the reference genome as described above, the CIGAR string,

and the realignment score. We report the alignments in a custom tab-delimited format, as well as in the BAM format (Li et al. 2009) to ensure compatibility with other downstream bioinformatics tools.

*Allelotyping*

The aim of the allelotyping step is to determine the most likely alleles of each STR locus by integrating information from all aligned reads and the expected stutter noise, which is created due to *in vitro* slippage events during sample preparation. This part of the program uses a BAM file as input and reports the allelotype calls.

By analyzing real sequencing data, we found that the length of the repeat unit is a major determinant of the stutter noise distribution (**Supplemental Methods**). In accordance with the mutation dynamics of STRs previously reported (Ellegren 2004), short repeat units are associated with higher stutter noise and long repeat units are more immune to noise (**Supplemental Figure 3A**). We did not find a significant association between stutter noise and the length of the STR (**Supplemental Figure 3B**) as was observed in past studies (Hauge and Litt 1993).

We developed a generative model for stutter noise that consists of two steps: (a) with probability $\pi(k)$, the read is a product of stutter noise, where $k$ denotes the repeat unit length (b) if the read is a product of stutter noise, then with probability $\mu(s; \lambda_k)$, the noisy read deviates by $s$ base pairs from the original allele, where $\mu(s; \lambda_k)$ is a Poisson distribution with parameter $\lambda_k$. The probabilities that the deviation is positive (repeat expansion) or negative (repeat contraction) are equal.

The user has two options to estimate the model parameters $\pi(k)$ and $\lambda_k$. In the case of a male genome, the user can instruct lobSTR to scan the hemizygous sex chromosomes to accumulate unambiguous data about stutter noise distribution. The algorithm observes the stutter probability for each repeat unit length and uses a logistic regression to infer $\pi(k)$ (**Supplemental Figure 3A**) and a Poission regression to learn $\lambda_k$. In the case of a female genome, users can use pre-computed values either from our observations or analyze male data in their collection.

Overall, the probability of generating a read with $L$ bp in the STR region from a hemizygous locus with an STR with $A$ bp in the STR region is:

$$P(L|A,k) = \begin{cases} \left(1-\pi(k)\right) & if\ L = A \\ \frac{\pi(k)}{2}\mu(|A-L|-1,\lambda_k) & otherwise \end{cases} \quad (5)$$

In a diploid STR locus with *A* and *B* repeat lengths, we use the following heuristic to approximate the likelihood of observing a read with length *L*:

$$P(L|A,B,k) = \max(P(L|A,k)\,, P(L|\,B,k)) \quad (6)$$

This heuristic was found to be more robust when the two STR alleles have large length differences.

Let $\vec{R}$ be a vector that describes the STR lengths of sequence reads from the same locus after removing PCR duplicates (**Supplemental Methods**). Since each remaining sequence read is a product of an independent series of PCR rounds, we assume that the stutter noise of different entries in $\vec{R}$ is independent. Accordingly:

$$\log P\left(\vec{R}\middle|A,B,k\right) = \sum_{L\in\vec{R}} \log P(L|A,B,k) \quad (7)$$

Thus, the most likely allelotype call is when Eq. (7) is maximized with respect to *A* and *B*. To find the best bi-allelic combination, we simply iterate over all possible pairs of STR lengths observed at the interrogated locus and compute the likelihood of generating the observed data given the noise model. For example, if $\vec{R} = (13,13,12,12,12)$, we calculate the log likelihood in Eq. 7 for the combinations: (*A*=12,*B*=12), (*A*=12,*B*=13), and (*A*=13,*B*=13). In addition to the log likelihood score, we require a minimum threshold of the variant allele in order to call a locus as heterozygous, with a default threshold of 20% and a minimum percentage of reads supporting the resulting allelotype, which defaults to 50%. In the case of sex chromosome loci for a male sample, only homozygous allelotypes are considered. The most likely (A, B) combination is reported.

For each STR locus, the allelotyping step returns the chromosome, start, and end of the locus, the STR motif and period, the reference repeat number from TRF, the allelotype call given as the number of base pairs difference from reference for each allele, coverage, number of reads agreeing with the allelotype call, the number of reads disagreeing with the allelotype call, and the number of reads supporting each observed allele.

## Technical Evaluation of lobSTR

*Comparison between lobSTR sensing step and the TRF tool*

Tandem Repeat Finder was developed to find repetitive elements in large sequence contigs. Conceptually, it could also process short reads and replace the lobSTR sensing step in characterizing STR repeats. To compare the performance of the two lobSTR sensing step and TRF, we challenged the two tools with a set of 5 million 101bp whole-genome Illumina reads. To make a fair comparison, TRF was restricted to a maximum repeat unit period of six nucleotides and lobSTR ran on a single CPU.

Our results indicate that lobSTR's sensing step is significantly more adequate for high throughput sequencing data. lobSTR running time was just under 8 minutes compared to 6.5 hours for TRF (about 50 times slower, **Supplemental Figure 4A**). This means that analyzing personal genomes would take weeks instead of half a day of running time. Moreover, 94% percent of reads that were flagged as informative by both methods were reported with the same repeat sequence (**Supplemental Figure 4B**). Most of the discordant results occurred in STRs of period 5 or 6 where lobSTR and TRF could not reach a consensus regarding the repeat unit of imperfect repeats. Last, lobSTR flagged as informative 75%-85% of the reads that were flagged by TRF, with higher sensitivity with increasing STR purity (**Supplemental Figure 4C**). Thus, while lobSTR cannot detect every read that is detected by TRF, it does reach high sensitivity with 1/50 of the running time which is more suited to the ultra-exponential trajectory of high throughput sequencing datasets.

*lobSTR performance with different sizes of STRs*

The size of the flanking regions determines the mappability of STR containing reads. In order to find the minimal flanking regions, we extracted genomic sequences of 100bp upstream and downstream of all STRs in the TRF table and organized them in prefix trees according to their canonical repeat unit. Then, we exhaustively aligned target STRs by allowing increased flanking region lengths and reporting the minimal length when a unique and correct alignment was achieved. Since this step is time prohibitive, we focused our analysis on a set of 2050 STR from the CODIS set, exonic regions, and genealogical Y-STR markers that were covered by 100bp reads. Our results show that a total of 8-9bp of upstream and downstream flanking regions is a lower bound for unique

alignment of 80% of tested STRs (**Supplemental Figure 5A**). This means that with 100bp reads, lobSTR can theoretically detect STR regions of up to 84nt.

We also determined the power of lobSTR to detect reads with very short STR regions due to strong repeat contraction. These reads have higher entropy and might not cross the threshold in the sensing step. To simulate this effect, we ran TRF on a set of 5 million input reads in a setting returning detected STRs as few as 12bp long. We then measured the performance of lobSTR to detect reads from these short STR loci and found that repetitive elements with 12nt were well captured (**Supplemental Figure 5B**. Our overall results suggest that lobSTR can perform well in detecting STRs of 12-84bp.

*lobSTR performance on various sequencing platforms*

To test lobSTR performance on other sequencing platforms than Illumina, we ran the algorithm on publicly available genomes from three different platforms: Sanger (Craig Venter genome) (Levy et al. 2007), 454 (Watson genome) (Wheeler et al. 2008), and IonTorrent (Moore genome) (Rothberg et al. 2011). In the absence of orthogonal information about STRs in these genomes, we estimated the performance of lobSTR by several parameters: (a) the ratio of aligned STR reads to the total input (b) the fraction of reads with a non-integer number of repeat units different from reference (c) the coverage of STR loci (**Supplemental Table 5**).

As expected from its long read length and high accuracy, Sanger sequencing showed the best performance. It produced the best ratio of reads that aligned to STR loci and showed the lowest fraction (7.3%) of STR reads with a non integer number of repeat units difference from reference. Importantly, the Sanger fraction of non-integer number of repeats was close to the Illumina fraction (8.0%). 454 produced more STR aligned reads per amount of sequencing data than Illumina but 25% of the STR reads showed a non-integer number of repeat units. IonTorrent showed the worst performance in both the ratio of STR reads and non-integer repeats. The high number of STR reads with non-integer repeat units is presumably because 454 and Ion Torrent exhibit indel error when sequencing homopolymer runs that are abundant in many types of repeats (e.g. AAAAC).

Our results show that lobSTR can process sequencing files from other high throughput sequencing platforms and report STR reads. However, the accuracy of the

STR calls is expected to be inferior to that reported for Illumina. We expect that improvement in homopolymer sequencing in 454 and Ion Torrent will make their datasets more amenable to STR profiling.

*STR coverage as a function of input libraries*

We sought to explore the function of STR coverage by lobSTR to the genome-wide coverage of autosomal regions. Using the genome sequenced to 126x coverage by Ajay, *et al*. described in the main text, we sampled from the BAM file produced by lobSTR for a range of desired coverage levels. We then allelotyped only this subset of reads and counted how many STR loci were covered by at least one informative read (**Supplemental Figure 6**).

As a rule of thumb for 100bp reads, we found that STRs obtain an average coverage of approximately one-fifth the genome-wide autosomal coverage. In addition, we found that around 60,000 to 80,000 STRs can be covered by at least a single sequence read even with a shallow genomic coverage of less than 5x. The number of STRs covered by at least 1 read rapidly plateaued to ~180,000 loci after a genome-wide coverage of around 40x.

Certain STR loci in the TRF table cannot be detected regardless of the coverage. The main limitation is that 100bp reads cannot span 16% of the STR entries in TRF. Other STR regions dwell in repetitive elements and generate non unique alignments, such as the Y-STR marker DYS464a/b/c/d/e/f, which has multiple locations (Kehdy and Pena 2010). Reads from these loci will be flagged as multi-mappers and will be removed from the analysis. Finally, some STR regions do not pass the entropy threshold due to their imperfect repeat structure and will not be detected using lobSTR default parameters. This can be circumvented by lowering the lobSTR entropy threshold but will require substantial running time.

*Coverage bias in heterozygous loci*

We found a slight but statistically significant bias of 1:1.06 in the number of reads towards the shorter alleles in heterozygous loci (one sided Mann-Whitney test, $p < 0.05$). For instance, there are on average ~2 more reads that support the shorter allele when

an STR is covered by 30 reads. This observation can be explained by a PCR bias as reported by a previous study (Wattier et al. 1998). Since this small effect only becomes visible in ultra-high coverage STRs, lobSTR does not currently correct for it.

*Hardware requirements*

With the given TRF reference, lobSTR reaches a peak memory footprint of 0.3Gb regardless of input size and can process about 0.6 million reads per minute on a single processor. On 25 processors, lobSTR took 26 hours to process the genome sequenced to 126x coverage described in the main text, rendering the hardware requirements of lobSTR well within the range of routinely performed bioinformatics tasks such as SNP calling and short read alignment. The processing times for several genomes analyzed in this paper are given in **Supplemental Table 2.**

## SUPPLEMENTAL METHODS

*Building an STR reference*

The STR reference was built according to the entries of the Simple Tandem Repeat Table for human reference genome build hg18, available from the UCSC genome browser (this reference was used for all other results as well) (Kent et al. 2002). The table was filtered to include STRs with repeat unit lengths of 2-6bp. Nearly half of the loci are dinucleotide repeats. The number of STR loci with each repeat unit length is given in **Supplemental Table 6.** The 10 most common repeat units are given in **Supplemental Table 7**. The median length of STR regions is near 40bp for each repeat unit length. The distribution of repeat region sizes increases slightly with the repeat period, and less than 6% of STR loci span more than 100bp (**Supplemental Figure 7**). The majority of reference STRs lie in intergenic regions. 1,221 reference loci overlap exonic regions.

STRs display cyclic ambiguity. For example, consider the following STR: GACGACGACGACGAC. This STR can be described in three ways (GAC)5, (ACG)5, or (CGA)5, as well as by (GTC)5, (CGT)5, or (TCG)5 on the reverse strand. The sequence repeats in the TRF table are reported in a redundant format that does not distinguish between cyclic shifts. We converted all repeat sequences in the table to a canonical form in which the repeat sequence is the lexicographically highest among all possible cyclic representations of the sequence and their reverse complements. STRs whose repeat sequences contradicted the canonical repeat unit length, such as TTT listed as period 3 instead of 1, were removed from the reference. lobSTR reports the period of the STR according to the canonical form.

For mapping Illumina reads, the reference consists of the ±150bp flanking regions of each STR locus. We grouped reference sequences from loci with the same canonical STR repeat unit into a single FASTA file and built a single BWT index using the BWA function “bwa index –a is” on each file.

*PCR duplicate removal*

By default all reads with the same 5' coordinate and length are flagged as PCR duplicates and collapsed into a single read. The user has the option to turn PCR duplicate removal off. If a group of PCR duplicate reads are associated with more than one STR length, lobSTR uses a majority vote to determine the STR length of the collapsed read. If the majority vote results in a tie, the STR length of the collapsed read

is determined according to the read with the highest average quality score. All reported sequencing coverage numbers are given after removing PCR duplicates.

*Building a model for stutter noise*

We analyzed Illumina reads from approximately 6,000 hemizygous loci on the sex chromosomes of a male individual from our lab collection. We assumed that the mode of the STR lengths in each locus was the true allele. All reads differing from the modal allele differed by either one (76% of noisy reads) or two (24% of noisy reads) repeats. Initial analysis of the stutter noise was done using R and was implemented in C++/R in the allelotyping script that is part of the lobSTR package.

*lobSTR implementation details*

lobSTR is written in C/C++ and calls on R for allelotyping step. We made an effort to use existing, highly optimized libraries for lobSTR implementation to increase the speed of the program. The spectral analysis in the detection step was implemented using FFTW (Frigo and Johnson 1998) and the alignment step uses extensive parts of the BWA code (Li and Durbin 2009) for BWT-indexing and the BamTools library (Barnett et al. 2011) for reading and writing BAM files.

From the user's perspective, lobSTR consists of running two simple programs: one command for sensing and alignment, followed by a command for allelotyping aligned reads from a BAM file. In the simplest setting, the user just needs to specify the input files, the prefix name of the output files, and the location of the reference, which is provided with the software. However, we also provide advanced options that include modification of the detection threshold, re-sizing the FFT windows, and increasing the tolerance to sequencing errors in the flanking regions (**Supplemental Table 1)**. The user can also build a custom reference using a tool in the lobSTR package.

*lobSTR comparison across sequencing platforms*

Raw reads for the Watson genome were downloaded from the NCBI short read archive with accession SRX000114. Reads for the Moore genome were downloaded from the European short read archive with accession ERS024569. Reads for the Venter genome were downloaded from TraceDB (Genbank accession ABBA00000000). For the Venter genome, we trimmed the first 50bp of every read due to the high error rate at the

beginning of Sanger sequence reads and discarded reads whose length after trimming was less than 100bp.

## SUPPLEMENTAL TABLES

### Supplemental Table 1

| Parameter | Default | Description |
|---|---|---|
| -p, --threads | 1 | Number of threads to use for alignment |
| --fft-window-size | 24 | Size of the FFT sliding window in the sensing step |
| --fft-window-step | 12 | Step size of the FFT sliding window in the sensing step |
| --entropy-threshold | 0.45 | Percentage of maximum entropy score for a read to pass the sensing step |
| --minperiod | 2 | Minimum period to detect |
| --maxperiod | 6 | Maximum period to detect |
| --minflank | 10 | Do not align reads that have either flanking region with length less than this value |
| --maxflank | 25 | Trim flanking region ends to this maximum value before alignment |
| --extend-flank | 6 | Extend the flanking regions this many bp into the STR region |
| --max-diff-ref | 50 | Discard reads different from the reference allele by more than this number of nucleotides in length |
| -u | False | Discard reads differing by a non-integer repeat number from reference |
| -m | -1 | Number of mismatches to allow in each flanking region. If set, -r is ignored. |
| -r | 0.01 | Fraction of missing alignments given a uniform 2% error rate (see BWA manual parameter –n) |
| -g | 1 | Number of gap opens to allow in each flanking region |
| -e | 1 | Number of gap extends to allow in each flanking region |
| --no-rmdup | False | Specify to remove PCR duplicates |

**Supplemental Table 1:** lobSTR program parameters.

## Supplemental Table 2

| Genome | Autosomal Coverage | Processing time |
|---|---|---|
| HGDP00778 | 5x | 1.3 hours |
| Male individual | 36x | 8.5 hours |
| Ajay, *et al.* | 126x | 26 hours |

**Supplemental Table 2**: Processing times of Illumina genomes at various coverage levels. Processing times as a result of running lobSTR with 25 processors (-p 25).

## Supplemental Table 3

| Sample | Coverage | STR Aligned reads | STRs ≥1x | STRs ≥ 3x |
|---|---|---|---|---|
| HGDP00456 (Mbuti Pygmy) | 1.4x | 70,424 | 50,505 | 3,339 |
| HGDP00998 (Karitiana Native American) | 1.3x | 65,236 | 48,481 | 2,553 |
| HGDP00665 (Sardinian) | 1.5x | 91,157 | 58,623 | 6,215 |
| HGDP00491 (Bougainville Melanesian) | 1.7x | 97,398 | 61,463 | 6,523 |
| HGDP00711 (Cambodian) | 1.9x | 104,594 | 66,566 | 7,263 |
| HGDP01224 (Mongolian) | 1.7x | 93,938 | 61,356 | 5,767 |
| HGDP00551 (Papuan) | 1.6x | 94,540 | 61,486 | 6,036 |
| HGDP00521 (French) | 5.9x | 184,437 | 91,813 | 17,855 |
| HGDP01029 (San) | 7.7x | 192,798 | 93,376 | 18,928 |
| HGDP00542 (Papuan) | 5.9x | 118,232 | 72,654 | 7,065 |
| HGDP00927 (Yoruba) | 4.8x | 155,136 | 84,828 | 12,774 |
| HGDP00778 (Han) | 5.0x | 141,522 | 80,631 | 10,815 |

**Supplemental Table 3:** HGDP sample coverage and lobSTR results

## Supplemental Table 4

| Sample | Period | Marker | Refseq (hg18 diff) | Coverage | Converted lobSTR alleles[a] | Converted HGDP alleles[a] | Status[b] |
|---|---|---|---|---|---|---|---|
| HGDP01029 | 4 | D11S2371 | (TATC)11 | 5 | 0,0 | 0,0 | 2 |
| HGDP01029 | 4 | D12S1300 | (TAGA)12 | 5 | 0,0 | 0,0 | 2 |
| HGDP00521 | 4 | D6S1009 | (TATC)11 | 5 | 1,1 | 1,4 | 1 |
| HGDP01029 | 4 | D2S405 | (TAGA)12 | 5 | -2,-2 | -2,0 | 1 |
| HGDP00778 | 4 | D8S1108 | (TCTA)11 | 4 | 0,0 | 0,0 | 2 |
| HGDP01029 | 4 | D15S818 | (TAGA)10 | 4 | 3,3 | 3,3 | 2 |
| HGDP00551 | 4 | D1S1653 | (TCTA)12 | 4 | -1,0 | -1,0 | 2 |
| HGDP00521 | 4 | D5S2500 | (ATAG)11 | 4 | 0,1 | 0,1 | 2 |
| HGDP00927 | 4 | D10S1426 | (TATC)11 | 4 | 0,2 | 0,2 | 2 |
| HGDP01029 | 4 | D17S1308 | (TGTA)11 (-1) | 4 | -1,-1 | -1,-1 | 2 |
| HGDP00521 | 4 | D17S1308 | (TGTA)11 (-1) | 4 | -1,0 | -1,0 | 2 |
| HGDP00542 | 4 | D17S1308 | (TGTA)11 (-1) | 4 | -1,-1 | -1,-1 | 2 |
| HGDP00927 | 2 | D3S3644 | (AC)16 | 4 | -1,0 | 0,0 | 0.5 |
| HGDP00521 | 2 | D9S1779 | (AC)14 | 4 | 0,0 | -2,-2 | 0 |
| HGDP00521 | 2 | D8S503 | (AC)17 | 4 | 2,4 | 3,6 | 0 |
| HGDP00521 | 2 | D1S2682 | (CA)10 | 3 | 0,10 | 0,10 | 2 |
| HGDP00998 | 4 | D2S427 | (GATA)9 | 3 | 0,0 | 0,0 | 2 |
| HGDP00542 | 4 | D8S1113 | (GGAA)12 | 3 | -6,-6 | -6,-6 | 2 |
| HGDP01029 | 4 | D10S1425 | (GATA)11 | 3 | -5,0 | -5,0 | 2 |
| HGDP01029 | 4 | D7S1824 | (TCTA)11 | 3 | -3,-2 | -3,-2 | 2 |
| HGDP00778 | 4 | D1S3669 | (TATC)10 | 3 | 1,1 | 1,1 | 2 |
| HGDP00711 | 4 | D3S2432 | (AGAT)15 (-3) | 3 | -3,0 | -3,0 | 2 |
| HGDP00491 | 4 | D2S405 | (TAGA)12 | 3 | 0,0 | 0,0 | 2 |
| HGDP00998 | 4 | D16S3253 | (TAGA)9 | 3 | 0,0 | 0,0 | 2 |
| HGDP00998 | 4 | D9S301 | (GATA)15 | 3 | -7,-1 | -7,-1 | 2 |
| HGDP00998 | 4 | D19S586 | (TAGA)12 (+2) | 3 | 1,2 | 1,2 | 2 |
| HGDP00711 | 2 | D15S165 | (AC)21 | 3 | -6,-6 | -6,-6 | 2 |
| HGDP00665 | 2 | D20S103 | (AC)16 | 3 | -1,-1 | -1,-1 | 2 |
| HGDP00456 | 3 | D4S2394 | (ATT)11 | 3 | 0,0 | 0,0 | 2 |
| HGDP00711 | 4 | D11S2371 | (TATC)11 | 3 | 1,1 | 1,1 | 2 |
| HGDP00927 | 4 | D10S1239 | (ATCT)11 | 3 | 0,1 | 0,1 | 2 |
| HGDP00521 | 4 | D14S1434 | (GATA)10 | 3 | 0,0 | 0,0 | 2 |
| HGDP00491 | 4 | D19S591 | (TAGA)10 (-2) | 3 | -1,0 | -1,0 | 2 |
| HGDP00521 | 4 | D19S591 | (TAGA)10 (-2) | 3 | -2,1 | -2,1 | 2 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| HGDP00542 | 4 | D19S591 | (TAGA)10 (-2) | 3 | -1,0 | -1,0 | 2 |
| HGDP00551 | 4 | D19S591 | (TAGA)10 (-2) | 3 | -2,0 | -2,0 | 2 |
| HGDP01224 | 4 | D19S591 | (TAGA)10 (-2) | 3 | -2,1 | -2,1 | 2 |
| HGDP00927 | 4 | D17S2196 | (AGAT)9 (-2) | 3 | 0,2 | 0,2 | 2 |
| HGDP01029 | 4 | D2S1391 | (ATCT)14 | 3 | -2,0 | -2,0 | 2 |
| HGDP00456 | 3 | D4S2361 | (TTA)13 | 3 | -1,-1 | -1,-1 | 2 |
| HGDP00665 | 4 | D1S1653 | (TCTA)12 | 3 | 0,0 | 0,0 | 2 |
| HGDP01224 | 4 | D1S1653 | (TCTA)12 | 3 | -2,-1 | -2,-1 | 2 |
| HGDP00542 | 4 | D5S2500 | (ATAG)11 | 3 | 0,0 | 0,0 | 2 |
| HGDP00778 | 4 | D10S1426 | (TATC)11 | 3 | 1,1 | 1,1 | 2 |
| HGDP01029 | 4 | D5S2500 | (ATAG)11 | 3 | -2,0 | -2,0 | 2 |
| HGDP00456 | 4 | D17S1298 | (TGAA)8 | 3 | 0,0 | 0,0 | 2 |
| HGDP00927 | 4 | D17S1298 | (TGAA)8 | 3 | 3,3 | 3,3 | 2 |
| HGDP00542 | 4 | D19S254 | (AGAT)13 (-6) | 3 | -1,1 | -1,1 | 2 |
| HGDP00711 | 2 | D1S2682 | (CA)10 | 3 | 0,0 | 0,10 | 1 |
| HGDP00998 | 4 | D5S1457 | (ATAG)9 | 3 | 0,0 | 0,1 | 1 |
| HGDP00521 | 2 | D20S103 | (AC)16 | 3 | 2,2 | -1,2 | 1 |
| HGDP00998 | 4 | D20S482 | (TCTA)14 | 3 | 0,0 | 0,1 | 1 |
| HGDP01224 | 3 | D9S910 | (ATA)14 | 3 | -7,-7 | -7,-7 | 1 |
| HGDP01224 | 4 | D11S2363 | (TATC)14 | 3 | -1,-1 | -1,9 | 1 |
| HGDP00711 | 4 | D19S591 | (TAGA)10 (-2) | 3 | -1,-1 | -1,0 | 1 |
| HGDP00927 | 2 | D18S1390 | (TG)18 | 3 | -1,0 | -2,-1 | 0.5 |
| HGDP00778 | 2 | D8S503 | (AC)17 | 3 | 2,3 | 3,3 | 0.5 |
| HGDP00778 | 4 | D12S1300 | (TAGA)12 | 3 | 2,4 | 2,2 | 0.5 |
| HGDP00491 | 2 | D9S1779 | (AC)14 | 3 | 0,9 | -1,6 | 0 |

**Supplemental Table 4:** Comparison of lobSTR allelotype calls to the CEPH-HGDP results. Differences between the RefSeq sequence and hg18 are indicated in parentheses. [a]Converted allelotypes given in number of repeat units different from the reference. [b]Status: 2 = both alleles called correctly, 1 = one allele of a heterozygous locus called correctly, 0.5 = one allele called correctly and one incorrectly, 0 = no correct alleles called.

## Supplemental Table 5

| Genome (platform) | Coverage | Input reads | Avg. Read length | STR Aligned reads / million bp input | % Non-unit Reads* | STRs ≥ 1x | STRs ≥3x |
|---|---|---|---|---|---|---|---|
| Venter (Sanger) | 7.5x | 12.5M | 996 | 24.78 | 7.3% | 127,017 | 41,261 |
| Watson (454) | 7.4x | 75M | 183 | 10.41 | 25.0% | 83,079 | 25,488 |
| Moore (IonTorrent) | 10.6x | 860M | 261 | 0.79 | 43.5% | 65,758 | 13,413 |
| Ajay, *et al.* (Illumina, (100bp) | 126x | 14B | 100 | 4.36 | 8.0% | 180,309 | 167,175 |

**Supplemental Table 5:** lobSTR performance on four sequencing platforms. *Reads differing by a non-integer number of copies of the STR motif from the reference. (M = million, B = billion).

**Supplemental Table 6**

| Repeat unit size | # STR loci | Percentage |
|---|---|---|
| 2 | 106,457 | 44% |
| 3 | 17,383 | 7% |
| 4 | 70,847 | 30% |
| 5 | 28,746 | 12% |
| 6 | 16,626 | 7% |
| **Total** | 240,059 | 100% |

**Supplemental Table 6:** STR reference repeat unit size distribution.

## Supplemental Table 7

| Repeat unit | # STR loci |
|---|---|
| AC | 66,992 |
| AT | 25,661 |
| AAAT | 20,319 |
| AG | 13,778 |
| AAAG | 12,553 |
| AAAAC | 10,015 |
| AAGG | 9,862 |
| AAAC | 8,842 |
| AGAT | 7,127 |
| AAAAT | 7,115 |

**Supplemental Table 7:** Most frequent reference STR repeat units.

## SUPPLEMENTAL REFERENCES


Barnett DW, Garrison EK, Quinlan AR, Stromberg MP, Marth GT. 2011. BamTools: a C++ API and toolkit for analyzing and managing BAM files. *Bioinformatics* 27(12): 1691-1692.

Benson G. 1999. Tandem repeats finder: a program to analyze DNA sequences. *Nucleic Acids Res* 27(2): 573-580.

Burrows M, Wheeler DJ. 1994. A block-sorting lossless data compression algorithm.

Ellegren H. 2004. Microsatellites: simple sequences with complex evolution. *Nat Rev Genet* 5(6): 435-445.

Frigo M, Johnson SG. 1998. FFTW: An adaptive software architecture for the FFT. *Proceedings of the 1998 Ieee International Conference on Acoustics, Speech and Signal Processing, Vols 1-6*: 1381-1384.

Hauge XY, Litt M. 1993. A study of the origin of 'shadow bands' seen when typing dinucleotide repeat polymorphisms by the PCR. *Hum Mol Genet* 2(4): 411-415.

Kehdy FS, Pena SD. 2010. Worldwide diversity of the Y-chromosome tetra-local microsatellite DYS464. *Genet Mol Res* 9(3): 1525-1534.

Kent WJ, Sugnet CW, Furey TS, Roskin KM, Pringle TH, Zahler AM, Haussler D. 2002. The human genome browser at UCSC. *Genome Res* 12(6): 996-1006.

Levy S, Sutton G, Ng PC, Feuk L, Halpern AL, Walenz BP, Axelrod N, Huang J, Kirkness EF, Denisov G et al. 2007. The diploid genome sequence of an individual human. *PLoS Biol* 5(10): e254.

Li H, Durbin R. 2009. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* 25(14): 1754-1760.

Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R. 2009. The Sequence Alignment/Map format and SAMtools. *Bioinformatics* 25(16): 2078-2079.

Needleman SB, Wunsch CD. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol* 48(3): 443-453.

Rothberg JM, Hinz W, Rearick TM, Schultz J, Mileski W, Davey M, Leamon JH, Johnson K, Milgrew MJ, Edwards M et al. 2011. An integrated semiconductor device enabling non-optical genome sequencing. *Nature* 475(7356): 348-352.

Sharma D, Issac B, Raghava GP, Ramaswamy R. 2004. Spectral Repeat Finder (SRF): identification of repetitive sequences using Fourier transformation. *Bioinformatics* 20(9): 1405-1412.

Wattier R, Engel CR, Saumitou-Laprade P, Valero M. 1998. Short allele dominance as a source of heterozygote deficiency at microsatellite loci: experimental evidence at the dinucleotide locus Gv1CT in Gracilaria gracilis (Rhodophyta). *Mol Ecol* 7(11): 1569-1573.

Wheeler DA, Srinivasan M, Egholm M, Shen Y, Chen L, McGuire A, He W, Chen YJ, Makhijani V, Roth GT et al. 2008. The complete genome of an individual by massively parallel DNA sequencing. *Nature* 452(7189): 872-876.

Zhou H, Du L, Yan H. 2009. Detection of tandem repeats in DNA sequences based on parametric spectral estimation. *IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society* 13(5): 747-755.