Advanced Strategies for Alignment-based Real-time Analysis and Data Protection in Next-Generation Sequencing

Dissertation zur Erlangung des Grades
eines Doktors der Naturwissenschaften (Dr. rer. nat.)
am Fachbereich Mathematik und Informatik
der Freien Universität Berlin

vorgelegt von

TOBIAS PASCAL LOKA

Berlin
Oktober 2019

Erstgutachter: Prof. Dr. Bernhard Renard
Zweitgutachterin: Prof. Dr. Caroline Friedel

Tag der Disputation: 16. Dezember 2019

Abstract

Next-generation sequencing (NGS), in particular Illumina sequencing, is the current state-of-the-art DNA sequencing technology. However, when it comes to time-critical analysis, Illumina sequencing lacks sufficiently short turnaround times due to the sequential paradigm of data acquisition and analysis. For clinical application and infectious disease outbreaks, a significant reduction of time needed from sample arrival to analysis outcome is crucial to optimally treat patients and to prevent further spread of disease. At the same time, nucleotide-level analysis is required to enable (sub-)species level classification and determination of organism-specific properties such as, for example, antimicrobial resistances. To accelerate the generation of NGS analysis results, the real-time read aligner HiLive was developed that performs read alignment while sequencing. Still, HiLive delivers results only at the end of the sequencing process and lacks sufficient resolution and scalability.

In this thesis, a novel real-time alignment algorithm is introduced that was implemented in HiLive2. Unlike its predecessor, HiLive2 provides results at any desired stage of sequencing at full nucleotide-level resolution. The novel approach is based on an FM-index and is more scalable with respect to reference database size and sample size. HiLive2 enables high-quality downstream analysis as shown by performing variant calling based on real-time alignments of human sequencing data. Further, PathoLive is presented, a pipeline for real-time pathogen identification from metagenomic datasets. Based on the output of HiLive2, PathoLive performs a weighted ranking of identified species. Thereby, sequences that typically do not occur in samples from non-infected human individuals are assumed to be of high clinical significance and therefore highlighted in the results. PathoLive also provides an intuitive and interactive visualization that significantly facilitates the interpretation of results. In a case study of a real-world sample from Sudan, PathoLive enables the correct identification of Crimean–Congo hemorrhagic fever virus based on only a few dozen related reads. Besides analytical challenges, samples from human individuals are problematic with respect to data protection as reads from a human host can be used for the identification of the patient. To address this issue, PriLive was developed that enables the irrevocable removal of human sequences from Illumina sequencing data during the ongoing sequencing process. This enables a much higher level of data protection than conventional *post hoc* host removal approaches as the human sequences are at no time available in full length.

Acknowledgements

First and foremost, I would like to thank my supervisor Bernhard Renard for providing the opportunity to work on this exciting project in his research group. Thank you for all the freedoms I had throughout all stages of my projects, but also for giving the best possible support and taking the time for extensive discussions whenever needed.

I wish to thank all my amazing colleagues that participated in my projects, in particular Martin Lindner who implemented the first version of HiLive and Simon Tausch for plenty of profound discussions and excellent collaborative effort. Further, many thanks to Kristina Kirsten and Jakob Schulze who were fantastic students and did great work, but also brought a lot of fun into the projects.

I am particularly grateful for the support of my colleagues at MF 1. Christine Jandrasits, Kathrin Trappe and Martina Fischer for the variety of scientific discussion and hours of 'coffee parties' in the attic office; Thilo Muth and Jakub Bartoszewicz for free-time conversation, common activities, a lot of honest feedback on my projects and proof-reading a plethora of scientific documents; Henning Schiebenhöfer for many fun hours at the soccer table; Vitor Piro for regular exchange of C++ and Bash programming knowledge; Mathias Kuhring for organizing many social activities, especially the board game evenings; many others who remain unmentioned but were always open for scientific and non-scientific discussion and made me always enjoy working in this fantastic group!

Of course, my research was also supported by many collaboration partners. Thanks to Prof. Knut Reinert and the SeqAn team for providing excellent support as well as fantastic workshops and discussions. Advices concerning Illumina sequencing given by MF 2, especially Andrea Thürmer and Aleksandar Radonić, were greatly appreciated. I thank Prof. Erwin Böttinger, Cindy Perscheid and Milena Kraus for making it possible to work with eight highly motivated students on transferring my research to well-engineered software. Further I would like to thank Prof. Caroline Friedel for agreeing to review my thesis.

I would like to thank my family, in particular my parents and brother, for their unconditional support. I am thankful to know that they are always standing by me no matter what happens.

I am truly grateful to Anna for her continuous love, patience and encouragement throughout the years, and for having endured (too) many hours of train travel. Thank you for always being there!

Contents

Terminology and Abbreviations

Terminology

Next-Generation Sequencing

In current discussion, the term next-generation sequencing (NGS) sometimes includes third-generation sequencing (TGS) technologies. In the context of this work, the term NGS describes only short-read sequencing technologies and can therefore be considered as a synonym for second-generation sequencing. In contrast, recent long-read sequencing technologies are referred to as TGS.

DNA Sequencing

Besides DNA, many different types of RNA can also be sequenced with all presented sequencing technologies. This thesis focuses on the sequencing of genomic material that is usually encoded by DNA, with RNA viruses being the only exception. While RNA sequencing usually requires specific protocols for sample preparation, the sequencing process itself is similar to DNA sequencing. Therefore, while referring to DNA sequencing, the methods described in this thesis can generally be adapted for RNA sequencing applications.

Abbreviations

APR	area under the precision-recall curve
AUC	area under the curve
bp	base pair(s)
BSL	biosafety level
BWT	Burrows-Wheeler transform
BWM	Burrows-Wheeler matrix
cDNA	complementary DNA
CPU	central processing unit
DNA	deoxyribonucleic acid
dNTP	deoxyribonucleoside triphosphate, a type of nucleotide
ds-cDNA	double-stranded cDNA
ERV	endogenous retrovirus
FM-index	full-text index in minute space
FPGA	field-programmable gate array
FPR	false positive rate
FTP	file transfer protocol
GB	gigabyte(s)
GHz	gigahertz
h	hour(s)
InDel	insertion or deletion, also summarized as gap
PCR	polymerase chain reaction
PR curve	precision-recall curve

kbp	kilobase pair(s)
MB	megabyte(s)
μL	microliter(s)
μM	micromole(s)
M	million(s)
ng	nanogram(s)
NGS	next-generation sequencing
ONT	Oxford Nanopore Technologies
RAM	random access memory
RNA	ribonucleic acid
ROC	receiver operating characteristic
rpWGS	rapid pulsed WGS
SBS	sequencing by synthesis
SMRT	single-molecule, real-time
SNP	single-nucleotide polymorphism
STR	short tandem repeat
TaxID	taxonomic identifier
tbp	terabase pair(s)
TGS	third-generation sequencing
TPR	true positive rate
WES	whole exome sequencing
WGS	whole genome sequencing
ZMW	zero-mode waveguide

1 Introduction

1.1 DNA Sequencing

The genetic information, specifying the structure and function of an organism, is conserved as deoxyribonucleic acid (DNA) which is composed of two connected strands of chained nucleotides that form a DNA double helix [1]. DNA contains four types of nucleotides that differ in their underlying nucleic bases: adenine (A), cytosine (C), guanine (G) and thymine (T). The DNA double helix is connected at each position by forming hydrogen bonds of two complementary nucleotides, i.e., A with T and C with G. Those pairs of bound nucleotides are called base pairs (bp). While DNA is the most frequent form of genomic nucleic acid, many viruses encode their genome using ribonucleic acid (RNA), a different type of nucleic acid that contains the nucleic base uracil (U) instead of thymine [2, 3]. RNA can generally occur in single- or double-stranded form and, besides encoding viral genomes, also plays an essential role in various cellular processes of all organisms.

The order of nucleotides in the genome of an organism serves as "a blueprint to direct a host of processes for building an embodied organism" [4]. Thus, the phenotype of an organism, defined by the set of observable characteristics, is generally based on its genetic information. However, many traits can be influenced by the environment such that a direct correlation of changes in the genome to a specific phenotype can be hard to observe. As genetic information is inheritable, closely related organisms share large proportions of their genetic code. This leads, by implication, to the general assumption that organisms with high similarities in the genomic material are more closely related than organisms with very distinct genetic information. Many different types of genetic analysis, for example taxonomic classification, genetic genealogy or the reconstruction of disease transmission networks, are based on this assumption.

DNA sequencing describes methods to analyze the order of nucleotides in a DNA sample. After the first attempts were made in the late 1960s and early 1970s [5, 6], two approaches that allowed the sequencing of hundreds of base pairs within a single day were published in 1977 [7, 8]. Unlike chemical sequencing of Maxam and Gilbert that has lost relevance over time, Sanger sequencing has been continuously improved by using fluorometric based detection [9–12] and the detection through capillary based electrophoresis [13, 14]. Based on these improvements, it was finally possible to use Sanger sequencing for the first sequencing

of the human genome that was published in 2001 [15, 16]. Until today, Sanger sequencing is widely used for many applications, especially for low-throughput targeted sequencing and as a gold standard for the confirmation of results obtained from other technologies [17–19]. However, new generations of sequencing evolved in the last two decades that provide much higher throughput at lower costs and time consumption and therefore enable a plethora of new sequencing-based applications.

1.1.1 Next-Generation Sequencing

From 2005 on, new sequencing devices became available that were based on substantially different technologies than the previously used Sanger sequencing. These new methods enabled the production of much higher amounts of sequencing data at lower cost per kilobase pair (kbp) [20]. This group of sequencing technologies is referred to as next-generation sequencing (NGS). The first commercially successful NGS sequencing device was the Roche 454 pyrosequencer that used the pyrosequencing method [21] and a bead emulsion amplification strategy [22]. Over several years, different competing technologies where brought to market including the Solexa Genome Analyzer (2006), ABI SOLiD system (2007), Ion Torrent Ion semiconductor system (2010) and the BGISEQ-500 (2016). Although each of those systems comes along with unique advantages and disadvantages, the currently dominating NGS technology is Illumina dye sequencing (or Illumina sequencing), which is similar to the technology used in the Solexa Genome Analyzer (Solexa was purchased by Illumina in 2007). With a market share in the field of DNA sequencing devices and services of around 90 % in 2018 [23], Illumina clearly provides the current state-of-the-art sequencing technology. Therefore, the proposed work focuses on this technology which is subsequently introduced in more detail. For extensive reviews on the history of DNA sequencing and NGS technologies, I refer to [17, 24–26].

Illumina Sequencing

Illumina sequencing is a realization of the sequencing by synthesis (SBS) approach. In this method, the DNA sequence is usually extended by a DNA polymerase or ligase, binding one deoxyribonucleoside triphosphate (dNTP), a type of nucleotide, per round (referred to as sequencing cycle). Whenever a nucleotide is bound, a measurable signal is produced which is specific for the respective nucleotide. In the case of original Illumina four-channel technology, this signal is produced by a specific fluorescent dye for each of the four nu-

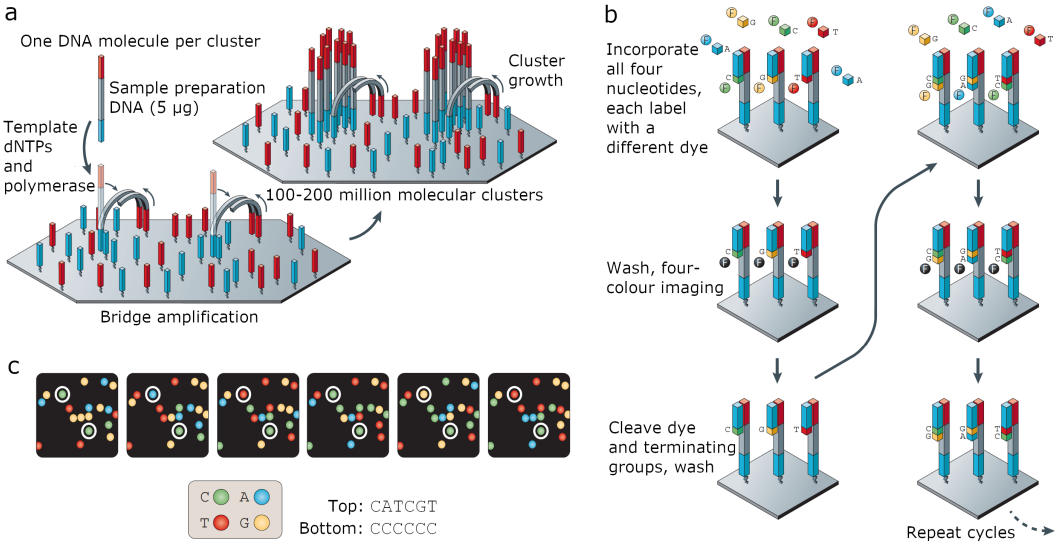


Figure 1.1: Overview of the Illumina sequencing technology. a Cluster generation: The DNA is cut into small fragments, bound to the flow cell and iteratively amplified via bridge amplification. This procedure leads to exponential growth of the cluster. **b** Sequencing by synthesis: Nucleotides labeled with different fluorescent dyes and a terminating group are used to extend the fragments of each cluster. After binding to the fragment, the nucleotide can be identified by the corresponding fluorescent signal. After the signal is recorded, the dyes and the terminating group are cleaved. This iterative procedure is performed a fixed number of times (sequencing cycles). **c** Base calling: The fluorescent signal of each cluster is recorded for each sequencing cycle. The order of signals corresponds to the nucleotide sequence of the fragments in one cluster. Figure adapted from [17], permission granted.[1]

cleotides (Figure 1.1). However, new devices use the one-channel approach (iSeq 100) or the two-channel approach (MiniSeq, NextSeq and NovaSeq Series), requiring only one or two dyes, respectively. For the one-channel approach, the single dye is bound to two nucleotides for the first step (A,T). In a second step, the dye is cleaved from A and added to C, making all nucleotides distinguishable when images from both steps are available [27]. For the two-channel approach, a mix of two dyes (represented in red and green color) is used of that none (G), one (C,T) or both (A) are bound to a specific nucleotide. When overlaying the images from both channels, the four different nucleotides can be distinguished [28]. Regardless of the differences in chemistry, the general sequencing approach is similar for all Illumina machines and can be divided in four general steps: Library preparation, cluster generation, SBS and base calling.

[1] Reprinted by permission from Springer Nature Customer Service Centre GmbH: Springer Nature, Nature Reviews Genetics [Sequencing technologies - the next generation, Metzker ML, ©2009 (2010)]

Library Preparation

The library preparation describes the preparation of one or more DNA samples for Illumina sequencing. Here, only the most important steps of library preparation that are relevant for this work are described. For a more extensive insight into the library preparation process, I refer to Bronner and Quail [29]. As a first step, the DNA of a single sample is quantified and fragmented into small DNA molecules, typically of size 200 - 500 bp for read lengths of 75 - 250 bp. These fragments are tagged with sequencing adapters and purified. Consequently, amplification via polymerase chain reaction (PCR) can be performed to enrich properly ligated template strands, increase the total amount of library and add oligonucleotide sequences to allow hybridization to the flow cell surface. However, alternative protocols that omit the PCR step exist to prevent PCR-induced sequencing bias. For both types of protocols (PCR and non-PCR), multiplex indices can be integrated in the sequencing adapters to enable identification of samples that are sequenced in the same sequencing run. Before sequencing, the library is purified and its quality and quantity are assessed. Especially the quantification step is crucial to prevent the generation of too many or few clusters. For multiplexed sequencing, several indexed libraries are pooled. The library pools can be stored or directly used for cluster generation [29, 30].

Cluster Generation

To obtain a signal of sufficient strength, each DNA molecule in the pool library is amplified. To enable a proper distinction of different fragments, the amplification takes place in a limited area of the flow cell that is separated from amplification products of other fragments. The resulting areas containing copies of the same DNA molecule, called clusters, are created via bridge amplification in Illumina sequencing (Figure 1.1 a). Therefore, a well-defined concentration of DNA from the pool library is bound to the flow cell. With both adapters being bound, a bridge is created and the fragment is amplified by a DNA polymerase from one end of the bridge to the other. When finished, the strands are separated and one adapter is unbound from the flow cell. In doing so, the number of identical copies of the initial fragment is doubled in each step such that the cluster growths exponentially until it has sufficient size to produce an interpretable signal [17, 31].

Sequencing by Synthesis

In the SBS step, all fragments of all clusters are amplified in parallel. Thereby, in four-channel technology, four nucleotides each marked with a different fluorescent dye are given to the flow cell. One of those four nucleotides binds to each fragment, producing a fluorescent signal. A reversible terminator group that is bound to the nucleotides prevents that more than one nucleotide can bind to each fragment. Unless amplification errors occurred, all fragments of one cluster are amplified with the same nucleotide to enhance the fluorescent signal. After analyzing the signals of one sequencing cycle, the dyes and terminating groups are cleaved. This iterative procedure is repeated a fixed number of times, subsequently being referred to as sequencing cycles (Figure 1.1 b) [17, 31]. The total number of sequencing cycles corresponds to the final read length. In paired-end sequencing, each fragment flips over after sequencing the first direction. This is done by forming a bridge and unbinding the adapter that was bound to the flow cell for sequencing the first read. In doing so, the fragment can be sequenced from both ends, enabling better results in subsequent analysis steps. For multiplex sequencing, the indices are sequenced in separate reactions at both ends of the fragment.

Base Calling

Base calling is often not regarded as a separate step in Illumina sequencing but rather described as a part of SBS, probably because it is performed for each sequencing cycle. However, as the transforming step from raw signals to human-readable and analyzable sequence information, it is a crucial part of the workflow and can strongly influence the sequencing and analysis results. This fact recently regained more attention in the community due to the high differences in the performance of base calling software for third-generation sequencing approaches (see Section 1.1.2). In Illumina sequencing, base calling is performed by a specialized software that analyzes the fluorescence images made by the sequencer. The first five sequencing cycles are used to distinct the clusters from each other. Once the cluster locations are identified, the fluorescent signals are analyzed to identify the nucleotide that was appended to the fragments of a cluster (Figure 1.1 c) [17, 31]. Thereby, the signals of neighboring clusters and of previous and subsequent sequencing cycles are considered in the calculation of the correct base call as they can influence the signal of the analyzed cluster. Based on these factors, the base calling software specifies the identified nucleotide including a quality score coding for the probability that the base call is correct. The base call quality is

provided as Phred score, a log-transformed quality value q defined as $q = -10 \cdot log_{10}(p)$, where p is the estimated error probability for the base call [32].

1.1.2 Third-Generation Sequencing

When compared to NGS, third-generation sequencing (TGS) comes with two major differences. First, TGS technologies perform sequencing on single DNA molecules rather than amplified clusters. Second, they provide much longer reads in the range of up to several kilobase pairs compared to a maximum of 2 x 300 bp for Illumina sequencing. There are currently two notable TGS technologies on the market. The first commercial product was released by Pacific Biosciences in 2011 using the single-molecule, real-time (SMRT) sequencing technology. This technology, like Illumina sequencing, follows the SBS approach. The sequencing library for SMRT sequencing consists of circular fragments of length 10 - 15 kbp and is created by the ligation of hairpin adapters to both ends of the DNA molecules. After binding sequencing primers and a DNA polymerase to the adapter, the library is loaded onto a flow cell that contains so-called zero-mode waveguides (ZMWs), each being occupied by one DNA polymerase and a single template DNA molecule. Similar to Illumina sequencing, the DNA polymerase replicates the fragment with fluorescently labeled nucleotides. The fluorescent signal produced during replication is enhanced by the ZMW while other signals, such as from neighboring reactions or labeled nucleotides in the solution, are repressed. The signal is measured by a camera system on the bottom of the flow cell. As modifications of the template DNA, such as methylation, decelerate the sequencing process, they can be identified through longer interruptions between two signals, the so-called longer inter-pulse duration (Figure 1.2 a-c) [33, 34].

In 2015, the portable nanopore sequencing device was released by ONT. This sequencer, not much larger than a USB stick, produces reads of arbitrary length, in general only limited by the length of the template DNA. The technology behind ONT sequencing is very different to the SBS approach of NGS and SMRT sequencing. For a standard 1D library, adapters with DNA protein complexes are bound to both ends of the DNA fragment. After binding to a single nanopore, the motor protein ensures the stepwise movement of a single strand of the DNA template through the nanopore. The different nucleotides of the template DNA lead to measurable changes in the voltage potential that is applied across the membrane. Thereby, the signal of modified nucleotides also differs when compared to unmodified nucleotides, meaning that modifications can implicitely be identified (Figure 1.2 d-f) [33, 34].

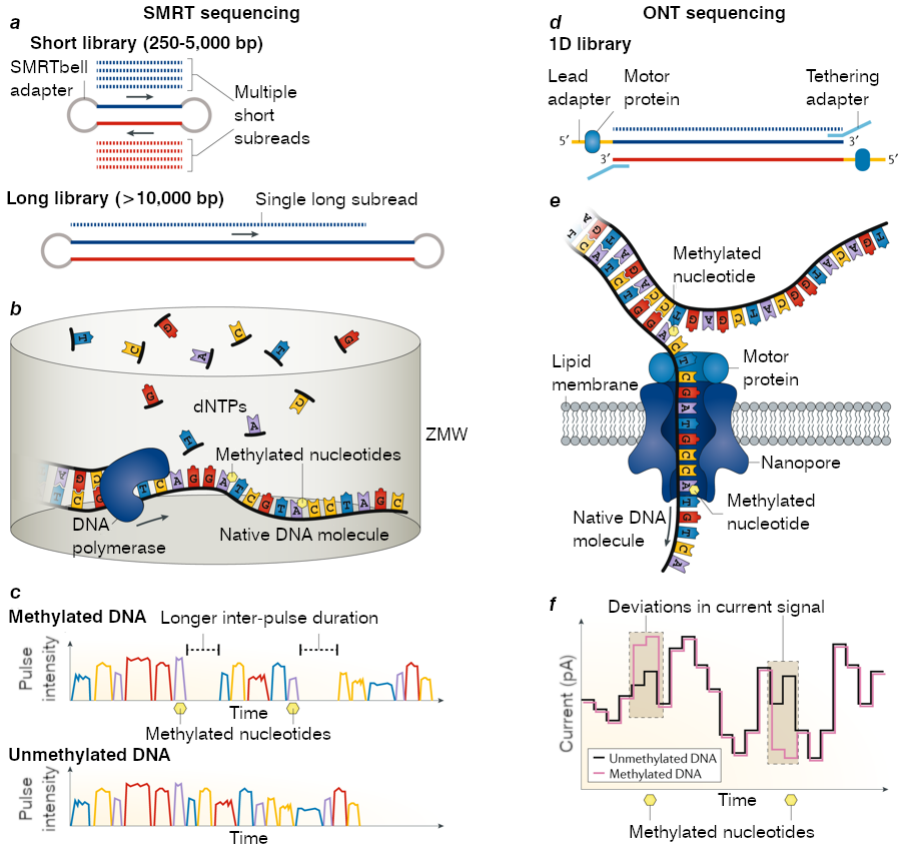


Figure 1.2: Overview of Pacific Biosciences SMRT sequencing (a-c) and Oxford Nanopore Technologies (ONT) sequencing (d-f). a-c A SMRT sequencing library containing double-stranded fragments of length 10-15 kbp is created by ligating hairpin adapters. For shorter fragments, one template may contain multiple subreads. The template DNA is sequenced using a DNA polymerase to replicate the template with fluorescently labeled dNTPs. The fluorescent signal of the incorporated dNTP is enhanced by a ZMW and measured by a laser and a camera system on the bottom of the flow cell. DNA modifications can be identified due to longer inter-pulse duration. **d-f** An ONT sequencing library contains DNA fragments of arbitrary length bound with adapters on both ends. The motor protein channels the fragment through the nanopore, producing measurable changes in the voltage potential of the membrane depending on the nucleotide sequence and potential modifications. Figure adapted from [34], permission granted.[2]

The translation of the voltage signal to human-readable and analyzable sequence outcome is not trivial and base calling approaches are still rapidly evolving. Therefore, and due to different approaches and training data, the performance of base callers and even different

[2]Reprinted by permission from Springer Nature Customer Service Centre GmbH: Springer Nature, Nature Reviews Genetics [Deciphering bacterial epigenomes using modern sequencing technologies, Beaulaurier J *et al.*, ©2018 (2018)]

versions of the same base caller differs profoundly in terms of base call accuracy, speed and detection of modifications [35].

Besides the plethora of new possibilities that TGS introduced with its highly increased read lengths, the ability to identify modifications and the implicit potential for real-time analyses, it also comes with several challenges. The biggest drawback of both technologies is the comparably high error rate of currently $\sim$ 11 - 15 % for SMRT sequencing with single long subreads [33, 36–38] and $\sim$ 5 - 20 % for ONT sequencing, depending on the type of molecules and library preparation methods [38–42]. However, in SMRT sequencing, smaller subreads can be used to create a circular consensus sequence achieving accuracies of > 99 %, but significantly reducing the maximum read length (Figure 1.2 a) [33, 43–45]. For ONT sequencing, besides ongoing improvements in chemistry for standard 1D sequencing, an increase of accuracy up to 97 % is reported by performing 2D (no longer available) or 1D squared ($1D^2$) sequencing that enables the sequencing of both strands of each template to obtain a consensus sequence [33, 46]. However, this method is reported to significantly reduce throughput and read length and was therefore summarized as "not very efficient" by van Dijk et al. [33]. The second general obstacle of TGS is scalability. The first devices of both technologies (Pacific Biosciences Sequel and Oxford Nanopore MinION) allowed massively less throughput compared to modern NGS devices. Today, both companies are working on high-throughput solutions that aim to be competitive to the throughput of Illumina sequencing technology. For example, Oxford Nanopore's PromethION P48 is specified to produce up to 7.6 tbp and up to 15 tbp being announced with future improvements [47]. While those numbers are at least comparable to Illumina's production-scale sequencer NovaSeq 6000, they are so far exclusively reported by the manufacturer and have to be proven and established by third party research laboratories in future.

1.2 Short Read Alignment

Before giving a formal introduction to short read alignment, a short overview of different terms for specific alignment problems is provided. Please note that the definitions given in this section refer to pairwise alignments, though many of them can easily be adapted for multiple alignments with more than two queries.

1.2.1 General Terminology

Read alignment, also called read mapping, is a concept to describe the transformation of a pattern P to a text T by single-character operations. Thereby, alignment algorithms aim to determine such a transformation and generally allow for differences in the compared regions of P and T, making it an approximate string matching problem. An alignment is called optimal with respect to a given scoring function (or string metric, see Section 1.2.2) if there is no other alignment with a higher score under the same function. Otherwise it is called suboptimal. Please note that there can be multiple optimal alignments for the same P and T. Alignments of a pattern P and a text T can be further grouped in three major categories: local alignments, global alignments and semi-global (also known as glocal) alignments. A global alignment is a so-called end-to-end alignment of complete P and complete T. In the field of computational biology, global alignments are usually used when two sequences are expected to be of high similarity over their full length, for example when searching for mutations in closely related genes. Local alignment algorithms aim at finding (optimal) alignments of a substring of P and a substring of T. This means, a local alignment can disregard both ends of P and T. The most prominent implementation of a local alignment algorithm is the Basic Local Alignment Search Tool (BLAST) [48]. Semi-global alignments aim at aligning the complete pattern P to a substring of T. This means that both ends of T can be disregarded while complete P must be aligned. Examples for these alignment categories are given in Figure 1.3. These examples also show that the choice of an appropriate alignment category and scoring function for a certain problem is crucial as it can substantially influence not only the alignment itself, but also the mapping position.

Two additional terms got commonly used in the last years, pseudoalignment and lightweight alignment. Thereby, a pseudoalignment specifies from which, but not where in a sequence a read originated. This concept is, for example, used for RNA sequencing quantification with kallisto [49]. In contrast, lightweight alignments provide partial alignment information including the mapping position and are used for quantification of transcript expression with Salmon [50]. However, the focus of this work are alignment-based approaches providing mapping positions and full nucleotide-level alignment resolution.

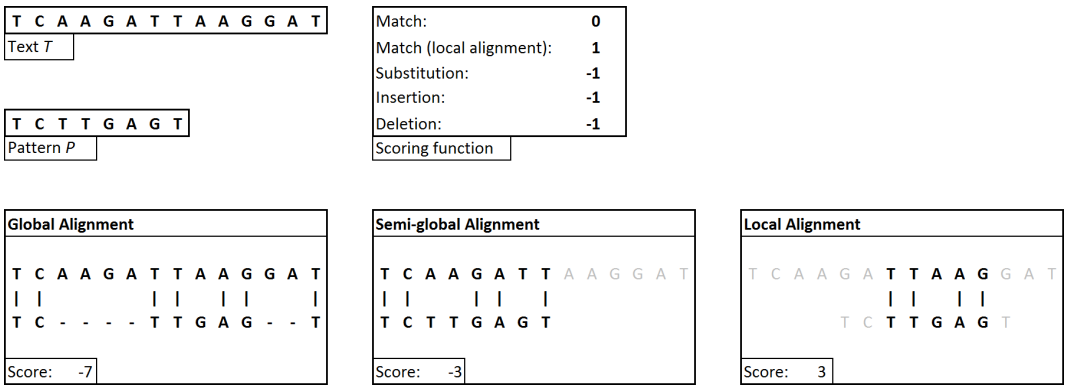


Figure 1.3: Examples for different alignment categories. The pattern P is aligned to text T under the given scoring function. Except for rewarding matches in local alignments with a score of 1, the scoring function is identical for all alignment categories. For global and semi-global alignments, the scoring function represents the negative Levenshtein distance (e.g., a score of -3 implies a Levenshtein distance of 3 between the aligned regions of P and T). One optimal alignment is shown for each alignment category. Disregarded regions for the alignments are shown in light gray. Matches between P and T are connected by vertical lines.

1.2.2 String Metrics

Short read alignment is a specialization of the approximate string matching problem that describes the retrieval of all occurrences of an input string (or pattern) P of length $|P| = n$ in a reference string (or text) T of length $|T| = m$ with a maximum distance d. Thus, short read alignment falls into the category of semi-global alignments. Both strings P and T contain characters of a common alphabet Σ. Basic single-character operations include substitutions, insertions and deletions of a character [51]. In general, arbitrary string metrics can be used to define the validity of an alignment. One of the most popular string metrics in computational biology is the Levenshtein distance that defines a cost of 1 for each operation [52]. Thus, an alignment with a Levenshtein distance of k corresponds to an alignment with k single-character operations, also called an alignment with k errors. The Levenshtein distance is often referred to as edit distance although this term originally describes a more general concept for any type of operations [53]. A second common string metric in the field of computational biology makes use of affine gap penalties. Thereby, the cost of an insertion or deletion (InDel) of length $l \geq 1$ is defined as $a + b \cdot l$, where a is the gap opening penalty and b is the gap extension penalty. Affine gap penalties result in better alignment scores for contiguous InDels when compared to non-contiguous InDels of the same total length, which often makes sense in the biological context.

In computational biology, the term short read alignment describes the alignment of NGS reads to a database of known reference genomes. For DNA, the alphabet usually consists of the four basic nucleotides and sometimes includes the ambiguous zero-quality base call N (i.e., $\Sigma = \{A, C, G, T, N\}$). The text T corresponds to a reference sequence or a database of reference sequences and P is the sequence of a read. The length $|P| = n$ is called read length.

1.2.3 Algorithms and Data Structures

A first solution of the approximate string matching problem in computational biology was the Needleman-Wunsch algorithm published in 1970 [54]. It uses a dynamic programming approach and is still widely used, especially when optimal results are required. However, the runtime complexity of $\mathcal{O}(m \cdot n)$ is too high to scale up for high-throughput applications such as NGS. After several optimizations and modifications, a breakthrough was the BLAST algorithm published in 1990, a heuristic seed-and-extend approach being an order of magnitude faster than previous approaches of similar sensitivity [48]. The seed-and-extend approach consists of two general parts. Seeding describes the first step, searching for alignments of short substrings of length q (called q-gram or k-mer) between the pattern P and the text T. In the following extension step, the alignments obtained from seeding (seeds) are extended in both directions. While the seeding step is often exact (i.e., having an alignment distance of 0), errors are usually considered during extension. Similar seed-and-extend strategies are still used today by many state-of-the-art short read alignment algorithms.

However, while BLAST is still widely used for many applications, the arrival of high-throughput NGS technologies led to the development of novel alignment approaches that are specialized to align millions of short reads several orders of magnitudes faster. While different approaches differ in algorithmic details, many of the most popular tools are based on similar general approaches and data structures that are introduced in the following paragraphs.

q-gram Index and Hashing Methods

A q-gram index is a data structure to store substrings of fixed length q from a given sequence. The most simple form to create such an index is the transformation of each q-gram of the reference genome to a representative number. When using a simple lookup table, this number corresponds to a specific entry that stores all occurrences of the respective q-gram of a reference sequence. In the seeding step, a read of length n can be segmented into $n - q + 1$

overlapping q-grams whose positions can efficiently be retrieved from the index. While q-gram indices are very efficient in build time and data retrieval, the required memory for a simple lookup table implementation of this approach is relatively high ($\mathcal{O}(|\Sigma|^q + m)$, m being the length of the reference genome). For large q (usually $q > 16$ for DNA) it is therefore necessary to apply hashing techniques to reduce memory consumption at the cost of retrieval time efficiency [55]. Other modifications were made to improve q-gram indices, e.g., using so-called spaced seeds that disregard specified positions in the q-gram to allow for non-consecutive matches. Spaced seeds were shown to significantly increase alignment sensitivity [56–58]. Many popular short read alignment tools are based on q-gram or hash table indices of the reference genome [59–69], others rely on indexing of the reads [48, 70–75] or double indexing (building indices of the reads and the reference database) [76].

FM-index

A very runtime-efficient index for the seeding step is the suffix tree that was introduced in 1973 [77] as it can find all occurrences of a pattern in optimal time while being built in linear time [55]. However, suffix trees come with a large memory footprint and can therefore hardly be used for large reference genomes. The suffix array [78] and enhanced suffix array [79] improved the memory footprint at the cost of runtime efficiency, but although the latter is used for short read alignment [80] the memory footprint is still not sufficiently small for the growing size of reference databases. Therefore, many modern short read aligners use the full-text index in minute space (FM-index) [81] that provides results in linear time $\mathcal{O}(n)$, n being the read length, with a much smaller memory footprint [82–90]. The FM-index is based on the Burrows-Wheeler transform (BWT) [91], a method that was originally developed for lossless data compression. In addition to the BWT, auxiliary tables are required to enable the traversal through the index (C and Occ) and the acquirement of the occurrence positions of the query in the reference (Pos). The construction of and traversal through the index is shown in Figure 1.4. The most prominent short read aligners using FM-index implementations are Bowtie 2 [85], BWA [89] and HISAT2 [90]. While Bowtie 2 and BWA use a single global FM-index, HISAT2 makes use of additional overlapping local FM-indices. Thereby, a single local FM-index in HISAT2 is small enough to fit into the CPU's cache memory, "which is substantially faster than standard RAM" [90] and thereby improves the overall alignment performance.

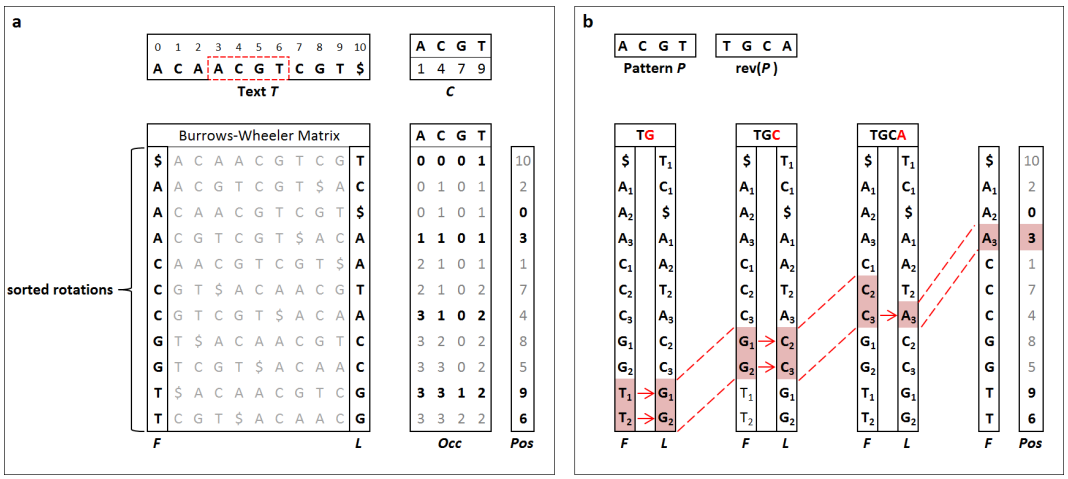


Figure 1.4: FM-index data structure and query search. **a** Example for the construction of an FM-index. The \$ character is appended at the end of text T. All possible rotations of the string are computed and lexicographically sorted, resulting in the Burrows-Wheeler matrix (BWM). Column L of the BWM is called Burrows-Wheeler transform (BWT). For the FM-index, F and L plus some additional tables are required. The prefix-sum table C stores the number of occurrences of lexicographically smaller letters in T for each character. The occurrence table Occ stores the occurrences of each character in a prefix of L, corresponding to the rank of the respective character in a given row. Occ and C can be used to find the related row in F for an entry of L. The position table Pos specifies the corresponding position of an entry in F in input string T. For memory efficiency, only every m-th row of Occ and n-th position of Pos are usually stored ($m = n = 3$ in this example, missing values being shown in light gray). Missing values can be efficiently calculated during runtime. **b** The reverse pattern $rev(P)$ is used to find all occurrences of P in T. For the sake of clarity, ranks are included as indices of all entries in F and L. For each entry in the current span of F, all entries in L that match the next character of $rev(P)$ are considered for the next iteration. Once the complete query was found, occurring positions can be calculated using the Pos vector. If no entry in L for the current span matches the next character, the query does not occur in the input string (not shown). The occurrence of the P in T is highlighted in subfigure a.

1.3 Thesis Outline

In this thesis, new algorithmic approaches for alignment-based real-time analysis of Illumina sequencing data are presented. The main goal of the work is to improve the previously existing method of real-time alignment implemented in HiLive [92]. Thereby, scalability of the approach is significantly increased to adapt it for the use with larger reference genomes and datasets. This is achieved by algorithmic developments and technical optimizations. A more efficient alignment algorithm is used as a basis for novel approaches to open up new areas of application for real-time analysis, including variant calling, data protection and pathogen detection. Throughout the projects, I was advised by Bernhard Renard who participated in the conceptional design of the projects and drafting of the manuscripts for publication.

In Chapter 2, I introduce HiLive2 which is the successor of HiLive. One major change I made for HiLive2 was the integration of a newly developed algorithm based on the FM-index while its predecessor made use of the q-gram index. This improvement made HiLive2 applicable for large reference genomes as shown for the human reference hg19. Additionally, the need of computational resources including CPU time and memory consumption was considerably reduced which allows the application for larger datasets. Further developments included the ability to produce intermediate results during runtime and improved functions to calculate the alignment score (affine gap costs) and mapping quality to improve follow-up analyses. Chapter 2 also shows a proof-of-principle to use real-time results of HiLive2 for variant calling with human whole exome sequencing data. In this contribution I conceptualized the FM-index based algorithm and did all related implementations. Simon Tausch extensively tested the functionality and performance of HiLive2 on various types of datasets and gave detailed feedback about misbehavior and usability. Simon Tausch was also involved in design decisions such as supported file formats, the choice of default parameters as well as the evaluation and prioritization of new features. Together with Bernhard Renard, I designed the proof-of-principle study for real-time alignment-based variant calling and I preformed the corresponding experiments. I drafted the manuscript with helpful input of the co-authors.

> **Loka TP**, Tausch SH, Renard BY. *Reliable variant calling during runtime of Illumina sequencing*. bioRxiv, 2018, doi: 10.1101/387662.
> Accepted for publication in Nature Scientific Reports.

In Chapter 3, I introduce PriLive which adds a local background alignment approach to the HiLive software. This functionality is primarily designed to remove human reads from

Illumina sequencing data to significantly increase the level of data protection when analyzing microbial samples from a human host. This is achieved by masking relevant sequence information in the raw base call data of the sequencing device even before sequencing has finished. The experiments on simulated and real data show the effectiveness of our approach with respect to data protection, even in early stages of sequencing. Based on the general idea of real-time filtering which was introduced by Bernhard Renard, I designed and implemented the local alignment approach via integration in HiLive. Simon Tausch was involved in the elaboration of algorithmic details and the technical strategy regarding the real-time modification of Illumina base call files. I preformed the computational experiments which were designed together with Bernhard Renard and Simon Tausch. Piotr Wojciech Dabrowski supported technical and infrastructural decisions with respect to the implementation of PriLive and reproducing the real-time experiment with dataset $HiSeq\_1$. Andreas Nitsche and Aleksandar Radonić designed the sequencing experiment of dataset $HiSeq\_1$ which was performed by Aleksandar Radonić. Andreas Nitsche further provided virological insights, especially concerning the expected proportions of viral and human reads. I wrote the manuscript with valuable contributions of all co-authors.

> **Loka TP**, Tausch SH, Dabrowski PW, Radonić A, Nitsche A, Renard BY. *PriLive: privacy-preserving real-time filtering for next-generation sequencing*. Bioinformatics, 2018, bty128, doi: 10.1093/bioinformatics/bty128

In Chapter 4, I show how the real-time alignment results of HiLive2 can be used for the detection of pathogens in metagenomic samples. The presented software PathoLive integrates HiLive2 and ranks the alignment results with respect to their occurrences in a database of known background signals in human samples. PathoLive further provides an intuitive visualization that highlights uncommon (and therefore interesting) signals of organisms with a high biosafety level which is usually assumed to correlate with their expected risk for humans. This visualization is provided for intermediate and final results and allows first interpretations even in early stages of sequencing. The concept of PathoLive was designed by Simon Tausch, Bernhard Renard and Andreas Nitsche. Andreas Nitsche supervised the in-house sequencing experiments and Jeanette Klenner produced the spiked dataset for benchmarking. I gave vast conceptional input concerning real-time applicability for the first implementation of PathoLive. Piotr Wojciech Dabrowski, Martin Lindner and Andreas Andrusch gave substantial input on algorithmics, parametrization, visualization, and scoring methods. Simon Tausch did the related implementations. I adapted PathoLive to run with HiLive2. Jakob Schulze and I adapted both versions of HiLive to fulfill specific requirements

of PathoLive and for the experiments shown in the manuscript. These changes included new major features such as, for example, the adapter trimming functionality which was required for the real-world sample from Sudan. The real-time experiments were performed by Simon Tausch, while I did all experiments with HiLive2 including the analysis of the sample from Sudan. The manuscript was drafted jointly by Simon Tausch and me while receiving valuable input from all co-authors. I particularly emphasize that the first version of the manuscript was also part of the doctoral thesis of Simon Tausch [93] in which I am not mentioned as a joint first author as many of my contributions were made in a later stage of the project.

> Tausch SH [*], **Loka TP** [*], Schulze JM, Andrusch A, Klenner J, Dabrowski PW, Lindner MS, Nitsche A, Renard BY. *PathoLive - Real time pathogen identification from metagenomic Illumina datasets*. bioRxiv, 2018, doi: 10.1101/402370. Submission in preparation.
>
> [*] These authors contributed equally to this work.

Chapter 5 gives a summary of the thesis and provides an outlook for potential future developments in the field of real-time analysis of sequencing data.

2 Reliable Variant Calling during Runtime of Illumina Sequencing

The sequential paradigm of data acquisition and analysis in NGS leads to high turnaround times for the generation of interpretable results. We combined a novel real-time read mapping algorithm with fast variant calling to obtain reliable variant calls still during the sequencing process. Thereby, our new algorithm allows for accurate read mapping results for intermediate cycles and supports large reference genomes such as the complete human reference. This enables the combination of real-time read mapping results with complex follow-up analysis. In this study, we showed the accuracy and scalability of our approach by applying real-time read mapping and variant calling to seven publicly available human whole exome sequencing (WES) datasets. Thereby, up to 89 % of all detected single-nucleotide polymorphisms (SNPs) were already identified after 40 sequencing cycles while showing similar precision as at the end of sequencing. Final results showed similar accuracy to those of conventional *post hoc* analysis methods. When compared to standard routines, our live approach enables considerably faster interventions in clinical applications and infectious disease outbreaks. Besides variant calling, our approach can be adapted for a plethora of other mapping-based analyses.

2.1 Background

Common workflows for the analysis of Illumina NGS data can only be applied after sequencing has finished. Besides the time needed for sample preparation, this sequential paradigm of data acquisition and analysis is one of the main bottlenecks leading to high turnaround times. For time-critical applications, it is crucial to massively reduce the time span from sample receipt to interpretable analysis results. Examples for such time-critical analyses range from the differential diagnosis of genetic disorders in infants [94–97], to the determination of *Mycobacterium tuberculosis* drug resistances [98], and to the identification of pathogens, virulence factors, drug resistances and paths of disease transmission in infectious disease outbreaks [99, 100]. While having considerably higher turnaround times than targeted approaches such as molecular tests, NGS provides a more open view as well as more extensive and reliable results. During bioinformatics analysis of NGS data, read mapping and variant calling are crucial steps to obtain genetic information that is essential for the treatment of a

patient, including strain level classification and drug resistances of a pathogen or the presence of genetic disorders that are known to be associated with specific disease characteristics.

While HiLive [92], the predecessor of our new algorithm HiLive2, delivered results at the end of sequencing, HiLive2 can produce read mapping output for arbitrary sequencing cycles while still sequencing. At the same time, the new algorithm is faster, more accurate and enables scalability to large reference genomes such as the complete human reference. The recently published software LiveKraken [101] already gives k-mer based taxonomic classification results for arbitrary sequencing cycles. However, while LiveKraken provides valuable information about the microbial composition of a sample, the results do not allow for complex reference-based follow-up analyses such as variant calling or the analysis of drug resistances. Alternative approaches to obtain read mapping results for Illumina data while still sequencing, such as rapid pulsed whole genome sequencing [97], lack sufficient scalability for high amounts of data and large reference genomes and are therefore only suitable for special use cases. At the same time, the incremental approach of HiLive2 provides higher flexibility in the choice of output cycles which can even be modified during the runtime of the sequencer. The use of specialized hardware, such as field-programmable gate arrays (FPGAs) that are for example used in the DRAGEN system [96] could generally overcome the lack of scalability and speed for intermediate analyses but come with additional costs, either for purchase and infrastructure of local solutions or for the use of a cloud system. At the same time, such approaches are usually not algorithmically optimized for analyzing incomplete data. Additionally, cloud solutions as provided by DRAGEN can be problematic with regard to data protection guidelines in many countries.

A different tool, TotalReCaller [102], implements a similar algorithmic idea as HiLive2. TotalReCaller uses an FM-index based alignment approach to perform reference-based base calling. While TotalReCaller's alignment approach only allowed for substitutions, its successor Gappy TotalReCaller [103] also considers insertions and deletions. However, as TotalReCaller's main objective is to improve base calling it does not produce actual alignment output. At the same time, the authors describe the need for an FPGA-implementation to make it suitable for real-time applications.

When compared to ONT sequencing, which enables real-time analysis by design, Illumina sequencing provides higher scalability at lower costs and with much lower error rates. Therefore, while being a promising technology for real-time analysis in the future, sequencing technology, protocols and computational analysis for ONT need to be further established and improved to become a viable alternative for many scenarios.

The workflow described in this study is based on real-time read mapping results with our novel algorithm HiLive2 followed by fast and accurate variant calling with xAtlas [104]. Thereby, live results can be obtained several hours before all data are written by the sequencer and provide increasing insights into the sample over sequencing time. This study describes the application of our workflow to human WES data, showing the scalability of our approach for high amounts of complex data and large reference genomes. However, our approach can generally be adapted for different types of sequencing methods such as whole genome sequencing (WGS) or amplicon sequencing and a plethora of different mapping-based analysis methods. While application to arbitrary sample types and reference genomes is possible in general, the latency of real-time results strongly depends on the size and complexity of the used reference genome, the number of reads per tile, the parameter settings made and the available hardware. Therefore, when planning real-time sequencing, it should be examined if the desired live analysis of a sample is possible under the given conditions. Our new real-time read mapping software HiLive2 is publicly available under BSD-3-clause on *https://gitlab.com/rki_bioinformatics/hilive2* and on Bioconda [105] for easy installation (conda install –c bioconda hilive2).

2.2 Methods

In this chapter, we provide a brief description of our workflow. All software versions are listed in the Software versions section.

Implementation of HiLive2

HiLive2 is based on a novel algorithm based on the efficient FM-index implementation of the SeqAn library [106]. HiLive2 provides five different modes to select a focus on runtime (fast or very-fast), accuracy (accurate or very-accurate) or a trade-off of both (balanced). Parameter decisions are made automatically by HiLive2 based on the selected mode, the size of the reference genome and the read length. The influenced parameters include the length of an initial, error-free k-mer (anchor-length), the intervals of creating new seeds (seeding-interval), the minimum alignment score (min-as) and the intervals of increasing the number of errors (error-interval). However, all parameters can be refined or completely manually set by the user to enable even faster or more accurate computations.

HiLive2 follows the seed-and-extend alignment approach. The first seeds are created when

sufficient bases are available for all reads to create an anchor of the specified length (initial seeding). Thereby, anchors are error-free matches of the most recently sequenced k-mer in the index. New alignment anchors of the same length are created in specified intervals (non-initial seeding). By default, the selected intervals produce overlapping anchors. In the following cycles after the creation of an anchor, the alignments are extended in the direction of sequencing only. This means that alignments originating from non-initial seeding have unaligned regions at the beginning of the reads which are reported as soft clips. During extension, the minimum score of the alignment decreases with ongoing sequencing. This means that more errors are permitted for longer sequences. This approach leads to a massive reduction of the search space in early sequencing cycles while having only minor effects on the final results as the mapping positions of reads with too many errors at the beginning can still be determined based on a non-initial seeding step. However, the actual alignment information for the erroneous regions is lost in this case but can be covered by other reads having these regions being placed in the middle or at the end of the read.

HiLive2 is an all-mapper by design. This means that all alignments within the search space specified by the parameter setting can be found and reported. However, the default output option of HiLive2 is to report only one best alignment for each read. For most analyses, this output option is the expected behavior. Output is written in the well-established BAM or SAM format. For all output cycles, temporary files are stored and can be used to efficiently produce output with different options (e.g., writing all alignments instead of one best alignment for each read). This functionality is particularly useful for explorative analyses. Additionally, these temporary files can be used as an entry point to continue the alignment if the process crashed or had to be interrupted.

Data Download and Conversion

The human reference genome hg19 was obtained from the National Center for Biotechnology Information (NCBI) and only considered chromosomes 1 - 22, X and Y. Alternative regions were omitted. The sequences were stored in a single multi-FASTA file. For the evaluation of variant calls with RTG Tools [107], the reference genome was converted to SDF format. The sequencing datasets of the individual NA12878 were downloaded from the European Bioinformatics Institute (EBI) in FASTQ format. For read mapping with HiLive2, read pairs were converted to Illumina base call file (BCL) format, distributed on one lane and 64 tiles. There were four different definitions for exome capture region definition required for the

different datasets (see Table 2.2 in the results section). The regions were obtained in BED format from the respective producer, if available. Whenever multiple definition files were provided, the primary target regions were selected.

Gold standard variants for the individual NA12878 were downloaded from the Genome in a Bottle (GIAB) consortium [108] and regularized with the vcfallelicprimitives tool of VCFtools [109]. SNPs and InDels of the gold standard were stored in two separated files and filtered out against the exome capture regions using BEDTools intersect [110]. The resulting files were used as the gold standard for datasets using the respective exome capture definition. During the evaluation of the results, only variant calls in high confidence homozygous regions which were obtained from GIAB were considered.

Real-time Read Alignment with HiLive2

The index of human reference genome hg19 for HiLive2 was built with default parameters. The creation of base call files by the sequencing machine was simulated using a script for sequencing simulation with a sequencing profile for HiSeq2500 machines in rapid mode and using dual barcodes. As no barcodes were present in our datasets, no data was written by the sequencing simulator for the respective cycles. HiLive2 was run in fast mode allowing faster turnaround times at the expense of slightly lower recall. Technical parameters as lanes, tiles and read length were set according to the datasets. In general, we chose cycles 30, 40, 55, 75 and 100 for each of the two reads as output cycles. For datasets with read lengths other than 2 x 100 bp, we adapted the output cycle numbers to 30, 40 and 50 (SRR292250) or 30, 40, 55 and 76 (SRR098401). We used the recommended number of threads (one thread per tile) for HiLive2 resulting in 64 threads for all datasets.

Read Alignment with Bowtie 2

The index of human reference genome hg19 for Bowtie 2 was built with default parameters. Read alignment with Bowtie 2 was performed with default parameters using ten threads.

Variant Calling with xAtlas

Variant calling with xAtlas was performed for each chromosome individually. Therefore, the alignment files of HiLive2 or Bowtie 2 were split in 24 files (one for each chromosome).

The resulting files were sorted and indexed using samtools [111]. Afterwards, variants were called with xAtlas for the respective exome capture regions using default parameters. Sorting, indexing and variant calling was performed with 24 threads (one per chromosome). The resulting VCF files were merged using vcflib vcf-concat (*https://github.com/vcflib/vcflib*) for SNPs and InDels separately.

Measure of Turnaround Time

The sequencing simulation script provides time stamps for each written sequencing cycle. These time stamps were compared to the system time stamps for the last modification of the alignment output files of HiLive2. The time span between both time stamps describes the alignment delay of HiLive2. Additionally, we measured the clock time of the xAtlas pipeline. The sum of sequencing time until the respective cycle, the alignment delay of HiLive2 and the clock time of xAtlas yields the overall turnaround times of our workflow.

Evaluation with RTG Tools

We used the vcfeval program of RTG Tools for the validation of variant calling results. We used the gold standard for the respective dataset (depending on the used exome kit) as baseline and the variant calling output of xAtlas or Genome Analysis Toolkit (GATK) as call. The human reference hg19 in SDF format was used as reference template. Only variant calls being included in the high-confidence regions for individual NA12878 provided by the GIAB consortium were considered for validation. We ran RTG Tools with 24 threads and used the squash-ploidy and all-records parameters. For variant calls produced by xAtlas, we additionally defined QUAL as the field for variant call quality. For GATK, the GQ field is chosen by default. RTG Tools vcfeval returns a list of statistical measures for different thresholds of the variant call quality field, including precision and recall. These values served as input for the precision-recall curves (PR curves) shown in Appendix Figure A1.1 and used for the calculation of the area under the precision-recall curve (APR).

Statistical Measures

We used precision and recall values for the validation of our approach. True positives (TP) describe the number of correctly detected variants. False negatives (FN) are the number of

undetected variants. False positives (FP) are the number of variants that were detected by our pipeline but are not contained in the gold standard.

Recall is the relative number of variants of the gold standard that were found by our approach $\frac{TP}{TP+FN}$. Precision is the fraction of variants called by our approach that are also present in the gold standard $\frac{TP}{TP+FP}$.

Software Versions

Table 2.1 shows the software versions used for this study.

Table 2.1: List of software used in this study. Software with source Bioconda was installed with the environment management software conda (*https://conda.io*) and obtained from the Bioconda channel [105].

Name	Version	Source	Used for
BEDTools	2.21.0	*https://github.com/arq5x/bedtools2*	VCF and BED file processing
Bowtie 2	2.3.4.1	Bioconda	Read alignment
GATK	3.8	Bioconda	Alignment file processing
HiLive2	2.0	*https://gitlab.com/rki_bioinformatics/hilive2*	Real-time read alignment
RTG Tools	3.9	Bioconda	Benchmark of variant calls
SAMtools	1.8	Bioconda	SAM / BAM file processing
vcflib	1.0.0_rc1	Bioconda	VCF file processing
VCFtools	0.1.12	*https://sourceforge.net/projects/vcftools/*	VCF file processing
xAtlas	0.1	Bioconda	Fast variant calling

Code Availability

The source code of HiLive2 is available for public download on *https://gitlab.com/rki_bioinformatics/hilive2* and comes with extensive documentation and sample data. HiLive2 is also available on Bioconda for easy installation (conda install –c bioconda hilive2).

Data Availability

Sequencing data of the individual NA12878 is publicly available on the NCBI Short Read Archive (SRA) and on the EBI FTP server. Gold standard variant calls are publicly available from the GIAB consortium. Human reference genome hg19 was obtained from the NCBI FTP server. Exome capture targets are available from the manufacturers or from third party resources.

2.3 Results

Implementation and Experimental Setup

To allow for faster NGS-based diagnosis and treatment, we developed a new real-time read mapping algorithm that generates high-quality results. We combined our new software with a fast variant caller to produce high-quality variant calls based on intermediate read mapping results, while sequencing is still running. This approach allows reliable and fast variant calling results without reducing the final sequencing coverage or quality. Therefore, we adapted our real-time read mapper HiLive [92] that gives output at the end of sequencing, using a novel algorithm based on the efficient FM-index [81] implementation of the SeqAn library [106] for continuously analyzing sequencing results during runtime. The new version (HiLive2) achieves scalability to larger indices such as the complete human reference genome. At the same time, the algorithm comes with improved performance in terms of runtime, memory and data storage and overcomes heuristic elements that were present in previous version of HiLive. The high scalability and accuracy of HiLive2 enable the combination of real-time read mapping results with complex follow-up analyses that has not been possible with the previous version. To demonstrate the power of such analyses, we performed variant calling on seven WES datasets of the human individual NA12878 from the CEPH Utah Reference Collection (Table 2.2) using the real-time read mapping results of HiLive2 as input data.

Table 2.2: Summary of datasets evaluated in this study. Information about sequencing platform, exome capture and coverage were adopted from Hwang et al. [112]

Accession No.	Platform	Exome capture	Exome coverage	Reads[1]	Read length
SRR098401	HiSeq2000	SureSelect v2	116.84 x	114 M	2 x 76 bp
SRR292250	HiSeq2000	SeqCap EZ v2	116.06 x	85 M	2 x 50 bp
SRR515199	HiSeq2000	SureSelect v4	298.45 x	167 M	2 x 100 bp
SRR1611178	HiSeq2000	SeqCap EZ v3	79.93 x	45 M	2 x 100 bp
SRR1611179	HiSeq2000	SeqCap EZ v3	79.84 x	45 M	2 x 100 bp
SRR1611183	HiSeq2500	SeqCap EZ v3	129.94 x	74 M	2 x 100 bp
SRR1611184	HiSeq2500	SeqCap EZ v3	111.90 x	64 M	2 x 100 bp

[1] M = millions

For variant calling, we used the fast variant caller xAtlas which shows comparable accuracy to established methods at much lower runtime [104]. We compared our results to read mapping with Bowtie 2 [85] and variant calling with either xAtlas or GATK 3.3.017 for the same datasets. For Bowtie 2 + GATK, we took the results from Hwang et al. [112] following the GATK best practice procedure using Picard ReorderSam (*http://broadinstitute.github.io/picard/*) and GATK IndelRealigner, BaseRecalibrator and HaplotypeCaller. Accuracy was determined by comparing the results to the well-established high-confident variant calls for the human individual NA12878 published by the GIAB consortium [108]. As benchmarking method we used the area under the precision-recall curve (APR).

Accuracy of Real-time Results

In Illumina sequencing, all reads are sequenced in parallel. In each so-called sequencing cycle, sequence information of one additional nucleotide is obtained for all reads. Thus, the current length of a read equals the number of the respective cycle (e.g., 40 nucleotides after cycle 40). To demonstrate the capability of our approach to provide interpretable results during runtime, we applied our workflow at different stages of sequencing. We expected

our live results to show higher accuracy for higher cycles due to the increasing amount of available sequence information. At the same time, we analyzed whether the detected variants in early sequencing cycles are as reliable as variants called at the end of sequencing. This is a crucial criterion for the proposed workflow since interpretation of live results is only meaningful when based on reliable variant calls. Therefore, besides comparing the APR values of different sequencing cycles, we also examined precision and recall separately. Figure 2.1 a shows the progression of the APR values for SNP calling in all analyzed datasets

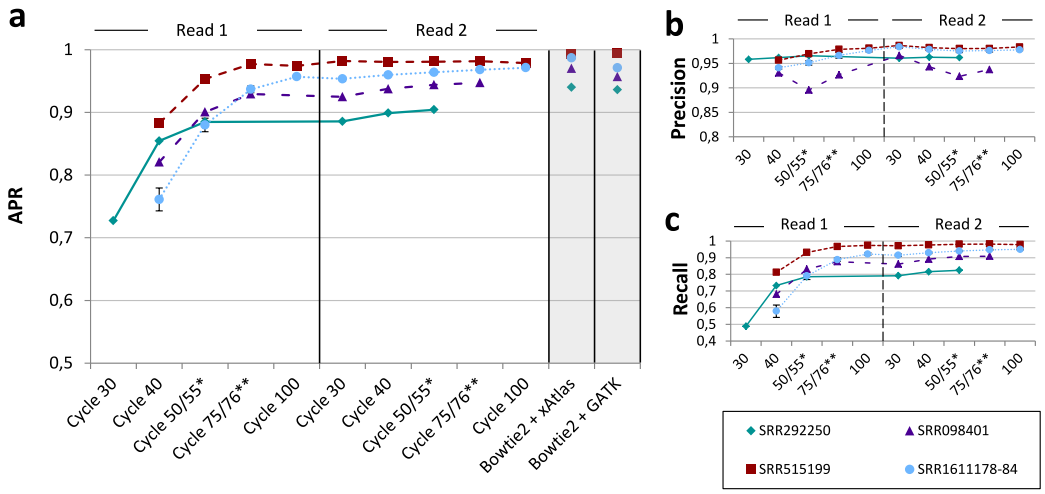


Figure 2.1: The area under the precision-recall curve (APR) for SNP calling in seven datasets at different sequencing cycles. SNP calling was performed with xAtlas using real-time read mapping results of HiLive2. Results for the samples SRR1611178, SRR1611179, SRR1611183 and SRR1611184 were combined to a single data series due to their high similarity (SRR1611178-84). Error bars for this data series show the standard deviation. Reads of SRR292250 and SRR098401 were shorter than 2 x 100 bp which leads to missing data points. The vertical, ticked line in the middle of the plot divides the first and second read. **a** The gray columns show APR values using Bowtie 2 for read mapping and xAtlas (left) and GATK (right) for variant calling. The data for Bowtie 2 + GATK were taken from Hwang et al. [112]. The real-time workflow with HiLive2 and xAtlas provides first results after 40 sequencing cycles (30 cycles for SRR292250). An APR greater than 0.9 is reached after 75 cycles for all datasets with a minimal read length of 75 bp. Until end of sequencing, there is a moderate increase of the APR. **b** Precision with a quality threshold of 1 for variant calling with xAtlas. The results show no precision lower than 0.89 for all sequencing cycles. In general, precision increases only slightly over time. This indicates that results in early sequencing cycles are already reliable. **c** Recall with a quality threshold of 1 for variant calling with xAtlas. The results show strong improvements from the first results available until the end of the first read. The progression of all curves is similar to that of the APR curve (Figure 2.1 a), indicating the correlation between those two measures. *Cycle 50 for SRR292250, cycle 55 for all other datasets. **Cycle 76 for SRR098401, cycle 75 for all other datasets.

with increasing sequencing time. In cycle 30, sequence information was not sufficient to call any variants with the given parameter settings for six of seven datasets. For dataset SRR292250, read mapping parameters were adapted by HiLive2 automatically due to the short read length of 50 bp. This led to earlier results after 30 cycles, while first results were available after cycle 40 for all other datasets. Results show a continuous increase of the APR values for all cycles of the first read. In cycle 75, an APR larger than 0.9 was achieved for all datasets with sufficient read length. Afterwards, the APR values continue increasing moderately. When regarding the progression of precision (Figure 2.1 b) and recall (Figure 2.1 c) over sequencing time separately, it can also be also observed that lower APR values for earlier sequencing cycles are mainly caused by a lower recall while precision changes only slightly with more sequence information available. The same conclusions are supported by the individual PR curves for all datasets which show a large increase of the recall but only minor changes of specificity over sequencing time (Appendix Figure A1.1). This indicates that live results are highly reliable and can therefore serve for early interpretation and problem-specific follow-up analyses. The increasing number of SNP calls in subsequent cycles provides additional information for complementing the previous interpretation of the data. However, the final results with HiLive2 show slightly lower maximum recall values than the same workflow applied to read mapping results of Bowtie 2 (Appendix Figure A1.1). This can be explained by the read mapping approach of HiLive2 which tolerates only a specified number of errors for a read. Thus, regions with a high number of variations may be lowly covered which leads to undetected variants. The same effect is somewhat stronger for InDels as the mapping algorithm only tolerates InDels with a maximum length of three nucleotides by default due to computational costs. While this behavior led to a lower recall than based on read mapping with Bowtie 2, the results showed comparable or higher precision (Appendix Figure A1.1). Thus, although focussing on SNPs in this study, our workflow can also provide valuable insights about small InDels.

Turnaround Time of the Workflow

Besides the accuracy of results, turnaround time is the second crucial factor for NGS-based real-time analyses. Thereby, live results should be available as soon as possible after the data of the respective sequencing cycle was written without showing significant delay in any stage of sequencing. We measured the turnaround time of real-time mapping with HiLive2 and subsequent variant calling with xAtlas for the same runs that delivered the accuracy

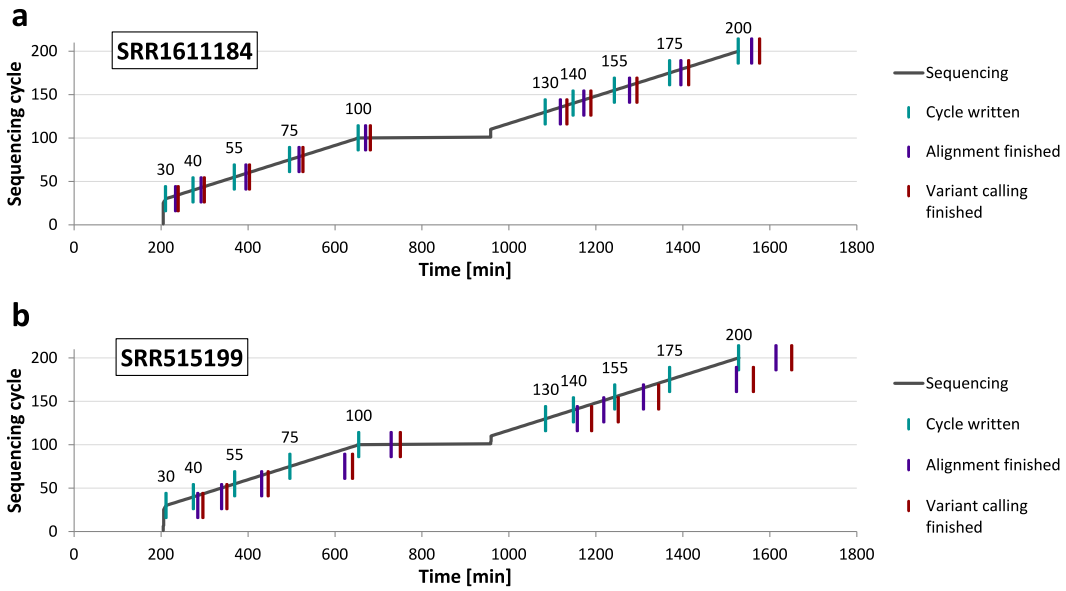


Figure 2.2: Turnaround time of our workflow for datasets SRR1611184 (a) and SRR515199 (b). For each cycle, the first vertical line indicates the time point when the data for the respective cycle was completely written. The second line shows when the alignment output of HiLive2 is written. The third line indicates the end of our workflow resulting in the output of variant calls for the respective cycle. Vertical lines with the same vertical position belong to the same output cycle.

results shown before. All computations were run on a 128-core machine (Intel Xeon CPU E5-4667 v4 @ 2.20 GHz, 45 MB Cache) with 500 GB RAM, using a maximum of 65 threads per dataset. Figure 2.2 shows an overview for the turnaround time of our workflow for different sequencing cycles for datasets SRR1611184 and SRR515199. For SRR1611184, variant calling results were available after a maximum of 35 minutes for all output cycles of the first read and a maximum of 52 minutes for all output cycles of the second read. With approximately 1 million reads per tile (or thread), these are realistic numbers for a real-case scenario using benchtop sequencing devices. For SRR515199, the latency is way higher reaching a maximum of more than three hours (193 minutes) for cycle 175. The higher latency originates from analyzing approximately 2.5 times as many reads per tile (or thread) as for dataset SRR1611184. This shows that the latency of real-time results strongly depends on the number of reads that are analyzed per thread which varies with the available hardware and the used sequencing device. In general, the analysis of high coverage datasets can be significantly reduced by adapting the alignment parameters at the cost of accuracy. However, for the sake of comparability, we ran all datasets with the same parameter settings in this

study leading to a higher latency for higher coverage datasets. Results for the other datasets are shown in Appendix Figure A1.2. Thereby, five of the seven datasets showed a maximum latency of less than one hour from data output to interpretable results for each output cycle.

2.4 Discussion of Results

In clinical applications and infectious disease outbreaks, the turnaround time of analyses is a critical factor for an effective treatment of patients. At the same time, a high analysis depth and an open perspective for unexpected findings are further crucial criteria in such scenarios. Therefore, despite its significantly higher turnaround times than alternative methods, NGS presents an established analysis method in several time-critical applications. For example, a comprehensive report of vancomycin resistant *Enterococcus faecium* infections in three patients was created in 48.5 hours including over-night culturing using an Illumina MiSeq benchtop sequencer [113]. Also in the field of acutely ill infants with suspected genetic diseases, there were impressive improvements in the applicability of NGS-based diagnostics including a 26 hours protocol for provisional molecular diagnosis [96]. However, even with a significant speed-up of the computational analysis using faster software or specialized hardware such as FPGAs, a decrease of turnaround time is strictly limited due to the sequential paradigm of data creation and analysis. Motivated by this, Miller et al. [96] introduced the idea to combine bioinformatics analysis using FPGAs, in particular the DRAGEN system, with the concept of rapid pulsed whole genome sequencing [97] to achieve near real-time analysis results. Such an approach would require a conversion of sequencing data to FASTQ or BAM / CRAM format for each desired output cycle as these are the only file formats supported by the DRAGEN system. At the same time, specialized hardware is required and licenses must be purchased. Alternatively, a cloud service can be used which can be problematic due to data protection guidelines and the required speed of data transfer. However, to the best of our knowledge there is no study describing a proof-of-principle for this idea. Another commercially available approach to speed up NGS analysis is implemented in the Sentieon Genomics Tools, an optimized version of BWA and GATK that overcomes the need of specialized hardware or cloud access [114]. On the other hand, this software is not specifically designed to produce real-time results and involves costs for licenses. Furthermore, both commercial systems are developed with a strong focus on variant calling for human samples. In contrast, the workflow presented in this study is open source, runs on a standard Linux machine and allows for easy and flexible adaption

of the workflow for different scenarios. At the same time it is highly scalable and provides high-quality analysis results without the need of acquiring specialized hardware, cloud access and licenses.

The results of our study demonstrate the enormous potential of our approach to reduce the turnaround time from sample arrival to meaningful analysis output by several hours up to days depending on the used sequencing device. Thereby, live results in very early stages of sequencing can already deliver highly confident results while the quantity of analysis results (i.e., the number of called variants in this study) increases with a growing number of sequenced nucleotides per read. Live analyses can therefore provide first relevant insights into the data while the analysis becomes more comprehensive with ongoing sequencing. The sensitivity of results in early sequencing cycles is thereby implicitly limited by the lower total coverage when compared to the full read length as well as the relatively high number of unambiguously mapped reads, especially in low complexity regions. To demonstrate the power of our approach, we showed its application to human WES data including real-time alignment of all reads to the full human reference genome hg19. We chose this type of data due to its complexity and computational demands as well as the availability of high-quality and extensively studied gold-standard datasets provided by the GIAB consortium. However, our approach is not restricted to human WES data and the presented use case of variant calling. We rather see an enormous potential of real-time read mapping to provide means for a wide range of complex follow-up analyses for various types of data. Still, due to current technical limitations of HiLive2 and runtime limitations of tools for subsequent analysis such as variant calling, certain types of analyses that require huge amounts of data are only feasible with limitations. For example, when applying the same SNP-calling workflow as shown for the WES data to a 30 x WGS dataset (SRR6808334) [115] of the human individual NA12878, the latency of real-time results was up to six hours while achieving a maximum recall of 87 % with 97 % precision (Appendix Figure A1.3). Even for such analyses, valuable insights could be gained for early sequencing cycles (e.g., 40 % recall with 90 % precision after 55 cycles and 62 % recall with 91 % precision after 75 cycles), but given the high latency and comparatively low accuracy we only recommend our workflow for such datasets in exceptional situations and when no other options are available. However, future improvements, such as a decrease of I/O operations, in-tile multithreading and acceleration of alignment output, could significantly reduce the latency of HiLive2 and make it more applicable for higher throughput applications.

A second limitation of our approach comes up when dealing with multiplexed sequencing

data which is usually applied for high-throughput applications to sequence more than one sample within one sequencing run. While HiLive2 provides demultiplexing functionality and can produce separate alignment files for each sample, the sequencing approach of Illumina limits the early assignment of reads by sequencing the barcodes after completing the first read. In doing so, it is not possible to distinguish between different samples before sequencing of the first read has finished. A change in the sequencing order is not trivial as sequencing the barcodes first would negatively influence the initial clustering that is performed during the first couple of sequencing cycles and relies on a wide variety of sequences which is not given for the barcodes, in particular when only having a low number of samples. When single-end sequencing is sufficient, this can be overcome by performing asymmetric paired-end sequencing. This means, that the first read is only sequenced for several base pairs to enable proper clustering, followed by the barcodes and the complete second read. While resulting in some additional runtime at the beginning of sequencing, this enables an early assignability of reads to the samples. A different conceivable solution could be to include a random sequence followed by inline barcodes at the beginning of the first read.

Alternative approaches to NGS for diagnosis can also be highly valuable for different scenarios. Molecular approaches are usually highly reliable and provide answers to specific questions in a very short timeframe and at much lower costs. For example, the detection of 25 genetic mutations in *M. tuberculosis* that confer to drug resistances can be finished in approximately two hours with a variation of the molecular GeneXpert test [98]. Even when providing live results, such short turnaround times are currently not feasible with NGS-based approaches due to the required time for sample preparation and clustering. Another interesting technology for time-critical applications is ONT sequencing. It was shown that metagenomic detection of viral pathogens can be achieved in less than six hours [116]. While ONT shows a high portability and much faster sample preparation as additional benefits, this and other current long-read technologies are still expensive and limited by their comparatively low coverage and high error rates. It is therefore hard to reliably identify lowly abundant pathogens, genetic variants, parallel infections or the presence of viral quasispecies. Thus, especially when it comes to these or other questions going beyond the identification of highly abundant pathogens in time-critical applications, real-time analyses for Illumina sequencing can be of great benefit.

Concluding, we consider our new real-time workflow for Illumina sequencing to be a complementary method to molecular tests and ultra-portable, long-read sequencing for time-critical analyses. It fills the current gap of short turnaround times, an open-view

perspective and high sequencing coverage which is essential for a plethora of applications such as pathogen identification and characterization, identification of acute genetic diseases or epidemiological analyses. Therefore, our approach is an important step for improving the ability for fast interventions in exceptional clinical situations, personalized medicine and infectious disease outbreaks.

3 PriLive: Privacy-preserving Real-time Filtering for Next-Generation Sequencing

In NGS, re-identification of individuals and other privacy-breaching strategies can be applied even for anonymized data. This also holds true for applications in which human DNA is acquired as a by-product, e.g., for viral or metagenomic samples from a human host. Conventional data protection strategies including cryptography and *post hoc* filtering are only appropriate for the final and processed sequencing data. This can result in an insufficient level of data protection and a considerable time delay in the further analysis workflow.
We present PriLive, a novel tool for the automated removal of sensitive data while the sequencing machine is running. Thereby, human sequence information can be detected and removed before being completely produced. This facilitates the compliance with strict data protection regulations. The unique characteristic to cause almost no time delay for further analyses is also a clear benefit for applications other than data protection. Especially if the sequencing data are dominated by known background signals, PriLive considerably accelerates consequent analyses by having only fractions of input data. Besides these conceptual advantages, PriLive achieves filtering results at least as accurate as conventional *post hoc* filtering tools.

3.1 Background

Over the last decade, the amount of publicly available genomic data has increased by several magnitudes. The development of new technologies that enable faster, cheaper and ultra-portable DNA sequencing further accelerates the growth of data generation; a total number of 100 million to 2 billion sequenced human genomes is estimated by 2025 [117], the latter corresponding to approximately 25 % of the current global population. With this forthcoming mass of produced sequencing data, the question of data protection becomes more and more important. Until today, no general concept to securely store, share and analyze these data with respect to data protection has been realized on a global scale. Consequently, researchers receive insufficient support when dealing with sensitive data despite a potential lack of instruction and knowledge. This does not only endanger the privacy of patients and their relatives but may also result in legal actions against researchers if existing data protection standards are not adequately respected.

Several types of data have been shown to enable violations of privacy even if the related metadata are anonymized. These include genome-wide association studies [118–120], clinical proteomics data [121], phenotype-genotype correlation studies [122] and personal genomes [123]. Thereby, a plethora of different privacy breaching strategies has been applied including statistical procedures, phenotypic prediction and data linkage. A detailed review of known privacy breaching strategies is provided by Erlich and Narayanan [124]. Most of these workflows "require a background in genetics and statistics and, importantly, a motivated adversary" [124]. Additionally, the results can be difficult to interpret and often involve a degree of uncertainty. However, with the increasing mass and quality of available data as well as technological advances, the reliability of privacy breaching techniques will be further improved and new methods will occur in the future. It is therefore highly desirable to remove sensitive information from the data, especially if it is not relevant for the analyses.

Most existing technical solutions for the protection of genomic data focus on human samples. Common approaches include cryptography [125–127], specialized data structures (e.g., based on Bloom filters [128, 129]), differential privacy [130, 131], selective data retrieval [132] or combinations of them. Besides human data, several types of non-human data exist that may contain sensitive information. Examples for these are human metagenomic datasets or viral sequencing data from a human host that can hardly be purified on a biological level [133, 134]. The protection of such data is essential and seems to be comparatively simple because the contained human information is usually of no interest for the analyses. Tools to detect and remove human reads from NGS samples have already been developed for genomic [135] and metagenomic [135–137] datasets. Nevertheless, the targeted removal of human data is often neglected for several reasons such as an increased analysis time or concerns about data loss. Another drawback of conventional data protection strategies is that they can only provide a limited level of genomic privacy by design: First, the original unsecured data is completely accessible in the timespan between data creation and the application of a privacy-preserving procedure. Second, consequently, the original data can be (and often is) stored further on besides the filtered data. Both aspects are potentially in conflict with legal rights of many countries, for instance the European Union and its 'Data Protection by Design and by Default' principle (Art. 25 EU General Data Protection Regulation). Moreover, this behavior may cause a lack of data protection if the internal access to the original data is not properly controlled or if the data are not sufficiently protected from attacks from the outside. Thus, there is an actual need for a new concept to remove sensitive information from NGS samples that (i) operates before the data is completely accessible, (ii) removes

sensitive data irreversibly, (iii) requires no additional analysis time and (iv) is independent of human interaction to enable institutional control for data protection. With PriLive we present a new approach that meets these requirements and therefore provides the highest level of genomic privacy for human-related NGS data.

3.2 Methods

In Illumina sequencing, hundreds of millions of short DNA fragments (reads) are analyzed in parallel. Thereby, the nucleotide sequences of all reads are identified and written by the sequencing machine base by base. Conventional sequence analysis tools for Illumina data, e.g., read mapping [55, 138] and (*de novo*) assembly software [139], cannot operate before the sequencing machine has finished and the raw data are converted to a human readable file format (usually FASTQ). To the best of our knowledge, all existing mapping-based, privacy-preserving read filtering strategies are based on conventional approaches that require the full sequence information as input. We present a novel tool, PriLive, to detect and remove human reads from Illumina HiSeq sequencing data (or similar) while the sequencing machine is running. We use a k-mer based real-time read mapping strategy that directly operates on the base call files that are produced by the sequencing machine. A prior conversion of the sequencing data to a human readable file format is not necessary. All available sequence information at a specific moment of the sequencing procedure is used to compute intermediate alignment candidates. These alignments are extended with each new base call that is produced by the sequencing machine [92]. Additionally to the mapping to a reference genome of interest (foreground alignment), we implemented a second alignment strategy to detect reads that should be removed from the data (background alignment). In the field of data protection, this usually is a mapping to a human reference genome or parts of it. As long as there is a promising foreground alignment candidate or the read maps to none of both reference genomes, the sequence information is retained. Only reads that do not map to the foreground reference genome but have a meaningful background alignment (e.g., human) are immediately removed from the sequencing data. The sequence information of all succeeding sequencing cycles of a detected read can be removed right after it was produced by the sequencing machine. By this approach, privacy-preserving real-time filtering with PriLive finishes only a few minutes later than the sequencing machine and therefore provides a significantly higher level of data protection than conventional tools.

Algorithm

PriLive is based on the basic functionality of the real-time read mapping software HiLive [92] which was designed for Illumina short read sequencing protocols (HiSeq or similar). In each sequencing cycle, the next nucleotide of each read is identified. The resulting sequence information is written to a connected hard drive in the binary BCL format. HiLive obtains these base call files as an input to perform read mapping to a set of reference genomes when the sequencing machine is still running rather than waiting for all data to be produced and converted to a human readable file format (FASTQ for most other read mappers). HiLive uses a k-mer approach for both alignment steps, i.e., to find candidate positions (seeding) and to extend the resulting seeds. Several heuristic approaches are used to identify the minimal number of errors in a candidate alignment. If a user-specified error threshold is reached, the respective seed is discarded.

For read filtering with PriLive, we implemented a local alignment strategy for a set of background reference genomes. This background alignment of a read runs in parallel to the foreground alignment when the minimal number of errors, i.e., the edit distance, for the foreground alignment ϵ is equal or larger than a specified threshold λ_1 ($\epsilon \geq \lambda_1$). If the minimal number of errors ϵ exceeds a second threshold $\lambda_2 \geq \lambda_1$ in the further sequencing procedure ($\epsilon \geq \lambda_2$) and there exist a significant background alignment, the respective sequence information will be removed from the sequencing data (Figure 3.1). By default, λ_1 and λ_2 depend on the user-specified parameter -e (or --min-errors) that describes the number of errors that are tolerated in the foreground alignment. This leads to the intuitive behavior that λ_1 and λ_2 are higher if more errors in the foreground alignment are tolerated. The minimal alignment score for the background alignment to remove a read depends on the read length r, k-mer size k and λ_1. The background alignment score itself describes the number of matching nucleotides in the local alignment. This includes an anchor of consecutive matches of length $a \geq k$ followed by an alignment strategy that allows for single nucleotide mismatches (substitutions, insertions and deletions of length 1). Several consecutive errors are not permitted. However, this limitation does not lead to a considerably lower sensitivity since reads with longer, consecutive mismatch regions can be identified at a different position of the read. The default parameters are designed for read filtering of genomic samples including a foreground reference genome. Recommended parameter adjustments when using PriLive without a foreground reference, e.g., for metagenomic samples, are described in the next section.

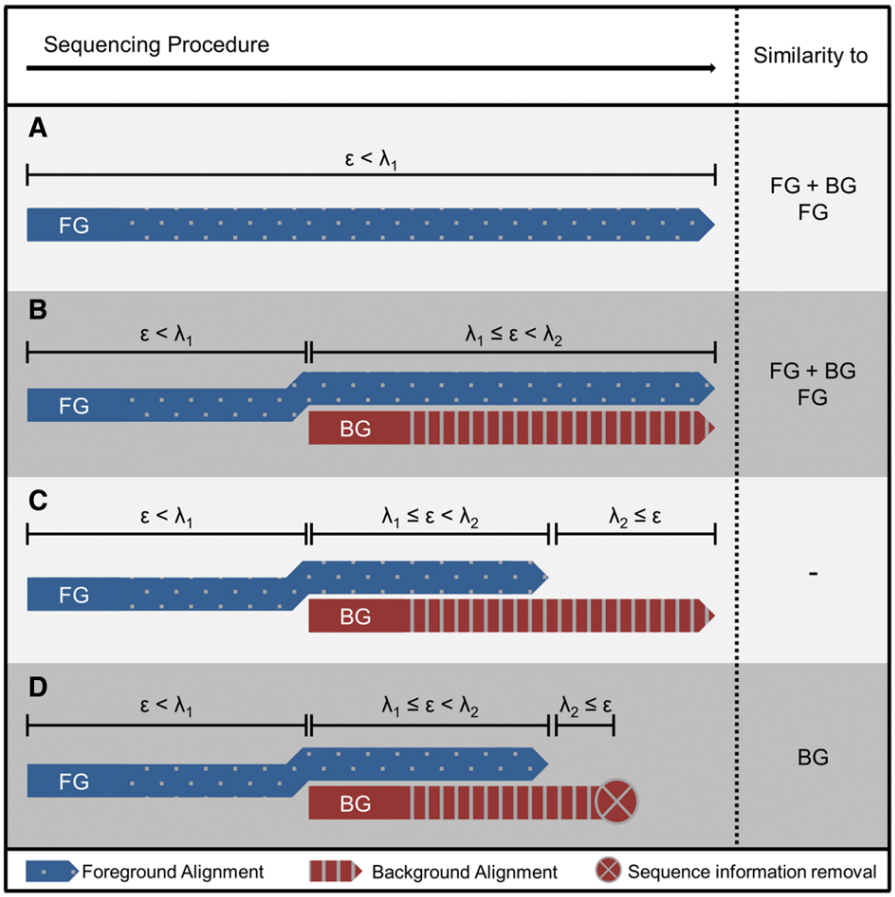


Figure 3.1: Alignment approach of PriLive. The similarity of a read to the foreground (FG) and background (BG) reference genome has a strong impact on how the read is handled by PriLive. ϵ is the minimal number of errors for a considered read when compared to FG at a given time of the alignment procedure. λ_1 and λ_2 are the error thresholds for the foreground alignment to start the background alignment in parallel and to allow the removal of sequence information, respectively. **(A)** High similarity to FG. The first threshold λ_1 is not reached such that the background alignment does not start. The sequence information of the read is retained. **(B)** Some similarity to FG. The first threshold λ_1 is reached such that the background alignment is started in parallel. The second threshold λ_2 is not reached, so the sequence information is retained regardless of the background alignment score. **(C)** No significant similarity to both FG and BG. Both thresholds λ_1 and λ_2 are reached such that the background alignment is started in parallel. Since there is no significant alignment to BG the sequence information is retained. (**D**) No significant similarity to FG. Both thresholds λ_1 and λ_2 are reached. Because of the significant alignment to BG the sequence information will be removed in real-time and the alignment procedure is not continued for the respective read.

Parameter Selection

Parameters of PriLive should be selected according to the given application. The default parameters of PriLive are designed for genomic samples of a known organism, i.e., when a foreground reference is given. Thus, if PriLive is used with a foreground reference genome, only minor changes of the parameter settings are necessary for most applications. The most common adjustments are setting the -e parameter for the number of tolerated errors in the foreground alignment according to the read length and the expected mutation rate of the organism and the increase or decrease of --bg-score for a lower or higher filtering sensitivity, respectively. If PriLive is used without a foreground reference, e.g., for metagenomic samples, we recommend to set the number of tolerated errors for the foreground alignment (-e) to 0. This implies $\lambda_1 = \lambda_2 = 0$ which means that the background alignment is started from the beginning of the read. Additionally, for samples without a foreground reference, we recommend to manually increase the value of --bg-score. This is necessary to achieve a sufficient filtering specificity (if a foreground reference is available, specificity is mainly achieved by keeping reads aligned to the foreground reference). At the same time, the value of --bg-score should always be lower than half the read length to allow for filtering reads with two or more consecutive errors in the middle of the read. An overview of the default values of important parameters is provided in Table 3.1.

Table 3.1: Overview of important algorithmic parameters in PriLive. FG is the foreground alignment, BG is the background alignment.

Parameter	Variable	Meaning	Default value
--min-errors	e	# tolerated errors for FG	Defined by the user (default: 2)
--CYCLES / -r	r	Read length	Defined by the user (required)
	k	k-mer size	Defined at compile-time (default: 15)
--bg-anchor	a	BG anchor length	$k+4$
--bg-start	λ_1	# errors in FG to start BG	$log_2(max(1,e))$
--bg-discard	λ_2	# errors in FG to discard a read	Min: $e+1-log_2(max(1,e))$; Max: $e+1$
--bg-score		BG score to discard a read	$max(30,\ 35 \cdot (log_{10}(r-(max(1,\lambda_1)-1)\cdot k))-1)$

Technical Details

Reads detected by PriLive are covered by calls of the ambiguous base N in the original base call files that are written during the sequencing procedure. This is done by replacing the respective bytes that encode the nucleotide and the quality of the base call by 0-bytes. It should be noted that quality values given in consequent data processing steps can be negatively affected by this. To ensure that PriLive works properly, the base call data must be organized as specified for Illumina HiSeq (bcl2fastq v1.8.4 User Guide, 2013). This must be especially considered when using base calling software other than provided by Illumina or special hardware set-ups. When using PriLive for decontamination or host removal other than human, the modification of the base call files can be deactivated. PriLive then only returns a list of filtered reads that can be considered in a later step of the analysis workflow. It is also possible to create copies of the original data in an encrypted (hybrid AES256/RSA encryption) or unencrypted manner.

Reference Genomes and Index Building

Index files were built for the tools PriLive, DeconSeq [135] and BMTagger [137]. For lobSTR [140], the index files were retrieved from the lobSTR project website. The human index files were generated for the human reference genome hg38. For the CPXV index we used the NCBI sequence for cowpox virus Brighton Red (NC_003663.2). For the index that contains the flanking regions of the short tandem repeat (STR) markers on the human Y-chromosome (Y-STRs), some preprocessing was necessary. We retrieved the positions of the Y-STRs from a BED file that is provided for Y-STR genotyping with lobSTR. Afterwards, we searched for these positions in the corresponding reference from the hg19 resource bundle for lobSTR. For each marker, we added the upstream and downstream flanking regions of 100 bp length to our final reference genome. The STRs themselves were not included in the reference.

Index building for PriLive was performed with default parameters for all reference genomes. For the human reference genome hg38, all k-mers that occur more than 1000 times were trimmed (-t 1000). Trimming was not used for all other reference genomes. Index building for BMTagger and DeconSeq was performed as recommended for the purpose of human host removal.

Datasets

Simulated dataset CPXV_1. 10 M reads of cowpox virus Brighton Red (CPXV) and the human reference genome hg38 were simulated with an edit distance of 0 to 9 (1 M each). The reads are single ended and of length 150 bp. Simulation was done with the mason read simulator [141] using different error rates. The reads for the final data subsets were selected by the NM:i tag of mason′s BAM output files. This procedure was performed in the same manner for CPXV and human reads.

Simulated dataset CAMI_1. One sample of the Toy Test Dataset High_Complexity (HC_Sample1) of the Critical Assessment of Metagenome Interpretation (CAMI) study [142] was used as metagenomic dataset. The dataset includes approx. 75 M metagenomic reads of length 2 x 100 bp (paired end). As background data, 10 M reads of the human reference genome hg38 were simulated with the mason read simulator. These reads of 2 x 100 bp (paired end) have an average error rate of 2 %.

Real dataset Venter_1. The original J. Craig Venter sequencing data (*ftp://ftp.ncbi.nih.gov/pub/TraceDB/Personal_Genomics/Venter/*) have been produced by Sanger sequencing. We removed the first 50 bp of each read for quality reasons (as suggested in Gymrek et al. [123]). The remaining sequence information was split in non-overlapping 150 bp long fragments and saved in FASTQ format. The paired-end information and quality of the Sanger reads were not considered. The resulting Illumina-like reads were used as an input for all analyses of the J. Craig Venter sequencing data. Although certain common effects in Illumina sequencing are not considered in this procedure, we expected these data to show a more realistic behavior than simulated data because of their real biological background.

Real dataset HiSeq_1. The dataset HiSeq_1 is a reproduction of an in-house sequencing run of the human HeLa derivative HEp-2 (ATCC CCL-23) that was infected with cowpox virus Brighton Red (ATCC VR-302) according to standard procedures [143]. The sequencing library was prepared with NexteraXT library generation. The sequencing procedure was performed on an Illumina HiSeq 1500 instrument (Illumina, San Diego, CA, USA) with 2 x 101 bp paired-end reads in rapid mode. The sample of interest was identified using the real-time demultiplexing functionality of PriLive. For the reproduction of the sequencing run, the original base call files were copied to the input directory for PriLive using the time stamps of the original sequencing procedure. In doing so, the base call files were written to the input directory of PriLive in the same intervals as they have been written by the sequencing machine. This guaranteed a similar behavior of PriLive as in a real-time

application. The computation was run on a 128-core machine (Intel Xeon CPU E5-4667 v4 @ 2.20 GHz, 45 MB Cache) with 500 GB random access memory (RAM). The original sequencing data are available at the NCBI SRA under accession number SRR5886855 (*https://www.ncbi.nlm.nih.gov/sra/?term=SRR5886855*).

Parameter Settings

BMTagger and DeconSeq were used with default parameters. PriLive was compiled with a k-mer size of 15 (default value, adapted from HiLive [92]). The remaining parameters were chosen according to the parameter selection guidelines described above. DeconSeq and BMTagger use algorithmic parameters relative to the read length. To allow for a better comparison of the tools, the number of tolerated errors for the foreground alignment (-e or --min-errors) was set according to approximately 3 % of the given read length which corresponds to the default value of PriLive for reads of length 75 bp. For samples without a foreground reference (CAMI_1 and Venter_1), -e was set to 0 as previously described. For the dataset CAMI_1, the minimal background alignment score (--bg-score) was set to 45 bp which corresponds to almost half of the read length (100 bp) to ensure a high specificity without missing background-related reads that have at least two consecutive errors. However, although running without a foreground reference, this parameter was not set manually for the dataset Venter_1 because of the special nature of the used background reference that contains only the flanking regions of the Y-STR markers. Through the gap between these two regions for each marker, the increase of the --bg-score parameter that only allows for non-consecutive mismatches would lead to a lower sensitivity which was not desirable for the given application case. For this special approach, parameter adjustments would also be necessary when using DeconSeq or BMTagger which was not done in this study. All other algorithmic parameters of PriLive were not adapted for any of the datasets described in this study. The lobSTR workflow for Y-STR genotyping was applied as recommended by the developers.

Statistical Measures

Sensitivity, specificity and F1 score were used for the validation of PriLive. True positives (TP) describe the number of correctly detected background-related reads (e.g., human). False negatives (FN) are the number of undetected background-related reads. False positives (FP)

are the number of foreground-related reads that are classified as background-related and true negatives (TN) are foreground-related reads that are correctly not detected as background data. Sensitivity is the relative number of correctly detected background-related reads from the complete background data $\frac{TP}{TP+FN}$. Specificity is the relative number of correctly not detected foreground-related reads $\frac{TN}{TN+FP}$. The F1 score is the balanced harmonic mean of precision $\frac{TP}{TP+FP}$ and sensitivity, calculated by $2 \cdot \frac{sensitivity \cdot precision}{sensitivity+precision}$.

3.3 Results

We compared the accuracy of PriLive in terms of sensitivity, specificity and F1 score to two conventional privacy-preserving read filtering tools, DeconSeq and BMTagger. DeconSeq was developed for genomic and metagenomic datasets. It is based on the read mapper BWA-SW [144] and supports foreground and background reference genomes. BMTagger was specifically designed for metagenomic datasets and therefore does not support foreground reference genomes. The algorithm of BMTagger makes filtering decisions using an alignment-free *k*-mer approach. Only if no clear decision was made by the alignment-free approach, a complete alignment is performed using the read mapping software SRPRISM [145].

We used two simulated and two real datasets to evaluate the performance of PriLive on genomic and metagenomic datasets (Table 3.2). The simulated viral dataset CPXV_1 contains reads of the human reference genome hg38 and cowpox virus Brighton Red (CPXV; Accession number NC_003663). It includes 10 M reads of length 150 bp and 0 – 9 errors (i.e., substitutions, insertions or deletions of length 1 bp) when compared to the respective reference genome for both organisms. The simulated metagenomic dataset CAMI_1 consists

Table 3.2: Overview of the datasets used for validation.

Dataset	Type	Foreground data	Background data	Reads
CPXV_1	sim.	CPXV / 10 M / 0 - 9 errors	hg38 / 10 M / 0 - 9 errors	150 bp
CAMI_1	sim.	Metagenomic / ~75 M	hg38 / 10 M	2 x 100 bp
Venter_1	real	JCV / ~135 M	Y-STRs	150 bp
HiSeq_1	real	CPXV	Human (HEp-2)	2 x 101 bp

of approximately 75 M reads of length 2 x 100 bp (paired-end) that were obtained from the CAMI study [142]. These data were mixed with 10 M simulated reads from the human reference genome hg38 with an average error rate of 2 %. As a real dataset, the sequencing data of J. Craig Venter were used to examine the ability of PriLive to prevent re-identification (Venter_1). Therefore, we reproduced a re-identification workflow [123] on the J. Craig Venter sequencing data before and after read filtering with PriLive. Finally we used PriLive to filter the data of an in-house Illumina HiSeq sequencing run. This dataset (HiSeq_1) contains reads of a CPXV-infected human cell line. With this reproduced real-time application we evaluated the ability of PriLive to finish read filtering only a few minutes later than the sequencing machine.

Simulated dataset CPXV_1. Figure 3.2 shows the results of PriLive, DeconSeq and BMTagger on the simulated date set CPXV_1. PriLive shows higher sensitivity than both other tools. This is especially the case for reads with high error rates. Even with an edit distance as high as 9 (6 % error rate), PriLive correctly identified 99.46 % of human reads whereas DeconSeq and BMTagger only achieved a sensitivity of 95.62 % and 96.44 %, respectively (Figure 3.2 a). At the same time, the specificity of PriLive was higher than 99.99 % up to the user-defined maximum number of tolerated errors in the foreground alignment (4 errors for the presented data). For reads with a higher error rate specificity was slightly worse but still in the same range as the specificity of DeconSeq (Figure 3.2 b). BMTagger, despite not considering cowpox virus as foreground reference in contrast to both other tools, showed the best specificity, especially for foreground-related reads with a high error rate. However, the overall accuracy of PriLive was at least as good as the accuracy of both other tools with a stronger focus on sensitivity to provide the highest possible level of data protection. If a higher specificity is required, this can be achieved by changing one single intuitive parameter (-e or --min-errors) which describes the number of tolerated errors for the foreground alignment. Besides the final accuracy, PriLive also achieved strong real-time results. After only half of the sequencing cycles, PriLive already had nearly full sensitivity for reads that contain at most one error when compared to the human reference genome. After 2 / 3 of the cycles, this also held true for reads with up to four errors. Additionally, more than 95 % of all reads with up to six errors have been detected at this point in time (Figure 3.2 c). These results show that the sequence information of most background-related reads can be removed even before it is entirely produced.

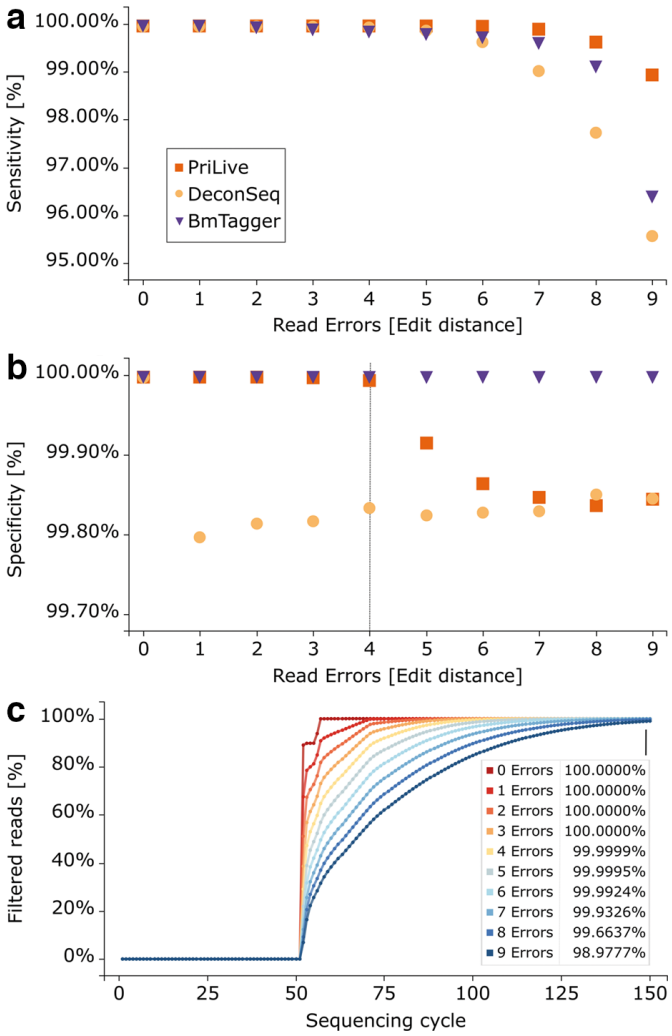


Figure 3.2: Removal of human reads for the simulated dataset CPXV_1. **(a)** Relative number of correctly detected human reads (sensitivity) of PriLive, DeconSeq and BMTagger. All tools achieve nearly full sensitivity for reads with a low number of mismatches. For reads with a higher number of errors, PriLive clearly outperforms both other tools. **(b)** Relative number of undetected non-human reads (specificity) of PriLive, DeconSeq and BMTagger. PriLive is set to consider all foreground alignments up to an edit distance of 4 (-e 4). It therefore achieves nearly full specificity for all cowpox virus reads with up to four errors (indicated by the dotted line). For a higher number of errors, the specificity of PriLive is still comparable to that of DeconSeq. **(c)** Relative number of human reads with an edit distance of 0 - 9 detected by PriLive in different cycles of the sequencing procedure. For human reads with a small number of errors, PriLive achieves nearly full sensitivity after only half of the sequencing cycles. Reads with a higher number of errors are detected later in the sequencing procedure. More than 99.5 % of all reads with up to eight errors are detected at the end of the sequencing procedure; full sensitivity is achieved for all reads up to three errors.

Simulated dataset CAMI_1. PriLive, BMTagger and DeconSeq were tested on the simulated metagenomic dataset CAMI_1 that contains a mixture of simulated human reads and metagenomic reads of the CAMI study. The human reference genome hg38 was used as background reference. No foreground reference was used. As described for metagenomic datasets (Section 3.2), the parameters -e and --bg-score were set to 0 and 45, respectively. In general, the results on the metagenomic dataset show similar tendencies as for genomic data (CPXV_1). While PriLive achieved the highest sensitivity, BMTagger has the highest specificity. DeconSeq showed intermediate sensitivity and lowest specificity of all tools. The overall results of PriLive - in terms of the F1 score - were better than that of both other tools for the given dataset (Table 3.3). This clearly shows the capability of PriLive to perform at least as good as conventional tools on metagenomic data while - as the only tool - performing in parallel to the sequencing machine.

Table 3.3: Read filtering results for the simulated metagenomic dataset CAMI_1.

	PriLive (%)	**DeconSeq (%)**	**BMTagger (%)**
Sensitivity	**99.98**	99.96	99.94
Specificity	99.9920	99.9903	**99.9958**
F1 score	**99.9614**	99.9417	99.9560

Note: The best value of each row is shown in bold.

Real dataset Venter_1. To ensure that the removal of human reads reliably includes the extinction of identification markers, we used PriLive to remove Y-STR marker regions from the J. Craig Venter sequencing data. This dataset has previously been shown to be retraceable to J. Craig Venter [123]. We converted the Sanger sequencing data (*ftp://ftp.ncbi.nih.gov/pub/TraceDB/Personal_Genomics/Venter/*) to approximately 135 M Illumina-like reads of length 150 bp. On these data, we reproduced the identification workflow. The tool lobSTR [140] was thereby used to determine the Y-STR genotype. With this genotype, a database query was performed on YSearch (*www.ysearch.org*) that provides related surnames and geographical information to the input data. When combined with metadata of the sequencing sample, this can enable the re-identification of the sample originator. We first performed the described workflow with the converted, unfiltered J. Craig Venter sequencing data. We found

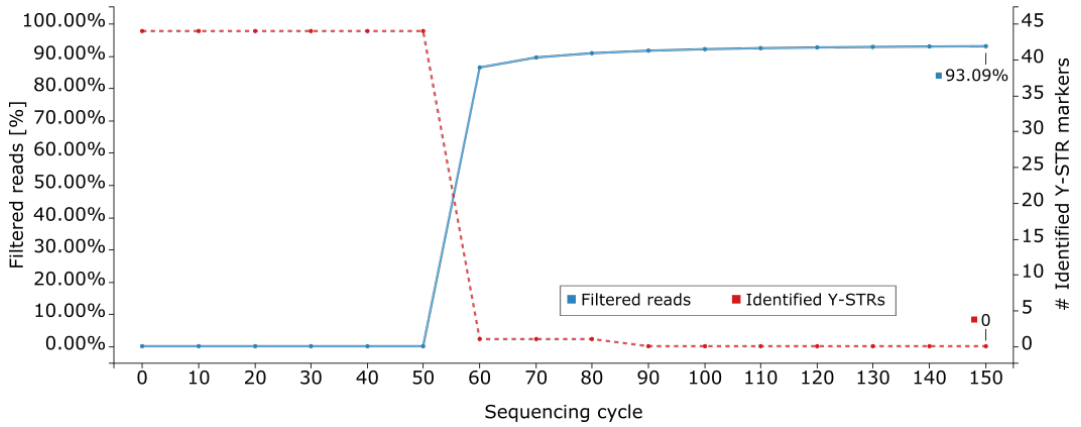


Figure 3.3: Influence of human read removal on the Y-STR marker abundance for the J. Craig Venter sequencing data. The solid blue line shows the relative amount of filtered reads. The dotted red line shows the absolute number of identified Y-STR markers. PriLive was used to remove all human reads from the J. Craig Venter sequencing data. Before the removal of human reads, the Venter entry on YSearch can be successfully found with 44 identified Y-STR markers. When using PriLive to remove all human reads, only 1 single marker remains in the data after only 60 sequencing cycles. After 90 cycles, all markers are removed from the data.

two matches on YSearch that belong to Venter with the Y-STR genotype that was obtained from the sequencing data: 30 of the 44 detected Y-STR markers matched the database entry of Venter which is based on the publications of Gymrek et al. [123] (YSearch User ID: 5BXHS). When compared with the original Venter database entry (YSearch User ID: VPBT4) we observed 29 matching and 2 non-matching markers. As expected, re-identification of J. Craig Venter was no longer possible after using PriLive to filter the entirety of human reads. All Y-STR markers were removed from the data after only 90 sequencing cycles (Figure 3.3). Besides removing all human-related reads we also performed a targeted removal of the Y-STR markers. Therefore, we created an index for PriLive that contains the flanking regions of known Y-STR marker sequences. When we used this index to filter the J. Craig Venter sequencing data, only 0.75 % of all reads were filtered. Thereby, after 70 cycles, half of the markers were already removed from the data. After 110 cycles, there were only six markers left which was no longer sufficient to perform a database query on YSearch. After 130 cycles, it was no longer possible to identify a single Y-STR marker (Figure 3.4). Both filtering procedures, either using the full human reference genome or only the Y-STR marker regions, demonstrate that our privacy-preserving read filtering approach can find and remove relevant identification markers from sequencing data even in early stages of the sequencing

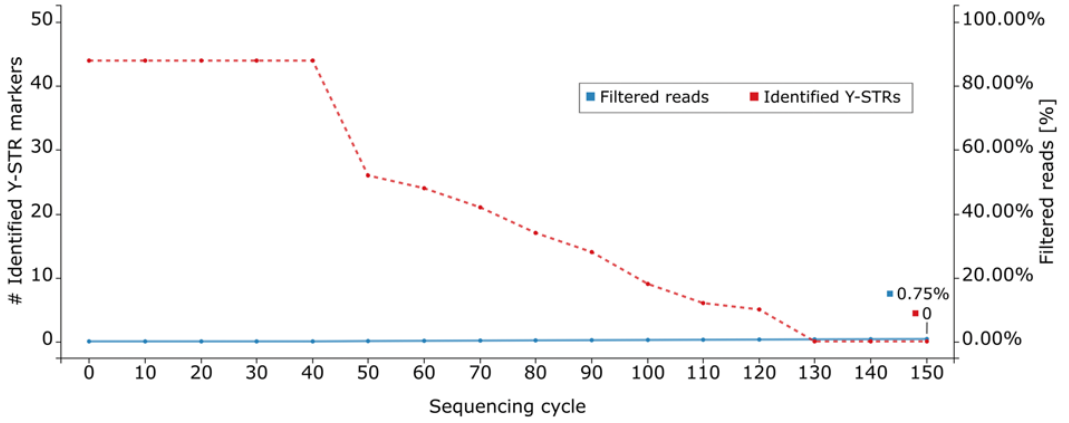


Figure 3.4: Targeted removal of Y-STR markers from the J. Craig Venter sequencing data (Venter_1). The plot shows the relative amount of filtered reads (solid line) and the absolute number of identified Y-STR markers (dotted line) when using PriLive during the sequencing procedure. For the original data, the Venter entry on YSearch can be successfully found with 44 identified Y-STR markers. When using PriLive, after 130 cycles no single marker could be identified from the data. At the end of the sequencing procedure, all Y-STR markers are undetected while more than 99 % of all other reads remain unaffected.

procedure. Thereby, the capability for the targeted removal of a defined set of identification markers also enables the use of PriLive for human datasets.

Real dataset HiSeq_1. We reproduced an in-house sequencing run to verify scalability of PriLive to a real sequencing experiment. The base call files in the input directory of PriLive were created in accordance with the respective time stamps of the original base call files from the sequencing procedure. PriLive was started with 64 threads and used a maximum of 120 GB RAM (allocated) and 190 GB disk space. The maximal delay of PriLive compared to the creation of the base call files was approximately 5 h for the cycles 35 to 40 of the first read. Afterwards the majority of human reads was identified and the algorithm was faster than the sequencing machine. At the end of the first read and for the rest of the sequencing procedure, PriLive operated in parallel to the sequencing machine and therefore only had a delay of a few minutes (Figure 3.5). Thus, data protection was ensured in real-time and the analyses were finished immediately after the sequencing procedure. Additionally, at the same point in time, we obtained the complete alignment output to the foreground reference genome (cowpox virus Brighton Red) in SAM format. From the final results, we determined the number of mapped and filtered reads. The analyzed lane of the sequencing procedure contained approximately 252 M reads in total. With consideration of the internal demultiplexing results,

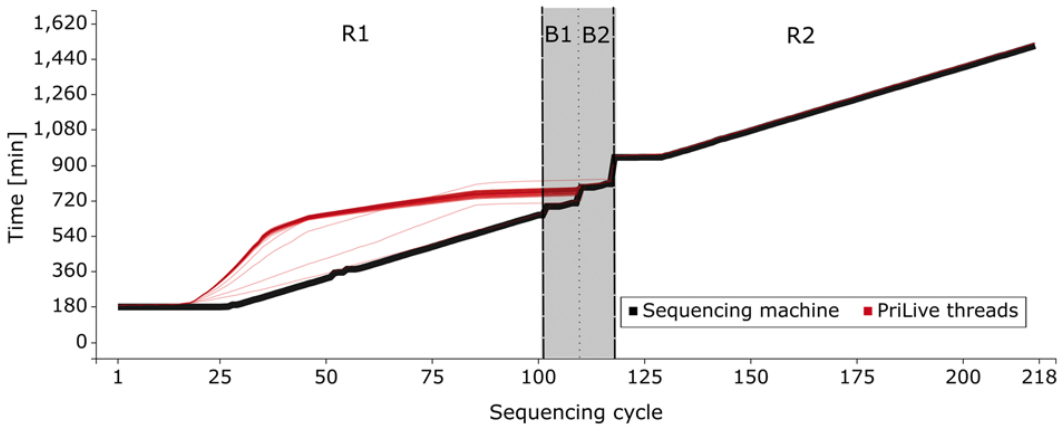


Figure 3.5: Turnaround time of PriLive in a reproduced real-time sequencing scenario. The black line indicates the point in time when the base call files for a cycle are written. The 64 red lines (most of them are bundled) represent the tiles that are analyzed independently from each other by PriLive. Some tiles are analyzed faster than the average because of a bad sequencing quality of the respective data. The first base call files are written by the sequencing machine after cycle 25 which leads to a delay of approximately 3 h from the start of the sequencing procedure for the first cycle. In average, PriLive is slower than the sequencing machine for the cycles 19 – 40 of the first read (R1). This delay is caught up when R1 is completely sequenced. In the middle of the procedure the sequencing machine needs additional initialization time for the barcodes (B1, B2) and the second read (R2). Since most of human reads is already detected after the first read (R1), PriLive is in real-time with the sequencing machine for the complete second read (R2). Also in single-end sequencing, PriLive would have finished immediately after the sequencing machine.

PriLive analyzed more than 155 M of these. Nearly 126 M reads were finally detected as human, for roughly 22 M sequences there was a foreground alignment output for at least one of both reads. The remaining unmapped reads included artifacts, low-quality reads and cowpox virus reads that did not fulfill the selected mapping criteria. All these numbers are in a range as expected for the given experimental setup. At the same time, there was a second, unrelated sample on the same lane of the sequencing procedure. This sample could be identified via a different barcode. It remained unaffected by the read filtering of PriLive as the Illumina demultiplexing output was identical for filtered and unfiltered data. This shows that the usage of PriLive poses no risk of data loss for other samples of the same sequencing run.

3.4 Discussion of Results

PriLive is a novel, powerful tool for data protection in human-related NGS procedures. Conventional tools wait for the sequencing machine to finish and the data to be processed to a human readable file format (e.g., FASTQ). PriLive is, to the best of our knowledge, the first privacy-preserving read filtering software for NGS that operates while the sequencing machine is running. This innovative approach facilitates the compliance with strict data protection guidelines for NGS procedures. Although the actual computation time of PriLive is higher than the runtime of conventional tools, final results can be provided even before other filtering software is started. This reduces the delay of analysis time that exists in conventional read filtering approaches to almost zero.

PriLive supports paired-end reads and live-demultiplexing to identify the data of interest in a mixed sample. The support of foreground reference genomes makes PriLive suitable for genomic and metagenomic applications. For both types of data, PriLive achieves comparable or better results as the conventional tools BMTagger and DeconSeq. By the local alignment approach, PriLive has a higher sensitivity than both tools, especially for high error rates. Specificity is almost 100 % up to a user-defined error rate for the foreground reference genome. Besides this, PriLive also provides high specificity for foreground-related reads above this threshold (> 99.8 %) and for metagenomic applications (> 99.98 %). Thus, there is only a minimal risk of losing relevant information.

Compared to conventional methods, the level of data protection is strongly increased when using PriLive since the sequence information of human reads is not completely available at a single point in time, neither in a human readable file format nor as raw data. In our study, more than 99 % of all human reads were filtered within the first 2/3 of the sequencing cycles. This is a highly relevant benefit since several scenarios for potential violations of data protection guidelines are addressed: attacks from outside and inside, lacks of data protection due to an uncontrolled spread of sensitive data (e.g., through service providers or cooperating institutions) and accidental findings during the analyses. We showed that PriLive reliably identifies reads that enable re-identification of individuals. When performing a targeted removal of the Y-STR markers from the J. Craig Venter sequencing data, PriLive removes relevant information for Y-STR genotyping while only filtering 0.75 % of the reads in total. Besides protecting the privacy of patients, PriLive also simplifies the handling of data for researchers. In conventional workflows they usually have full access to unfiltered data. Each single researcher is therefore responsible for data protection. With PriLive it is

possible to remove sensitive information even before the data is handed over for analysis. Thus, storing, analyzing, sharing and publishing data can be performed with looser data protection restrictions. When established, our strategy facilitates institutional control for a maximum of data protection.

While we showed that PriLive can help to significantly improve data protection, there always remains a trade-off between detecting as many human reads as possible and not losing relevant data. Since even small residues of human data in a sample may allow for re-identification [118, 146], the respective thresholds should be selected with care. Additionally, when PriLive is used to filter specific marker regions as demonstrated for the Y-STR markers of J. Craig Venter, the level of data protection is strongly dependent on the completeness of the selected markers. Therefore, in many cases re-identification may still be possible at present or in future. Depending on the experimental design and the given type of data, it should be considered to couple PriLive with its strength in real-time protection with further strategies such as cryptography.

The combination of high accuracy, a strong level of data protection and a minimal delay in analysis time makes PriLive perfectly suitable for a plethora of applications. This includes, but is not limited to, clinical and research studies, outbreak analyses and precision medicine. Additionally, PriLive can also be applied to use cases apart from data protection, for example the removal of data from expected contaminants, hosts (also other than human) or genomic regions that are not of interest. In combination with additionally provided functionality, e.g., real-time read mapping and demultiplexing, PriLive can speed up a variety of analysis workflows without a notable loss of data quality. This includes that conventional analyses can be performed immediately after the sequencing procedure without an additional filtering step but also that real-time analyses during the sequencing procedure can be further accelerated.

4 PathoLive - Real-time Pathogen Identification from Metagenomic Illumina Datasets

Over the past years, NGS has been applied in pathogen diagnostics with promising results. Yet, long turnaround times have to be accepted as the analysis can only be performed sequentially after sequencing has finished and the interpretation of results can be further complicated by contaminations, clinically irrelevant sequences, and the sheer amount and complexity of the data. We implemented PathoLive, a real-time diagnostics pipeline for the detection of pathogens from clinical samples hours before sequencing has finished. Based on real-time alignment with HiLive2, mappings are scored with respect to common contaminations, low-entropy areas, and sequences of widespread, non-pathogenic organisms. The results are visualized using an interactive taxonomic tree that provides an easily interpretable overview of the relevance of hits. For a human plasma sample that was spiked *in vitro* with six pathogenic viruses, all agents were clearly detected after 40 sequencing cycles. For a real-world sample from Sudan the results clearly indicated the presence of Crimean–Congo hemorrhagic fever virus which was confirmed via PCR. For both samples, clinically irrelevant hits were correctly not highlighted. The results indicate that our approach is valuable to obtain fast and accurate NGS-based pathogen identifications and correctly prioritize and visualize them based on their clinical significance.

4.1 Background

The identification of pathogens directly from patient samples is a major clinical need. While highly accurate pathogen detection methods such as PCR, cell culture, or amplicon sequencing exist, such routine procedures often fail to identify the underlying cause of a patient's symptoms due to their targeted behavior [147–150]. As a complementary approach, metagenomics NGS has been proposed as a valuable technique for clinical application. NGS facilitates the detection and characterization of pathogens without a priori knowledge about candidate species. Further it generates a sufficient amount of data to detect even lowly abundant pathogens without targeted amplification of specified sequences allowing for unbiased diagnostic analysis.

Current tools to address NGS-based pathogen identification can be divided into two major categories, either aiming to discover yet unknown genomes [151–168] or to detect known

species in a sample [49, 169–187]. From an algorithmic perspective, a further distinction can be made between alignment-based methods [49, 151–154, 169–180], alignment-free methods [155–158, 181–184] or combinations of both [159–168, 185, 186]. While alignment-free methods usually deliver faster results, they are mostly limited to the detection of sequences whereas alignment-based methods potentially allow for a more extensive characterization of the sample. Regardless of the algorithmic approach, existing methods based on unbiased metagenomics NGS face various obstacles, especially concerning the ranking of the results according to their clinical relevance and the long overall turnaround time [188–195]. The lack of good ranking methods is based on the fact that the distinction of clinically relevant and irrelevant data is not trivial. First, the dominating part of the sequences in a metagenomic patient sample usually originates from the host genome. Second, there are nucleic acids of various clinically irrelevant species such as endogenous retroviruses (ERVs) or non-pathogenic bacteria which commonly colonize a person. For these reasons, the number of reads hinting towards a relevant pathogen can be very limited and even be as low as a handful of individual reads. To put it more generally, it is a widespread misconception to rely only on quantitative measures when ranking the importance of candidate hits as not the amount but the uncommonness of a species in a given sample may give critical indications on its relevance. Based on the premise that a large proportion of the produced reads may stem from the host genome, species irrelevant for diagnosis, or common contaminations, even highly accurate methods struggle with false positive hits potentially concealing the relevant results. This central problem is getting even worse when considering that even microbial databases are contaminated with human sequences [196]. Existing pipelines tackle this problem in different ways. One common strategy is to ignore sequences that occur in a reference database of host and contaminating sequences [157, 162, 177, 180, 181, 185]. While facilitating cleaner results, it may lead to a premature rejection of relevant sequences and does not solve the problem of human contaminations in reference databases as those "derive primarily from high-copy human repeat regions, which themselves are not adequately represented in the current human reference genome" [196]. Thus, such sequences cannot be filtered based on a host database and could be falsely classified in the consequent workflow. Further, the definition of precise contamination databases proves rather difficult and has not yet been adequately solved. Thus deleting any results to gain a better overview comes at great risk of overlooking the true cause of an infection. A different strategy are intensity filters, as implemented, for example, in SLIMM [171], that disregard sequences with low genome coverage. As the author states, this step eliminates many genomes which introduces

the risk of losing information that might be relevant in the following diagnostic process. This problem even intensifies for marker-gene based methods such as MetaPhlAn2 [175], as large parts of the sequenced reads cannot be assigned due to the miniaturized reference database. While this may lead to a better ratio of seemingly relevant assigned reads to those from the background, it comes with the risk of disregarding relevant candidates. Another fundamental problem of NGS-based pathogen identification approaches is the fact that sequencing and analysis is very time consuming. Even when considering the enormous reduction of sequencing time in the last years, current mid- and high-throughput devices still have maximum runtimes of more than a day (NextSeq 550) and up to two (NovaSeq 6000) or three days (HiSeq X), respectively. The resulting turnaround times of two to four days including data processing and analysis are not short enough for many critical scenarios such as sepsis, postoperative and other life-threatening infections and infectious disease outbreaks. To obtain actionable results within an appropriate time frame, it is crucial to reduce the time span of the entire workflow from sample receipt to complete diagnosis. However, existing approaches to speed up NGS-based diagnostics come with significant disadvantages such as a highly reduced throughput and data quality [197], massive reduction of analyzed reads or targets [97] or the need of specialized hardware that involves additional costs and a low flexibility to adapt the workflow to a given scenario [96].

As a general complement to real-time analysis of short-read sequencing data, there are several promising studies for pathogen detection using the MinION handheld device which is particularly useful for field studies and produces longer reads of up to several hundred kilobase pairs. While allowing very fast throughput times, these devices yield only approximately a million reads with comparably low per-base qualities, limiting their areas of application to targeted sequencing so far [116, 197–200]. Therefore, from today's perspective, NGS is the only technology providing sufficient amount and quality of data for many applications in clinical diagnostics. The currently high turnaround times from sample arrival to final diagnosis make it necessary to develop efficient methods to generate, analyze, and understand large metagenomics datasets in an accurate and quick manner to pave the way for NGS as a standard tool for clinical diagnostics. This enforces NGS-based diagnostics workflows to generate and evaluate large numbers of reads to facilitate adequate sequencing depths while at the same time reducing the time span between sample receipt and diagnosis. To overcome the named obstacles, we present PathoLive, an NGS-based real-time pathogen detection tool. We present an innovative approach to handle the occurrence of common contaminations, background data and irrelevant species in a single step. To tackle the problem of long overall

turnaround times, we based our novel approach on the real-time read mapper HiLive2 that enables the analysis of sequencing data while an Illumina sequencer is still running [201]. This enables PathoLive to perform nucleotide-level analysis based on NGS providing an open view and high accuracy in short turnaround times while generating an intuitive and interactive visualization of results that highlights organisms of high clinical significance.

4.2 Methods

Implementation

In order to generate a quick, easy and robust pathogen diagnostics workflow, we implemented PathoLive. Our workflow follows a different paradigm than other frameworks to tackle the existing problems, as shown in Figure 4.1: **(i)** prepare informative, well defined reference databases, **(ii)** automatically define contaminating or non-pathogenic sequences beforehand, **(iii)** use HiLive2 for accurate real-time alignment of Illumina sequencing data, **(iv)** visualize the potential risk of candidate pathogens and present results in an intuitive, comprehensible manner. The details for each of these steps are provided in the following sections.

(i) Preparation of Reference Databases

In order to save computational effort during the post-processing of the real-time aligned reads, reference databases including the full taxonomic lineage of organisms are prepared before the first execution of PathoLive. For this purpose user selectable databases, for example the RefSeq Genomic Database [202], are downloaded from the file transfer protocol (FTP) servers of the NCBI and annotated accordingly with taxonomic information from the NCBI Taxonomy Database. While preserving the original NCBI annotation of each sequence, additional information is appended to the sequence header. This information consists of each taxonomic identifier (TaxID), rank and name of each taxon in the lineage of the respective organism. Afterwards, user definable sub-databases of taxonomic clades relevant for a distinct pathogen search are automatically created. The database updater used for this purpose is available at *https://gitlab.com/rki_bioinformatics/database-updater*. The viral database used in this manuscript can be downloaded as a single compressed FASTA file from Zenodo (*https://doi.org/10.5281/zenodo.2536788*) and is ready to use for viral diagnostics with PathoLive.

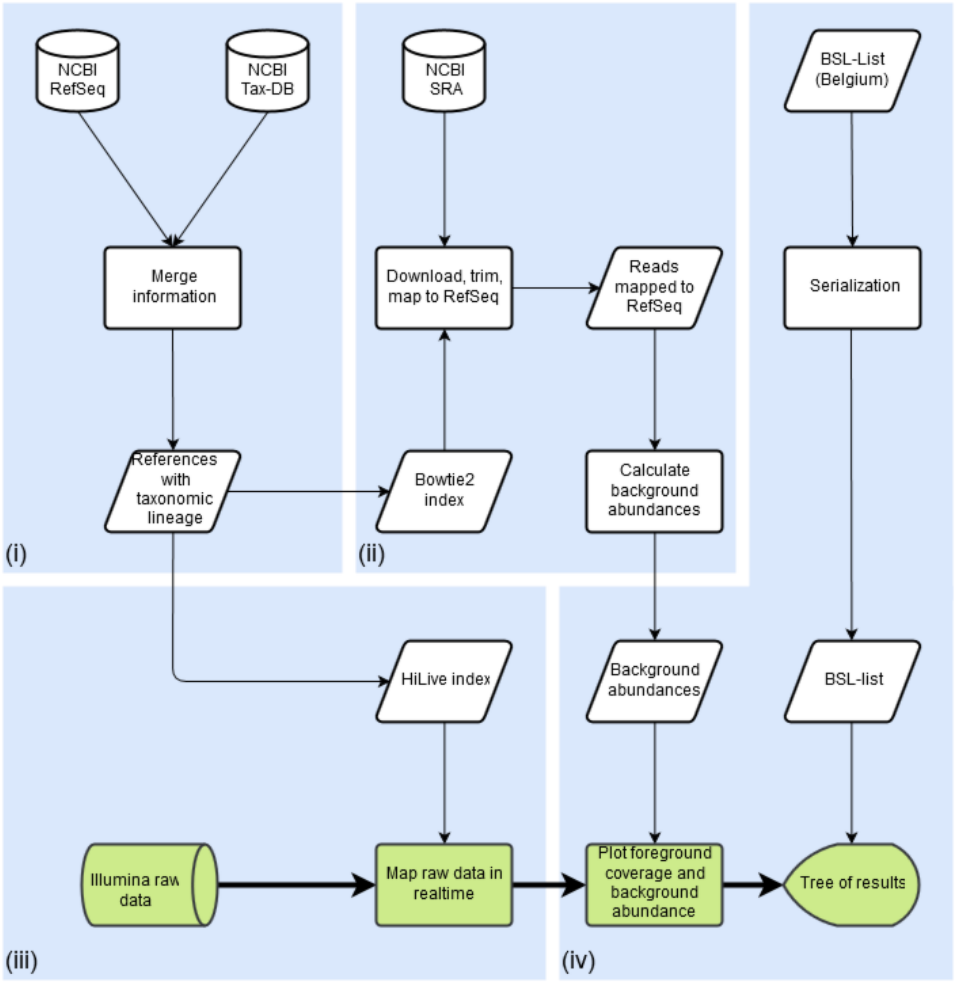


Figure 4.1: Workflow of PathoLive including four main modules. **(i)** Reference information from NCBI RefSeq is automatically downloaded and tagged with taxonomic information; **(ii)** NGS datasets from the 1000 Genomes Project are downloaded, trimmed and searched for sequences from the pathogen database from step (i), marking abundant stretches as clinically irrelevant; **(iii)** Reads from the clinical sample are mapped to the pathogen database obtained from (i) in real-time, producing intermediate alignment files in BAM format at predefined time points; **(iv)** results are visualized in an easily understandable manner, providing all available information while pointing to the most relevant results. Only the steps highlighted in green are calculated in execution time, steps in white are precomputation. Graphical results are presented only minutes after the sequencer finishes a cycle if desired.

(ii) Identification and Labeling of Clinically Irrelevant Hits

A main obstacle in NGS-based diagnostics is the large amount of background noise contained in the data. This includes various sources of contamination such as artificial sequences, ambiguous references and clinically irrelevant species, which hinder a quick evaluation of a dataset. Defining an exhaustive set of possible contaminations is a yet unachieved goal. Furthermore, deleting those sequences defined as irrelevant from the set of references carries the risk of losing ambiguous but relevant results. Since in this step, raw sequencing data from a human host is examined, the logical conclusion is to contrast it to comparable raw datasets instead of processed genomes. We implemented a method to define and mark all kinds of undesired signals on the basis of comparable datasets from freely available resources. For this purpose, raw data from 236 randomly selected datasets from the 1000 Genomes Project Phase 3 [203] were downloaded, assuming that a large majority of the participants in the 1000 Genomes Project were not acutely ill with an infectious disease. These reads are quality trimmed using Trimmomatic [204] and mapped to the selected pathogen reference database

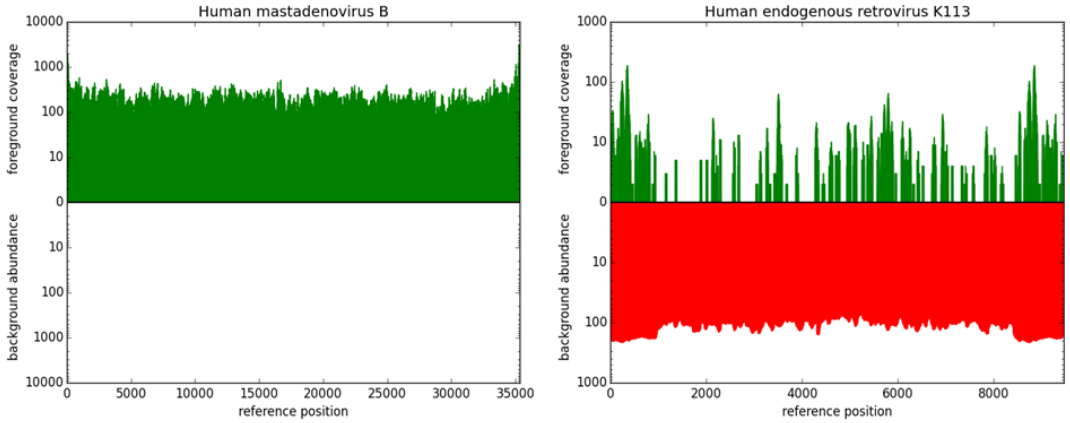


Figure 4.2: Two examples of fore- and background coverage plots. The upper, green bars show the coverage of a given genome in the foreground dataset, namely the reads sequenced from the patient sample. The lower, red part indicates in how many datasets from the 1000 Genomes Project a sequence is abundant. Bases covered in background datasets are regarded as less informative. Left: Fully covered genome of human mastadenovirus B, showing no hits resulting from data from the 1000 Genomes Project. Right: Coverage of human endogenous retrovirus (HERV) K113, partly covered in the patient dataset and completely covered in $\sim$ 110 datasets from the 1000 Genomes Project. Based on these illustrations, Human mastadenovirus B can be considered a relevant hit while HERV K113 is rightly found in the dataset, but not considered a clinically relevant candidate due to its common prevalence in healthy human individuals.

using Bowtie 2 [85]. Whenever a stretch of a sequence is covered once or more in a dataset from the 1000 Genomes Project, the overall background coverage of these bases is increased by one. Coverage maps of all references from the pathogen database hit at least by one dataset are stored in the serialized pickle file format. Stretches of DNA found in this data are marked as of lower clinical significance and visualized as such in later steps of the workflow. The coverage maps of the background abundances are thereto plotted in red color against the coverage maps of the reads from the patient dataset in green color on the same reference (Figure 4.2). This enables highlighting presumably relevant results without discarding other candidate pathogens, providing the best options to interpret the results in-depth but still in an efficient manner. The code for the generation of these databases is part of PathoLive.

(iii) Using HiLive2 for Real-time Alignment of Reads

We used HiLive2 (version 2.1) to produce real-time alignments of sequencing data. Thereby, the sequencing data is directly loaded in raw BCL format without the need to perform a file conversion step. Alignments are updated with each new sequencing cycle and output in BAM format can be created for any sequencing cycle. As changes in the mapping positions mainly occur in early sequencing cycles, we recommend to create output in shorter intervals at the beginning of sequencing. Options for integrated real-time demultiplexing and adapter trimming are available. For algorithmic details of HiLive2, see Section 2.2.

(iv) Visualization and Hazardousness Classification

A key hurdle in a rapid diagnostics workflow, which is often underestimated, is the presentation of results in an intuitive way. Many promising efforts have been made by different tools, e.g., providing coverage plots [180, 205] or interactive taxonomy explorers [152, 181]. While being hard to measure and thus often ignored, the time it takes for groups of experts to assess the results and come to a correct conclusion should be considered. Our browser-based, interactive visualization is implemented in JavaScript using the data visualization library D^3 [206]. For an example of the visualization, see Figure 4.4 in the results section. While providing all available information on demand, the structure of a taxonomic tree allows an intuitive overview at first glance. Detailed measures are available on genus, family, species and sequence level. For the calculation of scores for a given node n, we define $t(n)$ as the total number of read alignments to an underlying species of n. $b(n)$ is the total number of bases being covered by all reads with respect to n. Accordingly, $b_{bg}(n)$ describes the number of bases being covered by the background database and $b_{fg \setminus bg}(n)$ is the number of bases

being covered by the foreground but not by the background data. In total, we provide three different scores for each node n of the tree:

(a) Total Hits T_n, being the total number of hits to all underlying sequences in this branch:

$$T_n := t(n)$$

(b) Unambiguous Bases U_n, representing the total number of bases covered in the foreground data but not in any background dataset:

$$U_n := b_{fg \setminus bg}(n)$$

(c) Weighted Score W_n, being the ratio of unambiguous bases for the foreground data to the number of bases covered by the background database and logarithmically weighted by the total number of alignments:

$$W_n := \frac{U_n}{max(b_{bg}(n), 1)} \cdot log(T_n)$$

While the Total Hits T_n can be useful to get a general impression of the abundance of sequences in the sample, the Unambiguous Bases U_n provides a first comparison to the background dataset. The Weighted Score W_n introduces an intensified metric of how often a sequence is found in a healthy individual, and thereby allows drawing stricter conclusions from the background data. Not only exactly overlapping mappings of fore- and background are regarded, but also the overall abundance of a sequence within the background data is considered. The values of the selected scoring scheme are reflected in the thickness of the branches, which draws the visual focus to higher rated branches. Users can switch between the three scores via the respective buttons in the interactive visualization. In order to enable users to make early decisions regarding the handling of a sample as well as to further enhance the intuitive understanding of the results, the hazardousness of detected pathogens is color-coded based on a biosafety level (BSL) score list [207]. To improve BSL classification, minor changes were manually applied to improve matches to the organism names in the reference database. The BSL score gives information on the biological risk emanating from an organism. Therefore, it qualifies as a measure of hazardousness in this use case. The BSL-score is color-coded in green (no information / BSL-1), blue (BSL-2), yellow (BSL-3) or red (BSL-4), and the maximum hazardousness-level of a branch is propagated to the parent nodes. Phages are displayed in grey, as they cannot infect humans directly, but may imply

information on the presence of bacteria. Details about the sums of all three available scores of all underlying species are provided on mouse-over (see Figure 4.4 in the results section). When expanding a branch to sequence level, additional plots of the foreground coverage calculated in step (iii) as well as the abundance of bases in the background datasets calculated in step (ii) are shown when hovering the mouse over the node (Figure 4.2). These plots thus provide an intuitive visualization of the significance of a hit. The hits of a species in the patient dataset are shown in green while background hits are drawn in red on a correlating coverage plot. This way, it is easy to evaluate if a sequence is commonly found in non-ill humans and therefore can be considered less relevant, or if a detected sequence is very unique and could therefore lead to more certain conclusions.

Validation

We compared the results of PathoLive to two existing solutions, Clinical Pathoscope [177] and Bracken [208]. We selected Clinical Pathoscope for its very sophisticated read reassignment method, which promises a highly reliable rating of candidate hits. It also is perfectly tailored to this use case. Other promising pipelines such as SURPI [180] or Taxonomer [181] were not locally installable and had to be disregarded. Bracken, a method based on metagenomics classification with Kraken [184], was included in the benchmark as one of the fastest and best known classification tools which makes it one of the primary go-to methods for many users. The experiment is based on a real sequencing run on an Illumina HiSeq 1500 in High Output Mode. We designed an in-house generated sample in order to have a solid ground truth. We ran all tools using 40 threads, starting each at the earliest possible time point when the data was available from the sequencer in the expected input format. For the non-real-time tools, the base calling was executed via Illumina's standard tool bcl2fastq and the runtime was regarded in the overall turnaround time. Clinical Pathoscope and Bracken were both run with default parameters, apart from the multithreading. The reference database for PathoLive was built from the viral part of the NCBI RefSeq [209] downloaded on 2016-07-06. For Clinical Pathoscope we downloaded the associated database from *http://www.bu.edu/jlab/wp-assets/databases.tar.gz* on 2017-12-09 and used the provided viral database as foreground and the human database as background. The results of Bracken were generated based on the viral part of the NCBI RefSeq downloaded on 2017-12-18. The Bracken database was generated with default parameters and an expected read length of 100 bp. Please note that, in contrast to all other results shown in this manuscript, the live

analysis of the in-house sample was performed using read-mapping results of HiLive, the predecessor of HiLive2. However, we repeated the analysis using HiLive2 and obtained similar results with respect to accuracy (Appendix Figure A2.1 and Appendix Table A2.1).

To validate the functionality of PathoLive on real data, we applied it to a diagnostic human serum sample from an outbreak of hemorrhagic fever virus in Sudan in 2014 [210]. The sequencing data of this sample was only available in preprocessed FASTQ format. To convert the available FASTQ files back to BCL format, trimmed reads were extended to the required length with calls of the ambiguous nucleotide N. In general, this procedure could influence the results by introducing random hits. However, we observed that most reads still contained the adapter sequences. As all sequence information after a detected sequencing adapter is ignored from analysis when using the adapter trimming functionality of HiLive2, the applied procedure during format conversion should not have any significant effect on the results in this special case when compared to data directly coming from the sequencing machine. Still, even for reads missing an adapter sequence, this would lead to a decrease of aligned reads which further hampers the identification of pathogens and therefore does not limit the validity of the final results. The conversion step from FASTQ to BCL format itself was done by concatenating each read pair in a single FASTQ file and execution of the fastq2bcl script which is delivered with HiLive2. The total length of all reads was 2 x 301 bp, corresponding to a total of 602 sequencing cycles (multiplex barcodes were not included).

Sample Preparation

Viral RNA metagenomics studies were performed with a human plasma mix of six different RNA and DNA viruses as well-defined surrogate for clinical liquid specimen. The informed consent of the patient has been obtained. This 200 μL mix contained orthopoxvirus (Vaccinia virus VR-1536), flavivirus (yellow fever virus 17D vaccine), paramyxovirus (mumps virus vaccine), bunyavirus (rift valley fever virus MP12-vaccine), reovirus (T3 / Bat / Germany / 342 / 08) and adenovirus (human adenovirus 4) from cell culture supernatant at different concentrations. The sample also contains dependoparvovirus as proven via PCR. The sample was filtered through a 0.45 μM filter and nucleic acids were extracted using the Qiagen QIAamp Ultrasense Kit following the manufacturers' instructions. The extract was treated with Turbo DNA (Life Technologies, Darmstadt, Germany). The complementary DNA (cDNA) and double-stranded cDNA (ds-cDNA) synthesis were performed as previously described [211]. The ds-cDNA was purified with the Qiagen RNeasy MinElute Cleanup Kit.

The purification method takes ~ 6 h to complete. The Library preparation was performed with the Nextera XT DNA Sample Preparation Kit following the manufacturers' instructions. NGS libraries were quantified using the KAPA Library Quantification Kits for Illumina sequencing. If the starting amount of 1 ng of nucleic acid was not reached the entire sample volume was added to the library. The diagnostic sample from Sudan was prepared according to [210, 212], including inactivation of the human serum in Qiagen Buffer AVL, extraction with Qiagen QIAamp Viral RNA Mini Kit and DNA digestion using the Thermo Fisher TURBO DNA-free Kit. A sequencing library was created using the Illumina Nextera XT DNA Library Preparation Kit. The sample was sequenced on an Illumina MiSeq.

4.3 Results

Pathogen Detection in a Spiked Viral Mixture

The human plasma sample spiked with a viral mixture was subjected to sequencing on an Illumina HiSeq 1500 in High Output mode on one lane. PathoLive was executed from the beginning of the sequencing run using 40 threads. Results were produced after 40, 60, 80 and 100 cycles or after 36, 55, 74 and 93 hours, respectively. Raw reads usable for the testing of other tools were available only after 95 hours as they had to be converted to FASTQ format first. As a ground truth, we selected all sequences associated to the species described as abundant above. Turnaround time, runtime and results are shown in Table 4.1. The area

Table 4.1: Results of PathoLive, Clinical Pathoscope and Bracken on an Illumina HiSeq High Output run of a human plasma sample spiked with different viruses. Input data denotes the number of cycles the sequencer finished before results were generated. The turnaround time specifies total time passed from the start of the sequencer to result presentation, whereas tool runtime is the time taken to generate results from the input data. ROC-AUC denotes the area under the ROC-curve as a combined measure of TPR and FPR. Best values are printed bold.

	PathoLive				Pathoscope	Bracken
Input data [cycle]	40	60	80	100	100	100
Turnaround time [h]	**36**	55	74	93	95	95
Tool runtime [m]	22	25	18	**4**	25	13
ROC-AUC	**0.94**	0.92	0.92	0.90	0.88	0.45

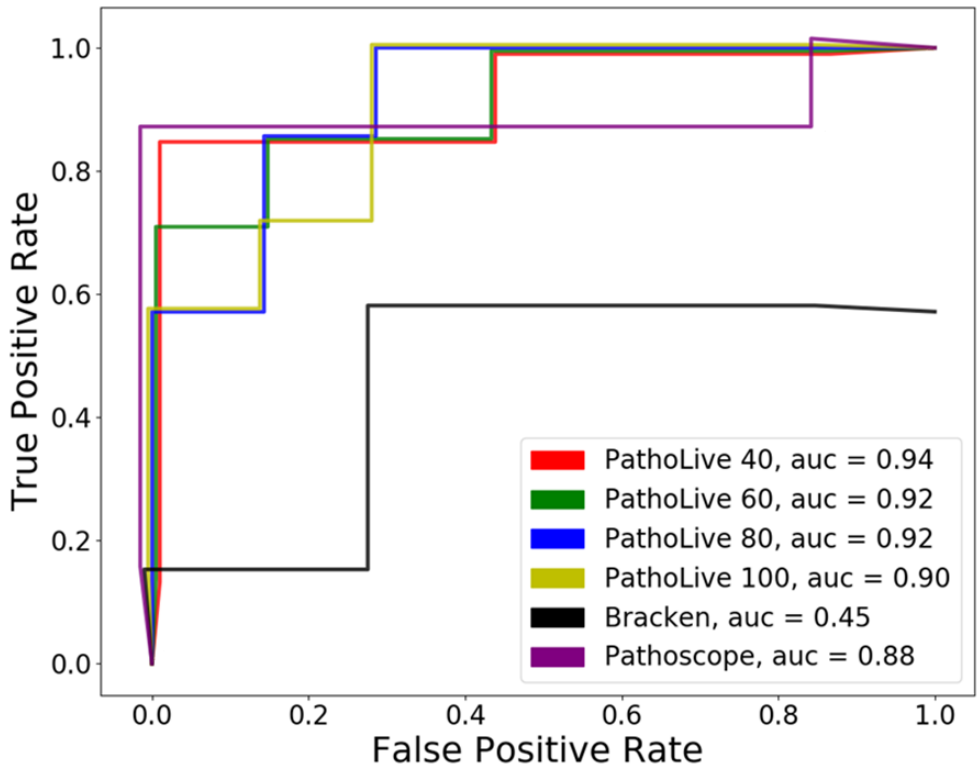


Figure 4.3: ROC-plot of benchmarked tools on a spiked dataset. Lines have slight offsets in x- and y-dimensions for reasons of distinguishability. We compared PathoLive to Clinical Pathoscope and Bracken on a real human sample containing 7 viruses. PathoLive performs best regarding the ROC-AUC at all sampled times (cycle 40, 60, 80 and 100) when compared to the results of the other tools after the sequencing run completed read 1 (cycle 100).

under the curve (AUC) of the receiver operating characteristic (ROC) was calculated using the 14 highest ranking species, as given by the tested tools. The top 14 of the identified species are considered because hits appearing after twice the number of true positives cannot be expected to be regarded by a user in this experiment. Furthermore, none of the tested tools found more true positives within the next 50 hits. The ROC-plot (Figure 4.3) denotes the true positive rate (TPR) and false positive rate (FPR) for each threshold $n \leq 14$, whereby a threshold n means that the best n hits are taken into account. This means that only the rank of the hits was considered while disregarding the actual score. For PathoLive, the ranks were determined by the Weighted Score W_n, for Clinical Pathoscope we used the 'final guess' metric and for Bracken, the species with most estimated reads were ranked highest.

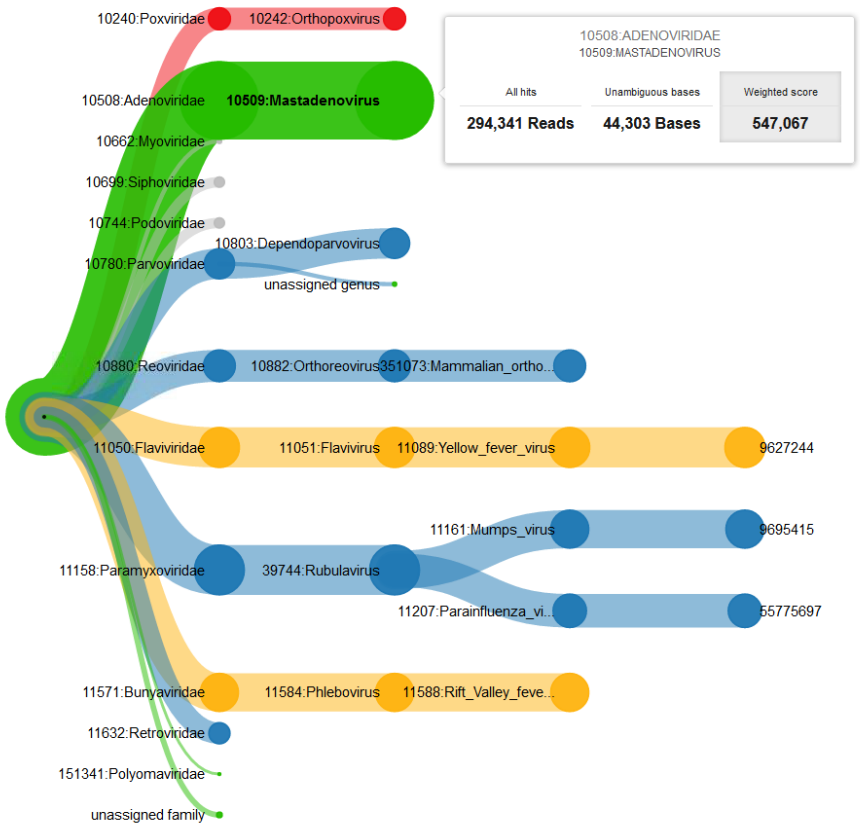


Figure 4.4: Example of the interactive taxonomic tree of results. It shows the visualized results of the described plasma sample at cycle 80 based on the weighted score. Thickness of the branches denotes the sum of scores of underlying sequences. The color codes for the maximum of the underlying BSL-levels (red=4, yellow=3, blue=2, green=1 or undefined; phages are shown in grey). On mouse-over, detailed information (here on genus Mastadenovirus) is displayed. The selected score (here: weighted score) is highlighted in grey. The visualization clearly highlights all spiked pathogens through the thickness of their clades, while other species are shown only in smaller clades and therefore ranked lower.

We were able to detect all abundant spiked species in the library after only 40 cycles of the sequencing run using PathoLive. While the overall number of false positive hits decreases with the sequencing time, the weighted score and the number of unambiguous bases yield accurate results throughout all reports. Reported phages are included in these numbers, although they are optically grayed out in the visualization, as they cannot infect vertebrates directly. As an example report, a screenshot of the resulting interactive tree of results after 80 cycles is shown in Figure 4.4.

Identification of Crimean-Congo Hemorrhagic Fever Virus in a Real Sample from Sudan

A central issue in pathogen identification, especially for viruses, is the potentially low number of pathogenic reads in the sample. Therefore, we demonstrated the performance of PathoLive on real data that is known to contain a low number of reads of interest. We analyzed a human serum sample from Sudan that was confirmed via PCR to contain Crimean–Congo hemorrhagic fever virus (CCHFV) but only shows a small amount of related reads in the corresponding Illumina sequencing data (45 out of 1,178,054 reads were reported by Andrusch et al. [212] to unambiguously belong to CCHFV). When running PathoLive with default parameters and having adapter trimming activated, Bunyaviridae was the family with the highest weighted score over the complete sequencing procedure when not considering phages and the 'unassigned family' branch. Thereby, the score of Bunyaviridae was consistently equal to the score of the underlying species CCHFV while other underlying species did not contribute to the overall score of the family. Figure 4.5 shows the development over time for all families that reach a score of 500 in at least one output cycle. It can be seen that the weighted score of CCHFV (represented by the family of

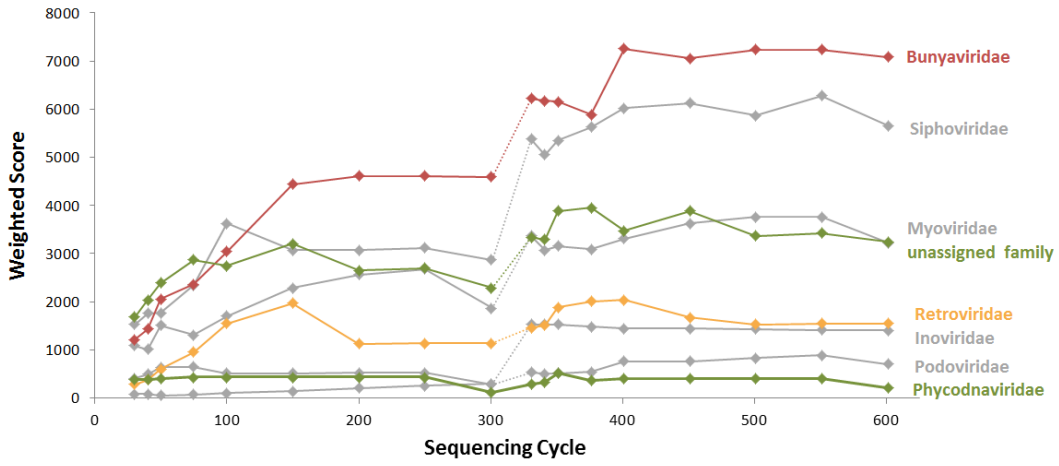


Figure 4.5: Development of the weighted score calculated by PathoLive over the sequencing procedure for all families reaching a score higher than 500 in at least one output cycle. Colors of the plots correspond to the underlying biosafety level in the last cycle, i.e., green for BSL-1, blue for BSL-2 and yellow for BSL-3. Phages are displayed in gray color. The dotted section of each line indicates the shift from the first to the second read of the 2 x 301 bp data.

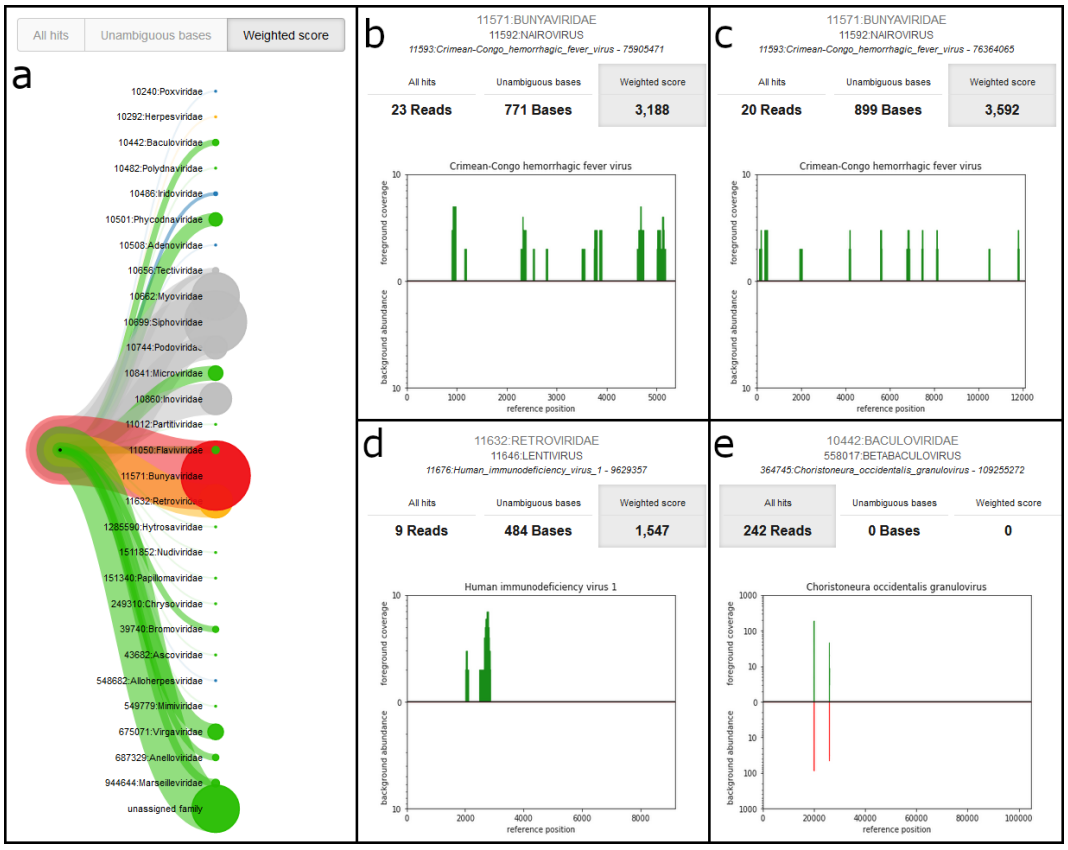


Figure 4.6: Visualization of the final results of PathoLive for cycle 602. **(a)** Tree structure on family level. **(b,c)** Tooltips for the sequence level of alignments for two CCHFV reference sequences of the Bunyaviridae family. **(d)** Tooltip for the sequence level of alignments for HIV 1 of the Retroviridae family. **(e)** Tooltip for the sequence level of alignments for a Granulovirus reference of the Baculoviridae family.

Bunyaviridae) is in the top three of all identified families after only 30 sequencing cycles which corresponds to 5 % of the sequencing procedure. At this time point, only 16 reads were aligned to CCHFV. Thus, indications for the correct finding are already possible within a short time span and based on only a couple of available reads while the result is more and more emphasized with ongoing sequencing. The only other family reaching a score higher than 500 and not exclusively containing phages was Retroviridae, being mainly driven by the species HIV 1. However, a more detailed view on the sequence level shows that all mappings to HIV 1 cluster in a small region of approximately 1,000 bp (Figure 4.6 d) while the alignments to CCHFV distribute over the complete genome (Figure 4.6 b,c). This

strongly indicates that CCHFV is more likely to be a true positive. Figure 4.6 further shows the family level visualization of the PathoLive tree structure (Figure 4.6 a) and an example for Granulovirus of the Baculoviridae family that shows a high total number of mappings, but all of those being located in regions that are covered in the background database leading to a weighted score of 0 (Figure 4.6 e). The overall results for this sample show the strength of PathoLive to pronounce interesting findings at first glance while still allowing for a more detailed perspective that is often important for interpretation.

4.4 Discussion of Results

NGS has been shown to be state of the art for pathogen detection, reaching out into clinical usage as well. Although TGS approaches are also becoming more and more influential, the discovery of lowly abundant pathogens is still problematic due to the relatively low number of reads. Additionally, the comparably low coverage and high error rates still hamper certain types of complex follow-up analyses such as the detection of antimicrobial resistances or the geographical origin of a pathogen. On the other hand, long-read sequencing technology show an immense potential for real-time diagnostics in the future, especially when considering the continuously decreasing error rates, shorter sample preparation times, arising higher throughput devices such as the PromethION, as well as valuable technology-specific features such as the read until functionality for that first attempts have been made to separate microbial reads from host DNA during the sequencing procedure [199, 213]. All these aspects considered we assume long-read sequencing technology a valuable complement to NGS-based diagnostics in future with distinct properties and therefore potentially different application areas.

The high turnaround time of NGS-based diagnostics is a major drawback compared to targeted molecular methods. Past efforts to speed up NGS-based diagnostics have been made but often come with significant disadvantages: Quick et al. introduced a fast sequencing protocol for Illumina sequencers that allows obtaining results after as little as 6 hours [197]. This speedup is accompanied by lower throughput and lower data quality, making it less suitable for whole genome shotgun sequencing approaches without a priori knowledge. Other approaches aiming at performing analyses of intermediate sequencing data, such as presented by Stranneheim et al. [97] and Miller et al. [96], require either a massive reduction of the amount of analyzed reads or targets [97] or the application of specialized hardware such as FPGAs which are, for example, used for the DRAGEN system [96]. Such specialized

hardware approaches come with additional costs, either for purchase and infrastructure of local solutions or for the use of a cloud system. At the same time, such approaches provide a low level of flexibility in the analysis and are not algorithmically optimized for working with incomplete data. PathoLive does not require the use of specialized hardware and provides accurate diagnostics results in real-time, illustrated with an easily understandable and interactive visualization. This strongly facilitates to get insights into a clinical sample before the sequencer has finished. Real-time output before the sequencing process of the first read has finished lacks information about multiplex indices, though. Therefore, early results of multiplexed sequencing runs can only be assigned to a specific sample after sequencing of the multiplex-indices. For paired-end sequencing runs, this still means analyses are still possible far before the sequencer ends, and single-end sequencing runs can produce results at the very moment the indices have been sequenced. A solution for this problem would be to sequence the indices before the first read, which attracts some problems for the sequencer regarding cluster identification, but is currently worked on. The algorithmic functionality for this is already available. As a working solution, many sequencing devices allow paired-end sequencing with different lengths for the first and second reads. It is thereby possible to sequence only a short fragment of the first read to get early access to the multiplex indices. Thus, this approach can be used to obtain *de facto* single-end reads (i.e., the full second read) while having the multiplex information available from the beginning of the read.

For pathogen identification, we changed the basis for the selection of clinically relevant hits from pure abundance or coverage-based measures towards a metric that takes information on the singularity of a detected pathogen into account. Still, we decided not to completely trust the algorithmic evaluation alone, but provide all available information to the user in an intuitive interactive taxonomic tree. While we assume that this form of presentation allows users to come to the right conclusions very quickly, more sophisticated methods for the abundance estimation especially on strain level exist. Implementing an additional abundance estimation approach comparable to the read reassignment of Clinical Pathoscope [177] or the abundance estimation of Bracken [208] could enable more accurate results, albeit this would not be applicable trivially to the overall conception of PathoLive.

The sensitivity and specificity of PathoLive varies with the time of a sequencing run. In the beginning, when only little sequence information is available, only a small number of nucleotides specify a candidate hit, leading to comparably high false positive rates. At the end of a sequencing run on the contrary, the number of sequence mismatches in the longer alignments may lead to the erroneous exclusion of hits, especially when sequencing quality

decreases. However, this behavior is implicitly considered by the HiLive2 algorithm which allows for an increasing number of mismatching nucleotides with increasing length of the reads. Still, the results can vary over runtime with the optimal outcome being measured at intermediate cycles if the alignment parameters are not well-suited for the specific sample or if the sequencing quality decreases stronger than expected.

Besides these challenges which are unique to PathoLive, we do also struggle with similar problems as conventional approaches. First, the definition of meaningful reference databases is difficult. No reference database can ever be exhaustive since not all existing organisms have been sequenced yet. Besides that, there may be erroneous information in the reference databases due to sequencing artifacts, contaminations or false taxonomic assignment. The definition of the hazardousness was especially complicated, as to our knowledge no well-established solution for the automated assignment of this information exists. Therefore, the basis for our BSL-leveling approach might not be exhaustive, leading to underestimated danger levels of pathogens that are missing in the list of organisms with a BSL of at least 2. Furthermore, in-house contaminations, some of which are known to be carried over from run to run on the sequencer while others may come from the lab, could interfere with the result interpretation of a sequencing run. Especially since no indices are sequenced for the first results of PathoLive, comparably large numbers of carry-over contaminations might lead to false conclusions. Candidate lab contaminations should therefore be thoroughly kept in mind when interpreting results.

Using in-house generated spiked human plasma samples, we were able to show the advantages of PathoLive not only concerning its unprecedented runtime but also the selection of relevant pathogens. Additionally, we also show the high sensitivity of our approach by identifying CCHFV in a real sample from Sudan based on only a few dozens of reads. While being very fast and accurate, a limitation of PathoLive lies in the discovery of yet unknown pathogens. This is due to the limited sensitivity of alignment-based methods in general, which hampers the correct assignment of highly deviant sequences. As this would imply tedious manual curation, it is not the core task of this tool.

Concluding, PathoLive is a helpful tool for accurate and yet rapid detection of pathogens in clinical NGS datasets. The key advantages are the real-time availability of analysis results as well as the intuitive and interactive visualization with down-prioritization of likely irrelevant candidates.

5 Summary and Outlook

5.1 Summary

NGS is currently the state-of-the-art technology when it comes to nucleotide-level DNA sequence analysis. It can be applied in a broad range of applications including, but not limited to, the detection of (rare) genetic diseases and heterogeneous inherited disorders [214–219], cancer classification and identification of therapeutic targets in tumor therapy [220–224] and the identification of disease-causing organisms and characterization of pathogens, especially with respect to potential antimicrobial resistances [225–227]. When compared to other methods, such as molecular testing or PCR, NGS provides a more open view and can be applied for untargeted analyses. In contrast, alternative methods are usually specific to a single biomarker and therefore require presumptions, even when combining multiple tests. However, while providing the highest throughput on the market, high-quality results and comparably low costs, the turnaround time of NGS-based approaches from sample arrival to interpretable analysis outcome is still critically high when considering life-threatening scenarios and public health threats such as infectious disease outbreaks. Several approaches for speeding up NGS analysis workflows have already been proposed, especially in the field of clinical diagnostics. This includes commercially available software such as the DRAGEN system [96] and Sentieon solutions for secondary DNA analysis [114]. However, many of these approaches rely on the utilization of FPGAs, are optimized for the analysis of human samples and only provide results after sequencing has finished. Therefore, the major bottleneck of sample preparation and sequencing time remains unchanged. Methodological approaches such as rapid pulsed WGS (rpWGS) have been developed to analyze intermediate sequencing results in certain intervals but rely on a massive reduction of data or database size [97].

In this thesis, methods for the real-time analysis of Illummina sequencing data are introduced. The new approaches enable shorter turnaround times from sample arrival to analysis results without the need of specialized hardware. All presented methods are based on the concept of real-time read alignment which was first implemented in HiLive [92]. Chapter 2 presents an improved version of this approach using a seed-and-extend algorithm based on the FM-index. The reduced computational requirements, higher scalability and high quality results enable more complex follow-up analyses than before, as demonstrated by a real-time workflow

performing variant calling on human WES data.

Chapter 3 introduces PriLive and shows the benefit to analyze the unprocessed base call data instead of converting them to a human-readable file format such as FASTQ. That way, it is possible to use PriLive for removing human sequences from raw data such that no copy in any file format containing sensitive information remains available, unless any unprocessed image files are permanently stored. As human data is removed immediately after being identified (i.e., before the end of sequencing), this enables real-time data protection which no other method could provide before. The results of the experiments in Chapter 3 show that PriLive performs at least as good as conventional *post hoc* approaches while providing real-time data protection and reliable preservation of foreground-related reads.

PathoLive, on the other hand, shows the use of real-time alignment results as a basis for pathogen identification. While many other methods rely on a ranking of results based on the pure abundance of alignment hits, PathoLive highlights hits of high interest by comparing the mapping positions with background signals that usually occur when sequencing human samples. Further, PathoLive provides an intuitive and interactive visualization of results that shows the pathogenic potential of identified organisms and a detailed view on mapping occurrences in a foreground and background database. Chapter 4 shows the performance of PathoLive on an *in vitro* spiked dataset of six viruses and on a real-world dataset from Sudan containing a low concentration of Crimean–Congo hemorrhagic fever virus. All relevant organisms were clearly identified while other hits, even with a higher number of alignments, were correctly reported to be of low significance.

Concluding, the presented tools provide NGS-based real-time analysis solutions for a plethora of time-critical applications. Thereby, the applicability of the approaches range from data protection to the identification of pathogens and even nucleotide-level analysis such as SNP calling. The probably most important characteristic of the real-time sequencing approach is the correctness of results even in early sequencing cycles. By showing high precision in every stage of sequencing, all presented methods provide reliable results that can serve as a basis for early interpretation and decision-making while giving more extensive insights with ongoing sequencing.

5.2 Outlook

This thesis describes advanced methods for real-time analysis of Illumina sequencing data. All approaches provide high-quality results and open up new possibilities in time-critical

sequencing applications. Until today, the presented methods were mostly applied for explorative analysis in exceptional situations such as infectious disease outbreaks. However, with future improvements, the concept shows the potential to be established in clinical routine application, for example in sepsis diagnostics, screening for post-operative infections and detection of antimicrobial resistances. Still, the successful transfer to clinical routine application requires further development of the presented methods. In this section, I provide an outlook about such potential future developments, including technical aspects of Illumina sequencing, improvements of existing approaches, new approaches and the integration of other technologies into the concept of NGS-based real-time analysis.

Illumina Sequencing

With the presented real-time analysis methods, results can be produced during runtime of the sequencing device. Still, Illumina sequencing comes with the additional bottleneck of high sample preparation times. While a 30 minutes library preparation kit exists for amplicon sequencing [228], the currently fastest available library preparation protocol for untargeted sequencing is the Nextera DNA Flex Library Prep Kit with a total preparation time of three to four hours from DNA extraction to normalized library [229]. Therefore, a further acceleration of the library preparation would be desirable to reduce the turnaround time from sample arrival to analysis output. When coupled with the methods for real-time analysis presented in this thesis, this could make NGS-based approaches applicable in situations that currently require the speed of molecular methods.

A second technical limitation for the application of the presented real-time methods affects the analysis of multiplexed sequencing data. Multiplexing is used to combine several samples in a single sequencing run which reduces costs and increases sample throughput. While the analysis of multiplexed data is generally supported by all presented methods, the barcodes are usually sequenced after the first read. Thus, reads can only be assigned to the correct sample after the first read was completely sequenced. Efforts should be made to design a sequencing protocol that allows to sequence the barcodes first without negatively affecting the cluster generation. Thereby, it would be possible to obtain real-time results for several samples of the same sequencing run even for the first read, making the application of real-time sequencing cheaper and more scalable. One potential approach to achieve this could be the introduction of a short random sequence followed by the barcode at the beginning of the first read. As an interim solution, some Illumina sequencing devices support asymmetric read lengths. This

function can be used to sequence only a small fraction of the first read and have the barcodes available at an earlier stage of sequencing.

Modular Toolkit for Real-time Analysis of Illumina Reads

Currently, all tools for the real-time analysis of Illumina sequencing data are developed as standalone software. Thus, high efforts are required to maintain and further develop the tools. The organization of all methods in a common modular toolkit would be a possible approach to overcome these obstacles. When sharing a common data structure and enabling an exchange of analysis results, this would increase the flexibility in application as several approaches could be reused and combined with each other. Additionally, the maintenance of the software would be significantly facilitated as new features would only need to be implemented once to be available for all tools. That way, common preprocessing steps such as adapter or quality trimming could be coupled with each type of analysis. Further, even the major analyses could be combined with each other. For example, PriLive could be used in combination with PathoLive to enable real-time data protection and pathogen identification at the same time. As a second example, reads that could not be mapped to a reference genome in real-time could be used to perform other explorative analyses, for example investigating their pathogenic potential or searching for similar sequences in a larger database. It would also be easier to integrate functionalities for smaller tasks such as a real-time version of bcl2fastq or real-time quality control that can be executed in parallel to any type of analysis. Concluding, the integration of all tools in a common modular toolkit shows a high potential to make real-time analysis of Illumina sequencing data more efficient, flexible and maintainable.

Scalability and User Interaction

For the new generation of Illumina high throughput sequencing devices, particularly the NovaSeq Series, further improvements of the scalability of the real-time alignment approach should be pursued. However, this will require some conceptional changes. For example, the number of reads per tile is continuously increasing with modern sequencers. Therefore, it would be useful to enable the possibility of in-tile multithreading while it is currently only possible to use a single thread per tile. Further, the output of alignment files is a crucial bottleneck with increasing amounts of sequencing data and should be reengineered. From

the algorithmic perspective, a support of even larger indices such as the bacterial RefSeq [230] could be achieved by the use of a distributed index approach such as the DREAM framework of the SeqAn library [231]. Additionally, such a distributed index would allow to introduce user-driven real-time analysis decisions, for example by prioritizing or omitting certain subindices based on intermediate results. Such a concept to enable user interactions would also imply an advanced graphical user interface for presenting the current analysis results and allow an intuitive adaption of parameters.

Alignment-free Methods

An approach that became very popular over the last years is the use of efficient alignment-free analysis methods. Such methods overcome the computationally expensive alignment step, usually by evaluating k-mer frequencies or informational content such as, for example, the Kolmogorov complexity [232]. For extensive reviews about alignment-free methods, I refer to Zielezinski et al. [233] and Ren et al. [234].

Prominent examples for alignment-free methods in the field of NGS analysis include Kallisto [49] for transcript quantification and Kraken [184] for metagenomics classification. Usually, alignment-free analyses are fast and memory-efficient while still providing accurate results. On the other hand, due to the lack of alignment information, full nucleotide-level results are not provided which limits their use to a certain range of applications.

In metagenomics classification, other application-specific limitations such as the so far memory- and runtime-inefficient building and updating process of reference databases are addressed by a new generation of tools, namely ganon [183] and Kraken 2 [182]. While a real-time version of Kraken has already been published [101], the recent improvements of alignment-free metagenomics classification tools provide the possibility to use much larger and up-to-date databases which has a massive impact for the analysis outcome [188]. The integration of these new, more scalable tools in the real-time approach would be highly desirable to make use of the full spectrum of available sequences.

Alignment-free approaches can also be used for many other types of analyses, for example for the comparison of two or more samples, prediction of virus-host interactions or the identification of horizontal gene transfer [234]. While some of these applications are usually not relevant in a time-critical context, others like the comparison of samples can be of high interest such as the comparison of samples that can be used to investigate whether infections of several individuals are related to each other or not. Therefore, such approaches should be

considered to be integrated into the concept of real-time sequencing.

Reference-free Methods

All real-time approaches for NGS presented in this thesis are based on a reference database. This can lead to incomplete or incorrect analysis outcome when analyzing samples that contain highly variable or yet unknown organisms, or organisms that are simply not contained in the selected reference database. To fill this gap, reference-free (or taxonomy-independent) binning approaches could be integrated either for complementary analysis or to analyze reads that could not be classified by the reference-based methods. Especially when followed by an assembly step, such an approach could serve as a basis to detect organisms that are not present in a reference database or to identify highly variable organisms in real-time. However, the reference-free real-time binning of sequences is algorithmically complicated as sequence similarities can massively change with ongoing sequencing. Thus, it is uncertain whether it is possible to find an approach that is efficient enough to be applied in real-time and still accurate in all stages of sequencing.

A completely different concept of reference-free analyses is the prediction of certain properties of an unidentified organism where a read originated from. For example, there exist deep-learning based methods to predict the pathogenic potential of a DNA sequence [235]. When applied in real-time, such an approach could identify previously unclassified reads that could be relevant in the context of an infection and therefore deliver candidates for further explorative analysis.

Integration with ONT Sequencing Data

In comparison to Illumina sequencing, third-generation sequencing technologies provide much longer reads and implicitely support the real-time analysis of sequencing results. Oxford Nanopore Technologies sequencing, namely the MinION, can additionally be used for sequencing and analysis in the field where no laboratory and computational hardware are available, as for example successfully proven in the Antarctic Dry Valleys [236] and on the International Space Station [237]. However, while the throughput of this technology is continuously increasing with newly released devices, there are still several obstacles for the establishment of ONT sequencing as a standalone sequencing solution for many applications, especially in clinics. First, base calling is computationally expensive and error-prone. It was

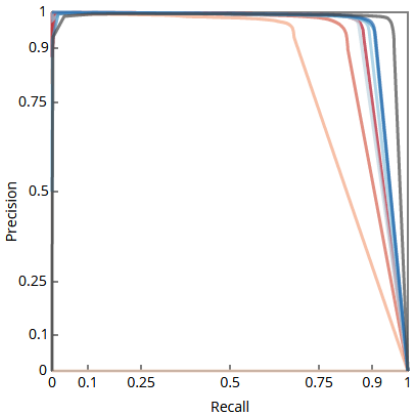
stated in several recent studies that the use of ONT requires "dramatic improvements" [35] to be used in public health laboratories, especially due to the high numbers of false positive SNP calls. This is an even greater issue in real-time analyses as these require rather fast than the most accurate methods. Additionally, many real-time applications require untargeted analysis workflows such that specifically tailored base calling approaches (e.g., trained for a specific organism) cannot be applied. Another issue derives from the general technical approach of ONT sequencing which can hardly resolve the length of homopolymers (length >4 bp) leading to a high number of false positive InDel calls in such regions. This can have major effects for certain analyses, including the identification of several antibiotic resistances that are associated with such homopolymeric regions [238, 239].

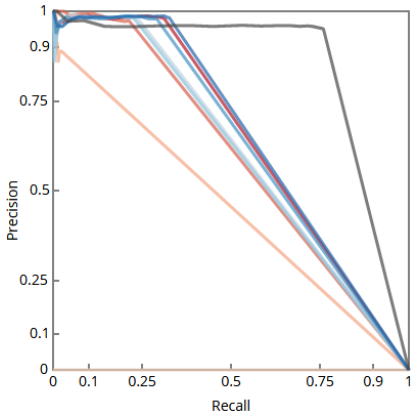
For these reasons, it makes generally sense to combine the strengths of both technologies by integrating Illumina and ONT sequencing results. In the field of genome assembly, software for such a combined approach already exists [240–242] and was shown to improve results when compared to assemblies using data of a single sequencing technology [243–246]. For many time-critical applications such an integration of both technologies in real-time could considerably increase the spectrum possible analyses, especially those based on assemblies. While using very short intermediate Illumina reads are not promising for standalone assembly, they could be very valuable when polishing real-time assemblies obtained from ONT sequencing to reduce the number of wrong base calls. Such assembly-based real-time analyses could improve the identification of antibiotic resistances, phylogenetic and epidemiological analyses or the identification and classification of yet unknown or highly variable pathogens.

A Appendix

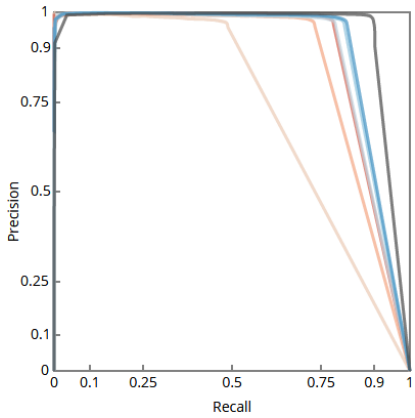
Appendix 1 - Reliable Variant Calling during Runtime of Illumina Sequencing

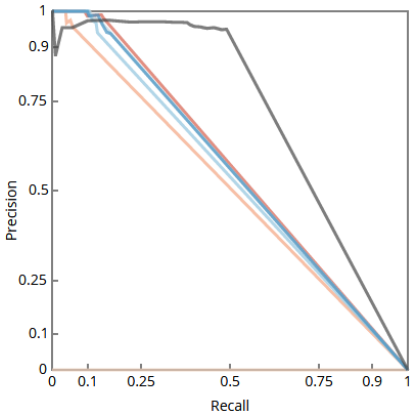
SRR098401



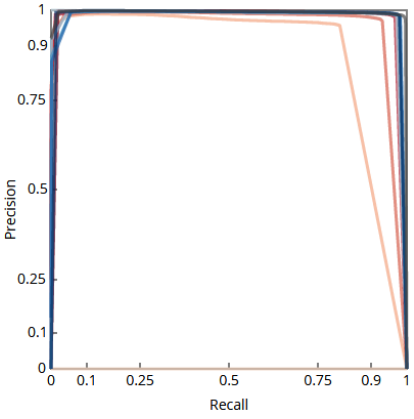


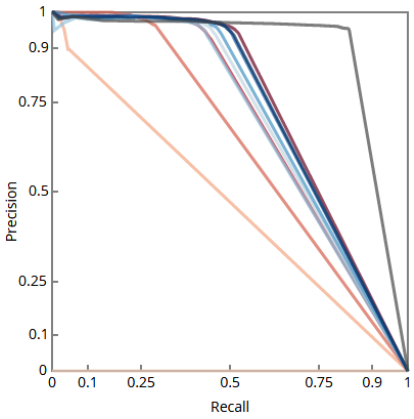
SRR292250



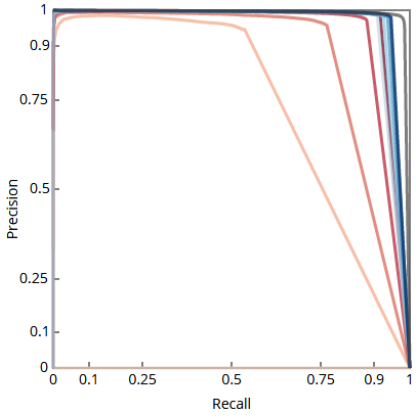


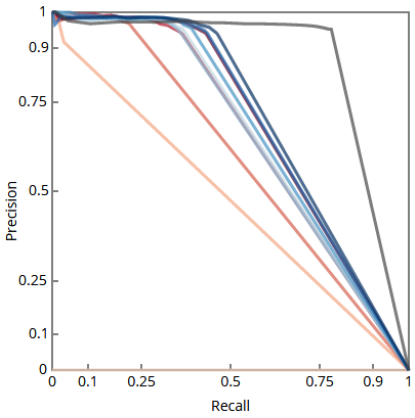
SRR515199



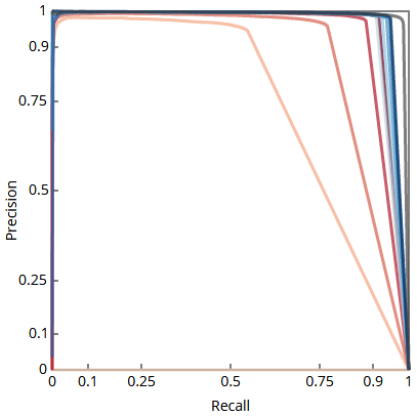


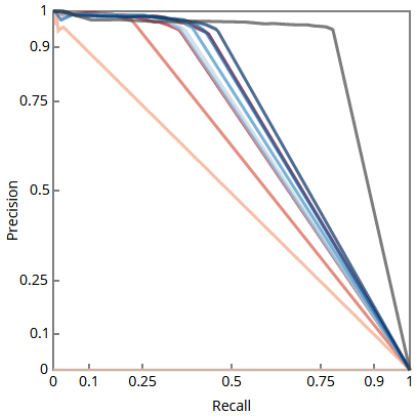
SRR1611178



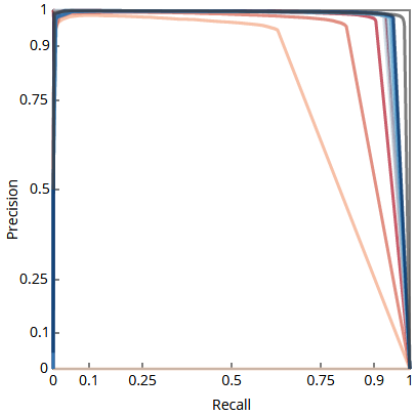


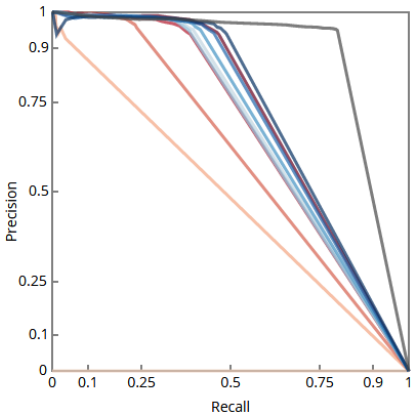
SRR1611179

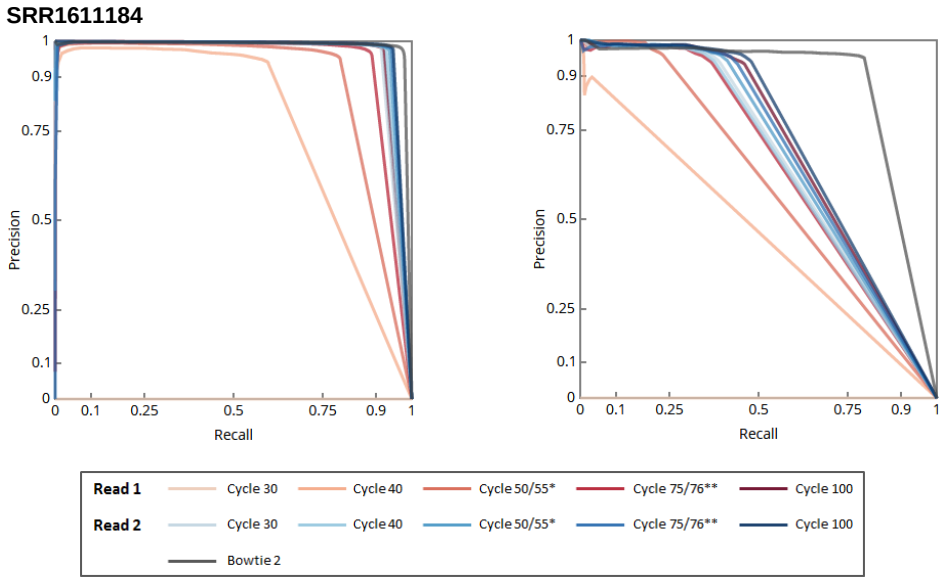




SRR1611183

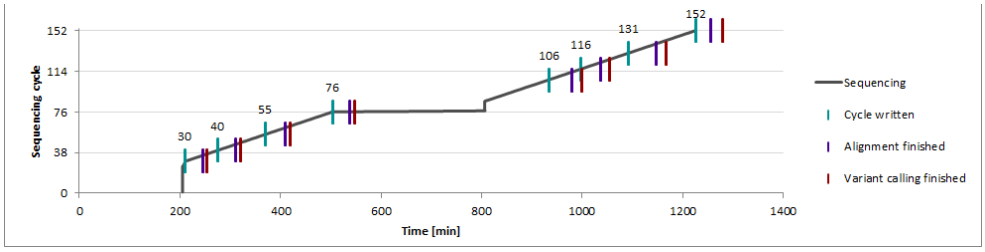




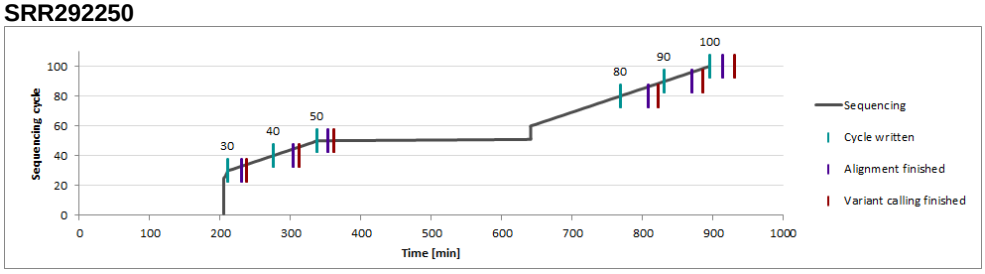


Appendix Figure A1.1: PR curves for seven WES datasets of human individual NA12878 for a real-time variant calling workflow using HiLive2 and xAtlas. Red curves show results for the first read, blue curves show results for the second read and the gray curve show results based on read mapping with Bowtie 2. The left figure for each dataset shows the PR curve for SNPs, the right figure shows PR curves for short InDels. For SNP calling, recall increases with longer sequencing time while the precision is high from the very beginning. Final results show at least the same precision as results based on Bowtie 2 but have slightly lower recall. For short InDels, an additional preprocessing-step was performed to left-align all InDel positions in the read mapping results. The general tendency of the results is very similar to those for SNPs but results show much lower recall than based on Bowtie 2. This is because HiLive2 can only find consecutive InDels up to a length of 3 with the used parameter settings. As longer InDels are included in the gold standard, 18 - 24 % of gold standard InDels cannot be identified based on read mapping results with HiLive2. The particularly low recall for InDels in dataset SRR292250 is because of the short read length of 50 bp.

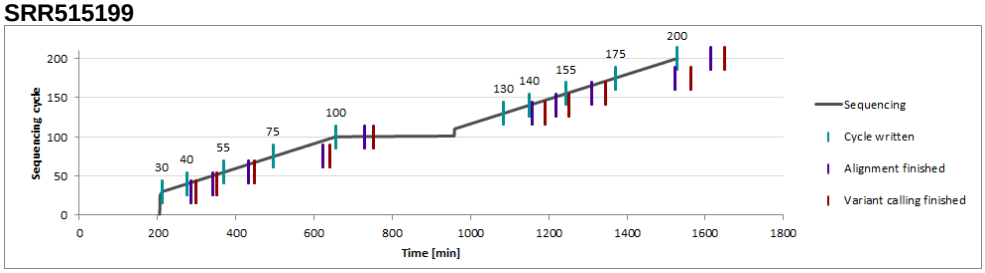
SRR098401



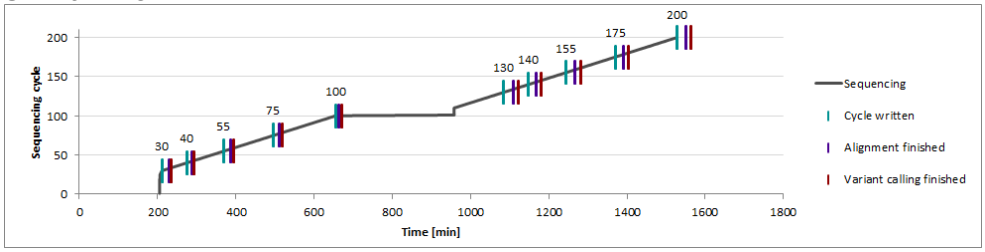
SRR292250



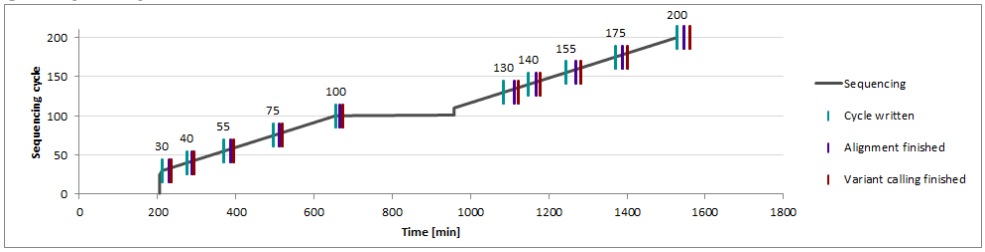
SRR515199

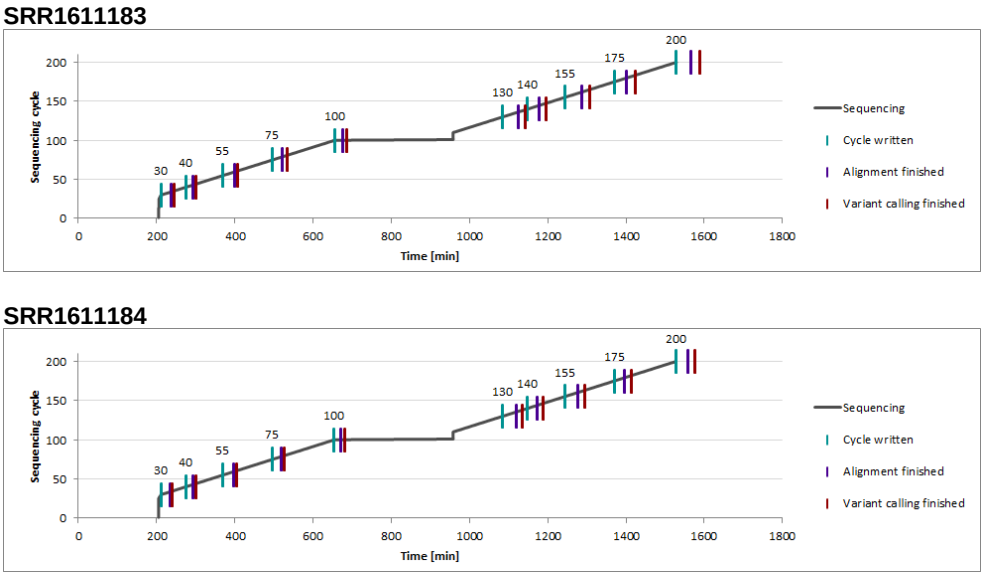


SRR1611178

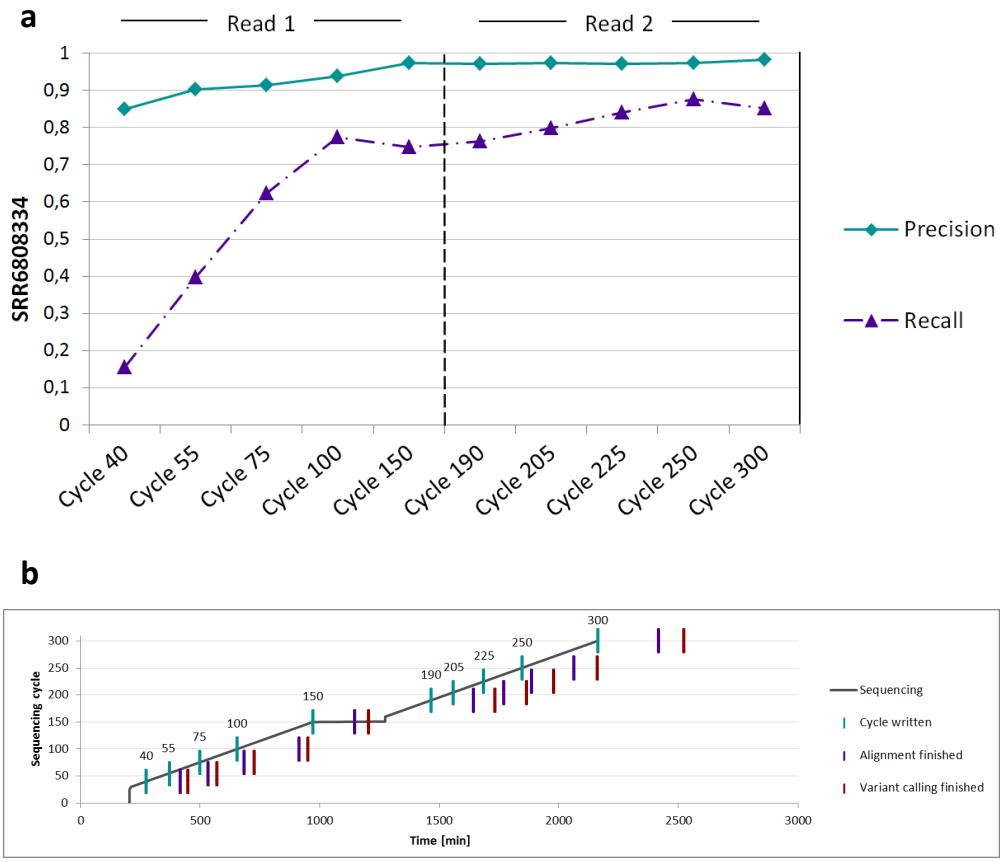


SRR1611179



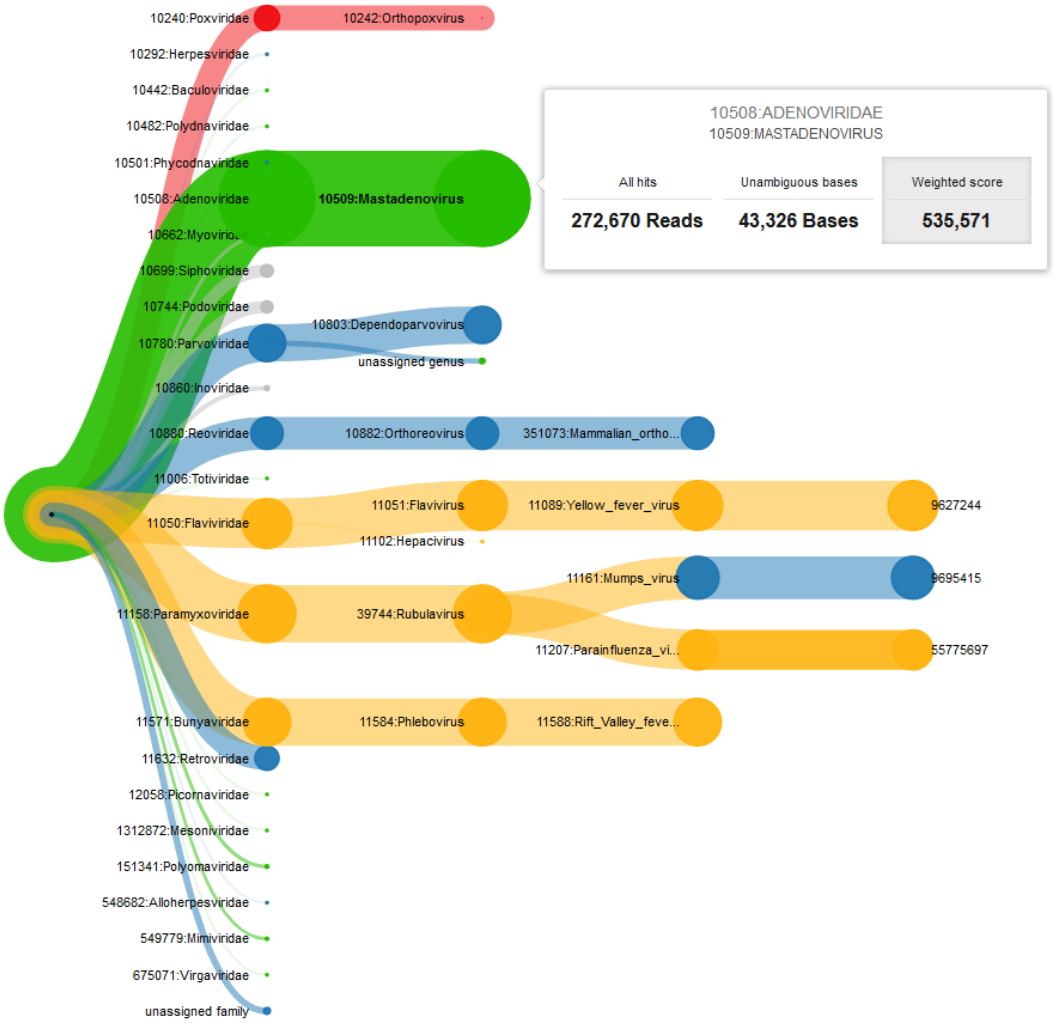


Appendix Figure A1.2: Turnaround times for the variant calling workflow for seven WES datasets and different sequencing cycles. The x-axis describes the turnaround time from starting the sequencer in minutes. The y-axis describes the sequencing cycle. The first vertical line of each data point indicates the time point when the sequencing cycle was written by the sequencing machine. The second vertical line shows when read mapping with HiLive2 finished and the third vertical line represents the availability of final variant calling results with xAtlas. The presented turnaround times do not include left-aligning InDels that was necessary to call InDels. Sequencing time was simulated with a simulator for Illumina sequencing.



Appendix Figure A1.3: Precision, recall (a) and turnaround times (b) for the variant calling workflow for the WGS dataset SRR6808334 and different sequencing cycles. **a** Precision and recall for SRR6808334 in different sequencing cycles. The x-axis denotes the output cycles. The y-axis denotes the values for precision and recall. **b** The x-axis shows the turnaround time from starting the sequencer in minutes. The y-axis describes the sequencing cycle. The first vertical line of each data point indicates the time point when the sequencing cycle was written by the sequencing machine. The second vertical line shows when read mapping with HiLive2 finished and the third vertical line represents the availability of final variant calling results with xAtlas. Sequencing time was simulated with a simulator for Illumina sequencing.

Appendix 2 - Real-time Pathogen Identification from Metagenomic Illumina Datasets



Appendix Figure A2.1: Example of the interactive taxonomic tree of results of PathoLive for a spiked dataset when using HiLive2 for read alignment. The visualized results of the described plasma sample at cycle 80 based on the weighted score are shown.

Appendix Table A2.1: Comparison of AUC values of PathoLive for a spiked dataset when using HiLive and HiLive2 for read alignment.

Cycle	AUC value with HiLive	AUC value with HiLive2
40	0.94	0.94
60	0.92	0.92
80	0.92	0.92
100	0.90	0.88

Accession Numbers

For the creation of the background database we used the datasets from the 1000 Genomes Project Phase 3 with the following accession numbers:

SRR190845, SRR068180, ERR251013, ERR251014, SRR099960, ERR229780, SRR189815, ERR015529, SRR099967, SRR099969, ERR251012, ERR251011, SRR099961, ERR013139, SRR099959, ERR013142, SRR701450, SRR098436, ERR018404, ERR015530, ERR251010, ERR251009, ERR015533, SRR098442, ERR015517, ERR013112, SRR701451, ERR015880, ERR019906, ERR015763, ERR013144, SRR707169, ERR015762, SRR099955, ERR018557, ERR015532, ERR013156, ERR015515, ERR013145, ERR013161, ERR013152, ERR016162, ERR013158, ERR018405, SRR098439, SRR043393, ERR018402, ERR018547, SRR707168, SRR741387, ERR018420, ERR016155, SRR062639, SRR062636, SRR741386, SRR101476, SRR101463, SRR101475, SRR043351, ERR015879, SRR101469, SRR718071, ERR016351, SRR062637, ERR016161, ERR018418, ERR018419, SRR101474, SRR060290, SRR037754, SRR037755, ERR031937, SRR101473, SRR051599, ERR031965, SRR060294, ERR016168, ERR013101, ERR016167, ERR031933, SRR101466, SRR101470, SRR764703, SRR037756, SRR101472, SRR035595, SRR038565, ERR016158, SRR060289, ERR016345, SRR037753, SRR764730, ERR016157, SRR035596, SRR101471, SRR101478, ERR016350, SRR701480, SRR044231, SRR765995, SRR101464, SRR044232, ERR031964, SRR101465, SRR035677, ERR034564, SRR060292, SRR060291, SRR044233, SRR766045, ERR031932, SRR707198, SRR060293, SRR101467,

SRR711355, ERR031936, ERR031935, SRR044235, SRR060295, SRR060296, ERR016160, SRR711356, SRR035676, SRR707196, SRR038561, SRR038564, ERR031934, SRR038563, SRR043360, SRR035673, SRR043357, SRR043396, SRR035600, SRR101477, SRR043410, SRR035674, SRR038562, SRR035675, SRR043354, SRR043384, SRR043392, SRR101468, SRR035594, SRR035593, SRR035672, SRR043379, SRR043372, SRR035591, SRR043378, SRR043381, SRR043386, SRR035592, SRR043370, SRR768526, SRR043382, ERR016005, SRR043405, SRR035590, SRR035601, SRR037782, SRR035589, ERR013146, SRR037783, ERR018521, ERR013131, SRR718072, SRR764729, SRR701483, SRR764704, SRR037777, ERR019904, SRR070801, ERR018523, SRR070516, ERR015527, SRR233084, SRR316803, SRR233083, SRR233086, SRR233075, SRR233102, SRR233105, SRR233085, SRR233088, SRR233069, SRR233079, SRR233087, SRR233074, SRR233101, SRR233082, ERR016166, ERR016159, ERR016156, ERR016169, ERR018403, ERR016163, ERR016165, ERR016164, SRR098444, SRR098432, SRR098438, SRR233073, SRR316801, SRR098437, SRR098441, SRR098433, SRR233107, SRR233106, SRR098435, SRR233097, SRR233104, SRR233094, SRR233078, SRR233091, SRR233096, SRR233071, SRR233100, SRR233099, SRR233089, SRR107017, SRR101146, SRR101150, SRR101144, SRR101145, SRR101147, SRR101148, SRR101149, SRR043361, SRR043362, SRR035485, SRR043383, SRR043408, SRR043367, SRR035484, SRR043356

Bibliography

[1] J. D. Watson and F. H. C. Crick. Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid. *Nature*, 171:737–8, 1953.

[2] B. Lewin. *Genes VII*. Oxford University Press, Chapter 1. Genes are DNA (DNA is the genetic material), 1999.

[3] S. Kumar and P. J. Bentley. *On Growth, Form and Computers*. Academic Press, New York, 1st edition, Chapter 1. An introduction to computational development (1.2.4 Genes - DNA), 2003.

[4] L. M. Zahn, B. A. Purnell, and C. Ash. The manifestation of the genome. *Science*, 365(6460):1394–1395, 2019.

[5] R. Wu and A. D. Kaiser. Structure and base sequence in the cohesive ends of bacteriophage lambda DNA. *Journal of Molecular Biology*, 35(3):523–37, 1968.

[6] W. Gilbert and A. Maxam. The nucleotide sequence of the lac operator. *Proceedings of the National Academy of Sciences of the United States of America*, 70(12):3581–4, 1973.

[7] A. M. Maxam and W. Gilbert. A new method for sequencing DNA. *Proceedings of the National Academy of Sciences of the United States of America*, 74(2):560–4, 1977.

[8] F. Sanger, S. Nicklen, and A. R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America*, 74(12):5463–7, 1977.

[9] L. M. Smith, S. Fung, M. W. Hunkapiller, T. J. Hunkapiller, and L. E. Hood. The synthesis of oligonucleotides containing an aliphatic amino group at the 5' terminus: synthesis of fluorescent DNA primers for use in DNA sequence analysis. *Nucleic Acids Research*, 13(7):2399–412, 1985.

[10] W. Ansorge, B. S. Sproat, J. Stegemann, and C. Schwager. A non-radioactive automated method for DNA sequence determination. *Journal of Biochemical and Biophysical Methods*, 13(6):315–23, 1986.

[11] W. Ansorge, B. Sproat, J. Stegemann, C. Schwager, and M. Zenke. Automated DNA sequencing: ultrasensitive detection of fluorescent bands during electrophoresis. *Nucleic Acids Research*, 15(11):4593–602, 1987.

[12] J. M. Prober, G. L. Trainor, R. J. Dam, F. W. Hobbs, C. W. Robertson, R. J. Zagursky, A. J. Cocuzza, M. A. Jensen, and K. Baumeister. A system for rapid DNA sequencing with fluorescent chain-terminating dideoxynucleotides. *Science*, 238(4825):336–41, 1987.

[13] H. Swerdlow and R. Gesteland. Capillary gel electrophoresis for rapid, high resolution DNA sequencing. *Nucleic Acids Research*, 18(6):1415–9, 1990.

[14] L. M. Smith. High-speed DNA sequencing by capillary gel electrophoresis. *Nature*, 349(6312):812–3, 1991.

[15] International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, 2001.

[16] J. C. Venter, M. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, et al. The sequence of the human genome. *Science*, 291(5507):1304–51, 2001.

[17] M. L. Metzker. Sequencing technologies - the next generation. *Nature Reviews Genetics*, 11(1):31–46, 2010.

[18] Y. S. Chang, H. D. Huang, K. T. Yeh, and J. G. Chang. Evaluation of whole exome sequencing by targeted gene sequencing and Sanger sequencing. *Clinica Chimica Acta*, 471:222–32, 2017.

[19] D. Muzzey, S. Kash, J. I. Johnson, L. M. Melroy, P. Kaleta, K. A. Pierce, K. Ready, H. P. Kang, and K. R. Haas. Software-assisted manual review of clinical next-generation sequencing data: An alternative to routine Sanger sequencing confirmation with equivalent results in > 15,000 germline DNA screens. *The Journal of Molecular Diagnostics*, 21(2):296–306, 2019.

[20] E. R. Mardis. A decade's perspective on DNA sequencing technology. *Nature*, 470 (7333):198–203, 2011.

[21] P. Nyrén, B. Pettersson, and M. Uhlén. Solid phase DNA minisequencing by an enzymatic luminometric inorganic pyrophosphate detection assay. *Analytical Biochemistry*, 208(1):171–5, 1993.

[22] J. Berka, Y. J. Chen, J. H. Leamon, S. Lefkowitz, K. L. Lohman, V. B. Makhijani, J. M. Rothberg, G. J. Sarkis, M. Srinivasan, and M. P. Weiner. Bead emulsion nucleic acid amplification. Patent Application; International Publication No. WO 2004/069849 A3. 2004.

[23] C. Scheid. Aktie der Woche: Illumina. *Capital*, Available Online from: *https://www.capital.de/geld-versicherungen/aktie-der-woche-illumina*; accessed 21.06.2019, 2018.

[24] S. Goodwin, J. D. McPherson, and W. R. McCombie. Coming of age: Ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, 17(6):333–51, 2016.

[25] J. M. Heather and B. Chain. The sequence of sequencers: The history of sequencing DNA. *Genomics*, 107(1):1–8, 2016.

[26] J. Shendure, S. Balasubramanian, G. M. Church, W. Gilbert, J. Rogers, J. A. Schloss, and R. H. Waterston. DNA sequencing at 40: past, present and future. *Nature*, 550 (7676):345–53, 2017.

[27] San Diego, CA: Illumina, Inc. Illumina CMOS chip and one-channel SBS chemistry, 2018. Pub. No. 770-2013-054-B QB 5474.

[28] San Diego, CA: Illumina, Inc. Illumina two-channel SBS sequencing technology, 2016. Pub. No. 770-2013-054.

[29] I. F. Bronner and M. A. Quail. Best practices for Illumina library preparation. *Current Protocols in Human Genetics*, 102(1):e86, 2019.

[30] Agilent Technologies. Protocol: SureSelect^{QXT} whole genome library prep for Illumina multiplexed sequencing (Version E0), 2018.

[31] San Diego, CA: Illumina, Inc. An introduction to next-generation sequencing technology, 2017. Pub. No. 770-2012-008-B.

[32] B. Ewing and P. Green. Base-calling of automated sequencer traces using phred. II. Error probabilities. *Genome Research*, 8(3):186–94, 1998.

[33] E. L. van Dijk, Y. Jaszczyszyn, D. Naquin, and C. Thermes. The third revolution in sequencing technology. *Trends in Genetics*, 34(9):666–81, 2018.

[34] J. Beaulaurier, E. E. Schadt, and F. G. Deciphering bacterial epigenomes using modern sequencing technologies. *Nature Reviews Genetics*, 20(3):157–72, 2019.

[35] R. R. Wick, L. M. Judd, and K. E. Holt. Performance of neural network basecalling tools for Oxford Nanopore sequencing. *Genome Biology*, 20(1):129, 2019.

[36] M. A. Quail, M. Smith, P. Coupland, T. D. Otto, S. R. Harris, T. R. Connor, A. Bertoni, H. P. Swerdlow, and Y. Gu. A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers. *BMC Genomics*, 13:341, 2012.

[37] J. Korlach. Understanding accuracy in SMRT sequencing. Technical Report PN 100-211-800-03, Pacific Biosciences of California, Inc., 2013.

[38] J. L. Weirather, M. de Cesare, Y. Wang, P. Piazza, V. Sebastiano, X. J. Wang, D. Buck, and K. F. Au. Comprehensive comparison of Pacific Biosciences and Oxford Nanopore Technologies and their applications to transcriptome analysis. *F1000Research*, 6, 2017.

[39] M. Jain, J. R. Tyson, M. Loose, C. L. C. Ip, D. A. Eccles, J. O'Grady, S. Malla, R. M. Leggett, et al. MinION Analysis and Reference Consortium: Phase 2 data release and analysis of R9.0 chemistry. *F1000Research*, 6:760, 2017.

[40] R. M. Leggett and M. D. Clark. A world of opportunities with nanopore sequencing. *Journal of Experimental Botany*, 68(20):5419–29, 2017.

[41] F. J. Rang, W. P. Kloosterman, and J. de Ridder. From squiggle to basepair: computational approaches for improving nanopore sequencing read accuracy. *Genome Biology*, 19(1):90, 2018.

[42] N. Kono and K. Arakawa. Nanopore sequencing: Review of potential applications in functional genomics. *Development Growth & Differentiation*, 61(5):316–26, 2019.

[43] J. Eid, A. Fehr, J. Gray, K. Luong, J. Lyle, G. Otto, P. Peluso, D. Rank, et al. Real-time DNA sequencing from single polymerase molecules. *Science*, 323(5910):133–8, 2009.

[44] K. J. Travers, C. S. Chin, D. R. Rank, J. S. Eid, and S. W. Turner. A flexible and efficient template format for circular consensus sequencing and SNP detection. *Nucleic Acids Research*, 38(15):e159, 2010.

[45] S. Ardui, A. Ameur, J. R. Vermeesch, and M. S. Hestand. Single molecule real-time (SMRT) sequencing comes of age: applications and utilities for medical diagnostics. *Nucleic Acids Research*, 46(5):2159–68, 2018.

[46] A. D. Tyler, L. Mataseje, C. J. Urfano, L. Schmidt, K. S. Antonation, M. R. Mulvey, and C. R. Corbett. Evaluation of Oxford Nanopore's MinION sequencing device for microbial whole genome sequencing applications. *Scientific Reports*, 8(1):10931, 2018.

[47] Oxford Nanopore Technologies. Promethion specification, 2019. URL *https://nanoporetech.com/products/promethion*. accessed 05.07.2019.

[48] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–10, 1990.

[49] N. L. Bray, H. Pimentel, P. Melsted, and L. Pachter. Near-optimal probabilistic RNA-seq quantification. *Nature Biotechnology*, 34(5):525–7, 2016.

[50] R. Patro, G. Duggal, M. I. Love, R. A. Irizarry, and C. Kingsford. Salmon provides fast and bias-aware quantification of transcript expression. *Nature Methods*, 14(4): 417–419, 2017.

[51] R. A. Baeza-Yates and C. H. Perleberg. Fast and practical approximate string matching. In *Combinatorial Pattern Matching*, pages 185–92, 1992. doi: 10.1016/0020-0190(96)00083-X.

[52] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Doklady Akademii Nauk SSSR*, 163(4):845–8, 1965.

[53] G. Navarro. A guided tour to approximate string matching. *ACM Computing Surveys (CSUR)*, 33(1):31–88, 2001.

[54] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–53, 1970.

[55] K. Reinert, B. Langmead, D. Weese, and D. J. Evers. Alignment of next-generation sequencing reads. *Annual Review of Genomics and Human Genetics*, 16(1):133–51, 2015.

[56] A. Califano and I. Rigoutsos. FLASH: a fast look-up algorithm for string homology. *Proceedings. International Conference on Intelligent Systems for Molecular Biology*, 1:56–64, 1993.

[57] B. Ma, J. Tromp, and M. Li. PatternHunter: faster and more sensitive homology search. *Bioinformatics*, 18(3):440–5, 2002.

[58] M. Li, B. Ma, D. Kisman, and J. Tromp. PatternHunter II: Highly sensitive and fast homology search. *Journal of Bioinformatics and Computational Biology*, 2(03): 417–39, 2004.

[59] W. P. Lee, M. P. Stromberg, A. Ward, C. Stewart, E. P. Garrison, and G. T. Marth. MOSAIK: a hash-based algorithm for accurate next-generation sequencing short-read mapping. *PloS One*, 9(3):e90581, 2014.

[60] R. Li, Y. Li, K. Kristiansen, and J. Wang. SOAP: short oligonucleotide alignment program. *Bioinformatics*, 24(5):713–4, 2008.

[61] H. Cheng, H. Jiang, J. Yang, Y. Xu, and Y. Shang. BitMapper: an efficient all-mapper based on bit-vector computing. *BMC Bioinformatics*, 16:192, 2015.

[62] A. Ahmadi, A. Behm, N. Honnalli, C. Li, L. Weng, and X. Xie. Hobbes: optimized gram-based methods for efficient read alignment. *Nucleic Acids Research*, 40(6), 2012.

[63] D. Weese, A. K. Emde, T. Rausch, A. Doring, and K. Reinert. RazerS - fast read mapping with sensitivity control. *Genome Research*, 19(9):1646–54, 2009.

[64] D. Weese, M. Holtgrewe, and K. Reinert. RazerS 3: faster, fully sensitive read mapping. *Bioinformatics*, 28(20):2592–99, 2012.

[65] N. Homer, B. Merriman, and S. F. Nelson. BFAST: an alignment tool for large scale genome resequencing. *PloS One*, 4(11):e7767, 2009.

[66] W. J. Kent. BLAT - the BLAST-like alignment tool. *Genome Research*, 12(4):656–64, 2002.

[67] H. Xin, D. Lee, F. Hormozdiari, S. Yedkar, O. Mutlu, and C. Alkan. Accelerating read mapping with FastHASH. *BMC Genomics*, 14 Suppl 1:S13, 2013.

[68] M. Zaharia, W. J. Bolosky, K. Curtis, A. Fox, D. A. Patterson, S. Shenker, I. Stoica, R. M. Karp, and T. Sittler. Faster and more accurate sequence alignment with SNAP. *arXiv*, http://arxiv.org/abs/1111.5572, 2011.

[69] Y. Liao, G. K. Smyth, and W. Shi. The Subread aligner: fast, accurate and scalable read mapping by seed-and-vote. *Nucleic Acids Research*, 41(10):e108, 2013.

[70] H. Li, J. Ruan, and R. Durbin. Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Research*, 18(11):1851–8, 2008.

[71] C. Alkan, J. M. Kidd, T. Marques-Bonet, G. Aksay, F. Antonacci, F. Hormozdiari, J. O. Kitzman, C. Baker, et al. Personalized copy number and segmental duplication maps using next-generation sequencing. *Nature Genetics*, 41(10):1061–67, 2009.

[72] F. Hach, F. Hormozdiari, C. Alkan, F. Hormozdiari, I. Birol, E. E. Eichler, and S. C. Sahinalp. mrsFAST: a cache-oblivious algorithm for short-read mapping. *Nature Methods*, 7(8):576–7, 2010.

[73] S. M. Rumble, P. Lacroute, A. V. Dalca, M. Fiume, A. Sidow, and M. Brudno. SHRiMP: accurate mapping of short color-space reads. *PloS Computational Biology*, 5(5):e1000386, 2009.

[74] M. David, M. Dzamba, D. Lister, L. Ilie, and M. Brudno. SHRiMP2: sensitive yet practical short read mapping. *Bioinformatics*, 27(7):1011–2, 2011.

[75] Novocraft Technologies Sdn Bhd. NovoAlign, http://www.novocraft.com. 2010.

[76] B. Buchfink, C. Xie, and D. H. Huson. Fast and sensitive protein alignment using DIAMOND. *Nature Methods*, 12(1):59–60, 2015.

[77] P. Weiner. Linear pattern matching algorithms. In *Switching and Automata Theory, 1973. SWAT'08. IEEE Conference Record of 14th Annual Symposium on*, pages 1–11, 1973.

[78] U. Manber and G. Myers. Suffix arrays: a new method for on-line string searches. *SIAM Journal on Computing*, 22(5):935–48, 1993.

[79] M. I. Abouelhoda, S. Kurtz, and E. Ohlebusch. Replacing suffix trees with enhanced suffix arrays. *Journal of Discrete Algorithms*, 2(1):53–86, The 9th International Symposium on String Processing and Information Retrieval, 2004.

[80] S. Hoffmann, C. Otto, S. Kurtz, C. M. Sharma, P. Khaitovich, J. Vogel, P. F. Stadler, and J. Hackermuller. Fast mapping of short sequences with mismatches, insertions and deletions using index structures. *PloS Computational Biology*, 5(9):e1000502, 2009.

[81] P. Ferragina and G. Manzini. Opportunistic data structures with applications. In *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, pages 390–8, 2000. doi: 10.1.1.23.7615.

[82] R. Li, C. Yu, Y. Li, T. W. Lam, S. M. Yiu, K. Kristiansen, and J. Wang. SOAP2: an improved ultrafast tool for short read alignment. *Bioinformatics*, 25(15):1966–7, 2009.

[83] C. M. Liu, T. Wong, E. Wu, R. Luo, S. M. Yiu, Y. Li, B. Wang, C. Yu, et al. SOAP3: ultra-fast GPU-based parallel alignment tool for short reads. *Bioinformatics*, 28(6): 878–9, 2012.

[84] B. Langmead. Aligning short sequencing reads with Bowtie. *Current Protocols in Bioinformatics*, Chapter 11, Unit 11.7, 2010.

[85] B. Langmead and S. L. Salzberg. Fast gapped-read alignment with Bowtie 2. *Nature Methods*, 9(4):357–9, 2012.

[86] E. Siragusa, D. Weese, and K. Reinert. Fast and accurate read mapping with approximate seeds and multiple backtracking. *Nucleic Acids Research*, 41(7):e78, 2013.

[87] E. Siragusa. *Approximate string matching for high-throughput sequencing*. PhD thesis, Free University of Berlin, 2015. doi: 10.17169/refubium-15562.

[88] S. Marco-Sola, M. Sammeth, R. Guigo, and P. Ribeca. The GEM mapper: fast, accurate and versatile alignment by filtration. *Nature Methods*, 9(12):1185–8, 2012.

[89] H. Li and R. Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–60, 2009.

[90] D. Kim, B. Langmead, and S. L. Salzberg. HISAT: a fast spliced aligner with low memory requirements. *Nature Methods*, 12(4):357–60, 2015.

[91] M. Burrows and D. J. Wheeler. A block-sorting lossless data compression algorithm. Technical report, Digital Equipment Corporation, Palo Alto, California, 1994. Technical Report 124.

[92] M. S. Lindner, B. Strauch, J. M. Schulze, S. H. Tausch, P. W. Dabrowski, A. Nitsche, and B. Y. Renard. HiLive: real-time mapping of illumina reads while sequencing. *Bioinformatics*, 33(6):917–9, 2017.

[93] S. H. Tausch. *Development of bioinformatics tools for the rapid and sensitive detection of known and unknown pathogens from next generation sequencing data*. PhD thesis, Free University of Berlin, 2019. doi: 10.17169/refubium-1579.

[94] S. E. Soden, C. J. Saunders, L. K. Willig, E. G. Farrow, L. D. Smith, J. E. Petrikin, J. B. LePichon, N. A. Miller, et al. Effectiveness of exome and genome sequencing guided by acuity of illness for diagnosis of neurodevelopmental disorders. *Science Translational Medicine*, 6(265):265ra168, 2014.

[95] C. J. Saunders, N. A. Miller, S. E. Soden, D. L. Dinwiddie, A. Noll, N. A. Alnadi, N. Andraws, M. L. Patterson, et al. Rapid whole-genome sequencing for genetic disease diagnosis in neonatal intensive care units. *Science Translational Medicine*, 4 (154):154ra135, 2012.

[96] N. A. Miller, E. G. Farrow, M. Gibson, L. K. Willig, G. Twist, B. Yoo, T. Marrs, S. Corder, et al. A 26-hour system of highly sensitive whole genome sequencing for emergency management of genetic diseases. *Genome Medicine*, 7:100, 2015.

[97] H. Stranneheim, M. Engvall, K. Naess, N. Lesko, P. Larsson, M. Dahlberg, R. Andeer, A. Wredenberg, et al. Rapid pulsed whole genome sequencing for comprehensive acute diagnostics of inborn errors of metabolism. *BMC Genomics*, 15:1090, 2014.

[98] E. J. Rubin. TB diagnosis from the Dark Ages to fluorescence. *Nature Microbiology*, 3(3):268–9, 2018.

[99] S. Quainoo, J. P. M. Coolen, S. A. F. T. van Hijum, M. A. Huynen, W. J. G. Melchers, W. van Schaik, and H. F. L. Wertheim. Whole-Genome Sequencing of Bacterial Pathogens: the Future of Nosocomial Outbreak Analysis. *Clinical Microbiology Reviews*, 30(4):1015–63, 2017.

[100] C. A. Gilchrist, S. D. Turner, M. F. Riley, W. A. Petri, and E. L. Hewlett. Whole-genome sequencing in outbreak analysis. *Clinical Microbiology Reviews*, 28(3): 541–63, 2015.

[101] S. H. Tausch, B. Strauch, A. Andrusch, T. P. Loka, M. S. Lindner, A. Nitsche, and B. Y. Renard. LiveKraken - real-time metagenomic classification of illumina data. *Bioinformatics*, 34(21):3750–2, 2018.

[102] F. Menges, G. Narzisi, and B. Mishra. TotalReCaller: improved accuracy and performance via integrated alignment and base-calling. *Bioinformatics*, 27(17):2330–7, 2011.

[103] B. Mishra. Gappy TotalReCaller for RNASeq base-calling and mapping. *bioRxiv*. doi: 10.1101/000489, 2013.

[104] J. Farek, D. Hughes, A. Mansfield, O. Krasheninina, W. Nasser, F. J. Sedlazeck, Z. Khan, E. Venner, et al. xAtlas: Scalable small variant calling across heterogeneous next-generation sequencing experiments. *bioRxiv*. doi: 10.1101/295071, 2018.

[105] B. Grüning, R. Dale, A. Sjodin, B. A. Chapman, J. Rowe, C. H. Tomkins-Tinch, R. Valieris, and J. Koster. Bioconda: sustainable and comprehensive software distribution for the life sciences. *Nature Methods*, 15(7):475–6, 2018.

[106] A. Döring, D. Weese, T. Rausch, and K. Reinert. SeqAn An efficient, generic C++ library for sequence analysis. *BMC Bioinformatics*, 9(1):11, 2008.

[107] J. G. Cleary, R. Braithwaite, K. Gaastra, B. S. Hilbush, S. Inglis, S. A. Irvine, A. Jackson, R. Littin, et al. Comparing variant call files for performance benchmarking of next-generation sequencing variant calling pipelines. *bioRxiv*. doi: 10.1101/023754, 2015.

[108] J. M. Zook, B. Chapman, J. Wang, D. Mittelman, O. Hofmann, W. Hide, and M. Salit. Integrating human sequence data sets provides a resource of benchmark SNP and indel genotype calls. *Nature Biotechnology*, 32(3):246–51, 2014.

[109] P. Danecek, A. Auton, G. Abecasis, C. A. Albers, E. Banks, M. A. DePristo, R. E. Handsaker, G. Lunter, et al. The variant call format and VCFtools. *Bioinformatics*, 27 (15):2156–8, 2011.

[110] A. R. Quinlan and I. M. Hall. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6):841–2, 2010.

[111] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, and R. Durbin. The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25(16):2078–9, 2009.

[112] S. Hwang, E. Kim, I. Lee, and E. M. Marcotte. Systematic comparison of variant calling pipelines using gold standard personal exome variants. *Scientific Reports*, 5: 17875, 2015.

[113] P. McGann, J. L. Bunin, E. Snesrud, S. Singh, R. Maybank, A. C. Ong, Y. I. Kwak, S. Seronello, et al. Real time application of whole genome sequencing for outbreak investigation - What is an achievable turnaround time? *Diagnostic Microbiology and Infectious Disease*, 85(3):277–82, 2016.

[114] D. Freed, R. Aldana, J. A. Weber, and J. S. Edwards. The Sentieon Genomics Tools - a fast and accurate solution to variant calling from next-generation sequence data. *bioRxiv*. doi: 10.1101/115717, 2017.

[115] A. Supernat, O. V. Vidarsson, V. M. Steen, and T. Stokowy. Comparison of three variant callers for human whole genome sequencing. *Scientific Reports*, 8(1):17851, 2018.

[116] A. L. Greninger, S. N. Naccache, S. Federman, G. Yu, P. Mbala, V. Bres, D. Stryke, J. Bouquet, et al. Rapid metagenomic identification of viral pathogens in clinical samples by real-time nanopore sequencing analysis. *Genome Medicine*, 7:99, 2015.

[117] Z. D. Stephens, S. Y. Lee, F. Faghri, R. H. Campbell, C. Zhai, M. J. Efron, R. Iyer, M. C. Schatz, S. Sinha, and G. E. Robinson. Big data: Astronomical or genomical? *PloS Biology*, 13(7):e1002195, 2015.

[118] N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J. V. Pearson, D. A. Stephan, S. F. Nelson, and D. W. Craig. Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays. *PloS Genetics*, 4(8):e1000167, 2008.

[119] K. B. Jacobs, M. Yeager, S. Wacholder, D. Craig, P. Kraft, D. J. Hunter, J. Paschal, T. A. Manolio, et al. A new statistic and its power to infer membership in a genome-wide association study using genotype frequencies. *Nature Genetics*, 41(11):1253–7, 2009.

[120] S. Sankararaman, G. Obozinski, M. I. Jordan, and E. Halperin. Genomic privacy and limits of individual detection in a pool. *Nature Genetics*, 41(9):965–7, 2009.

[121] S. Li, N. Bandeira, X. Wang, and H. Tang. On the privacy risks of sharing clinical proteomics data. *AMIA Summits on Translational Science Proceedings*, 2016:122–31, 2016.

[122] A. Harmanci and M. Gerstein. Quantification of private information leakage from phenotype-genotype data: linking attacks. *Nature Methods*, 13(3):251–6, 2016.

[123] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich. Identifying personal genomes by surname inference. *Science*, 339(6117):321–4, 2013.

[124] Y. Erlich and A. Narayanan. Routes for breaching and protecting genetic privacy. *Nature Reviews Genetics*, 15(6):409–21, 2014.

[125] E. Ayday, J. L. Raisaro, P. J. McLaren, J. Fellay, and J. P. Hubaux. Privacy-preserving computation of disease risk by using genomic, clinical, and environmental data. USENIX Security Workshop on Health Information Technologies, 2013.

[126] F. Chen, S. Wang, X. Jiang, S. Ding, Y. Lu, J. Kim, S. C. Sahinalp, C. Shimizu, et al. PRINCESS: privacy-protecting rare disease international network collaboration via encryption through software guard extensions. *Bioinformatics*, 33(6):871–8, 2017.

[127] K. Shimizu, K. Nuida, and G. Rätsch. Efficient privacy-preserving string search and an application in genomics. *Bioinformatics*, 32(11):1652–61, 2016.

[128] E. A. Durham, M. Kantarcioglu, Y. Xue, C. Toth, M. Kuzu, and B. Malin. Composite bloom filters for secure record linkage. *IEEE Transactions on Knowledge and Data Engineering*, 26(12):2956–68, 2014.

[129] R. Schnell, T. Bachteler, and J. Reiher. Privacy-preserving record linkage using Bloom filters. *BMC Medical Informatics and Decision Making*, 9(1):41, 2009.

[130] C. Dwork. Differential privacy. In M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, editors, *Automata, Languages and Programming: 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*, pages 1–12. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[131] A. Machanavajjhala, D. Kifer, J. Abowd, J. Gehrke, and L. Vilhuber. Privacy: Theory meets practice on the map. In *2008 IEEE 24th International Conference on Data Engineering*, pages 277–86, 2008.

[132] Z. Huang, E. Ayday, H. Lin, R. S. Aiyar, A. Molyneaux, Z. Xu, J. Fellay, L. M. Steinmetz, and J. P. Hubaux. A privacy-preserving solution for compressed storage and selective retrieval of genomic data. *Genome Research*, 2016.

[133] L. Barzon, E. Lavezzo, V. Militello, S. Toppo, and G. Palù. Applications of next-generation sequencing technologies to diagnostic virology. *International Journal of Molecular Sciences*, 12(11):7861, 2011.

[134] S. Datta, R. Budhauliya, B. Das, S. Chatterjee, Vanlalhmuaka, and V. Veer. Next-generation sequencing in clinical virology: discovery of new viruses. *World Journal of Virology*, 4(3):265–76, 2015.

[135] R. Schmieder and R. Edwards. Fast identification and removal of sequence contamination from genomic and metagenomic datasets. *PloS One*, 6(3):e17288, 2011.

[136] M. M. Haque, T. Bose, A. Dutta, C. V. S. K. Reddy, and S. S. Mande. CS-SCORE: rapid identification and removal of human genome contaminants from metagenomic datasets. *Genomics*, 106(2):116–21, 2015.

[137] K. Rotmistrovsky and R. Agarwala. BMTagger: Best match tagger for removing human reads from metagenomic datasets. *ftp://ftp.ncbi.nlm.nih.gov/pub/agarwala/bmtagger/*, accessed 17.07.2017, 2011.

[138] A. Hatem, D. Bozdağ, A. E. Toland, and . V. Çatalyürek. Benchmarking short sequence mapping tools. *BMC Bioinformatics*, 14(1):184, 2013.

[139] J. I. Sohn and J. W. Nam. The present and future of de novo whole-genome assembly. *Briefings in Bioinformatics*, 2016.

[140] M. Gymrek, D. Golan, S. Rosset, and Y. Erlich. lobSTR: a short tandem repeat profiler for personal genomes. *Genome Research*, 22(6):1154–62, 2012.

[141] M. Holtgrewe. Mason - a read simulator for second generation sequencing data. Technical Report TR-B-10-06, Free University of Berlin, 2010.

[142] A. Sczyrba, P. Hofmann, P. Belmann, D. Koslicki, S. Janssen, J. Droge, I. Gregor, S. Majda, et al. Critical assessment of metagenome interpretation - a benchmark of metagenomics software. *Nature Methods*, 14(11):1063–71, 2017.

[143] D. Bourquain and A. Nitsche. Cowpox virus but not vaccinia virus induces secretion of CXCL1, IL-8 and IL-6 and chemotaxis of monocytes in vitro. *Virus Research*, 171 (1):161–7, 2013.

[144] H. Li and R. Durbin. Fast and accurate long-read alignment with Burrows–Wheeler transform. *Bioinformatics*, 26(5):589–95, 2010.

[145] SRPRISM, *ftp://ftp.ncbi.nlm.nih.gov/pub/agarwala/srprism*. accessed 17.07.2017.

[146] J. L. Raisaro, F. Tramèr, Z. Ji, D. Bu, Y. Zhao, K. Carey, D. Lloyd, H. Sofia, et al. Addressing Beacon re-identification attacks: quantification and mitigation of privacy risks. *Journal of the American Medical Informatics Association*, 24(4):799–805, 2017.

[147] D. Bzhalava, H. Johansson, J. Ekstrom, H. Faust, B. Moller, C. Eklund, P. Nordin, B. Stenquist, et al. Unbiased approach for virus detection in skin lesions. *PloS One*, 8 (6):e65953, 2013.

[148] A. L. Greninger, D. M. Zerr, X. Qin, A. L. Adler, R. Sampoleo, J. M. Kuypers, J. A. Englund, and K. R. Jerome. Rapid metagenomic next-generation sequencing during an investigation of hospital-acquired Human Parainfluenza Virus 3 infections. *Journal of Clinical Microbiology*, 55(1):177–82, 2017.

[149] F. P. Breitwieser, C. A. Pardo, and S. L. Salzberg. Re-analysis of metagenomic sequences from acute flaccid myelitis patients reveals alternatives to enterovirus D68 infection. *F1000Research*, 4:180, 2015.

[150] S. L. Salzberg, F. P. Breitwieser, A. Kumar, H. Hao, P. Burger, F. J. Rodriguez, M. Lim, A. Quinones-Hinojosa, et al. Next-generation sequencing in neuropathologic diagnosis of infections of the nervous system. *Neurology Neuroimmunology & Neuroinflammation*, 3(4):e251, 2016.

[151] D. H. Huson, A. F. Auch, J. Qi, and S. C. Schuster. MEGAN analysis of metagenomic data. *Genome Research*, 17(3):377–86, 2007.

[152] D. H. Huson, S. Beier, I. Flade, A. Gorska, M. El-Hadidi, S. Mitra, H. J. Ruscheweyh, and R. Tappu. MEGAN Community Edition - interactive exploration and analysis of large-scale microbiome sequencing data. *PloS Computational Biology*, 12(6): e1004957, 2016.

[153] D. H. Huson and S. Mitra. Introduction to the analysis of environmental sequences: metagenomics with MEGAN. *Methods in Molecular Biology*, 856:415–29, 2012.

[154] D. H. Huson and N. Weber. Microbial community analysis using MEGAN. *Methods in Enzymology*, 531:465–85, 2013.

[155] S. Roux, J. Tournayre, A. Mahul, D. Debroas, and F. Enault. Metavir 2: new tools for viral metagenome comparison and assembled virome analysis. *BMC Bioinformatics*, 15:76, 2014.

[156] P. Skewes-Cox, T. J. Sharpton, K. S. Pollard, and J. L. DeRisi. Profile hidden Markov models for the detection of viruses within metagenomic sequence data. *PloS One*, 9 (8):e105067, 2014.

[157] B. E. Dutilh, R. Schmieder, J. Nulton, B. Felts, P. Salamon, R. A. Edwards, and J. L. Mokili. Reference-independent comparative metagenomics using cross-assembly: crAss. *Bioinformatics*, 28(24):3225–31, 2012.

[158] S. H. Tausch, B. Y. Renard, A. Nitsche, and P. W. Dabrowski. RAMBO-K: Rapid and sensitive removal of background sequences from next generation sequencing data. *PloS One*, 10(9):e0137896, 2015.

[159] Y. Li, H. Wang, K. Nie, C. Zhang, Y. Zhang, J. Wang, P. Niu, and X. Ma. VIP: an integrated pipeline for metagenomics of virus identification and discovery. *Scientific Reports*, 6:23774, 2016.

[160] S. Roux, M. Faubladier, A. Mahul, N. Paulhe, A. Bernard, D. Debroas, and F. Enault. Metavir: a web server dedicated to virome analysis. *Bioinformatics*, 27(21):3074–5, 2011.

[161] A. D. Kostic, A. I. Ojesina, C. S. Pedamallu, J. Jung, R. G. Verhaak, G. Getz, and M. Meyerson. PathSeq: software to identify or discover microbes by deep sequencing of human tissue. *Nature Biotechnology*, 29(5):393–6, 2011.

[162] K. E. Wommack, J. Bhavsar, S. W. Polson, J. Chen, M. Dumas, S. Srinivasiah, M. Furman, S. Jamindar, and D. J. Nasko. VIROME: a standard operating procedure for analysis of viral metagenome sequences. *Standards in Genomic Sciences*, 6(3): 427–39, 2012.

[163] G. Zhao, S. Krishnamurthy, Z. Cai, V. L. Popov, A. P. Travassos da Rosa, H. Guzman, S. Cao, H. W. Virgin, R. B. Tesh, and D. Wang. Identification of novel viruses using VirusHunter - an automated data analysis pipeline. *PloS One*, 8(10):e78470, 2013.

[164] M. Norling, O. E. Karlsson-Lindsjo, H. Gourle, E. Bongcam-Rudloff, and J. Hayer. MetLab: An in silico experimental design, simulation and analysis tool for viral metagenomics studies. *PloS One*, 11(8):e0160334, 2016.

[165] J. M. Alves, A. L. de Oliveira, T. O. Sandberg, J. L. Moreno-Gallego, M. A. de Toledo, E. M. de Moura, L. S. Oliveira, A. M. Durham, et al. GenSeed-HMM: A tool for progressive assembly using profile HMMs as seeds and its application in Alpavirinae viral discovery from metagenomic data. *Frontiers in Microbiology*, 7:269, 2016.

[166] G. Zhao, G. Wu, E. Lim, L. Droit, S. Krishnamurthy, D. H. Barouch, H. W. Virgin, and D. Wang. VirusSeeker, a computational pipeline for virus discovery and virome composition analysis. *Virology*, 503:21–30, 2017.

[167] X. Deng, S. N. Naccache, T. Ng, S. Federman, L. Li, C. Y. Chiu, and E. L. Delwart. An ensemble strategy that significantly improves de novo assembly of microbial genomes from metagenomic next-generation sequencing data. *Nucleic Acids Research*, 43(7): e46, 2015.

[168] V. C. Piro, M. Matschkowski, and B. Y. Renard. MetaMeta: integrating metagenome analysis tools to improve taxonomic profiling. *Microbiome*, 5(1):101, 2017.

[169] R. Naeem, M. Rashid, and A. Pain. READSCAN: a fast and scalable pathogen discovery program with accurate genome relative abundance estimation. *Bioinformatics*, 29(3):391–2, 2013.

[170] T. A. Freitas, P. E. Li, M. B. Scholz, and P. S. Chain. Accurate read-based metagenome characterization using a hierarchical suite of unique signatures. *Nucleic Acids Research*, 43(10):e69, 2015.

[171] T. H. Dadi, B. Y. Renard, L. H. Wieler, T. Semmler, and K. Reinert. SLIMM: species level identification of microorganisms from metagenomes. *PeerJ*, 5:e3138, 2017.

[172] A. Y. Lee, C. S. Lee, and R. N. Van Gelder. Scalable metagenomics alignment research tool (SMART): a scalable, rapid, and complete search heuristic for the classification of metagenomic sequences from complex sequence populations. *BMC Bioinformatics*, 17:292, 2016.

[173] B. Fosso, M. Santamaria, M. D'Antonio, D. Lovero, G. Corrado, E. Vizza, N. Passaro, A. R. Garbuglia, et al. MetaShot: an accurate workflow for taxon classification of host-associated microbiome from shotgun metagenomic data. *Bioinformatics*, 33(11): 1730–2, 2017.

[174] V. C. Piro, M. S. Lindner, and B. Y. Renard. DUDes: a top-down taxonomic profiler for metagenomics. *Bioinformatics*, 32(15):2272–80, 2016.

[175] D. T. Truong, E. A. Franzosa, T. L. Tickle, M. Scholz, G. Weingart, E. Pasolli, A. Tett, C. Huttenhower, and N. Segata. MetaPhlAn2 for enhanced metagenomic taxonomic profiling. *Nature Methods*, 12(10):902–3, 2015.

[176] C. Hong, S. Manimaran, Y. Shen, J. F. Perez-Rogers, A. L. Byrd, E. Castro-Nallar, K. A. Crandall, and W. E. Johnson. PathoScope 2.0: a complete computational framework for strain identification in environmental or clinical sequencing samples. *Microbiome*, 2:33, 2014.

[177] A. L. Byrd, J. F. Perez-Rogers, S. Manimaran, E. Castro-Nallar, I. Toma, T. McCaffrey, M. Siegel, G. Benson, K. A. Crandall, and W. E. Johnson. Clinical PathoScope: rapid alignment and filtration for accurate pathogen identification in clinical samples using unassembled sequencing data. *BMC Bioinformatics*, 15:262, 2014.

[178] O. E. Francis, M. Bendall, S. Manimaran, C. Hong, N. L. Clement, E. Castro-Nallar, Q. Snell, G. B. Schaalje, et al. Pathoscope: species identification and strain attribution with unassembled sequencing data. *Genome Research*, 23(10):1721–9, 2013.

[179] M. S. Lindner and B. Y. Renard. Metagenomic abundance estimation and diagnostic testing on species level. *Nucleic Acids Research*, 41(1):e10, 2013.

[180] S. N. Naccache, S. Federman, N. Veeraraghavan, M. Zaharia, D. Lee, E. Samayoa, J. Bouquet, A. L. Greninger, et al. A cloud-compatible bioinformatics pipeline for ultrarapid pathogen identification from next-generation sequencing of clinical samples. *Genome Research*, 24(7):1180–92, 2014.

[181] S. Flygare, K. Simmon, C. Miller, Y. Qiao, B. Kennedy, T. Di Sera, E. H. Graf, K. D. Tardif, et al. Taxonomer: an interactive metagenomics analysis portal for universal pathogen detection and host mRNA expression profiling. *Genome Biology*, 17(1):111, 2016.

[182] D. E. Wood, J. Lu, and B. Langmead. Improved metagenomic analysis with kraken 2. *bioRxiv*. doi: 10.1101/762302, 2019.

[183] V. C. Piro, T. H. Dadi, E. Seiler, K. Reinert, and B. Y. Renard. ganon: precise metagenomics classification against large and up-to-date sets of reference sequences. *bioRxiv*. doi: 10.1101/406017, 2019.

[184] D. E. Wood and S. L. Salzberg. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biology*, 15(3):R46, 2014.

[185] Y. Zheng, S. Gao, C. Padmanabhan, R. Li, M. Galvez, D. Gutierrez, S. Fuentes, K. S. Ling, J. Kreuze, and Z. Fei. VirusDetect: An automated pipeline for efficient virus discovery using deep sequencing of small RNAs. *Virology*, 500:130–8, 2017.

[186] M. Scheuch, D. Hoper, and M. Beer. RIEMS: a software pipeline for sensitive and comprehensive taxonomic classification of reads from metagenomics datasets. *BMC Bioinformatics*, 16:69, 2015.

[187] P. Menzel, K. L. Ng, and A. Krogh. Fast and sensitive taxonomic classification for metagenomics with Kaiju. *Nature Communications*, 7:11257, 2016.

[188] F. P. Breitwieser, J. Lu, and S. L. Salzberg. A review of methods and databases for metagenomic classification and assembly. *Briefings in Bioinformatics*, 2017.

[189] B. E. Dutilh, A. Reyes, R. J. Hall, and K. L. Whiteson. Editorial: Virus discovery by metagenomics: The (im)possibilities. *Frontiers in Microbiology*, 8:1710, 2017.

[190] K. G. Frey, J. E. Herrera-Galeano, C. Redden, T. V. Luu, S. L. Servetas, A. J. Mateczun, V. P. Mokashi, and K. A. Bishop-Lilly. Comparison of three next-generation sequencing platforms for metagenomic sequencing and identification of pathogens in blood. *BMC Genomics*, 15:96, 2014.

[191] M. Lecuit and M. Eloit. The potential of whole genome NGS for infectious disease diagnosis. *Expert Review of Molecular Diagnostics*, 15(12):1517–9, 2015.

[192] M. Lecuit and M. Eloit. The diagnosis of infectious diseases by whole genome next generation sequencing: a new era is opening. *Frontiers in Cellular and Infection Microbiology*, 4:25, 2014.

[193] J. L. Mokili, F. Rohwer, and B. E. Dutilh. Metagenomics and future perspectives in virus discovery. *Current Opinion in Virology*, 2(1):63–77, 2012.

[194] S. Roux, J. B. Emerson, E. A. Eloe-Fadrosh, and M. B. Sullivan. Benchmarking viromics: an in silico evaluation of metagenome-enabled estimates of viral community composition and diversity. *PeerJ*, 5:e3817, 2017.

[195] L. A. Snyder, N. Loman, M. J. Pallen, and C. W. Penn. Next-generation sequencing - the promise and perils of charting the great microbial unknown. *Microbial Ecology*, 57(1):1–3, 2009.

[196] F. P. Breitwieser, M. Pertea, A. V. Zimin, and S. L. Salzberg. Human contamination in bacterial genomes has created thousands of spurious proteins. *Genome Research*, 29(6):954–60, 2019.

[197] J. Quick, P. Ashton, S. Calus, C. Chatt, S. Gossain, J. Hawker, S. Nair, K. Neal, et al. Rapid draft sequencing and real-time nanopore sequencing in a hospital outbreak of Salmonella. *Genome Biology*, 16:114, 2015.

[198] M. D. Cao, D. Ganesamoorthy, A. G. Elliott, H. Zhang, M. A. Cooper, and L. J. Coin. Streaming algorithms for identification of pathogens and antibiotic resistance potential from real-time MinION sequencing. *GigaScience*, 5(1):32, 2016.

[199] M. Loose, S. Malla, and M. Stout. Real-time selective sequencing using nanopore technology. *Nature Methods*, 13(9):751–4, 2016.

[200] R. D. Stewart and M. Watson. poRe GUIs for parallel and real-time processing of MinION sequence data. *Bioinformatics*, 33(14):2207–8, 2017.

[201] T. P. Loka, S. H. Tausch, and B. Y. Renard. Reliable variant calling during runtime of Illumina sequencing. *bioRxiv*. doi: 10.1101/387662, 2018.

[202] J. R. Brister, D. Ako-Adjei, Y. Bao, and O. Blinkova. NCBI viral genomes resource. *Nucleic Acids Research*, 43:D571–7, 2015.

[203] Genomes Project Consortium, A. Auton, L. D. Brooks, R. M. Durbin, E. P. Garrison, H. M. Kang, J. O. Korbel, J. L. Marchini, et al. A global reference for human genetic variation. *Nature*, 526(7571):68–74, 2015.

[204] A. M. Bolger, M. Lohse, and B. Usadel. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics*, 30(15):2114–20, 2014.

[205] M. S. Lindner and B. Y. Renard. Metagenomic profiling of known and unknown microbes with microbeGPS. *PloS One*, 10(2):e0117711, 2015.

[206] M. Bostock, V. Ogievetsky, and J. Heer. D(3): Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–9, 2011.

[207] Belgian Biosafety Server. Belgian classification for micro-organisms based on their biological risks, https://www.biosafety.be/node/286. accessed 19.03.2018, 2008.

[208] J. Lu, F. P. Breitwieser, P. Thielen, and S. L. Salzberg. Bracken: estimating species abundance in metagenomics data. *PeerJ Computer Science*, 3:e104, 2017.

[209] N. A. O'Leary, M. W. Wright, J. R. Brister, S. Ciufo, D. Haddad, R. McVeigh, B. Rajput, B. Robbertse, et al. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation.

[210] C. Kohl, M. Eldegail, I. Mahmoud, L. Schrick, A. Radonic, P. Emmerich, T. Rieger, S. Gunther, A. Nitsche, and A. A. Osman. Crimean congo hemorrhagic fever, 2013 and 2014 sudan. *International Journal of Infectious Diseases*, 53:9, 2016.

[211] J. Klenner, C. Kohl, P. W. Dabrowski, and A. Nitsche. Comparing viral metagenomic extraction methods. *Current Issues in Molecular Biology*, 24:59–70, 2017.

[212] A. Andrusch, P. W. Dabrowski, J. Klenner, S. H. Tausch, C. Kohl, A. A. Osman, B. Y. Renard, and A. Nitsche. PAIPline: pathogen identification in metagenomic and clinical next generation sequencing samples. *Bioinformatics*, 34(17):i715–21, 2018.

[213] H. S. Edwards, R. Krishnakumar, A. Sinha, S. W. Bird, K. D. Patel, and M. S. Bartsch. Real-time selective sequencing with RUBRIC: Read until with basecall and reference-informed criteria. *bioRxiv*. doi: 10.1101/460014, 2019.

[214] K. Lohmann and C. Klein. Next generation sequencing and the future of genetic diagnosis. *Neurotherapeutics*, 11(4):699–707, 2014.

[215] R. Harripaul, A. Noor, M. Ayub, and J. B. Vincent. The use of next-generation sequencing for research and diagnostics for intellectual disability. *Cold Spring Harbor Perspectives in Medicine*, 7(3), 2017.

[216] Y. Cho, C. H. Lee, E. G. Jeong, M. H. Kim, J. H. Hong, Y. Ko, B. Lee, G. Yun, et al. Prevalence of rare genetic variations and their implications in NGS-data interpretation. *Scientific Reports*, 7(1):9810, 2017.

[217] A. Fernandez-Marmiesse, S. Gouveia, and M. L. Couce. NGS technologies as a turning point in rare disease research , diagnosis and treatment. *Current Medicinal Chemistry*, 25(3):404–32, 2018.

[218] C. Di Resta, S. Galbiati, P. Carrera, and M. Ferrari. Next-generation sequencing approach for the diagnosis of human diseases: open challenges and new opportunities. *eJIFCC*, 29(1):4–14, 2018.

[219] D. Bick, M. Jones, S. L. Taylor, R. J. Taft, and J. Belmont. Case for genome sequencing in infants and children with rare, undiagnosed or genetic diseases. *Journal of Medical Genetics*, 2019.

[220] T. J. Ley, E. R. Mardis, L. Ding, B. Fulton, M. D. McLellan, K. Chen, D. Dooling, B. H. Dunford-Shore, et al. DNA sequencing of a cytogenetically normal acute myeloid leukaemia genome. *Nature*, 456(7218):66–72, 2008.

[221] C. Meldrum, M. A. Doyle, and R. W. Tothill. Next-generation sequencing for cancer diagnostics: a practical perspective. *The Clinical Biochemist Reviews*, 32(4):177–95, 2011.

[222] L. A. Garraway and E. S. Lander. Lessons from the cancer genome. *Cell*, 153(1): 17–37, 2013.

[223] R. Kamps, R. D. Brandao, B. J. Bosch, A. D. Paulussen, S. Xanthoulea, M. J. Blok, and A. Romano. Next-generation sequencing in oncology: Genetic diagnosis, risk

prediction and cancer classification. *International Journal of Molecular Sciences*, 18 (2), 2017.

[224] M. F. Berger and E. R. Mardis. The emerging clinical relevance of genomics in cancer medicine. *Nature Reviews Clinical Oncology*, 15(6):353–65, 2018.

[225] M. R. Wilson, S. N. Naccache, E. Samayoa, M. Biagtan, H. Bashir, G. Yu, S. M. Salamat, S. Somasekar, et al. Actionable diagnosis of neuroleptospirosis by next-generation sequencing. *The New England Journal of Medicine*, 370(25):2408–17, 2014.

[226] M. J. Pallen. Diagnostic metagenomics: potential applications to bacterial, viral and parasitic infections. *Parasitology*, 141(14):1856–62, 2014.

[227] C. Y. Chiu and S. A. Miller. Clinical metagenomics. *Nature Reviews Genetics*, 20(6): 341–55, 2019.

[228] QIAGEN, Inc. QIAseq 1-Step Amplicon Library Preparation Handbook, 2016. Pub. No. HB-2028-001.

[229] San Diego, CA: Illumina, Inc. Nextera™ DNA Flex Library Preparation Kit, 2017. Pub. No. 770-2017-011-A.

[230] K. D. Pruitt, T. Tatusova, and D. R. Maglott. NCBI reference sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Research*, 33:D501–4, 2005.

[231] T. H. Dadi, E. Siragusa, V. C. Piro, A. Andrusch, E. Seiler, B. Y. Renard, and K. Reinert. DREAM-Yara: an exact read mapper for very large databases with short update time. *Bioinformatics*, 34(17):i766–72, 2018.

[232] A. N. Kolmogorov. On tables of random numbers. *Sankhyā: The Indian Journal of Statistics, Series A*, 25:369–76, 1963.

[233] A. Zielezinski, S. Vinga, J. Almeida, and W. M. Karlowski. Alignment-free sequence comparison: benefits, applications, and tools. *Genome Biology*, 18(1):186, 2017.

[234] J. Ren, X. Bai, Y. Y. Lu, K. Tang, Y. Wang, G. Reinert, and F. Sun. Alignment-free sequence analysis and applications. *Annual Review of Biomedical Data Science*, 1(1): 93–114, 2018.

[235] J. M. Bartoszewicz, A. Seidel, R. Rentzsch, and B. Y. Renard. DeePaC: Predicting pathogenic potential of novel DNA with reverse-complement neural networks. *Bioinformatics*, 2019.

[236] S. S. Johnson, E. Zaikova, D. S. Goerlitz, Y. Bai, and S. W. Tighe. Real-time DNA sequencing in the Antarctic Dry Valleys using the Oxford Nanopore sequencer. *Journal of Biomolecular Techniques*, 28(1):2–7, 2017.

[237] S. L. Castro-Wallace, C. Y. Chiu, K. K. John, S. E. Stahl, K. H. Rubins, A. B. R. McIntyre, J. P. Dworkin, M. L. Lupisella, et al. Nanopore DNA sequencing and genome assembly on the International Space Station. *Scientific Reports*, 7(1):18022, 2017.

[238] A. S. Gargis, B. Cherney, A. B. Conley, H. P. McLaughlin, and D. Sue. Rapid detection of genetic engineering, structural variation, and antimicrobial resistance markers in bacterial biothreat pathogens by nanopore sequencing. *Scientific Reports*, 9(1):13501, 2019.

[239] H. Safi, P. Gopal, S. Lingaraju, S. Ma, C. Levine, V. Dartois, M. Yee, L. Li, et al. Phase variation in Mycobacterium tuberculosis glpK produces transiently heritable drug tolerance. *Proceedings of the National Academy of Sciences of the United States of America*, 116(39):19665–74, 2019.

[240] D. Antipov, A. Korobeynikov, J. S. McLean, and P. A. Pevzner. hybridSPAdes: an algorithm for hybrid assembly of short and long reads. *Bioinformatics*, 32(7):1009–15, 2016.

[241] R. R. Wick, L. M. Judd, C. L. Gorrie, and K. E. Holt. Unicycler: Resolving bacterial genome assemblies from short and long sequencing reads. *PloS Computational Biology*, 13(6):e1005595, 2017.

[242] A. Di Genova, G. A. Ruz, M. F. Sagot, and A. Maass. Fast-SG: an alignment-free algorithm for hybrid assembly. *GigaScience*, 7(5), 2018.

[243] H. Lu, F. Giordano, and Z. Ning. Oxford Nanopore MinION sequencing and genome assembly. *Genomics, Proteomics & Bioinformatics*, 14(5):265–79, 2016.

[244] R. R. Wick, L. M. Judd, C. L. Gorrie, and K. E. Holt. Completing bacterial genome assemblies with multiplex MinION sequencing. *Microbial Genomics*, 3(10):e000132, 2017.

[245] M. H. Tan, C. M. Austin, M. P. Hammer, Y. P. Lee, L. J. Croft, and H. M. Gan. Finding Nemo: hybrid assembly with Oxford Nanopore and Illumina reads greatly improves the clownfish (Amphiprion ocellaris) genome assembly. *GigaScience*, 7(3):1–6, 2018.

[246] S. Goldstein, L. Beka, J. Graf, and J. L. Klassen. Evaluation of strategies for the assembly of diverse bacterial genomes using MinION long-read sequencing. *BMC Genomics*, 20(1):23, 2019.

Zusammenfassung

‘Next-Generation Sequencing’, im Speziellen die Illumina Sequenzierung, ist die derzeit meistgenutzte DNA-Sequenziertechnologie. Jedoch sind für zeitkritische Analysen aufgrund des sequentiellen Paradigmas der Datenerzeugung und -analyse die Durchlaufzeiten zu hoch. In der klinischen Anwendung und bei Ausbrüchen von Infektionskrankheiten ist es entscheidend, die Zeit vom Probeneingang zum Analyseergebnis zu verkürzen um Patienten optimal zu behandeln und einer weitere Krankheitsausbreitung zu verhindern. Gleichzeitig ist eine Analyse auf Nukleotidebene erforderlich um eine Spezies-Level-Klassifizierung und die Bestimmung spezifischer Eigenschaften, wie z.B. antimikrobiellen Resistenzen, zu ermöglichen. Um eine frühere Verfügbarkeit von Analyse-Ergebnissen zu erreichen wurde die Echtzeit-Alignierungssoftware HiLive entwickelt, welche DNA-Sequenzen während der Sequenzierung aligniert. Jedoch lieferte HiLive die Ergebnisse bislang nur am Ende eines Sequenzierlaufs und hatte keine ausreichende Auflösung und Skalierbarkeit.

In dieser Arbeit präsentiere ich einen neuen Echtzeit-Alignierungsalgorithmus, der in HiLive2 implementiert wurde. HiLive2 basiert auf dem FM-index, kann zu jedem Zeitpunkt der Sequenzierung Ergebnisse liefern und erreicht eine höhere Skalierbarkeit der Größe von Referenzdatenbank und Datensatz. Durch die Detektion von Varianten basierend auf den Echtzeit-Alignierungen von humanen Sequenzierdaten zeige ich, dass HiLive2 qualitativ hochwertige Folgeanalysen ermöglicht. Außerdem stelle ich PathoLive vor, eine Pipeline zur Echtzeit-Identifizierung von Krankheitserregern aus metagenomischen Datensätzen. Basierend auf den Ergebnissen von HiLive2 führt PathoLive eine gewichtete Einstufung der identifizierten Organismen durch. Dabei werden Sequenzen, die auch in Proben von gesunden Menschen vorkommen, in den Ergebnissen weniger stark berücksichtigt. PathoLive bietet eine intuitive und interaktive Visualisierung, welche die Interpretation der Ergebnisse erleichtert. Ich zeige, dass PathoLive basierend auf nur wenigen Dutzend Sequenzen die Identifizierung des Krim-Kongo-Hämorrhagisches-Fieber-Virus in einer Probe aus dem Sudan ermöglicht. Neben den analytischen Herausforderungen sind Patientenproben im Hinblick auf den Datenschutz problematisch, da die Daten des humanen Wirts zur Identifizierung des Patienten verwendet werden könnten. Für diese Problematik präsentiere ich PriLive, welches noch während des Sequenzierlaufs das Entfernen humaner Sequenzen aus den Rohdaten ermöglicht. Hierdurch kann ein deutlich höheres Datenschutzniveau erreicht werden als mit herkömmlichen *post hoc* Ansätzen, da die humanen Sequenzen auch während des Sequenzierungsprozesses zu keinem Zeitpunkt in voller Länge vorliegen.

Eigenständigkeitserklärung

Ich versichere, dass ich die hier vorgelegte Dissertation selbstständig angefertigt habe und die benutzten Quellen und Hilfsmittel vollständig angegeben sind. Ein Promotionsverfahren wurde zu keinem früheren Zeitpunkt an einer anderen in- oder ausländischen Hochschule oder bei einem anderen Fachbereich beantragt. Die Bestimmungen der Promotionsordnung sind mir bekannt.

Tobias Pascal Loka, Berlin, 29. Oktober 2019