Development of bioinformatics tools for the rapid and sensitive detection of known and unknown pathogens from next generation sequencing data

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

von Simon Tausch

Berlin
August 2018

Erstgutachter: PD. Dr. Bernhard Y. Renard
Zweitgutachter: Prof. Dr. Andreas Nitsche

Tag der Disputation: 12.12.2018

Betreuer: **PD Dr. Bernhard Renard**

Acknowledgements

First and foremost, I want to thank my supervisors, Andreas Nitsche and Bernhard Renard, for their active support, their advice and the excellent scientific environment they provided me with. While always having an open ear for my questions, they also gave me the freedom to follow my own ideas. Both of them never hesitated to entrust me with any project I showed interest in, which gave me the chance to gain insight into a variety of highly interesting topics. Beyond that, both of them provided a cordial atmosphere.

Furthermore, I want to thank my mentor Wojtek Dabrowski, without whom I would probably never have gotten in touch with the field of this work. He was never short on a good advice, nor of chocolate and coffee.

I also want to thank my fellow PhD students and other colleagues, who created a great working environment. Andreas Andrusch, with whom I could debate any idea that came to my mind; Tobias Loka, who was always up for a constructive discussion and worked most thoroughly on the code of HiLive; Benjamin Strauch for the implementation of LiveKraken and his readiness to help out in urgent situations; Claudia Kohl and Jeanette Klenner for a warm welcome and the opportunity to work on their data; Kathrin Trappe, who always was a great partner for a frank and honest discussion; Christine Jandrasits, who always found the time to organize extracurricular activities, helping to shape the great atmosphere in the group; Ursula Erikli for copy-editing my manuscripts and her help with any administrative burden; Ilka Schlenther, Marica Grossegesse, Annika Brinkmann, Aleksandar Radonic and many others who remain unmentioned but were around and kept me smiling over the last years.

Credit also goes to all students who contributed projects or theses to my research, namely Frederike Heinitz, Sophie Meier zu Ummeln, Jakob Schulze, Matthias Wajnberg and Kristina Kirsten. Although not all their work could be part of this thesis, it was a pleasure working with each of them and the generated knowledge has been of great value, whether for this thesis or for follow-up projects. Special thanks go to Frederike Heinitz for copyediting this work.

I would not have gained such a broad insight into various fields of work without the different collaboration partners, namely Prof. José Esparza, Prof. Clarissa Damaso, Prof. Manja Marz, Dr. Gudrun Wibbelt and Dr. Cesare Gruber among others. Each of them allowed me a glimpse at different topics, institutions and working cultures.

The help of the technical assistants shall not be left unmentioned: Jule Hinzmann, who has worked passionately on the squirrelpox project and was always open for a chat as well as Angelina Targosz and Jule Tesch, who sequenced countless samples together with Jule Hinzmann and therefore provided the basis for my work.

Last but not least, I am deeply grateful to my family – Ingrid, Hubert and Sarah Tausch for their wonderful support in whatever I did, do and will do. And to my friends – you know who you are – for always being there when I need to free my mind.

Greatest thanks to my beloved wife, Sandra Tausch, who is always there for me and made me the greatest present with the birth of our son Lion Tausch.

Lion, I also want to thank you. Even though not always making things easier, you provide me with unlimited motivation and joy. I dedicate this work to you.

Abstract

Infectious diseases still remain one of the main causes of death across the globe. Despite huge advances in clinical diagnostics, establishing a clear etiology remains impossible in a proportion of cases. Since the emergence of next generation sequencing (NGS), a multitude of new research fields based on this technology have evolved. Especially its application in metagenomics – denoting the research on genomic material taken directly from its environment – has led to a rapid development of new applications. Metagenomic NGS has proven to be a promising tool in the field of pathogen related research and diagnostics.

In this thesis, I present different approaches for the detection of known and the discovery of unknown pathogens from NGS data. These contributions subdivide into three newly developed methods and one publication on a real-world use case of methodology we developed and data analysis based on it.

First, I present LiveKraken, a real-time read classification tool based on the core algorithm of Kraken. LiveKraken uses streams of raw data from Illumina sequencers to classify reads taxonomically. This way, we are able to produce results identical to those of Kraken the moment the sequencer finishes. We are furthermore able to provide comparable results in early stages of a sequencing run, allowing saving up to a week of sequencing time. While the number of classified reads grows over time, false classifications appear in negligible numbers and proportions of identified taxa are only affected to a minor extent.

In the second project, we designed and implemented PathoLive, a real-time diagnostics pipeline which allows the detection of pathogens from clinical samples before the sequencing procedure is finished. We adapted the core algorithm of HiLive, a real-time read mapper, and enhanced its accuracy for our use case. Furthermore, probably irrelevant sequences automatically marked. The results are visualized in an interactive taxonomic tree that provides an intuitive overview and detailed metrics regarding the relevance of each identified pathogen. Testing PathoLive on the sequencing of a real plasma sample spiked with viruses, we could prove that we ranked the results more accurately throughout the complete sequencing run than any other tested tool did at the end of the sequencing run. With PathoLive, we shift the focus of NGS-based diagnostics from read quantification towards a more meaningful assessment of results in unprecedented turnaround time.

The third project aims at the detection of novel pathogens from NGS data. We developed RAMBO-K, a tool which allows rapid and sensitive removal of unwanted host sequences from NGS datasets. RAMBO-K is faster than any tool we tested, while showing a consistently high sensitivity and specificity across different datasets. RAMBO-K rapidly and reliably separates reads from different species. It is suitable as a straightforward standard solution for workflows dealing with mixed datasets.

In the fourth project, we used RAMBO-K as well as several other data analyses to discover Berlin squirrelpox virus, a deviant new poxvirus establishing a new genus of *poxviridae*. Near Berlin, Germany, several juvenile red squirrels (*Sciurus vulgaris*) were found with moist, crusty skin lesions. Histology, electron microscopy, and cell culture isolation revealed an *orthopoxvirus*-like infection. After standard workflows yielded no significant results, poxviral reads were assigned using RAMBO-K, enabling the assembly of the genome of the novel virus.

With these projects, we established three new application-related methods each of which closes different research gaps. Taken together, we enhance the available repertoire of NGS-based pathogen related research tools and alleviate and fasten a variety of research projects.

Contents

1 Introduction

1.1 Metagenomics

Metagenomics denotes research of genetic material taken directly from its environment without prior cultivation [1]. Metagenomic datasets are therefore mixtures of the genomes of heterogeneous communities of organisms. The term was introduced by Handelsman et al. in 1998 [2]. Using the literal translation from the Greek, it describes studies which are "beyond" genomics, aiming at analyzing more than a single genome at once [3].

Metagenomics have enabled answering a variety of scientific questions. Two of the most influential use cases I will focus on in particular are the study of microbial community composition and the study of genomes of organisms which cannot readily be cultured [4]. It is supposed that this applies to 99.8% of all microbes [5]. Mentionable, viruses can never be cultured independently of a host cell. But for the absolute majority of viruses, not even the host can be grown in culture [6]. Metagenomics therefore opens a door to a great unknown microbial diversity.

The first metagenomic studies were based on shotgun Sanger sequencing (s. Chapter 1.4) and therefore limited in their sensitivity [3]. To that time, most successful studies targeted low-complexity microbiomes such as these of geysers or deep sea water [4]. Still, Venter et al. were already able to study the highly complex freshwater microbiome of the Sargasso Sea in 2004 [7]. The progress in sequencing technology allowed studying more and more complex environments.

One of the best studied environments in the field of metagenomics is the human body. The collective of all microbes inhabiting the human body are called the human microbiome [8]. While the human body consists of approximately 10^{13} cells, it is populated by 10 times as many bacteria and even 100 times as many viruses [9]. Although a large proportion of the human microbiome is not yet understood in detail, it is already known to have a major impact on human health. Amongst many others, there is proven correlation of the human microbiome and that of the oral microbiome and dental caries [10] or the gut microbiome on obesity [11]. Notably, pathogenic organisms such as bacteria or viruses are detectable as part of the microbiome in case of an acute infection.

1.2 NGS-based pathogen detection

One field of research which is rapidly growing and already reaches out into clinical practice is NGS-based pathogen detection. In extraordinary cases of infectious diseases, diagnostic analyses can be extremely difficult. Despite huge advances in clinical diagnostics, at some medical conditions no clear etiology can be established for up to 60% of the cases [12]. Especially if the causative agent of an infection is unknown or an infection is caused by an unsuspected pathogen, common methods such as non-multiplex PCR or antibody detection are doomed to fail, as these are restricted to test for one species at a time [13].

With metagenomic NGS, a hypothesis and culture free pathogen detection method has emerged. It is based on the identification of a pathogen's nucleic acids, be it deoxyribonucleic acids (DNA) or ribonucleic acids (RNA), in a patient sample. A classical metagenomic sample is sequenced, producing millions to billions of reads which stem from all species in the sample. The first challenge of this approach is therefore, that the majority of reads stems from the host genome in most cases. This is owed to the random sampling of reads from all nucleic acids in the sample, where the host genome is generally more abundant and magnitudes larger than the pathogen's genome. It has been proven that 0.00001%-0.7% of the total reads from a sequencing run may have decisive influence on a successful diagnosis [12, 14]. To compensate for this, a high sequencing depth is desirable.

This leads directly to the second main issue of NGS-based pathogen detection: At very high sequencing depths, the data will contain reads from all kinds of sources. These may range from contaminations introduced with sample-taking over lab contaminations to commonly seen organisms which are regularly colonizing a person [12, 15]. As even single reads may be relevant for a diagnosis, it is extremely difficult to automatically reject any read without risking losing valuable information. Furthermore, it is often impossible to automatically distinguish a normal colonization from acute infection. As an example, human papillomavirus, known as the causative agent of a variety of different diseases including cancer, was detected in 95% of subjects in a study, with no correlation to their disease status [16].

Thirdly, the overall turnaround time of sequencing a sample at the necessary depth is too long to get actionable results in cases of acute infections or outbreak scenarios in reasonable time. Despite the massive advances in NGS, sequencing a metagenome at very high sequencing depth may still take up to 11 days on an Illumina HiSeq. This is of course too large a timespan for most diagnoses. Additionally, library preparation, data analysis and evaluation of results further prolong the overall turnaround time. In order to tackle this problem, fast sequencing protocols or usage of third generation sequencers have been proposed [17, 18]. These approaches have been shown to work in different diagnostic settings, but as all of them rely on a downscaled sequencing depth and lower quality data, the risk of missing relevant pathogens is increased.

As the diagnosis can only be considered as finished after a final decision for one or more causative agents has been made, the overall turnaround time does not end when the bioinformatic analysis is completed. Bioinformatic analyses can only bring a first structure into a metagenomic dataset. We are nowhere near the point where a fully automated diagnosis could rely on bioinformatic analyses alone. It is thus very important to present the results of the bioinformatic pipelines in an understandable form to enable researchers and clinicians to get to actionable results quickly [14].

Another problem that has been stated by Dutilh et al. in [16] is the data privacy of the patient, as large proportions of their genome are sequenced as a byproduct of the metagenome sequencing. Different approaches have been implemented to solve this problem by removing human background reads from the dataset [19, 20].

Despite all described difficulties, NGS has already greatly contributed to many successful diagnoses. A variety of tools for this purpose already exist, each of them tackling some of the above-mentioned problems [21-28]. Especially in the field of encephalitis diagnostics, where hundreds of pathogens have been proven to be candidate causative pathogens, NGS based diagnostics is at the frontier to be incorporated into routine clinical application. A number of successful diagnoses has been described by Brown et al. [29]. The outstanding success of NGS-based diagnostics in this field of diseases may be owed to the relative cleanness of brain related samples, be it from a biopsy or from liquor. This can be explained by the blood-brain barrier, which keeps the brain free of most organisms which colonize other regions of the body. Yet, other diseases have been successfully diagnosed using metagenomic NGS as well. Some of these cases have been listed by Simner et al. [12]. Even commercial companies have already successfully conducted clinical tests based on NGS. For example, a commercial test was able to diagnose a sepsis induced by *Capnocytophaga canimorsus* [30]. Still, different methods such as polymerase chain reaction (PCR) are necessary and widely used for the confirmation of the results of NGS in diagnostic setups to date.

Concluding, it can be stated that NGS as a basis for pathogen diagnostics can be expected to become a very influential method over the next years, although several issues still have to be overcome.

1.3 Metagenomic data for virus discovery

Novel human pathogenic viruses, distinct from any known species, are regularly emerging in the human population. These are often zoonotic agents which can spread to humans if they have contact with infected animals, as global epidemics as SARS coronavirus [31, 32] but also local outbursts like that of ebola virus in West Africa in 2014/2015 [33, 34] strikingly proved. Furthermore, the emergence of immunosuppressive therapies has led to infections with normally nonpathogenic viruses [35]. To date, many of these pathogens have only been discovered after having caused serious harm. It is expected that there exist about 320.000 mammal-associated viral species – only about 3200 of which are known today [36, 37]. In total, viruses are expected to be the most abundant group of organisms on the world, totaling to approximately to 5.2-7.5×10^{31} particles [9].

The possibilities of NGS enabled a particularly large increase of the number of newly discovered viruses [9, 38, 39]. Since isolating and examining novel viruses in the wet lab is eminently difficult, the possibility to obtain a viral sequence directly from an infected host alleviates these studies in particular, as has been proven by several research projects [35]. One of the best known results is the crAssphage. It has been named by the cross-assembly strategy (crAss [40]) which was used to unveil this phage. Interestingly, it is the most abundant phage in human feces, totaling 1.68% of all reads from human fecal metagenome sequencing sample – despite its relatively small genome of under 100 kbp [41]. Still, it had not been noticed before the regular use of NGS.

Tackling this task, a variety of tools allowing the discovery of novel viruses from metagenomic samples have been developed [40, 42-56]. Still, this field of research faces many difficulties. The key hurdle is that finding viral reads in sequencing data is a needle-in-a-haystack problem. The evaluation of sequencing data from purified viral particles is comparably easy. Unfortunately, this purification in the wet lab is markedly difficult for samples containing a complex mixture of organisms [39, 57]. Furthermore, even the computational classification of reads to the highest taxonomic ranks – eukaryotes, bacteria, archaea and viruses – is a yet unsolved problem [58]. With this in mind, tools for virus discovery need to be able to handle complex mixed datasets.

Furthermore, viral genomes mutate at comparably high speed. Starting at 10^{-8} mutations per base per generation for DNA-viruses, RNA-viruses may have mutation rates up to 10^{-3} per base per generation [59]. This may result in genomes highly deviant from the next known reference. To handle this problem, virus discovery tools need to be very sensitive.

Even though great advances in the development of virus discovery methods have been made, analyses are still connected with complex manual work and may even fail completely in various cases due to the aforementioned difficulties.

1.4 Genome sequencing

Genome sequencing generally denotes the determination of the order of nucleic acids in large molecules. Throughout the last 40 years, several techniques have been developed to decipher the genetic code of life.

All existing sequencing technologies rely on the measurement of a technology-specific signal which determines a piece of a nucleic acid sequence. This signal is then translated into an alphabet consisting of the corresponding bases of the nucleic acid sequenced. This step is called base calling and may also include the assignment of quality values which allow conclusions on the error-proneness of a specific base.

Since it is generally uncommon (yet not impossible anymore) to sequence a complete genome in one piece, usually a molecule of interest is fragmented and deciphered piece by piece. This is either achieved by selection and sequencing of a desired part of the nucleic acid sequence, or by shotgun sequencing. The latter denotes the idea of randomly shearing a sequence into small fragments and sequencing those fragments. Albeit sounding counterintuitive, this idea facilitated the successful completion of the Human Genome Project [60, 61].

The very first technique for DNA sequencing was developed by Sanger et al. in 1977. It is based on the chain-termination method. In simple terms, a DNA template is amplified and then divided into four subsamples. Each of these is then mixed with all four standard deoxynucleotides and DNA polymerase. Additionally, one of the four corresponding dideoxynucleotides is added. When a dideoxynucleotide is added to the growing chain of bases, the polymerase reaction gets interrupted. As this happens randomly, products of random lengths are produced. All produced fragments are then submitted to a gel electrophoresis. Via the lengths of the fragments on the different bands of the electrophoresis, the order of nucleotides can be determined. This method was later improved and used for the first automated sequencing machine by Applied Biosystems in 1987. Sanger sequencing can produce reads of length of up to 1000 bp and has extremely low error rates, which is why it is still a commonly used method. On the downside, it is comparably slow and expensive compared to newer sequencing methods when measured per base [62].

Over the past 13 years, NGS has gained influence in a variety of research fields. It enables sequencing multiple DNA molecules in parallel, which is why it is also denoted as massively parallel sequencing. This parallelization resulted in a drop of sequencing cost and a raise of sequencing capability in an exponential scale.

There is a variety of technologies referred to under the term NGS. The major division of these runs between second generation sequencing also denoted as massively parallel sequencing or short read sequencing, and third generation sequencing which denotes technologies facilitating long-read sequencing from single molecules.

Second generation sequencing

Second generation sequencing was introduced in 2005 [62]. For the first time, it was possible to sequence millions of short nucleic acid sequence fragments in parallel, leading to a massive increase of sequencing capabilities at once. As second generation sequencing can only produce comparably short reads, it is perfectly compatible with the idea of shotgun sequencing.

The first Solexa sequencers were for example limited to read lengths of <25 basepairs (bp). Different competitors pushed these boundaries, so that it is by now possible to sequence up to 400bp per read, generating as many as a trillion bases per run with error rates far below 1% [63]. Illumina, which acquired Solexa in 2007 [64] has become the market leader in the field of second generation sequencing [65].

Illumina dye sequencing technology

Large parts of our work are closely connected to the Illumina sequencing technology. Not only do we build up on Illumina sequencing data, but in some projects we interact with the sequencers directly while they are running. Therefore, I will give a short introduction on Illumina sequencing technology in the following paragraphs, which are structurally oriented at Illumina's own descriptions [66, 67] and Canzar et al. [68].

Illumina's sequencing workflow is subdivided into three basic steps, as shown in Figure 1: (i) Library Preparation, (ii) Cluster Generation and (iii) Sequencing itself [66, 68]. Every sequencing experiment starts with the purification of nucleic acids from a sample. DNA can be sequenced directly while RNA needs to be translated into cDNA first [69].

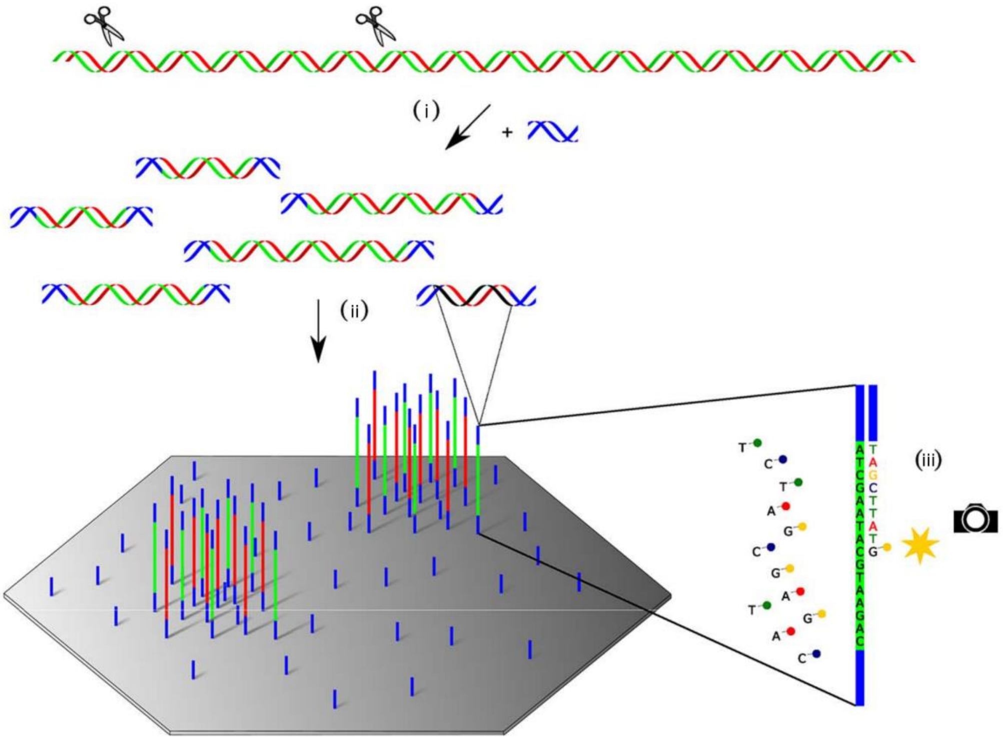


Figure 1. Workflow of the Illumina sequencing process. (i) Library preparation: Fragmentation of nucleic acids and ligation of adapter sequences, (ii) cluster generation on the flow cell, and (iii) sequencing by synthesis itself. ©2015 IEEE. Adapted and reprinted with friendly permission from Stefan Canzar and Steven Salzberg.

In step (i), the purified DNA is then randomly sheared into shorter fragments either physically, e.g. by sonication, or enzymatically [70]. Adapters, sequencing primers and indices are ligated on both ends of these fragments. Indices denote short nucleotide sequences which can be used to distinguish different samples which are sequenced together. This process is called multiplexing, while the assignment of reads to their indices after sequencing is called demultiplexing. Methods combining fragmentation and adapter ligation are also called "tagmentation" methods. The target DNA is located between the adapters and also denoted as "insert". The complete adapter-ligated fragments are amplified by PCR and get purified afterwards.

For the Cluster Generation step (ii), the prepared library is submitted to the sequencer. The chemical reaction underlying the sequencing process is conducted on a so called flow cell. A flow cell is a glass slide with a varying number of lanes, depending on the instrument and the sequencing mode. On the surface of the lanes, two types of short oligonucleotides are attached. Each of them is the complementary sequence of one of the adapters on the fragments. For hybridization, the first adapter is ligated to the first type of oligonucleotides. Then, a polymerase creates a complement of the hybridized template strand. The template strand is then washed away, leaving only the complementary strand. The second adapter region is then hybridized to the second type of oligonucleotides on the flow cell, forming a bridge, which is again turned into a double-stranded DNA by a polymerase. By repeating this so called bridge amplification step several times, large numbers of identical sequences and their complements are produced forming clusters on a flow cell. After the bridge amplification step, all complementary strands from the original template are washed off. All of this happens for all clusters simultaneously, resulting in one cluster of identical sequences for each of the billions of fragments.

The sequencing process itself (iii) begins with the binding of the first sequencing primer to the primer region of the template. A polymerase then synthesizes the complementary strand of the template, using fluorescently tagged nucleotides with a reversible terminator. This ensures that only one base is attached per cycle, which is always the complementary base of the corresponding base in the template. Then, the sequencing machine detects which base has been attached to a cluster in a given cycle by the fluorescent signal sent out by a complete cluster on excitation. Based on the measured signal of each cluster, the base call including a measure of base quality for a given cycle is calculated and saved. The first few bases of a template are used to identify the clusters on a flow cell, as their location is not predefined. Therefore, the signals sent out from a cluster in the early cycles must differ from those of a neighboring cluster to enable successful cluster distinction. Afterwards, the terminator is removed and the next cycle starts. After the first read has been sequenced, usually after 50-350 cycles, the product of the polymerase is removed. Then, the first index is sequenced in the same manner as the read before. After this product has been removed as well, the template is bound to the second type of oligonucleotides on the flow cell. Then, the second index is sequenced and the product is removed. A polymerase produces the complement of the bridged template and the original template is then removed, leaving only the complementary strands on the flow cell. For the last step of the procedure, the second read is sequenced in the same way as the first read at the beginning of the process. [66-68]

For single-end-read sequencing, protocols and other types of indexing, the procedure slightly differs.

Notably, the sequencing process is executed cycle-wise for all clusters at the same time, which is denoted as massively parallel sequencing. Billions of reads are therefore elongated by one base per cycle in parallel.

Third generation sequencing

Third generation sequencing was first introduced by PacBio in 2011 and enables read lengths of >10 kilobasepairs (kbp). Oxford Nanopore, the second big player in the field of third generation sequencing, claims to achieve read lengths up to 950 kbp [71]. This advantage comes at cost of comparably low throughput rates of less than 10 billion bases per run, high per base costs and error rates up to 10%.

Since third generation sequencers do not parallelize sequencing in a way comparable to second generation sequencing, they are able to make the reads available in real-time during the sequencing procedure. Therefore, these techniques are also called real-time long-read sequencing approaches [65]. This feature allows starting analyses even before the sequencing run has finished. Especially Oxford Nanopore has been widely used for real-time analyses of reads [17, 18].

With HiLive [72] and PriLive [19], we were able to show that real-time analyses are also possible using second generation sequencing devices. This combines the advantages of second generation sequencers, namely low per-base costs, low error rates and high sequencing depths, with a smaller overall turnaround time of a real-time experiment.

Both types of NGS have advantages and are suitable for certain tasks. Since all questions in this work could be answered best using second generation sequencing data from Illumina sequencers, the term NGS will by now be used interchangeably with Illumina's technology.

1.5 NGS data analysis

As shown above, NGS provides scientists with powerful tools to answer a variety of questions in different fields of research. In general, any sequence of nucleic acids can be sequenced. As each of the reads contains only very little information on its own, it is necessary to process NGS data with a variety of bioinformatics methods. I will describe some of the more basic methods relevant for my work in the following paragraphs 1.5.1 to 1.5.3. All the described algorithms have large impact on NGS-based pathogen related research.

1.5.1 Read mapping

If a reference sequence is known, it can be used as a basis for a broad variety of analyses. Read mapping, as one of the most prominent ones, allows determining the exact genomic position a read stems from. This information obviously also implies what references the reads in a sample stem from.

While the read mapping problem as such has just grown relevant with the rise of NGS, it can be thought of as an abstraction of the sequence alignment problem. Due to sequencing errors as well as genuine differences between a read and a reference sequence, the alignment of two sequences is anything but a trivial problem. Alignment methods useful for sequencing data analysis need to account for mismatches, insertions and deletions. Needleman and Wunsch proposed the first precursor of most modern alignment algorithms in 1970 [73]. It is used to calculate the global alignment between two sequences, meaning that the optimal alignment of the full length of both sequences is computed. In 1981, Smith and Waterman adapted the Needleman-Wunsch algorithm for local alignments, such that similarities between smaller parts of larger sequences could be detected as well [74].

The first big step from pure alignment algorithms towards read mapping was taken by Wilbur and Lipman [75] and later implemented in the FASTA program suite [76] and subsequently in BLAST [77]. Both of these rely on the principle of pre-filtering candidate sequences which have high chances of allowing a valid alignment. As the filtering step is computationally much more efficient than the actual alignment, these methods enabled new fields of applications in times of rapidly growing reference databases. On the other hand, most of the tools are based on heuristics and do no longer guarantee to find all valid alignments. With the growth of datasets as well as reference databases, using BLAST as a read mapper is not feasible anymore as it is too time consuming.

Today, there is a huge variety of read mappers which rely on different algorithmic principles [78]. What all of them have in common is that they include only small parts of a given reference database for the actual read alignment. Large parts of the reference database are excluded via highly efficient sequence searches, leaving only little candidate positions for the search of a read's origin via real alignments. I will shortly provide an introduction to read mapping at the example of HiLive [72], as it is the basis for some of my follow-up work. This illustration is based on Martin Lindner's description in the supplementary material of [72].

HiLive builds on a *k-mer*-based approach, where a *k-mer* is defined a subsequence of a read or a reference of length *k*. The reference database on which HiLive is built is a k-*mer* index: For every *k-mer* that occurs in the reference sequences, all positions of its occurrence are stored and can easily been looked up. For each read produced by the sequencer, candidate alignments are calculated in parallel. To account for better understandability, I will describe how these calculations are conducted for one read only. Whenever the sequencer provides a new base for the read, the alignments are calculated in the following steps: database lookup, seed extension, seed creation and seed filtering.

Database lookup

As soon as a read reaches a predefined length *k*, the last *k* bases of the read are considered the current *k-mer*. This *k-mer* is then searched in the *k-mer* index, returning a list of (genome ID, position)-pairs, denoted as matches. Negative positions denote reverse hits. All matches are then sorted by positions.

Seed extension

Existing seeds are being extended using the new matches. Whenever a new match lies in a window of *w* bases from an existing seed and shares the same genome ID, the new match is used to extend the existing seed. If more than one candidate database match exists, the one closest to the expected position is used for extension. Each database match may extend more than one seed.

Seed creation

All matches which cannot be used for seed extension are saved as new seeds, if there is a chance that they start an alignment that fulfills the user-specified criteria. In other words, after the completion of a certain cycle, no new seeds are created anymore.

Seed filtering

As all alignments must be kept in memory for the whole time, it is necessary to keep the number of existing seeds as low as possible. HiLive provides users with different instruments to discard seeds: exact filters, which exclude only alignments that cannot reach a given threshold and optional heuristics, which bear the risk of rejecting valid alignments. [72]

1.5.2 Read binning

One of the most fundamental steps in many metagenome workflows is read binning. This denotes grouping the reads into bins, which correlate to their operational taxonomic units (OTUs), which are defined as clusters of organisms with high sequence similarity.

Generally speaking, there are two kinds of binning tools: tools based on supervised [43, 46, 47, 79-85] and tools based on unsupervised [86-88] algorithms. Supervised methods rely on a predefined set of groups, to which the reads are assigned. In most cases, the sets are defined based on the taxonomic origin of sequences. E.g., a set of all sequences below a certain taxonomic rank may be grouped together and constitute one bin.

Supervised binning tools may either be based on alignments or make use of sequence characteristics such as GC-content, codon usage bias, *k-mer* frequencies and the like. In simple terms, certain sequence characteristics (including the nucleotide sequence itself for alignment-based methods) are measured in a set of reference sequences and assigned to predefined bins. In the next step, the reads are assigned to the bins based on the same characteristics.

Unsupervised methods are not provided with references but form bins without *a priori* information. For example, abundanceBin bins reads based on their *k-mer* frequencies which it compares to the overall frequency of *k-mer* abundances in the complete dataset [87].

Unsupervised binning tools, but also some of the more sensitive alignment free supervised binning tools, may be of great help when working with sequences for which no close reference is known. Relying on vague sequence characteristics, it is sometimes possible to correctly assign reads which do not even have an alignable counterpart in a reference. This may either happen due to a massive amount of single nucleotide polymorphisms or in case of structural variations such as insertions or deletions.

There are different underlying techniques used for read binning. Read mappers and other alignment-based methods can be very helpful for this purpose, as presented in MetaPhlAn2 [89], DiScRIBinATE [90], SPHINX [91], taxator-tk [92] and MEGAN [46]. Still, in some cases the use of a read mapper is not necessary or suitable in order to retrieve the desired information from an NGS dataset. The calculation of an actual alignment may take unnecessarily long. Especially calculating highly sensitive read alignments as performed by BLAST [77] is still computationally challenging. To circumvent this restriction, various methods based on different classification principles without the need for error-tolerant alignments have been proposed, which yield comparable or even better results in reasonable time frames.

Most of these methods rely on *k-mer*-based approaches [93-98]. I will shortly explain the principle of one *k-mer*-based classifier at the example of Kraken, as it was the first rapid metagenome classification tool and therefore serves as the reference tool for *k-mer* based classification approaches [99].

As a basis for the read classification, Kraken builds a *k-mer* database using the Jellyfish *k-mer* counter [100] with *k* being set to 31 bp as a default. For each occurring *k-mer*, the lowest common ancestor (LCA) of all sequences containing it is calculated. Here, this is defined as the lowest taxonomic clade under which all sequences containing the given *k-mer* are joined.

All *k-mers* occurring in a given read are searched within the *k-mer* database. All paths from the root to an LCA taxon of a read are combined to a pruned subtree. The count of each appearing in a read is stored. Next, the sums of these counts of all possible root-to-leaf (RTL)-paths are calculated. The leaf of the highest scoring RTL-path is selected as the resulting classification. If more than one RTL-path reaches the maximum score, the read is assigned to the LCA of their leafs. If a read contains no *k-mer* from the database it is left unclassified [94].

As Marchesi and Ravel [101] discussed, real metagenome classification is not to be confused with metataxonomics, a field of research relying on the amplification and sequencing of certain marker regions. The most wide spread example of metataxonomic methods is 16s RNA sequencing, which can be used for the study of microbial community composition analysis only [102, 103]. As there is an explicitly biased amplification step and large parts of most environments are excluded from sequencing, this is not a real metagenomic application. For example, there exists no universal marker region for viruses. Due to these limitations of metataxonomics, I will focus on true whole genome metagenome applications in this thesis.

Furthermore, we must distinguish between metagenome classification and abundance estimation methods. While metagenome classification means to assign each read to its taxon of origin, the latter aim at estimating the abundance of OTUs. For example, MetaPhlAn2 only uses a set of marker genes instead of a whole genome database, potentially leaving a large number of reads unassigned [89]. On the other hand, pure sequence classification tools do often not allow precise abundance estimation, as for groups with many rather similar subclades, reads may be stuck in higher taxonomic ranks [94]. Lu et al. have for this reason developed Bracken [104], a tool which enables precise abundance estimation based on the results of Kraken.

1.5.3 *De novo* assembly

If a nucleic acid sequence stems from a yet unknown source or is expected to be distinct from known references, it may be necessary to reconstruct the sequence from scratch. Although this is often more complicated than the described reference-based methods, it can avoid introducing biases into the data analysis and may thus be performed even if a reference is available.

As short reads carry very little information themselves, it is necessary to assemble the reads to a longer sequence in order to gain deeper insights. If no *a priori* information from a reference sequence is used, we speak of a *de novo* assembly. In principle, all *de novo* assemblers combine overlapping reads into contiguous sequences (contigs) using different strategies.

A first simplified theoretical concept of *de novo* assembly algorithms was published by Lander and Waterman in 1988 [105]. Algorithms based directly on this concept use so called greedy algorithms. An example of these is implemented in SSAKE [106]: All reads in a dataset are stored in a list of unclassified reads. Then, a first read is moved to a list of contigs and all unassembled reads are searched for overlaps of a given minimum length, which is reduced stepwise when no more overlaps of any read and the contig are found. These algorithms are comparably inefficient and prone to sequencing errors. Moreover, repetitive regions cannot be resolved using this concept.

The next bigger innovation was introduced through Overlap-Layout-Consensus (OLC)-based algorithms. These are also based on the idea of identifying overlaps between reads, but overlapping reads are not unconditionally accepted as contigs. Instead, an overlap graph is built, where every read is considered a node. Whenever two reads can be aligned and therefore overlap, their nodes are linked by an edge in the graph. In this overlap graph, a Hamilton path – a path visiting all nodes of a connected graph exactly once – is searched. This reduces the graph to a smaller number of nodes, where every node represents one contig. Finally, the consensus sequence of all reads constituting a contig is calculated.

Software based on this concept was for the first time able to assemble large genomes [107]. Still, it has the disadvantage to be computationally expensive, both concerning memory and computational power, as every read is a node in a graph and calculating Hamilton graphs is an NP-hard problem.

Most modern assembly algorithms such as SPAdes [108], EULER [109], Velvet [110]. ALLPATHS [111, 112] or SOAPdenovo2 [113] are therefore based on de Bruijn graphs. In contrast to the methods described previously, de Bruijn graph-based assemblers work without calculating an error-tolerant alignment between reads. Instead, reads are dissected into overlapping *k-mers*. These *k-mers* are used to build a de Bruijn graph. In contrast to the graphs from OLC-based algorithms, sequences do not represent nodes but edges in the graph. This switch allows searching for an Euler path instead of a Hamilton path. In an Euler path, not every node but every edge is visited exactly once. For the Eulerian path problem, algorithms which need only quadratic runtime are known. Furthermore, the size of the graph does not depend on the number of reads, as duplicate sequences are merged into one edge. As most *de novo* assemblies are based on a large number of short reads, de Bruijn-based algorithms can solve this problem far more efficiently than alignment-based methods. Since sequencing fewer but longer reads becomes more influential again, it is quite possible that OLC assemblers will as well experience a revival [114, 115].

All aforementioned assemblers are intended to be used on sequencing data from single organisms. Yet, they may of course be used on metagenomic datasets after the reads have been binned. These groups may then yield good assembly results when treated like sequencing data from isolated source organisms. Still, binning tools may falsely assign reads, or, more often and worse, leave reads unassigned which are then lacking in the assembly.

To circumvent preceding binning and its negative side effects such as the risk of misclassified reads, efforts have also been made in the field of direct metagenome assemblers [116-123]. The assemblers described previously often face difficulties when run on metagenomic data. These may have several causes including different abundance levels of different species, shared conserved regions between different organisms and mixtures of strains, combining both mixed abundances and highly similar genomes [116]. This may lead to falsely connected contigs or unnecessarily short contigs. Metagenome assemblers are designed to solve these challenges. They often rely on splitting de Bruijn graphs into smaller subgraphs based on features like coverage or graph connectivity [117, 118]. Therefore, they can assemble metagenomic data directly.

Most NGS-based metagenomic research questions can be answered by one or a combination of the aforementioned methods. The choice of methods does of course depend on the exact question, the type of the sample, the availability of reference sequences and more.

1.6 Terminology and abbreviations

auc	Area under the curve
bam	Binary sequence alignment/map
BerSQPV	Berlin squirrelpox virus
bp	Basepairs
BSL	Biosafety level
contig	Contiguous sequence
DNA	Deoxyribonucleic acid
ds-cDNA	Double-stranded cDNA
EM	Electron microscopy
ERV	Endogenous retrovirus
FTP	File Transfer Protocol
gbp	Gigabasepairs
GC	guanine-cytosine
HERV	Human endogenous retrovirus
kbp	Kilobasepairs
LCA	Lowest common ancestor
NCBI	National Center for Biotechnology Information
NGS	Next generation sequencing
OLC	Overlap-Layout-Consensus
OPV	Orthopoxvirus
OTU	Operational taxonomic unit
PCR	Polymerase chain reaction
PPV	Parapoxvirus
RAMBO-K	Read Assignment Method Based On K-mers
RNA	Ribonucleic acid
ROC	Receiver operating characteristic
RTL	Root to leaf
sam	Sequence alignment/map
SQPV	Squirrelpox virus
TaxID	Taxonomic identifier

1.7 Thesis outline

In this thesis, I will approach the problem of rapid and sensitive diagnostics for known and unknown pathogens based on NGS from different angles. In the following chapters 2, 3, 4 and 5, I show our contributions to the field of research. Each of them tackles different research gaps and thereby enables new applications or provides new insights. The chapters are arranged in the order in which the described applications could be used on a metagenomic NGS dataset if very little information is available beforehand. I will finish with a chapter which shows the application of some of my introduced work as well as several other bioinformatic analyses in a real setting.

In Chapter 2, I describe LiveKraken, a tool that enables the use of Kraken's core algorithm [94] on Illumina sequencing data in real-time. LiveKraken enables accurate classification of reads even up to days before the sequencer has finished. This very general approach facilitates getting a first insight into a dataset without investing extra time, making it suitable as a standard application for any sequencing project. I conceptualized the real-time reporting feature, the visualization and the user interface of LiveKraken, designed and performed the benchmarks and wrote the manuscript for this project. The tool was developed by Benjamin Strauch, who implemented the new data handling into Kraken's source code. Andreas Andrusch helped with the implementation of the visualization. Andreas Nitsche assisted writing the manuscript. Tobias Loka gave substantial input on the concept and helped writing the manuscript. Benjamin Strauch, Martin Lindner, and Bernhard Renard had the initial idea LiveKraken is based on. The project was performed under the guidance of Martin Lindner and Bernhard Renard. LiveKraken has been published at the journal Oxford Bioinformatics:

> **Simon H. Tausch**[1], Benjamin Strauch[1], Andreas Andrusch, Tobias P. Loka, Martin S. Lindner, Andreas Nitsche, Bernhard Y. Renard. *LiveKraken – Real-time metagenomic classification of Illumina data*. *Bioinformatics*, 2018. [124]

Chapter 3 addresses a more profound and exhaustive approach of a real-time application on Illumina data, which is tailored to accurate pathogen diagnostics. The described software tackles not only the problem of long turnaround times of NGS-based

[1] These authors contributed equally to the article

diagnostics using the enhanced core algorithm of HiLive [72], but also implements a new approach of contamination and background masking and visualizes the results in an intuitive way. This allows getting actionable results in minimal time. I conceptualized the workflow, implemented the visualization and the result evaluation including the pathogenicity rating, designed the scoring method and invented the novel background masking method. Tobias Loka, Jakob Schulze and Kristina Kirsten helped enhancing HiLive to enable real-time output, gapped *k-mer* functionality and implemented many more necessary features. Jeanette Klenner produced the dataset used for benchmarking. Andreas Nitsche gave virological insights and supervised the generation of the benchmarking datasets. Wojtek Dabrowski, Martin Lindner and Andreas Andrusch gave substantial input on general questions regarding algorithmics, parametrization, visualization, and scoring methods. Bernhard Renard led the project.

> **Simon H. Tausch**, Tobias P. Loka, Jakob M. Schulze, Andreas Andrusch, Kristina Kirsten, Jeanette Klenner, Piotr Wojciech Dabrowski, Martin S. Lindner, Andreas Nitsche, Bernhard Y. Renard. *PathoLive – Real-time pathogen identification from metagenomic Illumina datasets.* (Manuscript under final internal revision before submission)

Chapter 4 describes an efficient alignment-free read classifier. Especially when sequencing viruses, analyses are hampered by large proportions of background reads from the host cells. When no reference with appropriate similarity for a meaningful alignment exists, as it is common for viruses, a sensitive selection of relevant reads is difficult. RAMBO-K assigns reads to fore- and background using a Markov Chain-based classifier, yielding more precise classifications than its competitors in minimal time. I designed and implemented the Markov Chain-based scoring model in, the visualization, the read simulation and the user interface in Python and also designed and performed the benchmarks. Furthermore, I helped with writing the manuscript. Wojtek Dabrowski helped with the conceptualization and reimplemented the read scoring method and Markov Chain trainer in Java in a more efficient manner. Moreover, he guided the project and wrote the manuscript. His contributions have also been described in his doctoral thesis [115]. Bernhard Renard and Andreas Nitsche helped writing the manuscript and co-designed the idea of the tool. I have discussed the basic concept and implementation of RAMBO-K in my Master's thesis [125]. Beyond that, I guided the development of a Geneious [126] plugin for the tool, which is work in progress, with help of Wojtek Dabrowski, René Kmiecinski and Alona Tyshaieva. I also conceptualized

the idea for a high-level-binning based on RAMBO-K to classify reads as viral, bacterial or eukaryotic. This has been implemented by Sophie-Meier zu Ummeln but yielded no significant results.
RAMBO-K has been published in PlosOne:

> **Simon H. Tausch**, Bernhard Y. Renard, Andreas Nitsche, Piotr Wojciech Dabrowski. *RAMBO-K: Rapid and Sensitive Removal of Background Sequences from Next Generation Sequencing Data. PLoS One*, 2015. 10(9): p. e0137896. [85]

An example use case of RAMBO-K and a number of general bioinformatics analyses are shown in Chapter 5. There, the discovery of a novel squirrelpox virus which is believed to establish a whole new genus of *poxvirinae* is presented. Studying the dataset using the established default mapping tools, no viable results could be found. Only after I assigned the viral reads using RAMBO-K, I was able to assemble the genome. I also assembled and aligned resequenced samples from other specimens, helped with genomic analyses and wrote parts of the manuscript. Gudrun Wibbelt wrote major parts of the manuscript and conducted the electron microscopy as well as the histology. Olivia Kershaw provided the samples to RKI. Wojtek Dabrowski performed the phylogenetic analysis. Andreas Nitsche designed the PCRs and gave valuable input to all parts of the complete project. Livia Schrick guided the project. All authors contributed in writing the manuscript. This chapter has been published in the Journal Emerging Infectious Diseases:

> Gudrun Wibbelt[1], **Simon H. Tausch[1]**, Piotr Wojciech Dabrowski, Olivia Kershaw, Andreas Nitsche, Livia Schrick. *Berlin Squirrelpox Virus, a New Poxvirus in Red Squirrels, Berlin, Germany. Emerg Infect Dis*, 2017. **23**(10): p. 1726-1729. [127]

Further contributions

Besides these first author contributions which are presented in detail in this thesis, I also contributed methodology I developed and data analysis based on it in the following journal publications:

[1] These authors contributed equally to the article

Andreas Andrusch, Piotr W. Dabrowski, Jeanette Klenner, **Simon H. Tausch**, Claudia Kohl, Abdalla A. Osman, Bernhard Y. Renard, Andreas Nitsche. *PAIPline: Pathogen identification in metagenomic and clinical next generation sequencing samples. Bioinformatics,* 2018 (in press)

With PAIPline, we present a pathogen identification pipeline, enabling the alignment based taxonomic classification of metagenomic reads with a focus on clinical samples. I gave input on the conceptualization of the workflow and parametrization of the modules. Furthermore, I conducted extensive testing and proof-read the manuscript.

Tobias P. Loka, **Simon H. Tausch**, Piotr Wojciech Dabrowski, Aleksander Radonic, Andreas Nitsche, Bernhard Y. Renard. *PriLive: Privacy-preserving real-time filtering for Next Generation Sequencing. Bioinformatics*, 2018. [19]

In this project, we present a real-time filtering tool for Illumina sequencing data which removes host reads while they are being produced and at the same time minimize the risk of deleting relevant data. I gave input on the algorithmics of HiLive and especially on the adaption of the algorithm to real wet lab settings. I furthermore tested Tobias Loka's developments and proof-read the manuscript.

Livia Schrick, **Simon H. Tausch**, Piotr Wojciech Dabrowski, Clarissa R. Damaso, José Esparza, Andreas Nitsche*. An Early American Smallpox Vaccine Based on Horsepox. New England Journal of Medicine*, 2017. 377(15): p. 1491-1492. [128]

This letter in the New England Journal of Medicine is based on the sequencing of an ancient smallpox vaccine capillary from 1902. We were able to show that this vaccine was probably derived from horsepox instead of cowpox, gaining new insights to the mystery surrounding the history of smallpox vaccines. I analyzed the data from the highly fragmented genomic material using Trimmomatic [129], RAMBO-K [85] and SPAdes [108], which resulted in the complete genome of the ancient vaccine after manual correction. I furthermore calculated alignments using MAFFT [130] to related vaccinia, cowpox and horsepox genomes, finding major vaccinia-typical deletions in the ends of the ancient vaccine genome. Based on these, I designed figure 1B of the paper. I also wrote parts of the manuscript.

Martin S. Lindner, Benjamin Strauch, Jakob M. Schulze, **Simon H. Tausch**, Piotr Wojciech Dabrowski, Andreas Nitsche, Bernhard Y. Renard. *HiLive: real-time mapping of illumina reads while sequencing*. *Bioinformatics*, 2017. 33(6): p. 917-319. [72]

HiLive is the first real-time read mapper for Illumina data giving results by the end of a sequencing run. After extensive testing, I conceptualized several enhancements of HiLive, making it easier applicable to real world settings. My most relevant contributions were made on user interface design and efficient and understandable output including demultiplexing and the handling of paired-end reads.

Tobias P. Loka, **Simon H. Tausch**, Bernhard Y. Renard. *Reliable variant calling during runtime of Illumina sequencing.* (under review)

This paper introduces HiLive2, a follow-up version of HiLive with a novel underlying alignment method, and combines it with xAtlas to enable SNP-calling while the sequencer is still running. I continuously supported the development of HiLive2 in technical as well as algorithmic questions and evaluated the performance of HiLive2 on different types of data.

Cesare E. M. Gruber, Emanuela Giombini, Marina Selleri, **Simon H. Tausch**, Andreas Andrusch, Alona Tyshaieva, Giusy Cardeti, Raniero Lorenzetti, Giuseppe Manna, Fabrizio Carletti, Andreas Nitsche, Maria R. Capobianchi, Gian Luca Autorino, Concetta Castilletti. *Whole genome characterization of OPV Abatino, a zoonotic virus representing a putative novel clade of Old World Orthopoxviruses.* (Manuscript under final internal revision before submission)

In this project, we assembled and characterized the genome of a novel poxvirus from Macaques, revealing hints towards genomic recombination. I helped assembling and annotating the genome with the aid of RAMBO-K and a self-developed assembly pipeline. Furthermore, I designed and implemented data analyses measuring the similarity of open reading frames to genes of different related species and drew figures visualizing these results.

Simon H. Tausch, Andreas Andrusch, José Esparza, Andreas Nitsche[1], Clarissa R. Damaso[1]. *Genome analysis of the Mulford 1902 smallpox vaccine*. (Manuscript under final internal revision before submission)

In this project, we further characterize the genome of the ancient smallpox vaccine from [128]. This includes genome annotation, phylogenetic and metagenomic analyses as well as detailed investigations on gene level. I designed and implemented data analyses measuring the similarity of open reading frames to genes of different related species and drew figures visualizing these results. Moreover, I co-performed the metagenomic and gene level analyses and helped writing the manuscript.

[1] These authors contributed equally to the article

2 Real-time metagenomic classification using LiveKraken

In metagenomics, Kraken is one of the most widely used tools due to its robustness and speed. Yet, the overall turnaround time of metagenomic analysis is hampered by the sequential paradigm of wet and dry lab. In urgent experiments, it can be crucial to gain a timely insight into a dataset.

Here, we present LiveKraken, a real-time read classification tool based on the core algorithm of Kraken. LiveKraken uses streams of raw data from Illumina sequencers to classify reads taxonomically. This way, we are able to produce results identical to those of Kraken the moment the sequencer finishes. We are furthermore able to provide comparable results in early stages of a sequencing run, allowing saving up to a week of sequencing time on an Illumina HiSeq in High Output Mode. While the number of classified reads grows over time, false classifications appear in negligible numbers and proportions of identified taxa are only affected to a minor extent.

LiveKraken is available at https://gitlab.com/rki_bioinformatics/LiveKraken.

2.1 Introduction

Real-time analyses of genome sequencing data have been gaining particular attention over the last years, as they enable to analyze data while the sequencer is still running. Yet, the possibilities of live analysis approaches based on MinION sequencers are still limited due to low throughput rates and sequence qualities of these devices. With HiLive [72] we proposed the first method for real-time analyses of high-throughput sequencing data from Illumina machines, enabling a new field of applications. For metagenomic studies, classification tools such as Kraken [94] have also been used in time-relevant applications. These are, however, affected by the sequential paradigm of wet and dry lab, setting the lower limit of the overall duration of an experiment to the runtime of the sequencing machine. To tackle these limitations, we present LiveKraken, a real-time taxonomic classification tool based on the core algorithm of Kraken. We show that it yields results comparable to those of established tools long before the sequencer has even finished and that it guarantees results identical to those of Kraken as soon as a sequencing run has ended. LiveKraken has been tested on HiSeq and MiSeq systems and is as robust and easy to use as Kraken. The field of applications may range from controlling sample composition, contamination identification, or outbreak detection in real-time.

2.2 Materials and methods

Originally, Kraken has a linear workflow [94]. Sequencing reads are read from FASTA or FASTQ files and subsequently classified using a precomputed database. Since the reads are independent of each other, they can be processed in parallel. The LCA classification results found for each read are written to Kraken's tabular report file.

To make this workflow fit for the purpose of live taxonomic classification, similar to the approach taken in HiLive [72], a new sequence reader module was implemented which allows reading sequencing data from Illumina's binary BCL format. LiveKraken can be used to analyze continuously and refine the metagenomic sample composition, using the same database structure as the original Kraken.

Illumina sequencers process all reads in parallel in so called cycles, appending one base to all reads per cycle. For each cycle, basecall (BCL) files are produced in Illumina's BaseCalls directory, which is declared as input for LiveKraken instead of FASTA or FASTQ files. New data is collected by the BCL sequencing reader module in user-specified intervals of *j* sequencing cycles, starting with the first *k-mer* of size *k*. The collected data is sent to the classifier which refines the stored partial classification with the new sequence information. Temporary data structures of Kraken are stored for each read, such as the LCA list, a list of ambiguous nucleotides, and the number of *k-mer* occurrences in the database. This leads to an overall increase of memory consumption proportional to the number of LCAs found for each read sequence. Additionally, and crucial for the iterative refinement, a variable is stored that is holding the position up to which each read was classified. After each refinement step, output in the same tabular format as known from Kraken is produced. This enables early classification while also ensuring that the classification output after reading the data from the last sequencing cycle is exactly the same that Kraken would produce (cf. Figure 2a).

LiveKraken can be installed via the included script install_kraken.sh analogous to Kraken with an additional dependency to the boost library. It has been tested with gcc v. 4.9.2 and v. 7.2.0 and boost v. 1.5.8. Furthermore, a Conda package is available [131]. LiveKraken uses the same command line interface as Kraken.

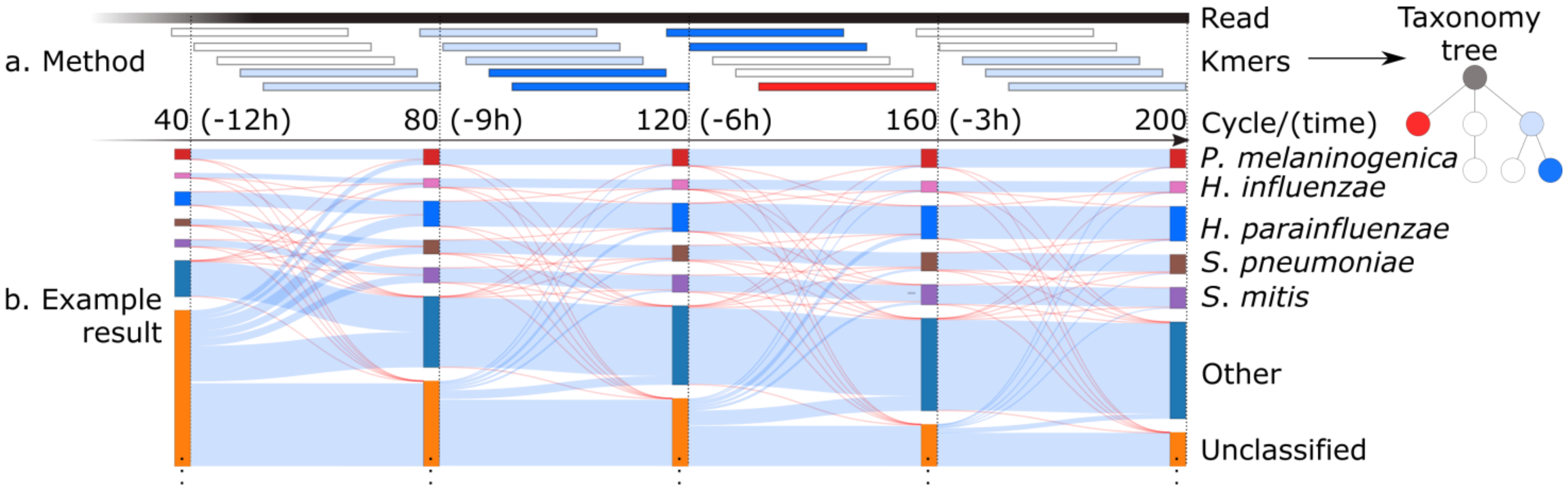


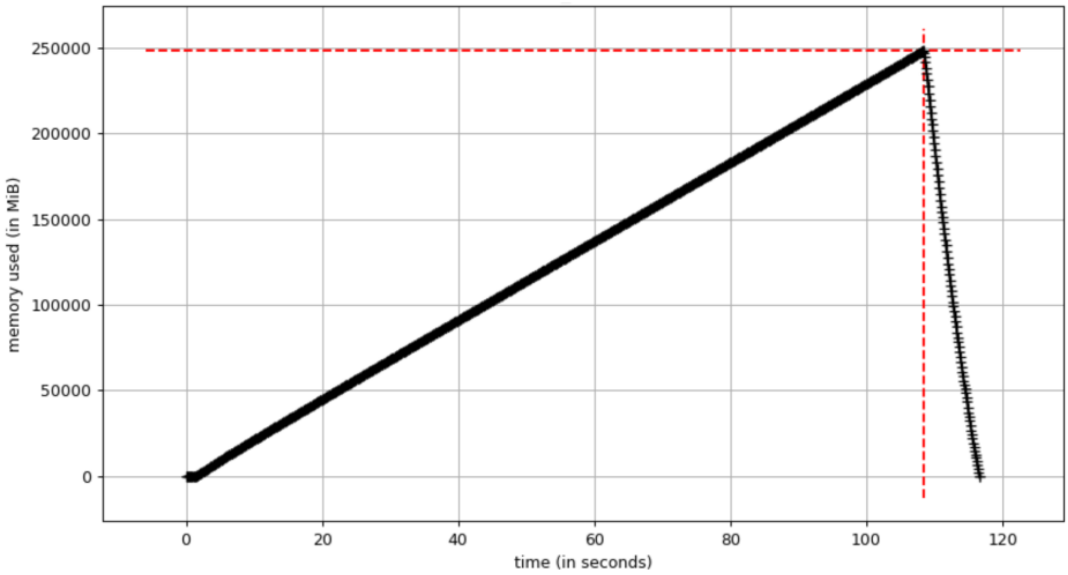
Figure 2 Timeline of LiveKraken: Upper part (a) showing the method, lower part (b) an exemplary result. (a) Method: Raw parts of sequenced reads are streamed directly from the sequencer into Kraken's classification algorithm. *K-mer*s are taxonomically classified using Kraken's precomputed map of each k-mer to the lowest common ancestor of all genomes containing the *k-mer*, as color coded in the taxonomy tree. The highest scoring path from the pruned subtree of the taxonomic tree is selected as classification of each read [88]. (b) Results: In this example (SRR062462), results are reported after 40, 80, 120, 160, and 200 sequencing cycles or approximately 12, 9, 6, 3, and 0 hours on an Illumina MiSeq before the sequencer finishes and data can be prepared for other tools to start. The results are visualized in a Sankey diagram of read classifications on species level after all cycles are reported. The top five groups with the most hits are shown, while groups with fewer hits are conflated as "other". Reads which cannot be assigned on species level are denoted as unclassified. The unclassified nodes are optically narrowed by approximately 1,500,000 reads each for better recognition of relevant groups. Thickness of the flows encodes the number of reads going from one node to another, where blue flows represent unchanged or new classifications and red ones show changed classifications. While the number of unclassified reads decreases, the overall proportions of taxa stay the same. Misclassifications occur in negligible magnitude. The visualization of results as an interactive Sankey-plot is part of LiveKraken.

2.3 Results and discussion

LiveKraken builds on the well-known tool Kraken. Hence, we show its results in comparison to the classic Kraken approach. While we guarantee identical results as Kraken with the end of a sequencing run, we also show that preliminary classifications allow a reliable estimate of the sample composition long before the sequencer has finished. We ran LiveKraken on three datasets from the NIH Human Microbiome Project [132] (cf. Table 1), returning results after every 40th sequencing cycle or approximately 12, 9, 6, 3, and 0 hours before the sequencer finished, respectively. As reference database we used all bacteria and archaea sequences from RefSeq [133] downloaded on June 2nd 2015. We compared the results to the output of Kraken on the full datasets (Table 1). An example is visualized in Figure 2b, showing that the number of unclassified reads decreases over time, but only a minor number of reads is misclassified in earlier stages. While the peak memory requirements of LiveKraken increase by <1% compared to Kraken in our experiments, speed decreases by 15% (cf. Figure 3). It is still orders of magnitude faster than the sequencer and therefore not the runtime bottleneck. Our results confirm the hypothesis that a classification is already possible long before classical metagenomic tools can even be started.

Table 1 Recall (tpr) and precision (ppv) of LiveKraken at different time points, based on read classification on species level at each cycle compared to Kraken classification after 200 cycles as ground truth.

Cycle	40		80		120		160	
Dataset	tpr	ppv	tpr	ppv	tpr	ppv	tpr	ppv
SRR062371	0.85	0.99	0.94	0.99	0.96	0.99	0.99	1
SRR062462	0.80	0.98	0.92	0.98	0.95	0.98	0.99	0.99
SRR062415	0.80	0.98	0.92	0.98	0.95	0.98	0.99	0.99



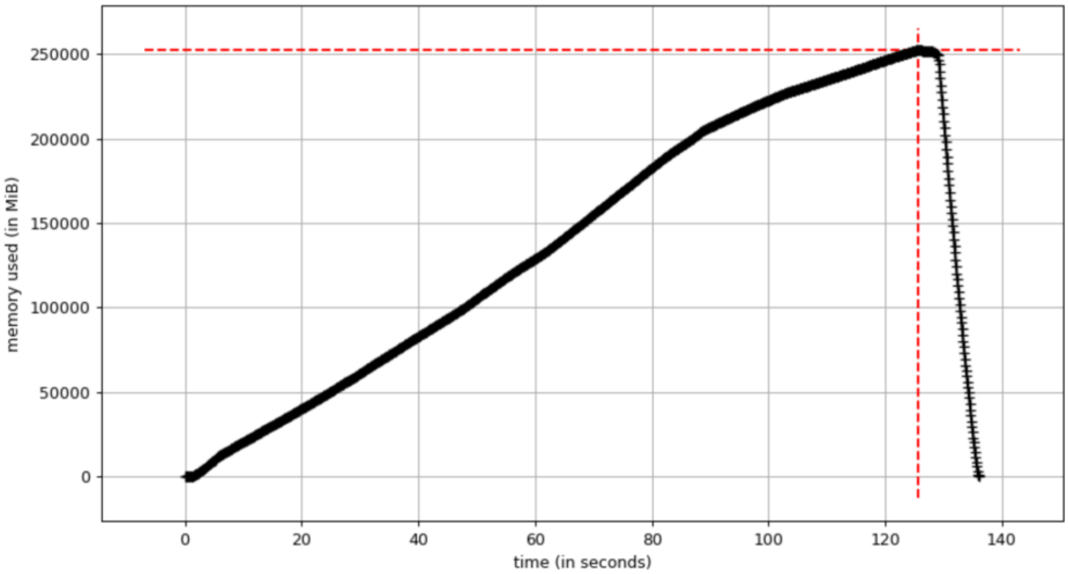


Figure 3. Comparison of runtime and memory consumption of Kraken (upper) and LiveKraken (lower) on dataset SRR062462. Both tools were run on 40 threads with default parameters. Kraken uses slightly more memory throughout the run. The computational runtime of LiveKraken increases by ~15% compared to Kraken.

3 Sensitive real-time pathogen diagnostics using PathoLive

Over the past years, NGS has been applied in time critical applications such as pathogen diagnostics with promising results. Yet, long turnaround times have to be accepted to generate sufficient data, as the analysis can only be performed sequentially after the sequencing has finished. Additionally, the interpretation of results can be further complicated by various types of contaminations, clinically irrelevant sequences, and the sheer amount and complexity of the data.
We designed and implemented PathoLive, a real-time diagnostics pipeline which allows the detection of pathogens from clinical samples up to several days before the sequencing procedure is even finished and currently available tools may start to run. We adapted the core algorithm of HiLive, a real-time read mapper, and enhanced its accuracy for our use case. Furthermore, common contaminations, low-entropy areas, and sequences of widespread, non-pathogenic organisms are automatically marked beforehand using NGS datasets from healthy humans as a baseline. The results are visualized in an interactive taxonomic tree that provides an intuitive overview and detailed measures regarding the relevance of each identified potential pathogen.
We applied the pipeline on a human plasma sample that was spiked *in vitro* with *vaccinia virus, yellow fever virus, mumps virus, Rift Valley fever virus, adenovirus,* and *mammalian orthoreovirus*. The sample was then sequenced on an Illumina HiSeq. All spiked agents were detected after the completion of only 12% of the sequencing procedure and were ranked more accurately throughout the run than by any of the tested tools on the complete data. We also found a large number of other sequences and these were correctly marked as clinically irrelevant in the resulting visualization. This tagging allows the user to obtain the correct assessment of the situation at first glance.
PathoLive is available at https://gitlab.com/rki_bioinformatics/PathoLive.

Sensitive real-time pathogen diagnostics using PathoLive

3.1 Introduction

The ability to sequence large amounts of nucleic acids in an unbiased manner through NGS is particularly interesting for metagenomics studies. Metagenomic NGS has been proposed as a valuable technique for clinical application. Nucleic acids of pathogens can be detected in metagenomic clinical samples even in cases where routine procedures fail to identify the underlying causes of a patient's symptoms [134-137]. Most other pathogen detection methods such as polymerase chain reaction (PCR), cell culture, or amplicon sequencing, aim to detect predefined organisms. On the contrary, NGS facilitates the detection and even characterization of pathogens without *a priori* knowledge about candidate species. NGS, unlike any other method, generates sufficient data to detect even lowly abundant pathogens without targeted amplification of defined sequences. Thus, it allows for an unbiased diagnostic analysis.

There is a variety of tool able to address NGS-based pathogen related questions with different focuses: either aiming to discover yet unknown genomes [40, 42-56, 85, 138] or to detect known species in a sample [21-28, 80, 89, 94, 96, 139-144]. Among both groups, there are different underlying algorithms, the main distinction running between alignment-based [21, 24-26, 28, 43, 45-47, 80, 89, 96, 140-144] and alignment-free methods [27, 40, 52, 55, 85, 94]. Many tools of course combine both approaches [22, 23, 42, 44, 48-51, 53, 54, 56, 138]. While being faster in most cases, alignment-free methods are limited to the detection of sequences, whereas alignment-based methods potentially allow for a more detailed characterization of genomes.

Existing approaches based on unbiased full genome sequencing of metagenomic samples are facing various obstacles, especially concerning the ranking of the results according to their clinical relevance and the long overall turnaround time [9, 16, 102, 145-149].

A central issue in NGS-based pathogen detection is that the clinically relevant data is very hard to identify. Not only is the host genome usually the dominating part in a metagenomic patient sample, but additionally there are nucleic acids of various clinically irrelevant species such as some *endogenous retroviruses* (ERV) or non-pathogenic bacteria which commonly colonize a person.

Even viruses may contain sequences of ERVs, as for example *gallid herpesvirus type 2* or *fowlpox virus*, potentially confusing the correct assignment of reads [150]. 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 compensate for the overwhelming amount of background sequences without introducing unwanted biases and thus risking a loss of signal, large numbers of reads are necessary. Still, there is no guarantee to get a sufficiently high coverage for the detection of a targeted pathogen genome.

To put it more generally, it is a widespread misconception to rely only on quantitative measures when ranking the importance of candidate hits. While the amount of nucleic acids of a pathogen in a sample may correlate with the phase or intensity of an infection, it may not be sufficient to select the most abundant species as the causative pathogen. On the contrary, not the amount but the uncommonness of a species in a given sample may give decisive 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. To date, there are several pipelines tackling this problem in different ways. Many pathogen detection pipelines propose to define a reference database of host and contaminating sequences [21, 22, 27, 28, 40, 51]. While facilitating cleaner results, it may lead to a premature rejection of relevant sequences. The definition of precise contamination databases proves rather difficult and has not yet been adequately solved. Thus, deletion of relevant hits and misinterpretation of irrelevant hits still remains a common problem.

Generally, handling high numbers of detected species with a low number of reads each makes it very difficult to get a clear definition of relevant and irrelevant hits. A presentation of all detected hits without any weighting would be hard to interpret, wasting precious time at the end of the workflow. Yet, deleting any results to gain a better overview comes at great risk of overlooking the true cause of an infection. Not only background and contamination removal introduces the risk of losing information that might be relevant in the following diagnostic process. Intensity filters, as implemented e.g. in SLIMM [141], disregard sequences with too small genome coverages. As the author states, this step eliminates many genomes. This problem even intensifies for marker-gene based methods such as MetaPhlAn2 [89], 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 actually relevant candidates.

Moreover, sequencing and analyzing the necessary amount of data is very time consuming. An Illumina HiSeq run in High Output Mode, potentially necessary to detect lowly abundant viruses, takes up to 11 days. Thus, in urgent cases or acute outbreak situations, standard workflows take too long to generate results in time to take the necessary measures. There is a plethora of infectious diseases which can be lethal, especially if not treated timely. For example, ebola patients who die from the disease die after 9.8 ± 0.7 days after the first symptoms occur on average [151]. To obtain actionable results within an appropriate time frame to help these patients and to prevent further spreading of the disease, it is crucial to reduce the time span of the entire workflow from sample receipt to complete diagnosis.

Efforts to speed up NGS based diagnostics have been made but 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 [17]. 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.

There are several promising approaches of pathogen detection using the MinION handheld device for in field studies. While allowing impressive 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 [17, 152-155]. Higher read numbers are indispensable for reliable pathogen detection. Therefore, the development of efficient methods to generate, analyze and understand large metagenomics datasets in an accurate and quick manner is crucial if NGS is to become 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 common contaminations, background data and irrelevant species all at once. Tackling the problem of slow overall turnaround times, we applied and enhanced our in-house developed real-time read mapper HiLive that enables analyzing sequencing data while an Illumina sequencer is still running [72].

3.2 Methods

3.2.1 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: **(i)** prepare informative, well defined reference databases, **(ii)** automatically define contaminating or non-pathogenic sequences beforehand, **(iii)** adapt HiLive, a real-time read mapper, to yield robust results even before the sequencer finishes, **(iv)** identify the hazardousness of candidate pathogens and present results in an intuitive, comprehensible manner. The details on the modules for each of these steps are provided in the following sections.

i. Prepare reference databases to be more efficient in runtime

In order to save computational effort during the post-processing of the live-mapped 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 [156], are downloaded from the File Transfer Protocol (FTP) servers of the National Center for Biotechnology Information (NCBI) and annotated accordingly with taxonomic information from the NCBI Taxonomy Database . The obtained data are then merged. 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 source organism of the sequence.
Afterwards, user definable subdatabases of taxonomic clades relevant for a distinct pathogen search are automatically created. For the experiments in this manuscript, we focused on viruses. The database updater used for this purpose is available at https://gitlab.com/rki_bioinformatics/database-updater.

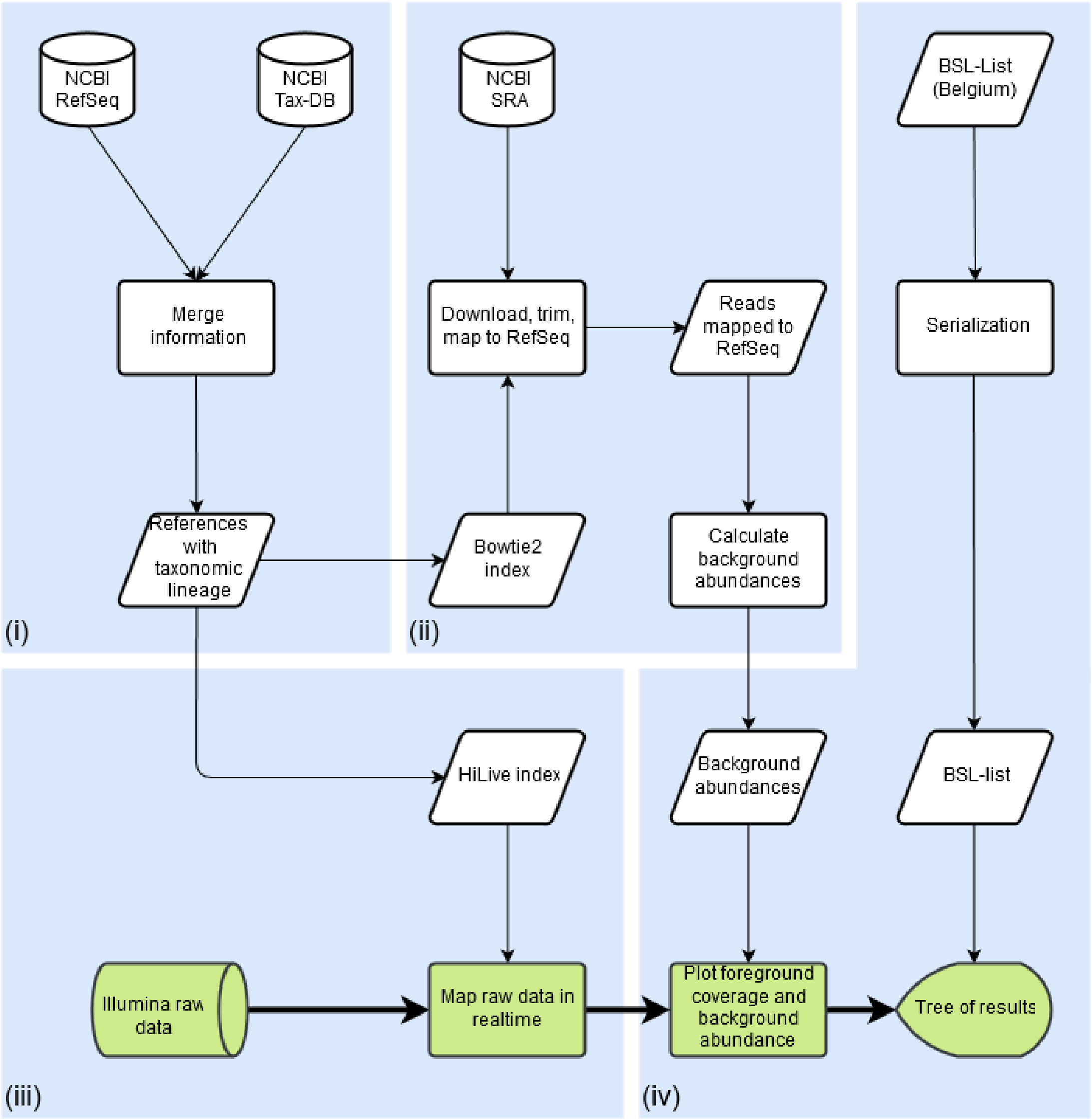


Figure 4: Workflow of PathoLive including four main modules: (i) Reference information from NCBI RefSeq is automatically downloaded and tagged with taxonomic information; (ii) NGS datasets from the 1000 Genomes Project are downloaded, trimmed and searched for sequences from the pathogen database from 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 the 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. Mark clinically irrelevant hits

A main obstacle in NGS based diagnostics is the large amount of background noise contained in the data. In this context, this refers to various sources of contamination including 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 [157] (s. section 7.1 in Supplementary material) were downloaded, assuming that a large majority of the participants in the 1000 Genomes Project was not acutely ill with an infectious disease. These reads are quality trimmed using Trimmomatic [129] and mapped to the selected pathogen reference database using Bowtie2 [158]. 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 clinically irrelevant and visualized as such in further steps of the workflow. The coverage maps of the background abundances are thereto plotted in red against the coverage maps of the reads from the patient dataset in green on the same reference (s. Figure 5). This enables highlighting presumably relevant results without discarding other candidate pathogens, giving the researcher 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.

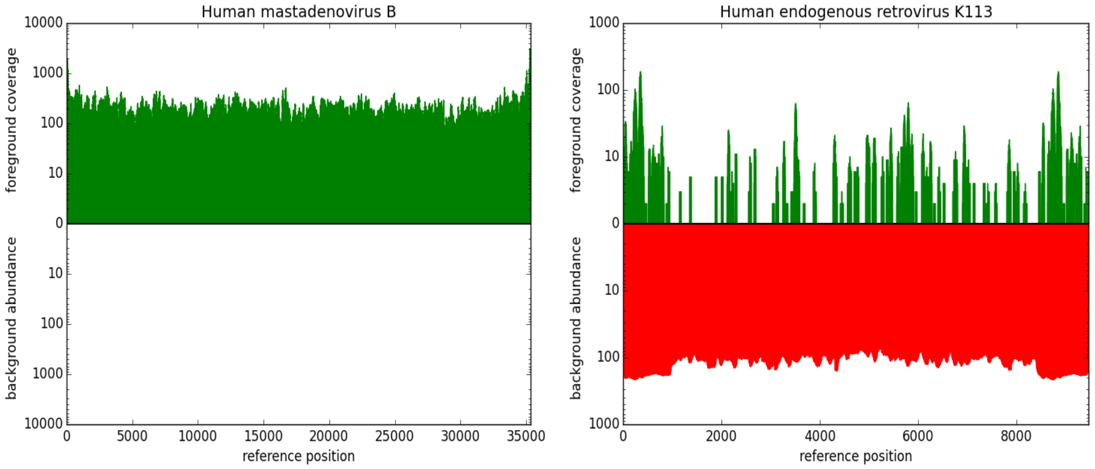


Figure 5: Two examples of fore- and background coverage plots. The upper, green bars show the coverage of a given genome in the foreground dataset, namely the reads sequenced from the patient sample. The lower, red part indicates in how many datasets from the 1000 Genomes Project a sequence is abundant. 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 ~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 abundance in non-ill humans.

iii. Adapt HiLive, Enhance to get results before sequencing finished

Due to the runtime requirements already mentioned, we aimed at breaking the sequential paradigm of wet and dry lab applications by parallelizing data generation and analysis. We used the real-time read mapper HiLive which yields results by the end of the sequencing run. To alleviate the high computational requirements to align all reads in parallel as they are sequenced, HiLive makes use of a highly efficient *k-mer* seed-and-extend approach. Therefore, errorless *k-mers* are looked up in a hash index. Each entry in the index contains matching positions for a *k-mer* in the database of reference genomes. Based on these *k-mer* positions, the q-gram lemma is applied to decide whether a certain *k-mer* position will be used to create, extend or discard an alignment candidate, referred to as seed [72]. Thereby, the user can decide how many errors to tolerate in an alignment. The algorithm results in a set of alignments for each read, including information about the matching genome and position but potentially missing detailed alignment information for regions with an accumulation of errors [72]. For the purpose of pathogen detection, we extended the current version (HiLive v0.3) by several features, resulting in a new version (HiLive v1.1). Instead of only obtaining results by the end of the sequencing run, HiLive now also contains the option to provide intermediate results at any point of a sequencing run with negligible delay. For the first time, this functionality allows not only to obtain mapping results at the same time the sequencing finishes but already during sequencing. The output of the mapping results was parallelized to handle even huge amounts of seeds that usually arise during intermediate steps. Additionally, we modified existing and created new output filters to reduce the number of random hits in the resulting alignment files. A separated executable can be used to create the output with different filter settings without re-executing the complete alignment algorithm. To further improve sensitivity, especially for the mapping results in early sequencing cycles, we adapted the core algorithm to support arbitrary gapped *k-mers*. This means that single or consecutive mismatches are tolerated within a single *k-mer*. As shown by Kucherov et al. [159], this concept results in significantly higher accuracy especially after few cycles of a sequencing run, even though the q-gram lemma does not hold for gapped *k-mers*. For our study, we used SpEEd to select an optimal *k-mer* gap pattern for seeds of weight 15 and an expected similarity of 0.95 on 40 basepairs, resulting in the pattern 11111100111101 [160].

PathoLive is implemented in a modular manner. Instead of the real-time read mapping using HiLive, any other read mapper providing sequence alignment/map (sam) or binary sequence alignment/map (bam) files can be used for the mapping step of the workflow.

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 [21, 161] or interactive taxonomy explorers [27, 46]. 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 visualization library D3 [162]. For an example of the visualization, see Figure 7. 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. We provide three scores for each node of the tree:

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

$$Total\ Hits = \#\ Reads\ mapped\ to\ clade$$

(b) Unambiguous Bases: the total number of bases covered in the patient dataset but not in any background dataset

$$Unambigious\ Bases = \#Bases\ covered\ by\ foreground\ but\ not\ by\ background\ data$$

(c) Weighted Score: the ratio of Unambiguous Bases to the number of bases covered by background reads

$$Weighted\ Score = \frac{Unambiguous\ Bases}{max(\#\ Bases\ covered\ by\ background\ data, 1)} \times \log(Total\ Hits)\ .$$

The weighted score introduces an intensified metric of how often a sequence is found in non-ill persons, therefore allowing drawing stricter conclusions from the background data. Not only exactly overlapping mappings of fore- and background are regarded, but the overall abundance of a sequence within the background data is considered.

The values of these scores are reflected in the thickness of the branches, which draws the visual focus to higher rated branches. By default, the visualization uses the weighted score, but users can switch between all three scores.

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 [163]. 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/BSL1), blue (BSL2), yellow (BSL3) or red (BSL4), 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 (Figure 7). When expanding a branch down 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 5). 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 very common genomes or parts of their sequences 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.

3.2.2 Validation

We compared the results of PathoLive to two existing solutions, Clinical Pathoscope [28] and Bracken [103]. 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 [21] or Taxonomer [27] were not locally installable and had to be disregarded. Bracken 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 cores, 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 BaseCalling 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 databases for PathoLive was built from the viral part of the NCBI RefSeq [133] 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 [133] downloaded on 2017-12-18. The Bracken database was generated with default parameters and an expected read length of 100 bp.

Sample preparation

Viral RNA metagenomics studies were performed with a human plasma mix of eight different RNA and DNA viruses as well-defined surrogate for clinical liquid specimen. The informed consent of the patient has been obtained. This 200µL 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 QIAamp Ultrasense Kit (Qiagen) following the manufacturers' instructions. The extract was treated with Turbo DNA (Life Technologies, Darmstadt, Germany). cDNA and double-stranded cDNA (ds-cDNA) synthesis were performed as previously described [164]. The ds-cDNA was purified with the RNeasy MinElute Cleanup Kit (Qiagen). The purification method takes ~6h to complete.

The Library preparation was performed with the Nextera XT DNA Sample Preparation Kit following the manufacturers' instructions (Illumina). NGS libraries were quantified using the KAPA Library Quantification Kits for Illumina sequencing (Kapa Biosystems). If the starting amount of 1 ng of nucleic acid was not reached the entire sample volume was added to the library.

3.3 Results

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. Intermediary results were taken 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 translated into the human readable 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 2. The area under the curve (auc) of the receiver operating characteristic (ROC) was calculated using the 16 highest ranking species, as given by the tested tools. The scores of all sequences attributed to a species were summed up. The top 16 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. For PathoLive, the weighted score is used, for Clinical Pathoscope we used the "final guess" metric and for Bracken, the species with most estimated reads were ranked highest. The corresponding ROC-plot is shown in Figure 6.

Table 2 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 the complete runtime of the sequencing from start of the sequencer to result presentation, whereas tool runtime is the time the tools take to generate results after all necessary input data has been provided. ROC-auc denotes the area under the ROC-curve as a combined measure of sensitivity and specificity. Best values are printed bold. PathoLive performs best according to all measures throughout the complete run.

	PathoLive				Pathoscope	Bracken
Input data [cycles]	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

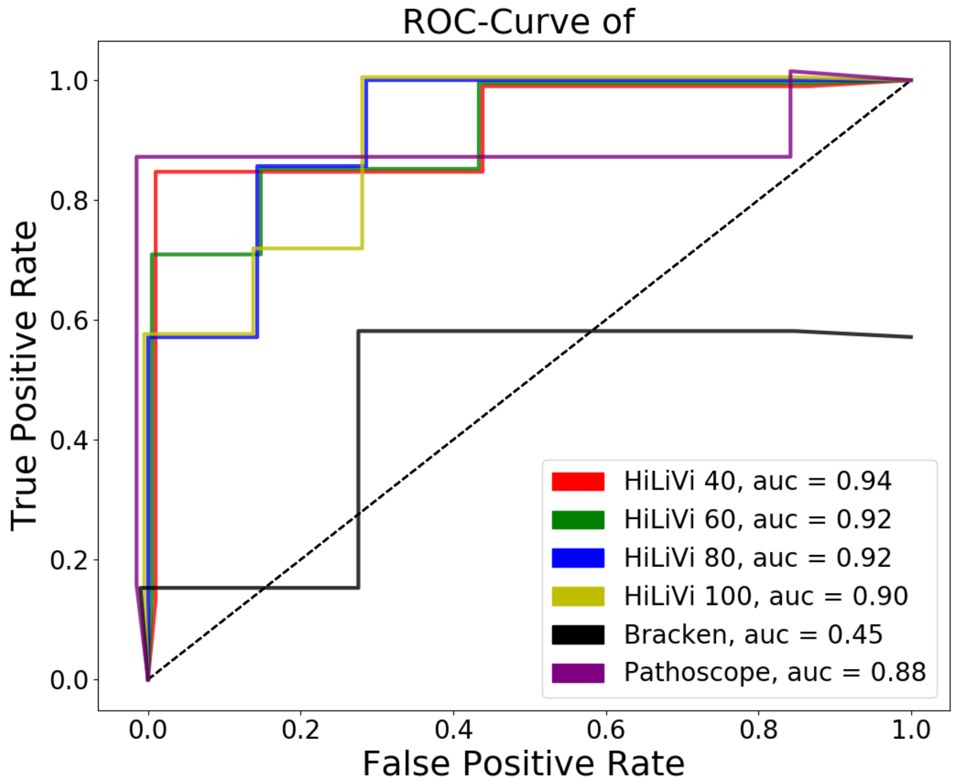
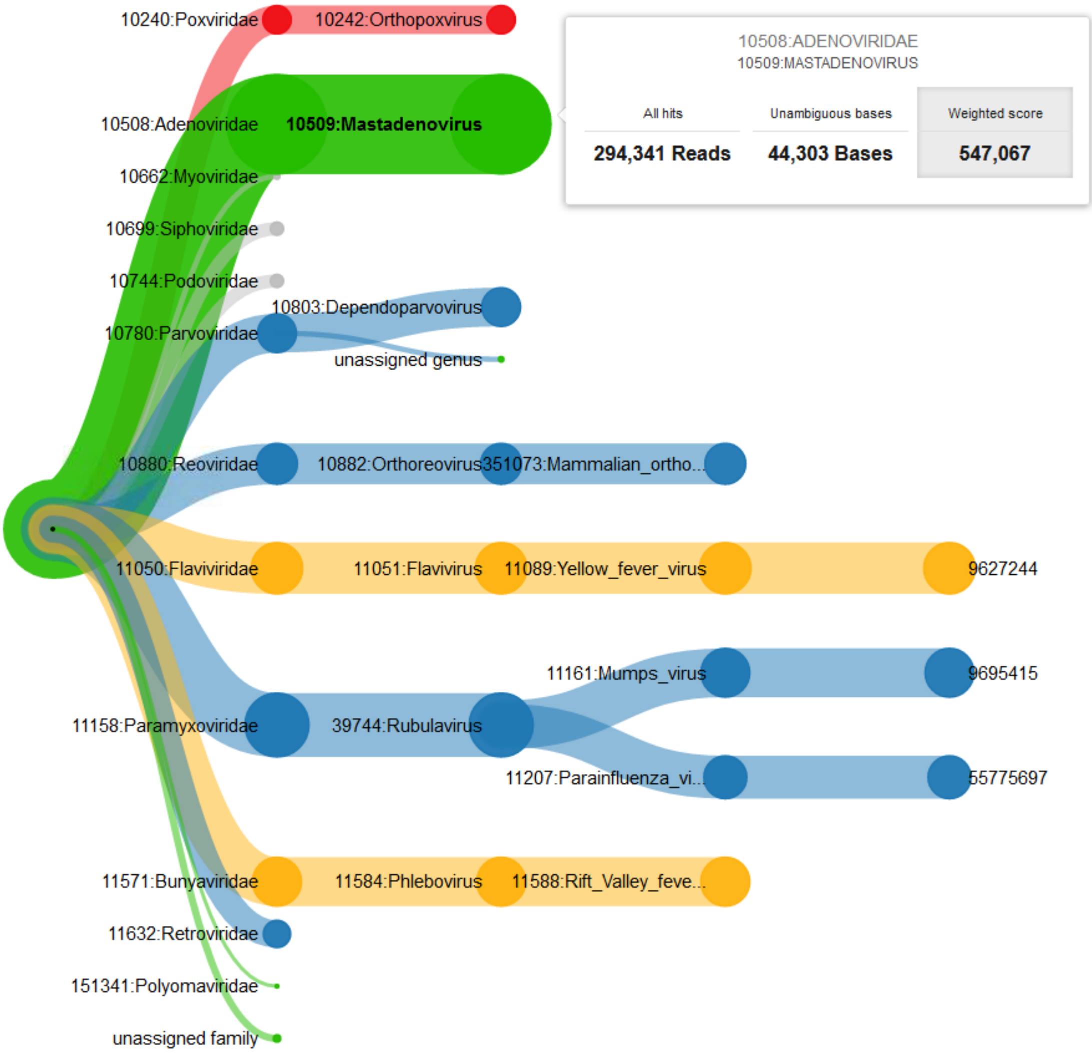


Figure 6 ROC-plot of benchmarked tools on a spiked dataset. Lines have slight offsets in x- and y-dimensions for reasons of distinguishability. We compared PathoLive to Clinical Pathoscope and Bracken on a real human sample containing 7 viruses. PathoLive performs best regarding the ROC-auc at all sampled 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).

We were able to detect all abundant spiked species in the library after only 40 cycles of the sequencing run. 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 7.



10508:ADENOVIRIDAE
10509:MASTADENOVIRUS

All hits	Unambiguous bases	Weighted score
294,341 Reads	44,303 Bases	547,067

Figure 7 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 BLS-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 emphasizes all spiked pathogens through the thickness of their clades, while other species are shown only in smaller clades and therefore ranked lower.

3.4 Discussion

NGS has been shown to be state of the art for pathogen detection, reaching out into clinical usage as well. Although Third Generation Sequencing approaches are also becoming more and more influential, the sequencing depth necessary for open-view diagnostics is only achievable via NGS. This does of course come at cost of higher overall throughput times. PathoLive is, to our knowledge, the first NGS-based diagnostics tool using a real-time approach, facilitating to gain 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, multiplexed sequencing runs can only be assessed 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.

We furthermore changed the basis for the selection of clinically relevant pathogens away 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 [28] or the abundance estimation of Bracken [103] 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, every matching *k-mer* must be regarded as 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. To cope with that, we recommend running PathoLive allowing high numbers of errors to ensure sensitive results at the end of a run and to report only reads with a low error-per-base ratio to exclude random hits at the beginning. This may however lead to the effect observed in our validation experiment, where the results vary over the runtime with the optimal outcome being measured at cycle 80.

Besides these challenges which are unique to PathoLive, we do of course struggle with the same problems as comparable tools. Firstly, 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 established solution for the automated assignment of this information exists. Therefore, the basis for our BSL-levelling approach might not be exhaustive, leading to underestimated danger levels of certain pathogens.

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 superiority of PathoLive not only concerning its unprecedented runtime but also the selection of relevant pathogens. 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.

We hope to provide a helpful tool for accurate and yet rapid detection of pathogens in clinical NGS datasets, overcoming many limitations of existing approaches.

4 Detection of novel pathogens using RAMBO-K

The assembly of viral or endosymbiont genomes from NGS data is often hampered by the predominant abundance of reads originating from the host organism. These reads increase the memory and CPU time usage of the assembler and can lead to misassemblies.

We developed RAMBO-K (Read Assignment Method Based On K-mers), a tool which allows rapid and sensitive removal of unwanted host sequences from NGS datasets. Reaching a speed of 10 Megabases/s on 4 CPU cores and a standard hard drive, RAMBO-K is faster than any tool we tested, while showing a consistently high sensitivity and specificity across different datasets.

RAMBO-K rapidly and reliably separates reads from different species without data preprocessing. It is suitable as a straightforward standard solution for workflows dealing with mixed datasets. Binaries and source code (Java and Python) are available from http://sourceforge.net/projects/rambok/.

4.1 Introduction

The rapid developments in NGS have allowed unprecedented numbers of different organisms to be sequenced. Thanks to the output of current generation sequencing machines, viral and endosymbiont genomes can even be directly sequenced from their host since the huge amount of data generated counterbalances the presence of host sequences. However, especially *de novo* assembly of genomes from datasets from mixed sources is complicated by the large number of background reads, necessitating some form of pre-filtering in order to identify the relevant foreground reads [165].

Here, we present RAMBO-K, a tool which allows the rapid and sensitive extraction of one organism's reads from a mixed dataset, thus facilitating downstream analysis.

4.2 Implementation

In order to separate reads, RAMBO-K uses a reference-driven approach. The user must provide FASTA files containing sequences related to both the foreground (usually the virus or endosymbiont of interest) and the background (usually the host organism). The reference sequences do not have to represent finished genomes; collections of contigs from a draft genome or lists of sequences from different related organisms can be provided if no exact reference is known. Based on these inputs, RAMBO-K performs the sorting of reads in three steps: simulation of reads from reference sequences (s. section 4.2.1); calculation of two Markov Chains, one for the foreground and one for the background, from the simulated reads (s. section 0); and classification of real reads based on their conformance with the Markov Chains (s. section 4.2.3). This workflow is visualized in Figure 8.

4.2.1 Simulation of reads

It is important to ensure that the training set used for the calculation of the Markov Chains is as similar to the real data set as possible. As such, in the first step the mean and the standard deviation of the read length are calculated from a user defined number of reads *n*. There is a trade-off involved in choosing the number of reads to simulate–while more simulated reads allow a better characterization of the foreground and background genomes, simulating more reads also takes more time. In our tests (data not shown), we have found 50’000 Reads to yield good results for the characterization of genomes of up to 3 gbp while not slowing down the calculation too much. We have thus chosen 50’000 as the default value for *n.*

The *n* reads matching the length characteristics of the raw data are generated–error-free and evenly distributed–from both the foreground and the background respectively by generating n sorted random positions in each reference file. Starting from each of these positions, a string of the length of a read is read and checked for non-base characters. If no such characters are found, the characters are saved as a simulated read. The number of successfully simulated reads *m* is saved in each iteration and *n*-*m* reads are generated in the next iteration until a total of *n* reads have been generated. This approach has been chosen since it substitutes reading the whole reference sequence from the hard drive with a series of seek operations which speeds up the read simulation on very large reference genomes while only slightly slowing down the simulation from small reference genomes, which is fast due to the small file size. The simulation process is repeated twice to generate both a training set and a test set.

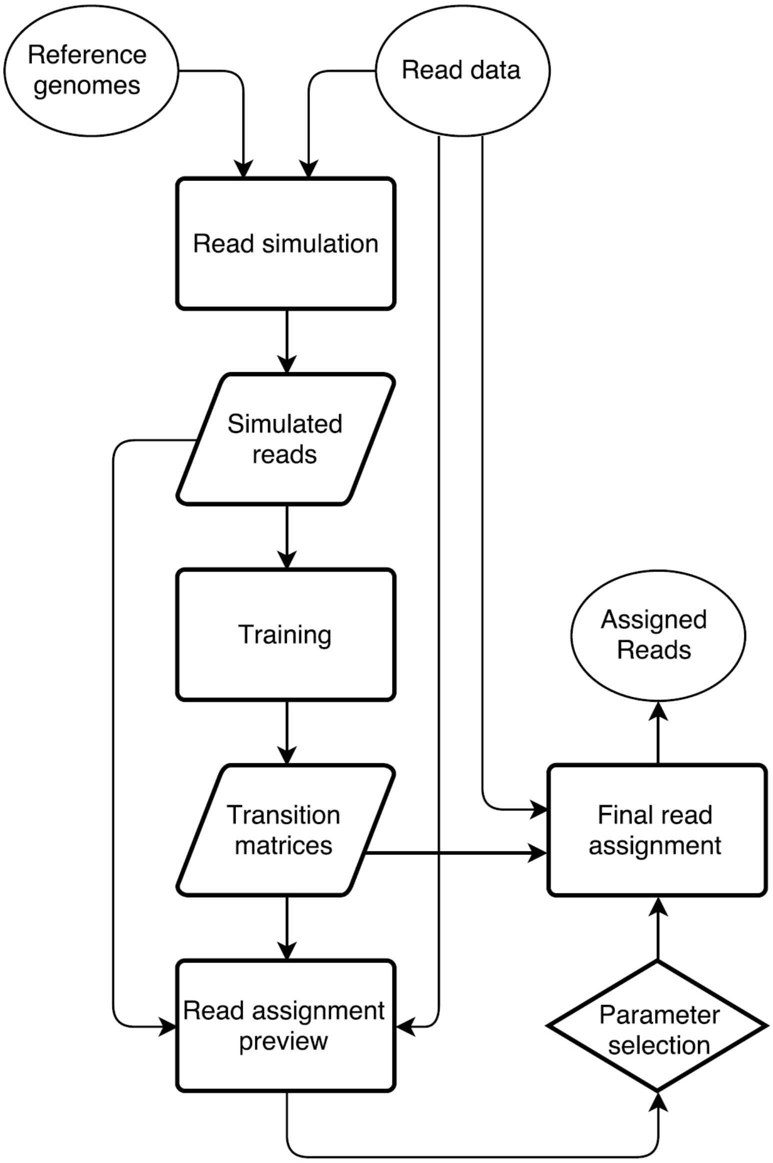


Figure 8 Graphical representation of RAMBO-K's workflow. Reads are simulated from the reference genomes and used to train a foreground and background Markov Chain. The simulated sequences and a subset of the real reads are assigned based on these matrices and a preview of the results is presented to the user. If this preview proves satisfactory, the same parameters are used to assign all reads.

4.2.2 Calculation of Markov Chains

Markov chains of user-specified order k are calculated from the foreground and background read training sets: for each *k-mer* of length $k-1$ the observed probability of being followed by A, G, T or C is calculated. Based on these Markov Chains, a score S for each read from the test set is calculated as follows:

$$S = \sum\nolimits_{i=k}^{l} \log\left(Pr_f\,(B_i|M_{i-1})\right) - \sum\nolimits_{i=k}^{l} \log\left(Pr_b\,(B_i|M_{i-1})\right)$$

where l is the read length, B_i is the base at position i, M_i is the *k-mer* ending at position i and Pr_f and Pr_b are the observed transition probabilities in the foreground and the background Markov Chain, respectively. Conceptually, this is the difference in how well the read is described by the foreground and the background Markov Chains. In order to avoid numeric complications which are likely to arise at higher orders, where the large number of possible *k-mers* leads to small observed probabilities, the logarithms of the probabilities are summed instead of multiplying the probabilities themselves [79].

The score is also calculated for the first 50,000 reads and the scores of both test sets and the reads are then plotted. This allows the user to choose a good cutoff for the subsequent classification (Figure 9). It also allows the user to assess whether separation of the reads is likely to succeed based on the provided reference sequences. If the score distributions from the simulated data overlap well with the score distributions from the real data, as is the case in the example shown in Figure 9, the separation is likely to be successful. In such a case, the plot also gives a first overview of the dataset's composition, since fitting the distributions of scores obtained from the test set to those from the reads allows RAMBO-K to provide a first estimation of the ratio of foreground to background reads in the data. On the other hand, a bad fit of the distribution of real and simulated read's scores indicates a potential problem. One reason could be that the organisms present in the sample are different from the organisms whose genomes were provided as references to RAMBO-K. Often though, it can indicate a poor quality of the data and the resulting need for trimming. In Figure 10, we have provided plots resulting from running RAMBO-K on the same dataset as used in Figure 9, but without first trimming the data.

Since the order of the Markov Chain strongly influences the performance of RAMBO-K, a range of orders for which the calculation is automatically repeated can also be provided (Figure 9). Additionally, ROC plots showing the performance on the simulated data for each *k* are provided.

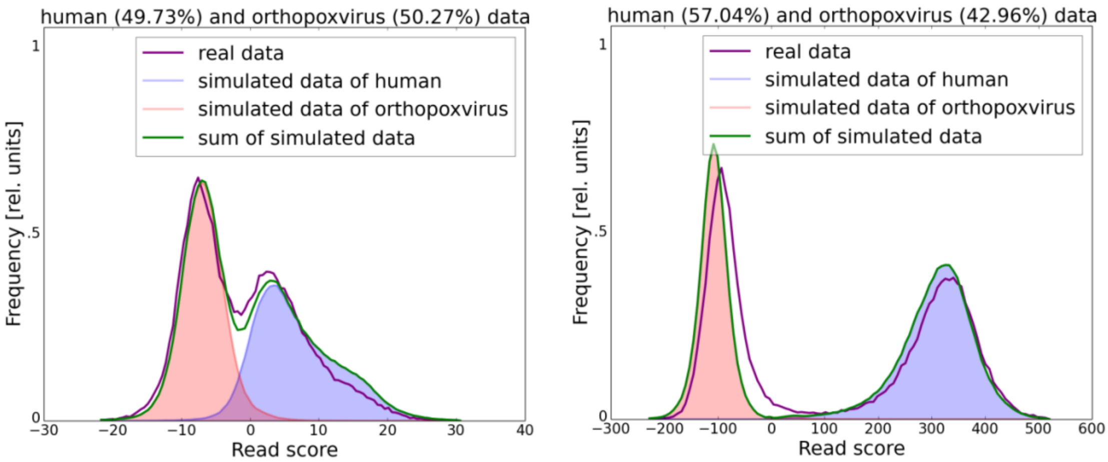


Figure 9 Example of the graphical output of RAMBO-K for a dataset containing human and orthopoxvirus sequences. The score distribution of both simulated and real reads is displayed for two different *k-mer* lengths (left: 4, right: 10), allowing the user to choose the best *k-mer* length and cutoff. In this case, a cutoff around -100 at a *k-mer* length of 10 would allow a clean separation of foreground and background reads, as visualized by the clearly separated peaks. The estimated abundance of foreground and background reads in the dataset is displayed in the figure title.

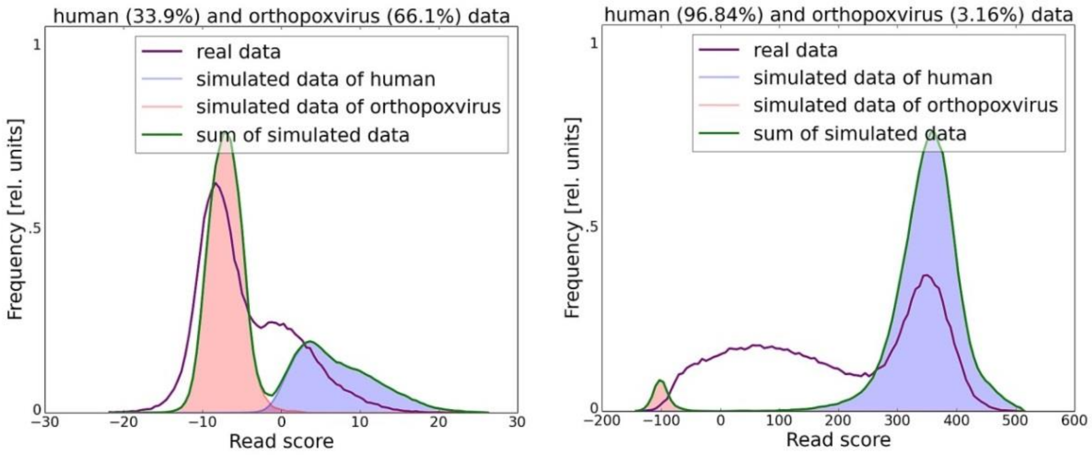


Figure 10 The dataset used in this graphic is the same one as used in Figure 9 and the results for the same *k-mer* lengths (left: 4, right: 10) are shown. However, in this case, the reads have not been trimmed. Two effects are visible: Firstly, the distribution of the real reads' scores deviates much more strongly from the distribution of the simulated reads' scores than is the case with trimmed data. Secondly, due to this discrepancy, RAMBO-K is not able to reliably estimate the relative abundance of reads from the two organisms and the estimate varies widely between the two *k-mer* sizes.

4.2.3 Classification of reads

Once the user has decided upon an upper or lower cutoff and a *k-mer* value, RAMBO-K can be run to classify the real reads based on the previously computed Markov Chains. A score is calculated for each read following the formula given in section 0 and a result file containing only the reads with scores below the upper or above the lower cutoff is created.

4.3 Benchmarking

In order to assess the usefulness of RAMBO-K, we compared its performance with that of several other tools. We used three datasets: (i) Vaccinia virus sequenced from cow lesions; (ii) Bat adenovirus sequenced from a bat, and (iii) Wolbachia endosymbiont sequenced from Drosophila. In addition to RAMBO-K, we used Kraken [94], AbundanceBin [166] and PhymmBL [79] to classify the datasets. While bowtie2 [158] is not a classifier per se, it is often used in preprocessing to either discard all reads not mapping to the foreground reference or to discard all reads mapping to a background reference. We have included both of these mapping-based approaches in our benchmark.

At the time of sequencing of the Bat adenovirus, the closest known genome was that of the distant canine adenovirus. We created our ground truth by mapping the reads to the now known Bat adenovirus genome, but gave all tools only a set of Adenovirus genomes known at the time of sequencing as references for benchmarking.

4.4 Results

As shown in Table 3, RAMBO-K is by far the fastest of all tested tools. Unlike the other tools we tested, which tend to excel either in the high sensitivity or in the low false positive rate department, RAMBO-K gives a high sensitivity at a low cost in terms of false positive assignments. Particularly when working with datasets where an exact reference is not known (such as the Bat adenovirus dataset) – which is becoming more common, especially with the expanding use of NGS in a clinical context – RAMBO-K performs better than current approaches.

A large advantage of RAMBO-K for the preprocessing of NGS data lies in the graphical feedback given to the user. This allows choosing the *k-mer* size and cutoff best suited for each run (

Figure 8). Together with its low runtime and easy installation (RAMBO-K requires only Java and Python 2.7+ with numpy and matplotlib); we believe that it represents a valuable and easy-to-implement step in the preprocessing of NGS data before assembly.

4.5 Additional developments

Several enhancements have been implemented on RAMBO-K since its publication.
Cesare Gruber worked on a function called "I'm feeling lucky" which allows full automation of RAMBO-K with optimally selected parameters. The parameters which cannot be set to meaningful default values easily are being set using two methods: The optimal *k-mer* size is calculated using a Wilcoxon test based on the score distributions of the assigned simulated reads. The cutoff is set based on the minimum difference of specificity and sensitivity from the ROC-plot. In various tests, these automatically set parameters yield good results (data not shown).
A high-level-binning based on RAMBO-K has been worked on to classify reads as viral, bacterial or eukaryotic. The standard Markov Chains from the RAMBO-K algorithm have been trained on sets of all bacterial, viral and eukaryotic genomes from the NCBI RefSeq [133]. Simulated reads from the same database have then been scored based on all three Markov Chains, resulting in three scores per read. These scores have then been scatter-plotted in a three-dimensional space to get a quick visual estimate of the capabilities of this approach. Although slight tendencies towards a signal were noticeable under perfect conditions using errorless reads, this approach would not be of much use in a real scenario due to its low accuracy and was therefore not progressed with.
For maximum availability and user-friendliness, RAMBO-K has been packetized and made available over the official Debian sources under https://packages.debian.org/sid/rambo-k by Andreas Tille.
Additionally, I have made RAMBO-K available on bioconda under https://bioconda.github.io/recipes/rambo-k/README.html, where it has been downloaded more than 850 times to date. A Docker container is available under https://quay.io/repository/biocontainers/rambo-k.

Table 3 Benchmark results. The best value for each dataset is in bold. While Bowtie2+ (keeping reads mapping to the foreground reference) generally gives the lowest false-positive rate (FPR) and Bowtie2- (discarding reads mapping to the background reference) the highest sensitivity (SEN), RAMBO-K shows the best balance, providing high SEN and low FPR (F-Score) with the consistently lowest run-time. RAMBO-K outperforms other methods by the largest margin when the nearest known reference has a low identity to the sequenced genome, as in the Bat adenovirus dataset.

	Cowpox (1.3 M reads, unpublished)				**Bat adenovirus (33 K reads, SRX856705)**				**Wolbachia (12 M reads, SRR1508956)**			
	Time [s]	SEN	FPR	F-Score	Time [s]	SEN	FPR	F-Score	Time [s]	SEN	FPR	F-Score
RAMBO-K	**31**	0,87	3,00E-04	**0,92**	**2,1**	0,79	0,05	**0,86**	**297**	1	4,00E-05	**1**
Kraken	157	0,83	2,00E-05	0,9	4,4	1	0,42	0,8	7004	0	0	N/A
AbundanceBin	20938	0	0	N/A	73	0,99	0,88	0,65	1,10E+06	0,5	0,48	0,07
PhymmBL	82556	0,68	1,00E-04	0,8	1,00E+05	0	0	N/A	1.7E7[a]	0,5	2E-3[a]	0.64[a]
Bowtie2+	146	0,85	**1,00E-05**	0,92	5,1	0,11	**0**	0,2	419	0,99	**3,00E-06**	0,99
Bowtie2-	550	**0,95**	0,76	0,03	93	**1**	0,91	0,65	1274	**1**	0,97	0,07

[a]The values for PhymmBL on the Wolbachia dataset were extrapolated based on the analysis of a subset of 5% of the reads

5 Discovery of a new *poxvirus* genus

Near Berlin, Germany, several juvenile red squirrels (*Sciurus vulgaris*) were found with moist, crusty skin lesions. Histology, electron microscopy, and cell culture isolation revealed an *orthopoxvirus*-like infection. Subsequent PCR and genome analysis identified a new *poxvirus* (Berlin squirrelpox virus) that could not be assigned to any known *poxvirus* genera.

5.1 Introduction

The Eurasian red squirrel (*Sciurus vulgaris*) is the only species of tree squirrels endemic throughout most of Europe. Although they are usually abundant, red squirrels are endangered or extinct in some regions in Great Britain and Ireland that are co-inhabited by invasive eastern gray squirrels (*Sciurus carolinensis*), which were introduced from North America in the late 19th century. One major threat is the transmission of *squirrelpox virus* (SQPV) from the gray squirrel reservoir host to red squirrels, which succumb to lethal infections [167]. SQPV had been assigned to the *parapoxviruses* due to morphological similarities [168], but the latest viral genome data placed it in a separate clade within the *poxvirus* family [169]. Recently, different *poxviruses* have been associated with similar lesions in American red squirrels (*Tamiasciurus hudsonicus*) from Canada [170], but except for a single case report from Spain [171], no poxvirus infections in squirrels have been reported in continental Europe.

5.2 The Study

In 2015 and 2016, at least 10 abandoned weak juvenile red squirrels were submitted to a sanctuary near Berlin, Germany. The animals had exudative and erosive-to-ulcerative dermatitis with serocellular crusts at auricles, noses, digits, tails, and genital/perianal regions. Skin specimens from affected animals were investigated by electron microscopy (EM) and PCR. Three animals that died under care were submitted for necropsy. We obtained samples of all organs for histological and PCR examination. We used 1 sample of a skin lesion for virus propagation in cell culture.

Discovery of a new poxvirus genus

EM-negative staining of skin lesions from all animals led to the discovery of brick-shaped poxvirus particles with irregular threadlike surface fibers and an average size of 294 nm × 221 nm (Figure 11). Pathological findings of corresponding skin lesions were consistent with poxvirus infection (ballooning degeneration of epidermal keratinocytes, numerous intracytoplasmic inclusion bodies, epidermal ulceration with suppurative inflammation, and secondary bacterial infection). All inner organs had either no pathological changes or lesions unrelated to poxvirus infection.

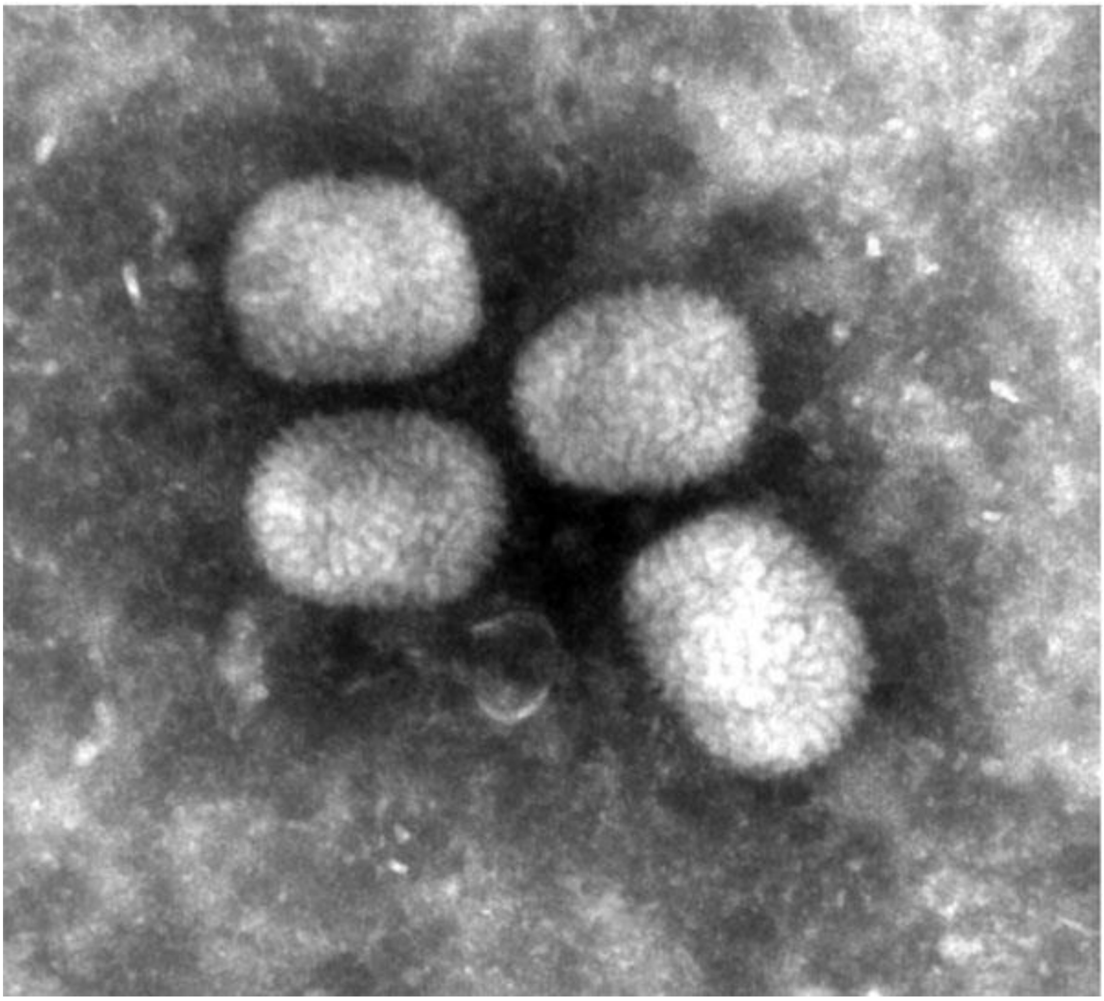


Figure 11 Ultrastructure of Berlin squirrelpox virus particles from skin lesions on a red squirrel in Berlin, Germany, visualized by negative staining (uranyl acetate) (original magnification ×68,000).

Discovery of a new poxvirus genus

To confirm the morphologic diagnosis, we extracted DNA from skin lesions and performed various PCRs. An *orthopoxvirus* (OPV)–specific PCR showed negative results [172]; a *parapoxvirus* (PPV)–specific PCR [172], a *leporipoxvirus*-specific PCR (A. Nitsche and L. Schrick, unpub. data), and a poxvirus-screening PCR [173] were positive for some samples. Obtained sequence fragments indicated poxviral relatedness but did not allow for the assignment to a *poxvirus* genus. Thus, we performed massively parallel sequencing. We directly subjected DNA extracted from a skin lesion on the foot of a dead animal to Nextera XT Library preparation and sequenced it on an Illumina HiSeq 1500 instrument (Illumina, San Diego, CA, USA), yielding 7,242,301 paired-end reads (150 + 150 bases, rapid run mode). Mapping [158] the obtained reads to all poxvirus reference sequences available in GenBank in high-sensitivity mode provided no notable results, which pointed to a virus with a highly deviant genome. Therefore, we separated poxviral reads from background data using RAMBO-K version 1.2 [85] and assembled the resulting 1,520,811 reads [174], yielding 1 single contig of 142,974 bp with ≈460-fold coverage after manual iterative mapping and scaffolding. We confirmed the genomic sequence by resequencing (Illumina MiSeq) of a Vero E6 cell-culture isolate obtained from a different skin specimen of the same animal. We named the new virus Berlin SQPV (BerSQPV), and uploaded the combined sequence information to GenBank (accession no. MF503315). Direct sequencing of DNA from skin samples of three other animals from the same origin yielded sequences with >99.9% identity to BerSQPV.

We compared characteristics of BerSQPV to related viruses and found that the EM structure shows features typical for OPV but the genome size of ≈143 kb is more consistent with PPV or SQPV from the United Kingdom [175] than with the large genome of OPV, whereas the guanine-cytosine (GC) content of 38.5% is more consistent with OPV and *leporipoxvirus* than with PPV and SQPV from the United Kingdom. Therefore, we explored the genomic relationship of BerSQPV to other *chordopoxviruses*. Pairwise alignments of each of the *chordopoxvirus* genomes available in GenBank with the BerSQPV genome resulted in a pairwise identity of at most 47% to tanapox virus isolate TPV-Kenya (accession no. EF420156.1). The retrieved phylogenetic tree (Figure 12Figure 12) demonstrates that BerSQPV cannot be assigned to any of the known poxvirus genera; moreover, it does not cluster with the only other squirrel poxvirus with a published genome sequence [175]. Further phylogenetic analyses based on conserved single genes frequently used for poxvirus tree calculations (A3L, F10L+F12L, F13L, E13L, E9L [VACV Copenhagen nomenclature]) showed similar results (A56R was not used for tree calculations because this open reading frame is too divergent among the *Chordopoxvirinae*), with BerSQPV forming a unique branch (data not shown). In addition, any partial sequences of SQPV available in GenBank were aligned to BerSQPV, showing a maximal sequence identity of 64.3% to gene E9L (GenBank accession no. AY340976.1), further emphasizing the uniqueness of this newly identified virus.

We designed a BerSQPV-specific quantitative PCR based on the genome sequence as a tool for future investigations (primer BerSQPV_F: ggAAgTTTTCCCATACCAACTgA, primer BerSQPV_R: ATCTCAAACCgCAgACggTA, probe BerSQPV_TM: FAM-ACTgTTATTCTTAgCgTAATT). Sensitivity was <10 genome equivalents per reaction amplifying plasmid dilution rows. We first validated the specificity in silico during the design process, revealing the highest identity of 88% to cowpox virus Kostroma (GenBank accession no. KY369926.1), with mismatches in crucial positions in the primer and probe binding sites. Squirrel poxvirus strain Red squirrel UK (GenBank accession no. NC_022563.1) showed only 84% identity, with additional mismatches in amplification-relevant positions. Practical PCR testing using DNA from *cowpox*, *monkeypox*, *ectromelia*, *parapox-ORF*, *myxoma*, *avipox*, and *molluscipox* viruses showed no cross-reactivity.

The new specific quantitative PCR was subsequently applied to DNA from skin lesions archived from 1 squirrel found dead in 2014 in the Berlin area, 2 live squirrels from 2015, and 5 live squirrels from 2016, as well as various organs from 3 affected squirrels necropsied in 2015 (Table 4). Organ tissues yielded high BerSQPV DNA loads in the affected skin but low viral DNA loads for inner organs, findings in concordance with pathological findings, indicating the detection of viral DNA in the blood homogenously distributed throughout the organs with specific tropism for the skin. Low virus loads in inner organs are usually observed in *poxvirus* infections that do not generalize. PCR results indicate that this virus has been circulating in the Berlin area over the past 10 years.

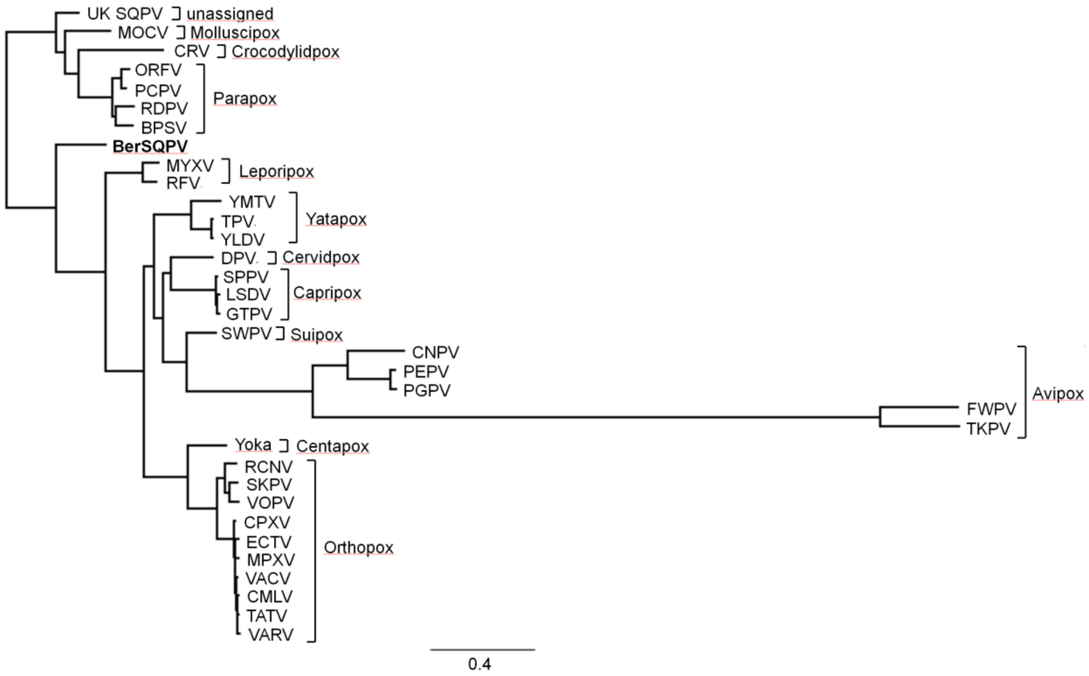


Figure 12 Phylogenetic position of BerSQPV (bold) from a red squirrel in Berlin, Germany, within the *Chordopoxvirinae*. We used MAFFT [130] to perform multiple alignments of all complete genome sequences within a species of the *Chordopoxvirinae* subfamily available in GenBank. The minimum pairwise identity found within any of these intraspecies alignments was 79.1%; the maximum pairwise identity of BerSQPV with any *chordopoxvirus* genome available was 47%. Because of this extreme difference in minimum pairwise identities, we selected individual prototype genomes for each species and the viruses with highest identity to BerSQPV for phylogenetic analysis (as indicated in figure). We performed a multiple alignment of these representative sequences with the BerSQPV genome and removed low-quality regions from the alignment using Gblocks version 0.9 [176], yielding a stripped alignment of 52,563 gap-free positions. The maximum-likelihood tree was then calculated using PhyML [177] (general time reversible plus gamma, 4 substitution rate categories, no invariable sites, BEST topology search, χ2-based parametric branch supports). Scale bar indicates nucleotide substitutions per site. BPSV, bovine papular stomatitis virus BV-AR02 (NC_005337); CMLV, camelpox virus CMS (AY009089); CNPV, canarypox virus Wheatley C93 (NC_005309); CPXV, cowpox virus Brighton Red (AF482758); CRV, Nile crocodilepox virus (NC_008030); DPV, deerpox virus W-848–83 (NC_006966); ECTV, ectromelia virus Moscow (AF012825); FWPV, fowlpox virus NVSL (NC_002188); GTPV, goatpox virus Pellor (NC_004003); LSDV, lumpy skin disease virus NI-2490 (NC_003027); MOCV, Molluscum contagiosum virus subtype 1 (NC_001731); MPXV, monkeypox virus Zaire-96-I-16 (AF380138); MYXV, myxoma virus Lausanne (NC_001132); ORFV, Orf virus OV-SA00 (NC_005336); PCPV, pseudocowpox virus VR634 (NC_013804); PEPV, penguinpox virus (KJ859677); PGPV, pigeonpox virus FeP2 (NC_024447); RCNV, raccoonpox virus Herman (NC_027213); RDPV, red deer pox virus (KM502564); RFV, rabbit fibroma virus Kasza (AF170722); SKPV, skunkpox virus (KU749310); SPPV, sheeppox virus 17077–99 (NC_004002); UK SQPV, squirrel poxvirus Red squirrel UK (HE601899); SWPV, swinepox virus 17077–99 (NC_003389); TATV, taterapox virus Dahomey 1968 (NC_008291); TKPV, turkeypox virus HU1124/2011 (KP728110); TPV, tanapox virus (EF420156); FukVACV, vaccinia virus Copenhagen (M35027); VARV, variola major virus Bangladesh-1975 (L22579); VPXV, volepox virus (KU749311); YLDV, Yaba-like disease virus (NC_002642); YMTV, Yaba monkey tumor virus (NC_005179); Yoka, Yokapox virus (NC_015960)].

Table 4 Results of PCRs of different tissues from seven live and four deceased squirrels showing *poxvirus* lesions*

Year of sampling	Live/ Deceased	Tissue	Cq BerSQPV	Cq c-myc	ΔCq (BerSQPV-c-myc)
2014	Deceased	Archived skin (paraffin)	23.7	34.8	-11.1
2015	Live	Crust‡	12.5	17.7	-5.2
2015	Live	Crust‡	14.8	18.3	-3.5
2015	Deceased	Skin (foot)‡	11.1	18.8	-7.7
		Skin (tail)	9.7	17.9	-8.2
		Skin (toe)†,‡	10.1	18.6	-8.5
		Lung	33.2	27.0	6.2
		Liver	34.7	23.1	11.6
		Spleen	34.9	23.9	11.0
		Brain	33.9	24.5	9.4
2015	Deceased	Skin (forefoot)‡	10.9	18.2	-7.3
		Skin	26.3	28.0	-1.7
		Lung	33.6	23.1	10.5
		Liver	neg	22.1	-
		Spleen	38.3	23.9	14.4
		Kidney	neg	24.1	-
		Small intestine	neg	21.8	-
		Large intestine	neg	24.4	-
		Brain	neg	25.3	-
2015	Deceased	Crust	19.0	23.2	-4.2
		Lung	35.2	25.4	9.8
		Liver	neg	20.8	-
		Spleen	34.0	25.1	8.9
		Kidney	neg	25.9	-
		Small intestine	36.4	21.6	14.8
		Large intestine	35.0	23.5	11.5
		Brain	neg	24.6	-
2016	Live	Crust	15.0	22.0	-7.0
2016	Live	Crust	12.1	18.6	-6.5
2016	Live	Crust	14.1	20.8	-6.7
2016	Live	Crust	13.2	17.7	-4.5
2016	Live	Crust	12.9	18.3	-5.4

*BerSQPV DNA was quantified in relation to cellular c-myc DNA; lower values for ΔCq indicate higher virus loads in a respective tissue. Cq, quantification cycle; neg, negative.

†Specimen used to obtain the cell culture isolate

‡Specimen applied to next generation sequencing

5.3 Conclusions

We describe a new *poxvirus*, BerSQPV, isolated from red squirrels in Berlin, Germany, that causes pathological changes consistent with other epidermal *poxvirus* infections. Genome analysis revealed a unique sequence within the *poxvirus* family, as BerSQPV is not clustering to other *poxvirus* genera, including UK SQPV from red squirrels from Great Britain. In contrast to UK SQPV, which resembles PPV ultrastructurally [168], the ultrastructure of BerSQPV is comparable to that of OPV. Two other *poxviruses* from tree squirrels with ultrastructural appearance similar to BerSQPV have been reported: a Eurasian red squirrel from Spain with epidermal *poxvirus* lesions [171] and American red squirrels from Canada [178]. Although no sequence information is available for the SQPV from Spain, the partial sequence analysis of SQPV from Canada showed the virus to also be distinct from all known mammalian *poxviruses* but most closely related to PPV, followed by UK SQPV [178].

BerSQPV is suspected to have been circulating for several years among Eurasian red squirrels in the greater Berlin area. Although diseased animals in care were handled in close contact, caretakers have remained asymptomatic, suggesting a negligible risk for human infection. Further detailed characterization of the isolated virus is ongoing.

6 Summary and conclusion

NGS has proven its applicability in a variety of research fields. It has led to an explosive increase in the number of sequenced species as the discovery of new genomes has become easier than ever before. With the growth of reference databases as well as sequencing capacities, the need for fast and efficient algorithms has become more urgent. Especially in clinical settings, the turnaround times of NGS-based experiments need to be accelerated. Furthermore, the sheer amount and the complexity of data being produced necessitate easily interpretable result presentation. The discovery of novel pathogens has been hampered by a lack of efficient, easy to use tools enabling to assemble the genomes of novel species in an easy way if there is no closely related reference sequence available.

In this thesis, I approached these problems from different angles. We developed three different tools which shorten the turnaround times and greatly simplify the evaluation of NGS-based pathogen related research projects. We were also able to show the possibilities and impact of our developments on a real case in which we combined one of our tools with a variety of other methods.

In Chapter 2, I presented LiveKraken. With this tool, we provide the first and only real-time metagenomic classifier for second generation sequencing data available to date. The method builds directly on the core algorithm of Kraken [94]. We showed that we guarantee to find identical results to those of Kraken by the end of a sequencing run. In this case, we still save the time for base-calling and the execution of Kraken. More importantly, we could show that we achieve comparable results to those of Kraken even at very early stages of a sequencing run. While we do have a slightly lower sensitivity at early stages of the run, the overall abundance ratio of the groups stays approximately the same at all time points. This allows saving up to several days of the overall turnaround time of an experiment for a first overview. LiveKraken is completely focused on minimizing the turnaround time of an experiment including drawing conclusions from its results. To simplify this last step, we have implemented an interactive, browser-based Sankey visualization. This visualization allows a good overview over the predominant taxonomic groups abundant in a sample over a range of time points in a sequencing run. A variety of possibilities opens up with LiveKraken, reaching from real-time quality control over contamination search to rapid selection of candidate pathogens in clinical settings.

A more specialized and sophisticated approach tailored to sensitive real-time pathogen diagnostics has been presented in Chapter 3. We implemented PathoLive, a tool based on the real-time mapper HiLive [72]. Besides being able to report first results days before the sequencer has finished, we also tackle most other challenges of NGS-based diagnostics pipelines with innovative developments. To our knowledge, PathoLive is the first tool integrating information on biosafety levels of candidate hits. Furthermore, instead of a classical background removal step, we mask commonly found sequences as clinically irrelevant. This way, we prevent the unintended deletion of relevant results. Furthermore, a user-defined set of background species will very probably always be incomplete. With our approach, we define an unbiased set of presumably irrelevant sequences independently of the species they belong to. This facilitates getting a more complete background database than those used in comparable tools. Integrating the information on the abundance of sequences in non-ill humans into the scoring of the hits, PathoLive moves the focus away from pure abundance estimation of candidate pathogens towards relevance estimation. This is backed by providing biosafety levels in the visualization. As the abundance alone is evidentially not a meaningful metric of relevance, we hope that our workflow contributes to a new understanding of NGS-based pathogen diagnostics. As in LiveKraken, we again tried to optimize not only the algorithm runtimes but to keep an eye on the overall turnaround time, including the final evaluation of the results. We implemented an interactive visualization showing a taxonomic tree encoding different optional scoring methods for all taxonomic levels. PathoLive furthermore includes coverage plots and color codes the biosafety level of each branch. We were able to show that PathoLive performs superior to the other tested tools at all time points of a real sequencing run – even days before the other tools could be started.

Given the strict focus on minimizing the turnaround times of an experiment with LiveKraken and PathoLive, both of these tools are not meant to address complex research questions which require time-consuming additional work. Although PathoLive does report real alignments and therefore enables deeper characterization of detected pathogens, this is not its key task.

In Chapter 4, I presented RAMBO-K, a tool for the binning of reads into fore- and background [85]. This enables the detection of reads of interest even if no close or complete reference is available. The algorithm is based on a *k-mer* Markov Chain which is used to determine the sequence characteristics of reference sequences as well as those of the reads. It is trained on user-provided sets of fore- and background sequences. According to these characteristics, all reads from a sequencing run are scored and afterwards assigned to either fore- or background. Since this method is tailored to highly complex cases with a large bandwidth of unique difficulties, we decided to provide the user with detailed visual feedback. This visualization allows a good estimate at the optimal parameter selection by combining simulated reads from the user-defined references as a simplified ground-truth with a subset of the real dataset. We showed that RAMBO-K outperforms other tools we tested in three real cases in terms of runtime and F-score. Especially for the discovery of novel pathogens, this method can simplify the overall project significantly. It has already been used to discover several new viral genomes [127, 128, 179, 180].

In Chapter 5 we made use of RAMBO-K as well as several other bioinformatics methods to discover what is believed to establish a new genus of *poxviruses*. The common techniques failed to identify significant amounts of viral material in the sequencing data from a squirrel infected with a yet unknown *poxvirus*. Only after we used RAMBO-K to select candidate poxviral reads, we found that there were viral reads which had a very low similarity to any known reference. Afterwards, we were able to assemble the genome of Berlin Squirrelpox Virus (BerSQPV) from these reads. Although morphologic attributes suggested that BerSQPV was part of the clade of *parapoxviruses*, genomic and phylogenetic analyses found that it rather establishes a whole new genus of *poxvirinae*.

Together, the developed methods enable the rapid detection of pathogens in different settings. LiveKraken may give a good first overview of any sequencing project, whereas PathoLive produces a full-featured foundation for pathogen diagnostics. With RAMBO-K, even novel pathogens can be discovered in a comparably simple manner, as showcased at the example of BerSQPV.

Future research

Although each of the proposed tools closes major research gaps in the field of NGS-based pathogen related research, the conducted experiments also indicate that further development promises even better results in some aspects. Additionally, these newly established methods open up a number of follow-up ideas and questions which should be explored.

LiveKraken proved to work just as well as the original Kraken and comparably well if provided with early-stage data of a sequencing run. Still, we are aware that Kraken was the first rapid metagenomic classifier of its kind in 2014. Although it is still widespread and commonly used as a default tool for many projects, there are enhanced methods available by now. Some of these methods outperform Kraken regarding its runtime, but due to the real-time implementation of LiveKraken this bottleneck can be ignored as long as sequencing runtimes do not outpace the read classification. More interestingly, Kraken is continuously worked on by the Salzberg Lab with Bracken being the most recent published result. Integrating the Bracken extension into LiveKraken would result in the first real-time abundance estimation tool. Although the reassignment step could only be started after a LiveKraken report for a given cycle has been finished, this would still be a great extension of LiveKraken's current functionality.

With PathoLive, we proposed a variety of innovations approaching different challenges of NGS-based pathogen detection pipelines. While we already showed that it enables unprecedented turnaround times and highly accurate results, these novelties are also meant to be a proof of principle for a new perspective on NGS-based diagnostics.

The presentation of BSL-levels of detected taxonomic clades is one example for this. While giving information on the hazardousness of abundant pathogens helps to put focus on the relevant conclusions and thus getting actionable results quickly, the BSL-level is still a too superficial measure. Instead, more sophisticated metrics should be implemented. They should include clinical symptoms and other anamnesis data of a patient, e.g. on the travel history, age, risk factors and many more. While computer programs for the integration of these data into diagnostics already exist, they have to our knowledge not been implemented in NGS-based diagnostics pipelines so far [181]. Including anamnesis data into PathoLive would allow emphasizing promising candidate pathogens even better.

Regarding the masking of clinically irrelevant sequences, there is also room for improvement left. While the data from which we derive our model is appropriate, it is not directly meant to be used for this kind of conclusions. We cannot guarantee that all participants of the underlying experiments were really free of potentially relevant infectious material. Furthermore, the data may stem from different tissues. With the number of available raw sequencing datasets steadily growing, a more specific choice of datasets to define a baseline can be made. Sampling tissue specific sequencing datasets for masking may further reduce the amount of falsely masked bases. As an example, we currently mask parts of several herpes viruses as they are commonly carried by healthy humans. Nevertheless, these viruses may cause serious medical conditions if they reach the brain. In order to cope with these cases, we currently provide different metrics, so that such an event would not remain undetected at a second glance. Yet, having more detailed information on the baseline data could alleviate this problem even further.

Another problem which has to our knowledge not been addressed by any of the established methods including PathoLive is the handling of lab-specific contaminations. In different experiments, we found that the inevitable device-specific carry-over contaminations on Illumina sequencers are detectable for as many as five subsequent runs (data not shown). As each and every read in a dataset must be considered relevant, even low-level contaminations from previous sequencing runs may confound the results of an experiment. These lab-specific sequences cannot be captured by our background masking, which only relies on publically available data from a multitude of different sequencing facilities and machines. Instead, a number of preceding sequencing runs on a specific machine should be monitored for recurring sequences. This could enable the prediction of a machine-specific baseline of expected contaminating sequences for a given run. Implementing such a contamination detection feature in the result presentation could help to prevent misdiagnoses.

The aforementioned ideas for refinements are based on the understanding that NGS-based diagnostics should be more than just metagenome abundance estimation, as the abundance of a species is evidentially not a meaningful metric of clinical relevance. Additionally to the suggested possible refinements of the relevance estimation of detected sequences, there is also potential for algorithmic improvement. As stated before, the final evaluation of results is designed to be made by clinicians or researchers, as we believe a fully automated selection is not yet feasible. Therefore, PathoLive does not follow a sophisticated method for read reassignment. Ambiguous hits can thus sometimes not be meaningfully classified. Although strain-level classification is not the core task of PathoLive and generally difficult on early stages of a sequencing run due to lacking sequence information, implementing a read reassignment step may yield even more precise results.

Although theoretically already possible, PathoLive has not yet been tested with bacterial or eukaryotic pathogens. The generation of a more complete reference database for testing should generally be unproblematic. Especially for complex bacterial communities such as the gut microbiome, pure abundance estimation yields mostly clinically irrelevant species. We expect our proposed background masking to have an even bigger positive impact on these datasets if a meaningful baseline is selected.

Some general questions came up when we started implementing real-time NGS tools. For example, reporting demultiplexed results is only possible after the index has been sequenced. Although algorithmically already possible, sequencing indices before the first read has not yet been tested by us and is expected to be problematic, as the first sequenced bases of a run are used for cluster detection by the sequencer. If these clusters contain too many similar bases at the same positions and therefore send out the same fluorescent signal, neighboring clusters cannot be distinguished reliably. As this is the case with most low-complex sequence sets like multiplex-indices, the sequencing run may fail altogether. Thus, the base composition of index sets would have to be taken into account for the switch of the sequencing order to work.

As reads are processed while sequencing, it is impossible to prefix any quality control programs. To cope with this, a real-time read trimmer or at least the inclusion of base qualities into the existing live tools could enable further improvements of all existing real-time NGS tools. While pure quality based read trimming is trivial to achieve, further quality control measures like adapter trimming would need to be based on an alignment step. Using HiLive in combination with a database of expected adapters could provide a solution and allow more sophisticated real-time quality control.

Finally, the functionality of real-time analyses of NGS data is still restricted to metagenome classification and mapping-based applications as provided by HiLive [72], PathoLive and PriLive [19]. Like all reference-based methods, these require the availability of somehow similar sequences. Alignment-free methods such as *de novo* assemblers enable the discovery of deviant or even completely novel species, as proven by crAss proved [40]. Having a real-time *de novo* assembler could enable reporting a pathogen's genome at the end of a sequencing run. Furthermore, even if full genome assembly does not work in every case, working with longer contigs instead of short reads may simplify other follow-up analyses. Since deBruijn graphs are a data structure which is based on the decomposition of reads into shorter *k-mers* independently of their order, their usage for the real-time assembly of massively parallel sequenced reads seems trivial. As an example, Faucet proves that streaming bases into a de Bruijn graph-based assembler is possible [182]. Although the proposed approach is structurally different to that of Faucet, which streams read by read instead of cycle by cycle, we expect the concept to be transferable. With this, we could shut another major methodological gap of real-time NGS-based analysis tools for pathogen detection.

The development of the read assignment tool RAMBO-K aimed at facilitating efficient and highly sensitive selection of reads of interest from mixed datasets. It proved itself in practice especially for the distinction of viral reads from host reads in virus sequencing projects. The limitation to two groups in one dataset is not a problem in these cases. Still, enabling read-assignment for a larger number of groups could open up new applications. The visualization is currently limited to two-dimensional score distributions. With larger numbers of bins, the number of dimensions would increase, complicating the concept of the visualization. An automated parameter selection as proposed by Cesare Gruber in personal communication could still be implemented for higher dimensions.

As RAMBO-K relies on a *k-mer* based approach, it could rather easily be implemented as a real-time NGS tool as well. There is no big conceptual difference to LiveKraken. It could therefore act as a read preselection tool for the real-time assembler proposed above. In cases where no similar reference sequence is expected to be available but a hint towards a candidate group of pathogens exists, this workflow could potentially yield the pathogen's genome with the end of a sequencing run. RAMBO-K could in this workflow reduce the memory requirements and increase the chances of assembling the desired genome. This could have for example been used for the assembly of BerSQPV, where we expected to find some *poxvirus* but lacked detailed information on the species in the sample.

Finally, the discovery of BerSQPV has of course opened up further research questions. Although we have already conducted research and published results beyond the absolute basics of the discovery, further characterization of the virus and the genome are work in progress. One important basis for deeper characterization on the genomic level is the annotation of the genome. Unfortunately, the low similarity of BerSQPV to any annotated reference sequence hampers this step. None of the annotation tools we have tested so far yielded satisfying results. Therefore, a new genome annotation pipeline is being implemented. The general workflow is comparable to available solutions, but instead of selecting candidate open reading frames with high probability of being a coding sequence, we BLAST [77] all available ORFs in the genome against all annotated coding sequences from a given set of genomes. Our new annotation pipeline will therefore be designed to be more sensitive than any comparable tool. Besides enabling the annotation of BerSQPV, it could as well help with the annotation of other novel genomes without closely related and therefore highly similar reference sequences.

7 Supplementary material

7.1 List of evaluated datasets

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

8 Bibliography

1. Allen, E.E. and J.F. Banfield, *Community genomics in microbial ecology and evolution.* Nat Rev Microbiol, 2005. **3**(6): p. 489-98.
2. Handelsman, J., et al., *Molecular biological access to the chemistry of unknown soil microbes: a new frontier for natural products.* Chem Biol, 1998. **5**(10): p. R245-9.
3. Gilbert, J.A. and C.L. Dupont, *Microbial metagenomics: beyond the genome.* Ann Rev Mar Sci, 2011. **3**: p. 347-71.
4. Lindner, M.M.S., *Computational methods for the identification and quantification of microbial organisms in metagenomes*. 2014.
5. Streit, W.R. and R.A. Schmitz, *Metagenomics - the key to the uncultured microbes.* Curr Opin Microbiol, 2004. **7**(5): p. 492-8.
6. Edwards, R.A. and F. Rohwer, *Viral metagenomics.* Nat Rev Microbiol, 2005. **3**(6): p. 504-10.
7. Venter, J.C., et al., *Environmental genome shotgun sequencing of the Sargasso Sea.* Science, 2004. **304**(5667): p. 66-74.
8. Turnbaugh, P.J., et al., *The human microbiome project.* Nature, 2007. **449**(7164): p. 804-10.
9. Mokili, J.L., F. Rohwer, and B.E. Dutilh, *Metagenomics and future perspectives in virus discovery.* Curr Opin Virol, 2012. **2**(1): p. 63-77.
10. Ribeiro, A.A., et al., *The oral bacterial microbiome of occlusal surfaces in children and its association with diet and caries.* PLoS One, 2017. **12**(7): p. e0180621.
11. Liu, R., et al., *Gut microbiome and serum metabolome alterations in obesity and after weight-loss intervention.* Nat Med, 2017. **23**(7): p. 859-868.
12. Simner, P.J., S. Miller, and K.C. Carroll, *Understanding the Promises and Hurdles of Metagenomic Next-Generation Sequencing as a Diagnostic Tool for Infectious Diseases.* Clin Infect Dis, 2018. **66**(5): p. 778-788.
13. Pallen, M.J., *Diagnostic metagenomics: potential applications to bacterial, viral and parasitic infections.* Parasitology, 2014. **141**(14): p. 1856-62.
14. Schlaberg, R., et al., *Validation of Metagenomic Next-Generation Sequencing Tests for Universal Pathogen Detection.* Arch Pathol Lab Med, 2017. **141**(6): p. 776-786.
15. Hall, R.J., et al., *Beyond research: a primer for considerations on using viral metagenomics in the field and clinic.* Front Microbiol, 2015. **6**: p. 224.
16. Dutilh, B.E., et al., *Editorial: Virus Discovery by Metagenomics: The (Im)possibilities.* Front Microbiol, 2017. **8**: p. 1710.
17. Quick, J., et al., *Rapid draft sequencing and real-time nanopore sequencing in a hospital outbreak of Salmonella.* Genome Biol, 2015. **16**: p. 114.
18. Pendleton, K.M., et al., *Rapid Pathogen Identification in Bacterial Pneumonia Using Real-Time Metagenomics.* Am J Respir Crit Care Med, 2017. **196**(12): p. 1610-1612.
19. Loka, T.P., et al., *PriLive: Privacy-preserving real-time filtering for Next Generation Sequencing.* Bioinformatics, 2018.

20. Schmieder, R. and R. Edwards, *Fast identification and removal of sequence contamination from genomic and metagenomic datasets.* PLoS One, 2011. **6**(3): p. e17288.
21. Naccache, S.N., et al., *A cloud-compatible bioinformatics pipeline for ultrarapid pathogen identification from next-generation sequencing of clinical samples.* Genome Res, 2014. **24**(7): p. 1180-92.
22. Zheng, Y., et al., *VirusDetect: An automated pipeline for efficient virus discovery using deep sequencing of small RNAs.* Virology, 2017. **500**: p. 130-138.
23. Scheuch, M., D. Hoper, and M. Beer, *RIEMS: a software pipeline for sensitive and comprehensive taxonomic classification of reads from metagenomics datasets.* BMC Bioinformatics, 2015. **16**: p. 69.
24. Naeem, R., M. Rashid, and A. Pain, *READSCAN: a fast and scalable pathogen discovery program with accurate genome relative abundance estimation.* Bioinformatics, 2013. **29**(3): p. 391-2.
25. Hong, C., et al., *PathoScope 2.0: a complete computational framework for strain identification in environmental or clinical sequencing samples.* Microbiome, 2014. **2**: p. 33.
26. Francis, O.E., et al., *Pathoscope: species identification and strain attribution with unassembled sequencing data.* Genome Res, 2013. **23**(10): p. 1721-9.
27. Flygare, S., et al., *Taxonomer: an interactive metagenomics analysis portal for universal pathogen detection and host mRNA expression profiling.* Genome Biol, 2016. **17**(1): p. 111.
28. Byrd, A.L., et al., *Clinical PathoScope: rapid alignment and filtration for accurate pathogen identification in clinical samples using unassembled sequencing data.* BMC Bioinformatics, 2014. **15**: p. 262.
29. Brown, J.R., T. Bharucha, and J. Breuer, *Encephalitis diagnosis using metagenomics: application of next generation sequencing for undiagnosed cases.* J Infect, 2018. **76**(3): p. 225-240.
30. Abril, M.K., et al., *Diagnosis of Capnocytophaga canimorsus Sepsis by Whole-Genome Next-Generation Sequencing.* Open Forum Infect Dis, 2016. **3**(3): p. ofw144.
31. Drosten, C., et al., *Identification of a novel coronavirus in patients with severe acute respiratory syndrome.* N Engl J Med, 2003. **348**(20): p. 1967-76.
32. Ksiazek, T.G., et al., *A novel coronavirus associated with severe acute respiratory syndrome.* N Engl J Med, 2003. **348**(20): p. 1953-66.
33. Carroll, M.W., et al., *Temporal and spatial analysis of the 2014-2015 Ebola virus outbreak in West Africa.* Nature, 2015. **524**(7563): p. 97-101.
34. Mari Saez, A., et al., *Investigating the zoonotic origin of the West African Ebola epidemic.* EMBO Mol Med, 2015. **7**(1): p. 17-23.
35. Tang, P. and C. Chiu, *Metagenomics for the discovery of novel human viruses.* Future Microbiol, 2010. **5**(2): p. 177-89.
36. Anthony, S.J., et al., *A strategy to estimate unknown viral diversity in mammals.* MBio, 2013. **4**(5): p. e00598-13.
37. Hufsky, F., et al., *Virologists-Heroes need weapons.* PLoS Pathog, 2018. **14**(2): p. e1006771.

38. Radford, A.D., et al., *Application of next-generation sequencing technologies in virology.* J Gen Virol, 2012. **93**(Pt 9): p. 1853-68.
39. Datta, S., et al., *Next-generation sequencing in clinical virology: Discovery of new viruses.* World J Virol, 2015. **4**(3): p. 265-76.
40. Dutilh, B.E., et al., *Reference-independent comparative metagenomics using cross-assembly: crAss.* Bioinformatics, 2012. **28**(24): p. 3225-31.
41. Dutilh, B.E., et al., *A highly abundant bacteriophage discovered in the unknown sequences of human faecal metagenomes.* Nat Commun, 2014. **5**: p. 4498.
42. Zhao, G., et al., *VirusSeeker, a computational pipeline for virus discovery and virome composition analysis.* Virology, 2017. **503**: p. 21-30.
43. Huson, D.H. and N. Weber, *Microbial community analysis using MEGAN.* Methods Enzymol, 2013. **531**: p. 465-85.
44. Deng, X., et al., *An ensemble strategy that significantly improves de novo assembly of microbial genomes from metagenomic next-generation sequencing data.* Nucleic Acids Res, 2015. **43**(7): p. e46.
45. Huson, D.H. and S. Mitra, *Introduction to the analysis of environmental sequences: metagenomics with MEGAN.* Methods Mol Biol, 2012. **856**: p. 415-29.
46. Huson, D.H., et al., *MEGAN Community Edition - Interactive Exploration and Analysis of Large-Scale Microbiome Sequencing Data.* PLoS Comput Biol, 2016. **12**(6): p. e1004957.
47. Huson, D.H., et al., *MEGAN analysis of metagenomic data.* Genome Res, 2007. **17**(3): p. 377-86.
48. Alves, J.M., et al., *GenSeed-HMM: A Tool for Progressive Assembly Using Profile HMMs as Seeds and its Application in Alpavirinae Viral Discovery from Metagenomic Data.* Front Microbiol, 2016. **7**: p. 269.
49. Norling, M., et al., *MetLab: An In Silico Experimental Design, Simulation and Analysis Tool for Viral Metagenomics Studies.* PLoS One, 2016. **11**(8): p. e0160334.
50. Zhao, G., et al., *Identification of novel viruses using VirusHunter - an automated data analysis pipeline.* PLoS One, 2013. **8**(10): p. e78470.
51. Wommack, K.E., et al., *VIROME: a standard operating procedure for analysis of viral metagenome sequences.* Stand Genomic Sci, 2012. **6**(3): p. 427-39.
52. Skewes-Cox, P., et al., *Profile hidden Markov models for the detection of viruses within metagenomic sequence data.* PLoS One, 2014. **9**(8): p. e105067.
53. Kostic, A.D., et al., *PathSeq: software to identify or discover microbes by deep sequencing of human tissue.* Nat Biotechnol, 2011. **29**(5): p. 393-6.
54. Roux, S., et al., *Metavir: a web server dedicated to virome analysis.* Bioinformatics, 2011. **27**(21): p. 3074-5.
55. Roux, S., et al., *Metavir 2: new tools for viral metagenome comparison and assembled virome analysis.* BMC Bioinformatics, 2014. **15**: p. 76.
56. Li, Y., et al., *VIP: an integrated pipeline for metagenomics of virus identification and discovery.* Sci Rep, 2016. **6**: p. 23774.
57. Barzon, L., et al., *Applications of next-generation sequencing technologies to diagnostic virology.* Int J Mol Sci, 2011. **12**(11): p. 7861-84.

58. Soueidan, H., et al., *Finding and identifying the viral needle in the metagenomic haystack: trends and challenges.* Front Microbiol, 2014. **5**: p. 739.
59. Drake, J.W., et al., *Rates of spontaneous mutation.* Genetics, 1998. **148**(4): p. 1667-86.
60. Weber, J.L. and E.W. Myers, *Human whole-genome shotgun sequencing.* Genome Res, 1997. **7**(5): p. 401-9.
61. Venter, J.C., et al., *The sequence of the human genome.* Science, 2001. **291**(5507): p. 1304-51.
62. Liu, L., et al., *Comparison of next-generation sequencing systems.* J Biomed Biotechnol, 2012. **2012**: p. 251364.
63. Schirmer, M., et al., *Illumina error profiles: resolving fine-scale variation in metagenomic sequencing data.* BMC Bioinformatics, 2016. **17**: p. 125.
64. Illumina, I. *Evolution of Illumina Sequencing*. 2018 12.03.2018]; Available from: https://www.illumina.com/science/technology/next-generation-sequencing/illumina-sequencing-history.html.
65. Goodwin, S., J.D. McPherson, and W.R. McCombie, *Coming of age: ten years of next-generation sequencing technologies.* Nat Rev Genet, 2016. **17**(6): p. 333-51.
66. Inc., I. *An introduction to Next-Generation Sequencing Technology*. 2017 [cited 2018 28.03.2018]; Available from: https://www.illumina.com/content/dam/illumina-marketing/documents/products/illumina_sequencing_introduction.pdf.
67. Inc., I. *Illumina Sequencing Technology*. 2013 23.10.2013 [cited 2018 28.03.2018]; Available from: https://www.youtube.com/watch?v=womKfikWlxM.
68. Canzar, S. and S.L. Salzberg, *Short Read Mapping: An Algorithmic Tour.* Proc IEEE Inst Electr Electron Eng, 2017. **105**(3): p. 436-458.
69. Ozsolak, F. and P.M. Milos, *RNA sequencing: advances, challenges and opportunities.* Nat Rev Genet, 2011. **12**(2): p. 87-98.
70. Profaizer, T., et al., *1012-Lbp: A comparison of physical and enzymatic fragmentation methods in library preparation for HLA sequencing on the illumina MISEQ.* Human Immunology, 2014. **75**(6): p. 485.
71. Technologies, O.N. *Sequencing with Nanopore Technology*. 2018 12.03.2018]; Available from: https://nanoporetech.com/learn-more.
72. Lindner, M.S., et al., *HiLive: real-time mapping of illumina reads while sequencing.* Bioinformatics, 2017. **33**(6): p. 917-319.
73. Needleman, S.B. and C.D. Wunsch, *A general method applicable to the search for similarities in the amino acid sequence of two proteins.* J Mol Biol, 1970. **48**(3): p. 443-53.
74. Smith, T.F. and M.S. Waterman, *Identification of common molecular subsequences.* J Mol Biol, 1981. **147**(1): p. 195-7.
75. Wilbur, W.J. and D.J. Lipman, *Rapid similarity searches of nucleic acid and protein data banks.* Proc Natl Acad Sci U S A, 1983. **80**(3): p. 726-30.
76. Pearson, W.R. and D.J. Lipman, *Improved tools for biological sequence comparison.* Proc Natl Acad Sci U S A, 1988. **85**(8): p. 2444-8.

77. Altschul, S.F., et al., *Basic local alignment search tool.* J Mol Biol, 1990. **215**(3): p. 403-10.
78. Fonseca, N.A., et al., *Tools for mapping high-throughput sequencing data.* Bioinformatics, 2012. **28**(24): p. 3169-77.
79. Brady, A. and S.L. Salzberg, *Phymm and PhymmBL: metagenomic phylogenetic classification with interpolated Markov models.* Nature Methods, 2009. **6**: p. 673-676.
80. Lee, A.Y., 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, 2016. **17**: p. 292.
81. Monzoorul Haque, M., et al., *SOrt-ITEMS: Sequence orthology based approach for improved taxonomic estimation of metagenomic sequences.* Bioinformatics, 2009. **25**(14): p. 1722-30.
82. Ghosh, T.S., et al., *ProViDE: A software tool for accurate estimation of viral diversity in metagenomic samples.* Bioinformation, 2011. **6**(2): p. 91-4.
83. Horton, M., N. Bodenhausen, and J. Bergelson, *MARTA: a suite of Java-based tools for assigning taxonomic status to DNA sequences.* Bioinformatics, 2010. **26**(4): p. 568-9.
84. Schreiber, F., et al., *Treephyler: fast taxonomic profiling of metagenomes.* Bioinformatics, 2010. **26**(7): p. 960-1.
85. Tausch, S.H., et al., *RAMBO-K: Rapid and Sensitive Removal of Background Sequences from Next Generation Sequencing Data.* PLoS One, 2015. **10**(9): p. e0137896.
86. Kislyuk, A., et al., *Unsupervised statistical clustering of environmental shotgun sequences.* BMC Bioinformatics, 2009. **10**: p. 316.
87. Wu, Y.W. and Y. Ye, *A novel abundance-based algorithm for binning metagenomic sequences using l-tuples.* J Comput Biol, 2011. **18**(3): p. 523-34.
88. Teeling, H., et al., *TETRA: a web-service and a stand-alone program for the analysis and comparison of tetranucleotide usage patterns in DNA sequences.* BMC Bioinformatics, 2004. **5**: p. 163.
89. Truong, D.T., et al., *MetaPhlAn2 for enhanced metagenomic taxonomic profiling.* Nat Methods, 2015. **12**(10): p. 902-3.
90. Ghosh, T.S., M. Monzoorul Haque, and S.S. Mande, *DiScRIBinATE: a rapid method for accurate taxonomic classification of metagenomic sequences.* BMC Bioinformatics, 2010. **11 Suppl 7**: p. S14.
91. Mohammed, M.H., et al., *SPHINX - an algorithm for taxonomic binning of metagenomic sequences.* Bioinformatics, 2011. **27**(1): p. 22-30.
92. Droge, J., I. Gregor, and A.C. McHardy, *Taxator-tk: precise taxonomic assignment of metagenomes by fast approximation of evolutionary neighborhoods.* Bioinformatics, 2015. **31**(6): p. 817-24.
93. Ounit, R., et al., *CLARK: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers.* BMC Genomics, 2015. **16**: p. 236.
94. Wood, D.E. and S.L. Salzberg, *Kraken: ultrafast metagenomic sequence classification using exact alignments.* Genome Biol, 2014. **15**(3): p. R46.

95. Ainsworth, D., et al., *k-SLAM: accurate and ultra-fast taxonomic classification and gene identification for large metagenomic data sets.* Nucleic Acids Res, 2017. **45**(4): p. 1649-1656.
96. Bray, N.L., et al., *Near-optimal probabilistic RNA-seq quantification.* Nat Biotechnol, 2016. **34**(5): p. 525-7.
97. Schaeffer, L., et al., *Pseudoalignment for metagenomic read assignment.* Bioinformatics, 2017. **33**(14): p. 2082-2088.
98. Ames, S.K., et al., *Scalable metagenomic taxonomy classification using a reference genome database.* Bioinformatics, 2013. **29**(18): p. 2253-60.
99. Břinda, K., *Novel computational techniques for mapping and classifying Next-Generation Sequencing data*. 2016, Université Paris-Est: Zenodo.
100. Marcais, G. and C. Kingsford, *A fast, lock-free approach for efficient parallel counting of occurrences of k-mers.* Bioinformatics, 2011. **27**(6): p. 764-70.
101. Marchesi, J.R. and J. Ravel, *The vocabulary of microbiome research: a proposal.* Microbiome, 2015. **3**: p. 31.
102. Breitwieser, F.P., J. Lu, and S.L. Salzberg, *A review of methods and databases for metagenomic classification and assembly.* Brief Bioinform, 2017.
103. Lu, J., F.P. Breitwieser, and S.L. Salzberg, *Bracken: estimating species abundance in metagenomics data.* PeerJ Computer Science. **3**(e104).
104. Lu, J., et al., *Bracken: estimating species abundance in metagenomics data.* PeerJ Computer Science, 2017. **3**: p. e104.
105. Lander, E.S. and M.S. Waterman, *Genomic mapping by fingerprinting random clones: a mathematical analysis.* Genomics, 1988. **2**(3): p. 231-9.
106. Warren, R.L., et al., *Assembling millions of short DNA sequences using SSAKE.* Bioinformatics, 2007. **23**(4): p. 500-1.
107. Adams, M.D., et al., *The genome sequence of Drosophila melanogaster.* Science, 2000. **287**(5461): p. 2185-95.
108. Bankevich, A., et al., *SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing.* J Comput Biol, 2012. **19**(5): p. 455-77.
109. Chaisson, M.J. and P.A. Pevzner, *Short read fragment assembly of bacterial genomes.* Genome Res, 2008. **18**(2): p. 324-30.
110. Zerbino, D.R. and E. Birney, *Velvet: algorithms for de novo short read assembly using de Bruijn graphs.* Genome Res, 2008. **18**(5): p. 821-9.
111. Maccallum, I., et al., *ALLPATHS 2: small genomes assembled accurately and with high continuity from short paired reads.* Genome Biol, 2009. **10**(10): p. R103.
112. Butler, J., et al., *ALLPATHS: de novo assembly of whole-genome shotgun microreads.* Genome Res, 2008. **18**(5): p. 810-20.
113. Luo, R., et al., *SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler.* Gigascience, 2012. **1**(1): p. 18.
114. Compeau, P.E., P.A. Pevzner, and G. Tesler, *How to apply de Bruijn graphs to genome assembly.* Nat Biotechnol, 2011. **29**(11): p. 987-91.
115. Dąbrowski, P.W., *Verbesserte Auswertung viraler Next Generation Sequencing-Daten am Beispiel von Kuhpockenviren*, in *Fakultät III - Prozesswissenschaften*. 2015, Technische Universität Berlin.
116. Nurk, S., et al., *metaSPAdes: a new versatile metagenomic assembler.* Genome Res, 2017. **27**(5): p. 824-834.

117. Afiahayati, K. Sato, and Y. Sakakibara, *MetaVelvet-SL: an extension of the Velvet assembler to a de novo metagenomic assembler utilizing supervised learning.* DNA Res, 2015. **22**(1): p. 69-77.

118. Namiki, T., et al., *MetaVelvet: an extension of Velvet assembler to de novo metagenome assembly from short sequence reads.* Nucleic Acids Res, 2012. **40**(20): p. e155.

119. Peng, Y., et al., *IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth.* Bioinformatics, 2012. **28**(11): p. 1420-8.

120. Boisvert, S., et al., *Ray Meta: scalable de novo metagenome assembly and profiling.* Genome Biol, 2012. **13**(12): p. R122.

121. Pell, J., et al., *Scaling metagenome sequence assembly with probabilistic de Bruijn graphs.* Proc Natl Acad Sci U S A, 2012. **109**(33): p. 13272-7.

122. Li, D., et al., *MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph.* Bioinformatics, 2015. **31**(10): p. 1674-6.

123. Li, D., et al., *MEGAHIT v1.0: A fast and scalable metagenome assembler driven by advanced methodologies and community practices.* Methods, 2016. **102**: p. 3-11.

124. Tausch, S.H., et al., *LiveKraken – Real-time metagenomic classification of Illumina data.* Bioinformatics, 2018: p. bty433-bty433.

125. Tausch, S.H., *Rapid and sensitive binning of Next Generation Sequencing data from two species*, in *Centre for Biological Threats and Special Pathogens*. 2015, Free University of Berlin.

126. Kearse, M., et al., *Geneious Basic: an integrated and extendable desktop software platform for the organization and analysis of sequence data.* Bioinformatics, 2012. **28**(12): p. 1647-9.

127. Wibbelt, G., et al., *Berlin Squirrelpox Virus, a New Poxvirus in Red Squirrels, Berlin, Germany.* Emerg Infect Dis, 2017. **23**(10): p. 1726-1729.

128. Schrick, L., et al., *An Early American Smallpox Vaccine Based on Horsepox.* N Engl J Med, 2017. **377**(15): p. 1491-1492.

129. Bolger, A.M., M. Lohse, and B. Usadel, *Trimmomatic: a flexible trimmer for Illumina sequence data.* Bioinformatics, 2014. **30**(15): p. 2114-20.

130. Katoh, K. and D.M. Standley, *MAFFT Multiple Sequence Alignment Software Version 7: Improvements in Performance and Usability.* Molecular Biology and Evolution, 2013. **30**: p. 772-780.

131. Grüning, B., et al., *Bioconda: A sustainable and comprehensive software distribution for the life sciences.* bioRxiv, 2017.

132. Group, N.H.W., et al., *The NIH Human Microbiome Project.* Genome Res, 2009. **19**(12): p. 2317-23.

133. O'Leary, N.A., et al., *Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation.* Nucleic Acids Res, 2016. **44**(D1): p. D733-45.

134. Bzhalava, D., et al., *Unbiased approach for virus detection in skin lesions.* PLoS One, 2013. **8**(6): p. e65953.

135. Greninger, A.L., et al., *Rapid Metagenomic Next-Generation Sequencing during an Investigation of Hospital-Acquired Human Parainfluenza Virus 3 Infections.* J Clin Microbiol, 2017. **55**(1): p. 177-182.
136. Breitwieser, F.P., C.A. Pardo, and S.L. Salzberg, *Re-analysis of metagenomic sequences from acute flaccid myelitis patients reveals alternatives to enterovirus D68 infection.* F1000Res, 2015. **4**: p. 180.
137. Salzberg, S.L., et al., *Next-generation sequencing in neuropathologic diagnosis of infections of the nervous system.* Neurol Neuroimmunol Neuroinflamm, 2016. **3**(4): p. e251.
138. Piro, V.C., M. Matschkowski, and B.Y. Renard, *MetaMeta: integrating metagenome analysis tools to improve taxonomic profiling.* Microbiome, 2017. **5**(1): p. 101.
139. Menzel, P., K.L. Ng, and A. Krogh, *Fast and sensitive taxonomic classification for metagenomics with Kaiju.* Nat Commun, 2016. **7**: p. 11257.
140. Freitas, T.A., et al., *Accurate read-based metagenome characterization using a hierarchical suite of unique signatures.* Nucleic Acids Res, 2015. **43**(10): p. e69.
141. Dadi, T.H., et al., *SLIMM: species level identification of microorganisms from metagenomes.* PeerJ, 2017. **5**: p. e3138.
142. Fosso, B., et al., *MetaShot: an accurate workflow for taxon classification of host-associated microbiome from shotgun metagenomic data.* Bioinformatics, 2017. **33**(11): p. 1730-1732.
143. Piro, V.C., M.S. Lindner, and B.Y. Renard, *DUDes: a top-down taxonomic profiler for metagenomics.* Bioinformatics, 2016. **32**(15): p. 2272-80.
144. Lindner, M.S. and B.Y. Renard, *Metagenomic abundance estimation and diagnostic testing on species level.* Nucleic Acids Res, 2013. **41**(1): p. e10.
145. Frey, K.G., et al., *Comparison of three next-generation sequencing platforms for metagenomic sequencing and identification of pathogens in blood.* BMC Genomics, 2014. **15**: p. 96.
146. Lecuit, M. and M. Eloit, *The potential of whole genome NGS for infectious disease diagnosis.* Expert Rev Mol Diagn, 2015. **15**(12): p. 1517-9.
147. Lecuit, M. and M. Eloit, *The diagnosis of infectious diseases by whole genome next generation sequencing: a new era is opening.* Front Cell Infect Microbiol, 2014. **4**: p. 25.
148. Roux, S., et al., *Benchmarking viromics: an in silico evaluation of metagenome-enabled estimates of viral community composition and diversity.* PeerJ, 2017. **5**: p. e3817.
149. Snyder, L.A., et al., *Next-generation sequencing - the promise and perils of charting the great microbial unknown.* Microb Ecol, 2009. **57**(1): p. 1-3.
150. Niewiadomska, A.M. and R.J. Gifford, *The extraordinary evolutionary history of the reticuloendotheliosis viruses.* PLoS Biol, 2013. **11**(8): p. e1001642.
151. Schieffelin, J.S., et al., *Clinical illness and outcomes in patients with Ebola in Sierra Leone.* N Engl J Med, 2014. **371**(22): p. 2092-100.
152. Cao, M.D., et al., *Streaming algorithms for identification of pathogens and antibiotic resistance potential from real-time MinION(TM) sequencing.* Gigascience, 2016. **5**(1): p. 32.

153. Greninger, A.L., et al., *Rapid metagenomic identification of viral pathogens in clinical samples by real-time nanopore sequencing analysis.* Genome Med, 2015. **7**: p. 99.

154. Loose, M., S. Malla, and M. Stout, *Real-time selective sequencing using nanopore technology.* Nat Methods, 2016. **13**(9): p. 751-4.

155. Stewart, R.D. and M. Watson, *poRe GUIs for parallel and real-time processing of MinION sequence data.* Bioinformatics, 2017. **33**(14): p. 2207-2208.

156. Brister, J.R., et al., *NCBI viral genomes resource.* Nucleic Acids Res, 2015. **43**(Database issue): p. D571-7.

157. Genomes Project, C., et al., *A global reference for human genetic variation.* Nature, 2015. **526**(7571): p. 68-74.

158. Langmead, B. and S.L. Salzberg, *Fast gapped-read alignment with Bowtie 2.* Nat Methods, 2012. **9**(4): p. 357-9.

159. Kucherov, G., L. Noe, and M. Roytberg, *Multiseed lossless filtration.* IEEE/ACM Trans Comput Biol Bioinform, 2005. **2**(1): p. 51-61.

160. Ilie, L., S. Ilie, and A.M. Bigvand, *SpEED: fast computation of sensitive spaced seeds.* Bioinformatics, 2011. **27**(17): p. 2433-4.

161. Lindner, M.S. and B.Y. Renard, *Metagenomic profiling of known and unknown microbes with microbeGPS.* PLoS One, 2015. **10**(2): p. e0117711.

162. Bostock, M., V. Ogievetsky, and J. Heer, *D(3): Data-Driven Documents.* IEEE Trans Vis Comput Graph, 2011. **17**(12): p. 2301-9.

163. Unit., B.a.B., *Belgian classifications for micro-organisms based on their biological risks - Definitions*. 2008: https://my.absa.org/Riskgroups.

164. Klenner, J., et al., *Comparing Viral Metagenomic Extraction Methods.* Curr Issues Mol Biol, 2017. **24**: p. 59-70.

165. Metcalf, J.A., et al., *Recent genome reduction of Wolbachia in Drosophila recens targets phage WO and narrows candidates for reproductive parasitism.* PeerJ, 2014. **2**: p. e529.

166. Wu, Y.-W. and Y. Ye, *A novel abundance-based algorithm for binning metagenomic sequences using l-tuples.* Journal of Computational Biology: A Journal of Computational Molecular Cell Biology, 2011. **18**: p. 523-534.

167. Sainsbury, A.W., et al., *Poxviral Disease in Red Squirrels Sciurus vulgaris in the UK: Spatial and Temporal Trends of an Emerging Threat.* EcoHealth, 2008. **5**: p. 305.

168. Scott, A.C., I.F. Keymer, and J. Labram, *Parapoxvirus infection of the red squirrel (Sciurus vulgaris).* Veterinary Record, 1981. **109**: p. 202-202.

169. McInnes, C.J., et al., *Genomic characterization of a novel poxvirus contributing to the decline of the red squirrel (Sciurus vulgaris) in the UK.* Journal of General Virology, 2006. **87**: p. 2115-2125.

170. Himsworth, C.G., et al., *Poxvirus Infection in an American Red Squirrel (Tamiasciurus hudsonicus) from Northwestern Canada.* Journal of Wildlife Diseases, 2009. **45**: p. 1143-1149.

171. Obon, E., et al., *Poxvirus identified in a red squirrel (Sciurus vulgaris) from Spain.* Veterinary Record, 2011. **168**: p. 86-86.

172. Kurth, A. and A. Nitsche, *Detection of Human-Pathogenic Poxviruses*, in *Diagnostic Virology Protocols*. 2010, Humana Press, Totowa, NJ. p. 257-278.

173. Li, Y., et al., *GC Content-Based Pan-Pox Universal PCR Assays for Poxvirus Detection.* Journal of Clinical Microbiology, 2010. **48**: p. 268-276.
174. Zerbino, D.R. and E. Birney, *Velvet: Algorithms for de novo short read assembly using de Bruijn graphs.* Genome Research, 2008. **18**: p. 821-829.
175. Darby, A.C., et al., *Novel Host-Related Virulence Factors Are Encoded by Squirrelpox Virus, the Main Causative Agent of Epidemic Disease in Red Squirrels in the UK.* PLOS ONE, 2014. **9**: p. e96439.
176. Talavera, G., et al., *Improvement of Phylogenies after Removing Divergent and Ambiguously Aligned Blocks from Protein Sequence Alignments.* Systematic Biology, 2007. **56**: p. 564-577.
177. Guindon, S., et al., *New Algorithms and Methods to Estimate Maximum-Likelihood Phylogenies: Assessing the Performance of PhyML 3.0.* Systematic Biology, 2010. **59**: p. 307-321.
178. Himsworth, C.G., et al., *Characterization of a Novel Poxvirus in a North American Red Squirrel (Tamiasciurus hudsonicus).* Journal of Wildlife Diseases, 2013. **49**: p. 173-179.
179. Weiss, S., et al., *A novel Coltivirus-related virus isolated from free-tailed bats from Cote d'Ivoire is able to infect human cells in vitro.* Virol J, 2017. **14**(1): p. 181.
180. Stagegaard, J., et al., *Seasonal recurrence of cowpox virus outbreaks in captive cheetahs (Acinonyx jubatus).* PLoS One, 2017. **12**(11): p. e0187089.
181. Berger, S.A., *GIDEON: a comprehensive Web-based resource for geographic medicine.* Int J Health Geogr, 2005. **4**(1): p. 10.
182. Rozov, R., et al., *Faucet: streaming de novo assembly graph construction.* Bioinformatics, 2018. **34**(1): p. 147-154.

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.

____

Simon Tausch, Berlin den 10. August 2018

Lebenslauf

Der Lebenslauf ist in der Online-Version aus Gründen des Datenschutzes nicht enthalten.

Publikationen

Andreas Andrusch, Piotr Wojciech Dabrowski, Jeanette Klenner, **Simon H. Tausch**, Claudia Kohl, Abdalla A. Osman, Bernhard Y. Renard, Andreas Nitsche. *PAIPline: Pathogen identification in metagenomic and clinical next generation sequencing samples. Bioinformatics* (2018)

Simon H. Tausch[1], Benjamin Strauch[1], Andreas Andrusch, Tobias P. Loka, Martin S. Lindner, Andreas Nitsche, Bernhard Y. Renard. *LiveKraken – Real-time metagenomic classification of Illumina data*. *Bioinformatics* (2018)

Tobias P. Loka, **Simon H. Tausch**, Piotr Wojciech Dabrowski, Aleksandar Radonic, Andreas Nitsche, Bernhard Y. Renard. *PriLive: Privacy-preserving real-time filtering for Next Generation Sequencing. Bioinformatics* (2018)

Livia Schrick**, Simon H. Tausch**, Piotr Wojciech Dabrowski, Clarissa R. Damaso, José Esparza, Andreas Nitsche. *An Early American Smallpox Vaccine Based on Horsepox*. *New England Journal of Medicine* (2017)

Gudrun Wibbelt[1], **Simon H. Tausch[1]**, Piotr Wojciech Dabrowski, Olivia Kershaw, Andreas Nitsche, Livia Schrick. *Berlin Squirrelpox Virus, a New Poxvirus in Red Squirrels, Berlin, Germany. Emerging Infectious Diseases* (2017)

Martin S. Lindner, Benjamin Strauch, Jakob M. Schulze, **Simon H. Tausch**, Piotr Wojciech Dabrowski, Andreas Nitsche, Bernhard Y. Renard. *HiLive: real-time mapping of illumina reads while sequencing*. *Bioinformatics* (2017)

Simon H. Tausch, Bernhard Y. Renard, Andreas Nitsche, Piotr Wojciech Dabrowski. *RAMBO-K: Rapid and Sensitive Removal of Background Sequences from Next Generation Sequencing Data. PLoS One* (2015)

[1] Diese Autoren haben gleichermaßen zur Publikation beigetragen

Tobias P. Loka, **Simon H. Tausch**, Bernhard Y. Renard. *Reliable variant calling during runtime of Illumina sequencing.* (Im Reviewprozess)

Simon H. Tausch, Tobias P. Loka, Jakob M. Schulze, Andreas Andrusch, Kristina Kirsten, Jeanette Klenner, Piotr Wojciech Dabrowski, Martin S. Lindner, Andreas Nitsche, Bernhard Y. Renard. *PathoLive – Real-time pathogen identification from metagenomic Illumina datasets.* (Manuskript unter finaler interner Durchsicht vor der Einreichung)

Simon H. Tausch, Andreas Andrusch, José Esparza, Andreas Nitsche[1], Clarissa R. Damaso[1]. *Genome analysis of the Mulford 1902 smallpox vaccine*. (Manuskript unter finaler interner Durchsicht vor der Einreichung)

Cesare E. M. Gruber, Emanuela Giombini, Marina Selleri, **Simon H. Tausch**, Andreas Andrusch, Alona Tyshaieva, Giusy Cardeti, Raniero Lorenzetti, Giuseppe Manna, Fabrizio Carletti, Andreas Nitsche, Maria R. Capobianchi, Gian Luca Autorino, Concetta Castilletti. *Whole genome characterization of OPV Abatino, a zoonotic virus representing a putative novel clade of Old World Orthopoxviruses.* (Manuskript unter finaler interner Durchsicht vor der Einreichung)

[1] Diese Autoren haben gleichermaßen zur Publikation beigetragen

Zusammenfassung

Infektionskrankheiten sind bis heute eine der häufigsten Todesursachen weltweit. Trotz großer Fortschritte im Bereich der klinischen Diagnostik ist es in vielen Fällen nicht möglich, eine eindeutige Ätiologie zu erstellen. Seit Aufkommen des Next Generation Sequencing (NGS) 2006 ist eine Vielzahl neuer Forschungsfelder entstanden, die auf dieser Technologie basieren. Insbesondere die Anwendung von NGS in der Metagenomic – der Forschung an genomischem Material, das direkt aus seiner Umwelt genommen wird – hat zur sprunghaften Entstehung neuer Anwendungsfelder geführt. Metagenomisches NGS hat sich als vielversprechendes Werkzeug im Feld der pathogenbezogenen Forschung erwiesen.

In dieser Arbeit präsentiere ich unterschiedliche Ansätze zur Detektion bekannter und Entdeckung unbekannter Pathogene anhand von NGS-Daten. Die gezeigten Beiträge lassen sich unterteilen in drei neu entwickelte Methoden sowie einen realen Anwendungsfall dieser Methodologie und darauf aufbauender Datenauswertung.

Zuerst präsentieren wir LiveKraken, ein Echtzeit-Klassifikationswerkzeug, das auf dem Kernalgorithmus von Kraken aufbaut. LiveKraken nutzt Ströme von Rohdaten von Illumina-Sequenzierern, um Reads taxonomisch zu klassifizieren. Dadurch sind wir in der Lage, mit dem Ende eines Sequenzierlaufs identische Ergebnisse wie Kraken zu generieren. Darüber hinaus lassen sich vergleichbare Ergebnisse in frühen Stadien eines Sequenzierlaufs produzieren, wodurch bis zu eine Woche Sequenzierzeit eingespart werden kann. Während die Anzahl der klassifizierten Reads mit der Zeit zunimmt, kommt es nur zu einer vernachlässigbaren Zahl falscher Klassifizierungen. Die Mehrheitsverhältnisse der identifizierten Taxa schwanken nur geringfügig.

Im zweiten Projekt haben wir PathoLive, ein Echtzeit-Diagnostikpipeline entworfen und entwickelt, die die Detektion von Pathogenen aus klinischen Proben schon vor Ende eines Sequenzierlaufs ermöglicht. Wir haben den Kernalgorithmus von HiLive, einem Echtzeit-Read-Mapper, angepasst und seine Genauigkeit für unseren Anwendungsfall optimiert. Darüber hinaus werden Sequenzen, die wahrscheinlich irrelevant sind, im Vorfeld markiert. Die Ergebnisse werden in einem interaktiven taxonomischen Baum visualisiert, der einen intuitiven Gesamtüberblick gibt. Des Weiteren werden detaillierte Metriken bezüglich der Relevanz jedes detektierten Pathogens ausgegeben. Ein Testlauf von PathoLive während der Sequenzierung einer mit Viren versetzten realen humanen Plasmaprobe zeigte, dass wir die Ergebnisse zu jedem gemessenen Zeitpunkt der Sequenzierung präziser einstufen als jedes getestete Tool nach Ende der Sequenzierung. Mit PathoLive verschieben wir den Fokus NGS-basierter Diagnostik weg von der reinen Readquantifizierung hin zu einer aussagekräftigeren Beurteilung der Ergebnisse.

Das dritte Projekt hat zum Ziel, neue Pathogene aus NGS-Daten zu detektieren. Wir haben mit RAMBO-K ein Werkzeug entwickelt, das schnelles und sensitives Entfernen von unerwünschten Wirtssequenzen aus NGS-Daten erlaubt. RAMBO-K ist schneller als alle anderen von uns getesteten Werkzeuge, während es durchgehend hohe Sensitivität und Spezifität auf unterschiedlichen Datensätzen erreicht. RAMBO-K unterscheidet schnell und zuverlässig zwischen Reads von verschiedenen Spezies. Es ist als unkomplizierte Standardanwendung in Arbeitsabläufen geeignet, die sich mit gemischten Datensätzen auseinandersetzen.

Im vierten Projekt haben wir durch RAMBO-K sowie mehrere darauf aufbauende Datenanalysen das Berlin Squirrelpox Virus entdeckt. Dies ist ein weit entferntes Pockenvirus, das ein neues Genus der Familie der Pockenviren begründet. In der Nähe von Berlin, Deutschland wurden mehrere junge rote Eichhörnchen (*Sciurus vulgaris*) mit feuchten, krustigen Läsionen gefunden. Histologie, Elektronenmikroskopie und Zellkulturisolate offenbarten eine Orthopoxvirus-ähnliche Infektion. Nachdem die gängigen Standardanalysen keine signifikanten Ergebnisse erbrachten, wurden pockenvirale Reads mit RAMBO-K zugeordnet, wodurch das Assemblieren des Genoms des neuen Virus ermöglicht wurde.

Mit diesen Projekten haben wir drei anwendungsnahe Werkzeuge entwickelt, die verschiedene Forschungslücken schließen. Zusammengenommen erweitern wir das Repertoire verfügbarer NGS-basierter pathogenbezogener Forschungswerkzeuge und erleichtern und beschleunigen eine Vielzahl von Forschungsprojekten.