RawHash

Enabling Fast and Accurate Real-Time Analysis of Raw Nanopore Signals for Large Genomes

Can Firtina

Nika Mansouri Ghiasi
Meryem Banu Cavlak
Joel Lindegger
Haiyu Mao
Gagandeep Singh
Onur Mutlu

Paper
Code

SAFARI

ETH Zürich
Nanopore Sequencing

Nanopore Sequencing: a widely used sequencing technology

- Can sequence large fragments of nucleic acid molecules (up to >2Mbp)
- Offers high throughput
- Cost-effective
- Enables real-time genome analysis
Real-Time Analysis with Nanopore Sequencing

**Raw Signals:** Ionic current measurements generated at a certain throughput

**Real-Time Analysis:** Analyzing all raw signals by matching the throughput

**Real-Time Decisions:** Stopping sequencing early based on real-time analysis
Benefits of Real-Time Genome Analysis

- **Reducing latency** by overlapping the sequencing and analysis steps

- **Reducing sequencing time and cost** by stopping sequencing early

**SEQUENCING & REAL-TIME ANALYSIS**

Sequencing is stopped early with a real-time decision.
Challenges in Real-Time Genome Analysis

- **Rapid analysis** to match the nanopore sequencer throughput
- **Timely decisions** to stop sequencing as early as possible
- **Accurate analysis** from noisy raw signal data
- **Power-efficient** computation for scalability and portability
Executive Summary

**Problem:** Real-time analysis of nanopore raw signals is **inaccurate** and **inefficient** for large genomes.

**Goal:** Enable **fast** and **accurate** real-time analysis of raw signals for **large genomes**.

**Key Contributions:**
1) The first **hash-based mechanism** that can quickly and accurately analyze raw nanopore signals for **large genomes**.
2) The novel **Sequence Until** technique can accurately and **dynamically stop the entire sequencing of all reads at once** if further sequencing is not necessary.

**Key Results:** Across 3 use cases and 5 genomes of varying sizes, RawHash provides
- **25.8×** and **3.4×** better average **throughput** compared to two state-of-the-art works
- **1.14×** – **2.13×** more accurate **mapping results** for **large genomes**
- Sequence Until **reduces the sequencing time and cost by 15×**
Existing Solutions

1. Deep neural networks (DNNs) for translating **signals** to **bases**

   - **Real-Time Analysis**
     - **Basecalling**
     - **Read Mapping**

   - Less noisy analysis from basecalled sequences
   - **Costly and power-hungry** computational requirements

2. Mapping **signals** to reference genomes **without** basecalling

   - **Real-Time Analysis**
     - **Mapping Raw Signals**

   - Raw signals contain richer information than bases
   - Efficient analysis with better scalability and portability
The Problem – Mapping Raw Signals

**Small Reference Genome**
- Fewer candidate regions in *small genomes*
- Accurate mapping
- High throughput

**Large Reference Genome (Human)**
- Substantially *larger number of regions* to check per read as the genome size increases

**Problem:** Probabilistic mechanisms on many regions → inaccurate mapping

**Problem:** Distance calculation on many regions → reduced throughput
The Problem – Mapping Raw Signals

- Small Reference Genome
  - Fewer candidate regions
  - Accurate mapping
- Large Reference Genome (Human)
  - Substantially larger number of regions to check per read as the genome size increases

Problem:
- Probabilistic mechanisms on many regions ⇒ inaccurate mapping
- Distance calculation on many regions ⇒ reduced throughput

Existing solutions are inaccurate or inefficient for large genomes
Outline

Background

RawHash

Evaluation

Conclusion
Enable **fast and accurate real-time analysis** of raw nanopore signals **for large genomes**.
The **first hash-based search mechanism** to quickly and accurately map raw nanopore signals to reference genomes

**Sequence Until** can accurately and **dynamically stop the entire sequencing run at once** if further sequencing is unnecessary.
The first hash-based search mechanism to quickly and accurately map raw nanopore signals to reference genomes.
RawHash – Key Idea

**Key Observation:** Identical nucleotides generate similar raw signals

**Challenge #1:** Generating the same hash value for similar enough signals

**Challenge #2:** Accurately finding similar regions as few as possible
RawHash Overview

Reference Genome

...GCTATTACCTTAATGTG...

1. Reference-to-Event Conversion

2. Quantization

3. Hashing

4. Matching Regions & Mapping

Raw Nanopore Signal

Signal-to-Event Conversion

1. Signal-to-Event Conversion

2. Quantization

3. Hashing

4. Mapping Positions

Indexing (Offline) & Mapping (Real-Time)

SAFARI
RawHash Overview

Reference Genome

...GCTATTACCTTAATGTG...

1. Reference-to-Event Conversion

2.21 -0.9 1.15

Raw Nanopore Signal

Signal-to-Event Conversion

2.22 -0.91 1.18
Events in Raw Nanopore Signals

- **Event**: A segment of the raw signal
  - Corresponds to a particular k-mer

- **Event detection** finds these segments to identify k-mers
  - Start and end positions are marked by abrupt signal changes
  - Statistical methods identify these abrupt changes
  - **Event value**: average of signals within an event

![Diagram showing event detection in raw nanopore signals](image-url)
Reference-to-Event Conversion

- **K-mer model**: Provides *expected* event values for each k-mer - Preconstructed based on nanopore sequencer characteristics

- Use the k-mer model to convert all k-mers of a reference genome to their *expected* event values

Reference Genome

- GCTATT
- CTATTA
- TATTAC
- ATTACC

K-mer Model (Lookup Table)

- GCTATT: 105.757390
- CTATTA: 81.740642
- TATTAC: 103.170091
- ATTACC: 101.082485

Normalized Event Values

- GCTATT: 2.21
- CTATTA: -0.09
- TATTAC: 1.11
- ATTACC: 1.15

SAFARI
Signal-to-Event Conversion

- **Event detection**: Identifies signal regions corresponding to specific k-mers
  - Uses statistical test (**segmentation**) to spot abrupt signal changes

• Consecutive events ➔ consecutive k-mers
Signal-to-Event Conversion

• **Event detection**: Identifies signal regions corresponding to specific k-mers
  - Uses statistical test *(segmentation)* to spot abrupt signal changes

---

Can we match events (k-mers) between reference genome and raw signals?

• Consecutive events ➔ consecutive k-mers
RawHash Overview

Reference Genome

Reference-to-Event Conversion

Signal-to-Event Conversion

Quantization

<table>
<thead>
<tr>
<th></th>
<th>Raw Nanopore Signal</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.22</td>
<td>-0.91</td>
</tr>
<tr>
<td>1.18</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Reference Genome</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.21</td>
<td>-0.9</td>
</tr>
<tr>
<td>1.15</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>28</td>
<td>6</td>
</tr>
<tr>
<td>18</td>
<td></td>
</tr>
</tbody>
</table>

SAFARI
Quantizing the Event Values

- **Observation:** Slight differences in raw signals from identical k-mers
  - **Challenge:** Direct event value matching is not feasible and accurate

- **Key Idea:** Quantize the event values
  - Enables assigning *identical quantized values* to *similar event values*

**Diagram:**

- CTATTA
- Normalized event values from the same k-mer:
  - -0.091
  - -0.084
  - -0.09
  - -0.086
- Quantized event values (in binary):
  - 11001
  - 11001
  - 11001
  - 11001
RawHash Overview

Reference Genome

...GCTATTACCTTAATGTG...

Reference-to-Event Conversion

Signal-to-Event Conversion

Quantization

Hashing

Store

Query

Hash Table

2.21 -0.9  1.15

2.22 -0.91  1.18

28 6 18

28 6 18

0x01

0x01
Hashing for Fast Similarity Search

- Each event usually represents a very small k-mer (6 to 9 characters)
  - **Challenge:** Short k-mers are likely to appear in many locations

- **Key Idea:** Create longer k-mers from many **consecutive events**
- **Key Benefit:** Directly match hash values to quickly identify similarities

---

**Consecutive k-mers**

- CTATTA
- TATTAC
- ATTACC

**Consecutive events**

- CTATTA
  - -0.09
  - Quantize
  - 11001
- TATTAC
  - 1.15
  - Quantize
  - 00110
- ATTACC
  - 1.11
  - Quantize
  - 00101

**Pack**

1100100110

**Hash value of consecutive events**

0x400D70A4

**Hash**

01001
RawHash Overview

Reference Genome

...GCTATTACCTTAATGTG...

Reference-to-Event Conversion

1. 2.21 -0.9 1.15

Signal-to-Event Conversion

2. 2.22 -0.91 1.18

Quantization

28 6 18

Quantization

28 6 18

Hashing

Hashing

0x01

Store

Query

0x01

Hash Table

Matching Regions

Chaining & Mapping

Mapping Positions

Indexing (Offline)

Mapping (Real-Time)
Real-Time Mapping using Hash-based Indexing

**Indexing (Offline)**

- **Reference Genome**
  - ...GCTATTACCTTAATGTG...
- **Reference-to-Event Conversion**
- **Quantization**
- **Hashing**
- **Store**
  - 0x01
- **Hash Table**

**Mapping (Real-time)**

- **Raw Nanopore Signal**
- **Signal-to-Event Conversion**
- **Quantization**
- **Hashing**
- **Query**
  - 0x01
- **Chaining & Mapping**

**Matching Positions**

- Read Until or Run Until
- Yes: Process the next chunk
- No: Stop mapping
- Continue Mapping?
The **first hash-based search mechanism** to quickly and accurately map raw nanopore signals to reference genomes

Sequence Until can accurately and *dynamically* stop the entire sequencing run at once if further sequencing is unnecessary
The **first hash-based search mechanism** to quickly and accurately map raw nanopore signals to reference genomes.

**Sequence Until** can accurately and **dynamically stop** the entire sequencing run at once if further sequencing is unnecessary.
Outline

Background

RawHash

Evaluation

Conclusion
Evaluation Methodology

• Compared to **UNCALLED** [Kovaka+, Nat. Biotech. 2021] and **Sigmap** [Zhang+, ISMB/ECCB 2021]

- **CPU baseline:** AMD EPYC 7742 @2.26GHz
- **32 threads** for each tool

• **Use cases** for real-time genome analysis:
  1. Read mapping
  2. Relative abundance estimation
    • **Benefits of Sequence Until**
  3. Contamination analysis
Evaluation Methodology

- **Evaluation metrics:**
  - **Throughput** (bases processed per second)
  - Potential reduction in **sequencing time and cost**
  - **Accuracy**
    - **Baseline:** Mapping basecalled reads using minimap2
    - Precision, recall, and F1 scores
    - Relative abundance estimation distance to ground truth

- **Datasets:**

<table>
<thead>
<tr>
<th>Organism</th>
<th>Reads (#)</th>
<th>Bases (#)</th>
<th>Genome Size</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Read Mapping</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D1 SARS-CoV-2</td>
<td>1,382,016</td>
<td>594M</td>
<td>29,903</td>
</tr>
<tr>
<td>D2 E. coli</td>
<td>353,317</td>
<td>2,365M</td>
<td>5M</td>
</tr>
<tr>
<td>D3 Yeast</td>
<td>49,989</td>
<td>380M</td>
<td>12M</td>
</tr>
<tr>
<td>D4 Green Algae</td>
<td>29,933</td>
<td>609M</td>
<td>111M</td>
</tr>
<tr>
<td>D5 Human HG001</td>
<td>269,507</td>
<td>1,584M</td>
<td>3,117M</td>
</tr>
<tr>
<td><strong>Relative Abundance Estimation</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D1-D5</td>
<td>2,084,762</td>
<td>5,531M</td>
<td>3,246M</td>
</tr>
<tr>
<td><strong>Contamination Analysis</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D1 and D5</td>
<td>1,651,523</td>
<td>2,178M</td>
<td>29,903</td>
</tr>
</tbody>
</table>
Throughput

**Real-time analysis requires** faster throughput than sequencer

- Throughput of a nanopore sequencer: \( \sim 450 \text{ bp/sec (data generation speed)} \)

\[
\begin{array}{cccc}
\text{Relative Abundance} & \text{Throughput (bp/sec)} \\
\text{D1 SARS-CoV-2} & 13892X & & \\
\text{D2 E. coli} & 10342X & 1467X & \\
\text{D3 Yeast} & 141X & 1467X & 3.7X \\
\text{D4 Green Algae} & 172X & 172X & 5.3X \\
\text{D5 Human} & 4.1X & 15.5X & 0.6X \\
\text{Contamination} & 14947X & 9141X & \\
\text{Relative Abundance} & 21X & 18.5X & 0.6X \\
\end{array}
\]

**25.8\times** and **3.4\times** better average throughput compared to **UNCALLED** and **Sigmap**, respectively

**Sigmap cannot** perform real-time analysis **for large genomes**
Sequencing Time

• Fewer bases to sequence ➔
  - Reduction in sequencing time and cost

RawHash reduces sequencing time and cost for large genomes up to 1.3× compared to UNCALLED.
# Mapping Accuracy

- Read mapping accuracy of each tool and each use case

<table>
<thead>
<tr>
<th>Dataset</th>
<th>UNCALLED</th>
<th>Sigmap</th>
<th>RawHash</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>Precision: 0.9547</td>
<td>0.9929</td>
<td>0.9868</td>
</tr>
<tr>
<td>SARS-CoV-2</td>
<td>Recall: 0.9910</td>
<td>0.5540</td>
<td>0.8735</td>
</tr>
<tr>
<td></td>
<td>$F_1$: 0.9725</td>
<td>0.7112</td>
<td>0.9267</td>
</tr>
<tr>
<td>D2</td>
<td>Precision: 0.9816</td>
<td>0.9842</td>
<td>0.9573</td>
</tr>
<tr>
<td>E. coli</td>
<td>Recall: 0.9647</td>
<td>0.9504</td>
<td>0.9009</td>
</tr>
<tr>
<td></td>
<td>$F_1$: 0.9731</td>
<td>0.9670</td>
<td>0.9282</td>
</tr>
<tr>
<td>D3</td>
<td>Precision: 0.9459</td>
<td>0.9856</td>
<td>0.9862</td>
</tr>
<tr>
<td>Yeast</td>
<td>Recall: 0.9366</td>
<td>0.9123</td>
<td>0.8412</td>
</tr>
<tr>
<td></td>
<td>$F_1$: 0.9412</td>
<td>0.9475</td>
<td>0.9079</td>
</tr>
<tr>
<td>D4</td>
<td>Precision: 0.8836</td>
<td>0.9741</td>
<td>0.9691</td>
</tr>
<tr>
<td>Green Algae</td>
<td>Recall: 0.7778</td>
<td>0.8987</td>
<td>0.7015</td>
</tr>
<tr>
<td></td>
<td>$F_1$: 0.8273</td>
<td>0.9349</td>
<td>0.8139</td>
</tr>
<tr>
<td>D5</td>
<td>Precision: 0.4867</td>
<td>0.4287</td>
<td>0.8959</td>
</tr>
<tr>
<td>Human HG001</td>
<td>Recall: 0.2379</td>
<td>0.2641</td>
<td>0.4054</td>
</tr>
<tr>
<td></td>
<td>$F_1$: 0.3196</td>
<td>0.3268</td>
<td>0.5582</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Relative Abundance Estimation</th>
<th>RawHash</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1-D5</td>
<td>Precision: 0.7683</td>
<td>0.7928</td>
</tr>
<tr>
<td></td>
<td>Recall: 0.1273</td>
<td>0.2739</td>
</tr>
<tr>
<td></td>
<td>$F_1$: 0.2184</td>
<td>0.4072</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Contamination Analysis</th>
<th>RawHash</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1, D5</td>
<td>Precision: 0.9378</td>
<td>0.7856</td>
</tr>
<tr>
<td></td>
<td>Recall: 0.9910</td>
<td>0.5540</td>
</tr>
<tr>
<td></td>
<td>$F_1$: 0.9637</td>
<td>0.6498</td>
</tr>
</tbody>
</table>

**For Large Genomes:** RawHash provides the best accuracy in all metrics, resulting in $1.14 \times - 2.13 \times$ improvement in $F_1$ score.
Relative Abundance Estimation Accuracy

• Estimating the ratio of genomes in a sample in real-time
  - **Distance**: Euclidean distance compared to the ground truth distance
  - The dataset includes a large reference genome

<table>
<thead>
<tr>
<th>Tool</th>
<th>Estimated Relative Abundance Ratios</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>SARS-CoV-2</strong></td>
<td><strong>E. coli</strong></td>
</tr>
<tr>
<td>Ground Truth</td>
<td>0.0929</td>
<td>0.4365</td>
</tr>
<tr>
<td>UNCALLED</td>
<td>0.0026</td>
<td>0.5884</td>
</tr>
<tr>
<td>Sigmap</td>
<td>0.0419</td>
<td>0.4191</td>
</tr>
<tr>
<td>RawHash</td>
<td>0.1249</td>
<td>0.4701</td>
</tr>
</tbody>
</table>

RawHash provides the **best relative abundance estimation** closest to the ground truth estimation.
Real Implementation of Sequence Until

- Running RawHash by using
  - **RawHash (100%)**: The entire sample **without Sequence Until**
  - **RawHash (7%)**: RawHash **with Sequence Until** where Sequence Until dynamically stops the entire sequencing after sequencing **7% of the sample**

<table>
<thead>
<tr>
<th>Tool</th>
<th>Estimated Relative Abundance Ratios in 50,000 Random Reads</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SARS-CoV-2</td>
</tr>
<tr>
<td>RawHash (100%)</td>
<td>0.0270</td>
</tr>
<tr>
<td>RawHash + Sequence Until (7%)</td>
<td><strong>0.0283</strong></td>
</tr>
</tbody>
</table>

Sequence Until enables sequencing **only 7% (~1/15)** of the entire sample **with high accuracy**
Simulating Sequence Until

- Real relative abundance results using the entire set of reads

<table>
<thead>
<tr>
<th>Tool</th>
<th>SARS-CoV-2</th>
<th>E. coli</th>
<th>Yeast</th>
<th>Green Algae</th>
<th>Human</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ground Truth</td>
<td>0.0929</td>
<td>0.4365</td>
<td>0.0698</td>
<td>0.1179</td>
<td>0.2828</td>
<td>N/A</td>
</tr>
<tr>
<td>UNCALLED</td>
<td>0.0026</td>
<td>0.5884</td>
<td>0.0615</td>
<td>0.1313</td>
<td>0.2161</td>
<td>0.1895</td>
</tr>
<tr>
<td>Sigmap</td>
<td>0.0419</td>
<td>0.4191</td>
<td>0.1038</td>
<td>0.0962</td>
<td>0.3390</td>
<td>0.0877</td>
</tr>
<tr>
<td>RawHash</td>
<td>0.1249</td>
<td>0.4701</td>
<td>0.0957</td>
<td>0.0629</td>
<td>0.2464</td>
<td>0.0847</td>
</tr>
</tbody>
</table>

- Simulating the benefits of Sequence Until by
  - Using a random portion (25%, 10%, 1%, ...) of the sample

<table>
<thead>
<tr>
<th>Tool</th>
<th>SARS-CoV-2</th>
<th>E. coli</th>
<th>Yeast</th>
<th>Green Algae</th>
<th>Human</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ground Truth</td>
<td>0.0929</td>
<td>0.4365</td>
<td>0.0698</td>
<td>0.1179</td>
<td>0.2828</td>
<td>N/A</td>
</tr>
<tr>
<td>UNCALLED (25%)</td>
<td>0.0026</td>
<td>0.5890</td>
<td>0.0613</td>
<td>0.1332</td>
<td>0.2139</td>
<td>0.1910</td>
</tr>
<tr>
<td>RawHash (25%)</td>
<td>0.0271</td>
<td>0.4853</td>
<td>0.0920</td>
<td>0.0786</td>
<td>0.3170</td>
<td>0.0995</td>
</tr>
<tr>
<td>UNCALLED (10%)</td>
<td>0.0026</td>
<td>0.5906</td>
<td>0.0611</td>
<td>0.1316</td>
<td>0.2141</td>
<td>0.1920</td>
</tr>
<tr>
<td>RawHash (10%)</td>
<td>0.0273</td>
<td>0.4869</td>
<td>0.0963</td>
<td>0.0772</td>
<td>0.3124</td>
<td>0.1004</td>
</tr>
<tr>
<td>UNCALLED (1%)</td>
<td>0.0026</td>
<td>0.5750</td>
<td>0.0616</td>
<td>0.1506</td>
<td>0.2103</td>
<td>0.1836</td>
</tr>
<tr>
<td>RawHash (1%)</td>
<td>0.0259</td>
<td>0.4783</td>
<td>0.0987</td>
<td>0.0882</td>
<td>0.3088</td>
<td>0.0928</td>
</tr>
<tr>
<td>UNCALLED (0.1%)</td>
<td>0.0040</td>
<td>0.4565</td>
<td>0.0380</td>
<td>0.1910</td>
<td>0.3105</td>
<td>0.1242</td>
</tr>
<tr>
<td>RawHash (0.1%)</td>
<td>0.0212</td>
<td>0.5045</td>
<td>0.1120</td>
<td>0.0810</td>
<td>0.2814</td>
<td>0.1136</td>
</tr>
<tr>
<td>UNCALLED (0.01%)</td>
<td>0.0000</td>
<td>0.5551</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.4449</td>
<td>0.2602</td>
</tr>
<tr>
<td>RawHash (0.01%)</td>
<td>0.0906</td>
<td>0.6122</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.2972</td>
<td>0.2232</td>
</tr>
</tbody>
</table>
Simulating Sequence Until

- Real relative abundance results using the entire set of reads

<table>
<thead>
<tr>
<th>Tool</th>
<th>SARS-CoV-2</th>
<th>E. coli</th>
<th>Yeast</th>
<th>Green Algae</th>
<th>Human</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ground Truth</td>
<td>0.0929</td>
<td>0.4365</td>
<td>0.0698</td>
<td>0.1179</td>
<td>0.2828</td>
<td>N/A</td>
</tr>
<tr>
<td>UNCALLED</td>
<td>0.0026</td>
<td>0.5884</td>
<td>0.0615</td>
<td>0.1313</td>
<td>0.2161</td>
<td>0.1895</td>
</tr>
<tr>
<td>Sigmap</td>
<td>0.0419</td>
<td>0.4191</td>
<td>0.1038</td>
<td>0.0962</td>
<td>0.3390</td>
<td>0.0877</td>
</tr>
</tbody>
</table>

**UNCALLED and RawHash** benefit from **Sequence Until** significantly **by up to 100× reductions in sequencing time and costs**
More in the Paper

• **More Results**
  - **Mapping time** per read
  - Overall *computational resources* required by each tool
    • Peak memory usage, CPU time and real time in the indexing and mapping steps
  - **Performance breakdown** of the steps in RawHash

• **Details of all mechanisms and configurations**
  - Details of the *quantization* and *hashing* mechanism
  - Details of the *parameter configurations*
  - Trade-offs between the *DNN-based approaches* and raw signal mapping approaches
RawHash

- Can Firtina, Nika Mansouri Ghiasi, Joel Lindegger, Gagandeep Singh, Meryem Banu Cavlak, Haiyu Mao, and Onur Mutlu,

"RawHash: Enabling Fast and Accurate Real-Time Analysis of Raw Nanopore Signals for Large Genomes"

Proceedings of the 31st Annual Conference on Intelligent Systems for Molecular Biology (ISMB) and the 22nd European Conference on Computational Biology (ECCB), Jul 2023

[arXiv preprint]
[Source Code]

---

**RawHash: enabling fast and accurate real-time analysis of raw nanopore signals for large genomes**

Can Firtina 1,*, Nika Mansouri Ghiasi 1, Joel Lindegger 1, Gagandeep Singh 1, Meryem Banu Cavlak 1, Haiyu Mao 1, Onur Mutlu 1,.*

1Department of Information Technology and Electrical Engineering, ETH Zurich, 8092 Zurich, Switzerland

*Corresponding author. Department of Information Technology and Electrical Engineering, ETH Zurich, Gloriastrasse 35, 8092 Zurich, Switzerland.

E-mail: firtinac@ethz.ch (C.F.), omutlu@ethz.ch (O.M.)
RawHash Source Code

- Supports **all major raw signal file formats and flow cell versions**
  - FAST5, POD5, S/BLOW5 file formats
- Easy-to-use scripts
  - To download all the datasets
  - To reproduce all of our results
- You can write your outlier function for Sequence Until
  - Easily integrate Sequence Until
- Upcoming Feature:
  - Integrating the MinKNOW API

[GitHub Repository](https://github.com/CMU-SAFARI/RawHash)
Sketching with Hash-based Indexing

**Indexing (Offline)**

- Reference Genome
  - ...GCTATTACCTTAATGTG...
- Reference-to-Event Conversion
- Quantization
- Hashing
- Sketch
- Hash Table
- Store
  - 0x01
- Matching Positions

**Mapping (Real-time)**

- Raw Nanopore Signal
- Signal-to-Event Conversion
- Quantization
- Hashing
- Sketch
- 0x01
- Query
- Chaining & Mapping
- Continue Mapping?
- Yes: Process the next chunk
- No: Stop mapping
- Read Until or Run Until

All k-mers, Minimizers, Strobemers, BLEND, ...

SAFARI
Outline

Background

RawHash

Evaluation

Conclusion
Conclusion

Key Contributions:
1) The **first hash-based mechanism** that can quickly and accurately analyze raw nanopore signals for **large genomes**

2) The novel **Sequence Until** technique can accurately and **dynamically stop the entire sequencing of all reads at once** if further sequencing is not necessary

Key Results: Across 3 use cases and 5 genomes of varying sizes, RawHash provides
- **25.8× and 3.4× better average throughput** compared to two state-of-the-art works
- **1.14× – 2.13× more accurate mapping results** for **large genomes**
- Sequence Until **reduces the sequencing time and cost by 15×**

Many opportunities for analyzing raw nanopore signals in real-time:
- Many hash-based **sketching techniques** can now be used for raw signals
- **Indexing is very cheap**: Many future use cases with the on-the-fly index construction
- We should rethink the algorithms to perform downstream analysis fully using raw signals
Enabling Fast and Accurate Real-Time Analysis of Raw Nanopore Signals for Large Genomes

Can Firtina

Nika Mansouri Ghiasi
Meryem Banu Cavlak
Joel Lindegger
Haiyu Mao
Gagandeep Singh
Onur Mutlu

Paper
Code
Fast and Accurate Real-Time Genome Analysis

- Can Firtina, Melina Soysal, Joel Lindegger, and Onur Mutlu,
  "RawHash2: Accurate and Fast Mapping of Raw Nanopore Signals using a Hash-based Seeding Mechanism"
  [arXiv version]
  [RawHash2 Source Code]

RawHash2: Accurate and Fast Mapping of Raw Nanopore Signals using a Hash-based Seeding Mechanism

Can Firtina  Melina Soysal  Joel Lindegger  Onur Mutlu

ETH Zürich
Optimizations in RawHash2 (1)

- **More sensitive** chaining implementation with penalty scores
  - **Benefits:** Enables filtering dissimilar regions quickly
  - **Downside:** Additional computations with costly log operations

- Weighted mapping decisions
  - **Benefit #1:** ‘Learned’ mapping decisions based on the weights chosen from empirical analysis
  - **Benefit #2:** Faster and more accurate decisions

- Frequency filters
  - Filters the seeds that frequently appear before chaining
  - **Benefits:** Reduced workload on chaining without significantly affecting accuracy
  - **Downside:** Less sensitive mapping due to removed seeds
Optimizations in RawHash2 (2)

- New sketching techniques such as **minimizers** and **BLEND**
  - Enables integration of widely studied sketching techniques
  - **Benefits:** Can take advantage of these techniques (e.g., reduced storage requirements)

- Support for the recent improvements in the technology
  - Support for **new data formats:** POD5 and S/BLOW5
  - Support for **newer nanopore chemistry** versions: R10.4
Results – Throughput

- **Real-time analysis requires** faster throughput than sequencer
  - Throughput of a nanopore sequencer: ~**450 bp/sec** (data generation speed)

2.3× better average throughput RawHash
### Results – Accuracy

<table>
<thead>
<tr>
<th>Dataset</th>
<th>UNCALLED</th>
<th>Siganp</th>
<th>RawHash</th>
<th>RawHash2</th>
<th>RawHash2-Minimizer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Precision</td>
<td>Recall</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>D1</strong></td>
<td>0.9947</td>
<td>0.9929</td>
<td>0.9868</td>
<td>0.9857</td>
<td>0.9602</td>
</tr>
<tr>
<td>SARS-CoV-2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>D2</strong></td>
<td>0.9816</td>
<td>0.9842</td>
<td>0.9573</td>
<td><strong>0.9864</strong></td>
<td>0.9761</td>
</tr>
<tr>
<td>E. coli</td>
<td><strong>0.9647</strong></td>
<td>0.9504</td>
<td>0.9009</td>
<td>0.8934</td>
<td>0.7805</td>
</tr>
<tr>
<td><strong>D3</strong></td>
<td>0.9459</td>
<td>0.9856</td>
<td>0.9862</td>
<td>0.9567</td>
<td>0.9547</td>
</tr>
<tr>
<td>Yeast</td>
<td><strong>0.9366</strong></td>
<td>0.9123</td>
<td>0.8412</td>
<td>0.8942</td>
<td>0.7792</td>
</tr>
<tr>
<td><strong>D4</strong></td>
<td>0.9412</td>
<td><strong>0.9475</strong></td>
<td>0.9079</td>
<td>0.9244</td>
<td>0.8581</td>
</tr>
<tr>
<td>Green Algae</td>
<td>0.8836</td>
<td><strong>0.9741</strong></td>
<td>0.9691</td>
<td>0.9264</td>
<td>0.9198</td>
</tr>
<tr>
<td><strong>D5</strong></td>
<td>0.8778</td>
<td><strong>0.8987</strong></td>
<td>0.7015</td>
<td>0.8659</td>
<td>0.6711</td>
</tr>
<tr>
<td>Human HG001</td>
<td>0.8273</td>
<td><strong>0.9349</strong></td>
<td>0.8139</td>
<td>0.8951</td>
<td>0.7760</td>
</tr>
<tr>
<td></td>
<td>Contamination</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D1 and D5</td>
<td>0.9378</td>
<td>0.7856</td>
<td>0.8733</td>
<td><strong>0.9393</strong></td>
<td>0.9330</td>
</tr>
</tbody>
</table>

RawHash2 is more accurate than RawHash in all cases.
## Results – Average Sequencing Length

<table>
<thead>
<tr>
<th>Tool</th>
<th>SARS-CoV-2</th>
<th>E. coli</th>
<th>Yeast</th>
<th>Green Algae</th>
<th>Human</th>
<th>Contamination</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Average sequenced base length per read</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNCALLED</td>
<td>184.51</td>
<td>580.52</td>
<td>1,233.20</td>
<td>5,300.15</td>
<td>6,060.23</td>
<td>1,582.63</td>
</tr>
<tr>
<td>RawHash</td>
<td>513.95</td>
<td>1,376.14</td>
<td>2,565.09</td>
<td>4,760.59</td>
<td>4,773.58</td>
<td>742.56</td>
</tr>
<tr>
<td>RawHash2</td>
<td>488.46</td>
<td>1,234.39</td>
<td>1,715.31</td>
<td>2,077.39</td>
<td>3,441.43</td>
<td>681.94</td>
</tr>
<tr>
<td>RawHash2-Minimizer</td>
<td>566.42</td>
<td>1,763.76</td>
<td>2,339.41</td>
<td>2,891.55</td>
<td>4,090.68</td>
<td>787.82</td>
</tr>
<tr>
<td><strong>Average sequenced number of chunks per read</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sigmoid</td>
<td>1.01</td>
<td>2.11</td>
<td>4.14</td>
<td>5.76</td>
<td>10.40</td>
<td>2.06</td>
</tr>
<tr>
<td>RawHash</td>
<td>1.24</td>
<td>3.20</td>
<td>5.83</td>
<td>10.72</td>
<td>10.70</td>
<td>2.41</td>
</tr>
<tr>
<td>RawHash2</td>
<td>1.18</td>
<td>2.93</td>
<td>4.02</td>
<td>4.84</td>
<td>7.78</td>
<td>1.68</td>
</tr>
<tr>
<td>RawHash2-Minimizer</td>
<td>1.39</td>
<td>4.16</td>
<td>5.45</td>
<td>6.66</td>
<td>9.17</td>
<td>1.89</td>
</tr>
</tbody>
</table>

RawHash2 uses fewer bases to sequence than RawHash in all cases.

RawHash2 uses the smallest number of bases to sequence for larger genomes.
Fast and Accurate Real-Time Genome Analysis

- Can Firtina, Melina Soysal, Joel Lindegger, and Onur Mutlu,
  "RawHash2: Accurate and Fast Mapping of Raw Nanopore Signals using a Hash-based Seeding Mechanism"
  [arXiv version]
  [RawHash2 Source Code]

RawHash2: Accurate and Fast Mapping of Raw Nanopore Signals using a Hash-based Seeding Mechanism

Can Firtina    Melina Soysal    Joel Lindegger    Onur Mutlu

*ETH Zürich*
Backup Slides
Practical Similarity Identification

Reference

Read

K-mers

Determining potential matching regions (seeds) in the reference genome

Prune some seeds in the reference genome

Determine the exact differences between the read and the reference genome

Seeding

Seed Filtering (e.g., Chaining)

Alignment

SAFARI
Existing Solutions – Real-time Basecalling

Deep neural networks (DNNs) for translating signals to bases

DNNs provide less noisy analysis from basecalled sequences

Costly and power-hungry computational requirements
The Problem

The existing solutions are ineffective for large genomes

Costly and energy-hungry computations to basecall each read:
Portable sequencing becomes challenging with resource-constrained devices

Larger number of reference regions cannot be handled accurately or quickly, rendering existing solutions ineffective for large genomes
Applications of Read Until

**Depletion:** Reads mapping to a particular reference genome is ejected

- Removing contaminated reads from a sample
- Relative abundance estimation
- Controlling low/high-abundance genomes in a sample
- Controlling the sequencing of depth of a genome

**Enrichment:** Reads **not** mapping to a particular reference genome is ejected

- Purifying the sample to ensure it contains only the selected genomes
- Removing the host genome (e.g., human) in contamination analysis
Applications of Run Until and Sequence Until

**Run Until:** Stopping the sequencing without informative decision from analysis

- Stopping when reads reach to a particular depth of coverage

- Stopping when the abundance of all genomes reach a particular threshold

**Sequence Until:** Stopping the sequencing based on information decision

- Stopping when relative abundance estimations do not change substantially (for high-abundance genomes)

- Stopping when finding that the sample is contaminated with a particular set of genomes

- ...

SAFARI
Details: Quantizing the Event Values

- **Observation:** Identical k-mers generate similar raw signals
- **Challenge:** Their corresponding event values can be slightly different

- **Key Idea:** Quantize the event values
  - To enable assigning the **same quantized value** to the **similar event values**

![Diagram](attachment://image.png)

-0.091 in binary:

```
1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1...
```

- Most significant $Q = 9$ bits:
  - Pruning $p = 4$ bits:
    - Matching Quantized Event Values: 1 0 0 1 1

-0.084 in binary:

```
1 0 1 1 1 1 0 1 1 0 1 1 0 1 0 1 0...
```

- Most significant $Q = 9$ bits:
  - Pruning $p = 4$ bits:
    - Matching Quantized Event Values: 1 0 0 1 1

**SAFARI**
## Average Sequenced Bases and Chunks

<table>
<thead>
<tr>
<th>Tool</th>
<th>SARS-CoV-2</th>
<th>E. coli</th>
<th>Yeast</th>
<th>Green Algae</th>
<th>Human</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Average sequenced base length per read</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNCALLED</td>
<td>184.51</td>
<td>580.52</td>
<td>1,233.20</td>
<td>5,300.15</td>
<td>6,060.23</td>
</tr>
<tr>
<td>RawHash</td>
<td>513.95</td>
<td>1,376.14</td>
<td>2,565.09</td>
<td>4,760.59</td>
<td>4,773.58</td>
</tr>
<tr>
<td></td>
<td>Average sequenced number of chunks per read</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sigmap</td>
<td>1.01</td>
<td>2.11</td>
<td>4.14</td>
<td>5.76</td>
<td>10.40</td>
</tr>
<tr>
<td>RawHash</td>
<td>1.24</td>
<td>3.20</td>
<td>5.83</td>
<td>10.72</td>
<td>10.70</td>
</tr>
</tbody>
</table>

RawHash **reduces sequencing time and cost for large genomes** up to $1.3 \times$ compared to UNCALLED.

Although Sigmap processes less number of chunks than RawHash, it fails to provide real-time analysis capabilities for large genomes.
# Breakdown Analysis of the RawHash Steps

The entire runtime is **bottlenecked by the chaining step**

<table>
<thead>
<tr>
<th>Tool</th>
<th>Fraction of entire runtime (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SARS-CoV-2</td>
</tr>
<tr>
<td>File I/O</td>
<td>0.00</td>
</tr>
<tr>
<td>Signal-to-Event</td>
<td>21.75</td>
</tr>
<tr>
<td>Sketching</td>
<td>0.74</td>
</tr>
<tr>
<td>Seeding</td>
<td>3.86</td>
</tr>
<tr>
<td>Chaining</td>
<td>73.50</td>
</tr>
<tr>
<td>Seeding + Chaining</td>
<td>77.36</td>
</tr>
</tbody>
</table>
# Required Computation Resources in Indexing

The indexing step of RawHash is **orders of magnitude faster** than the indexing steps of UNCALLED and Sigmap, especially for large genomes.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Contamination</th>
<th>SARS-CoV-2</th>
<th>E. coli</th>
<th>Yeast</th>
<th>Green Algae</th>
<th>Human</th>
<th>Relative Abundance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CPU Time (sec)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNCALLED</td>
<td>8.72</td>
<td>9.00</td>
<td>11.08</td>
<td>18.62</td>
<td>285.88</td>
<td>4,148.10</td>
<td>4,382.38</td>
</tr>
<tr>
<td>Sigmap</td>
<td>0.02</td>
<td>0.04</td>
<td>8.66</td>
<td>24.57</td>
<td>449.29</td>
<td>36,765.24</td>
<td>40,926.76</td>
</tr>
<tr>
<td>RawHash</td>
<td>0.18</td>
<td>0.13</td>
<td>2.62</td>
<td>4.48</td>
<td>34.18</td>
<td>1,184.42</td>
<td>788.88</td>
</tr>
<tr>
<td></td>
<td>Real time (sec)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNCALLED</td>
<td>1.01</td>
<td>1.04</td>
<td>2.67</td>
<td>7.79</td>
<td>280.27</td>
<td>4,190.00</td>
<td>4,471.82</td>
</tr>
<tr>
<td>Sigmap</td>
<td>0.13</td>
<td>0.25</td>
<td>9.31</td>
<td>25.86</td>
<td>458.46</td>
<td>37,136.61</td>
<td>41,340.16</td>
</tr>
<tr>
<td>RawHash</td>
<td>0.14</td>
<td>0.10</td>
<td>1.70</td>
<td>2.06</td>
<td>15.82</td>
<td>278.69</td>
<td>154.68</td>
</tr>
<tr>
<td></td>
<td>Peak memory (GB)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNCALLED</td>
<td>0.07</td>
<td>0.07</td>
<td>0.13</td>
<td>0.31</td>
<td>11.96</td>
<td>48.44</td>
<td>47.81</td>
</tr>
<tr>
<td>Sigmap</td>
<td>0.01</td>
<td>0.01</td>
<td>0.40</td>
<td>1.04</td>
<td>8.63</td>
<td>227.77</td>
<td>238.32</td>
</tr>
<tr>
<td>RawHash</td>
<td>0.01</td>
<td>0.01</td>
<td>0.35</td>
<td>0.76</td>
<td>5.33</td>
<td>83.09</td>
<td>152.80</td>
</tr>
</tbody>
</table>

RawHash requires **larger memory space** than UNCALLED.
Required Computation Resources in Mapping

<table>
<thead>
<tr>
<th>Tool</th>
<th>Contamination</th>
<th>SARS-CoV-2</th>
<th>E. coli</th>
<th>Yeast</th>
<th>Green Algae</th>
<th>Human</th>
<th>Relative Abundance</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Time (sec)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNCALLED</td>
<td>265,902.26</td>
<td>36,667.26</td>
<td>35,821.14</td>
<td>8,933.52</td>
<td>16,769.09</td>
<td>262,597.83</td>
<td>586,561.54</td>
</tr>
<tr>
<td>Sigmap</td>
<td>4,573.18</td>
<td>1,997.84</td>
<td>23,894.70</td>
<td>11,168.96</td>
<td>31,544.55</td>
<td>4,837,058.90</td>
<td>11,027,652.91</td>
</tr>
<tr>
<td>RawHash</td>
<td>3,721.62</td>
<td>1,832.56</td>
<td>8,212.17</td>
<td>4,906.70</td>
<td>25,215.23</td>
<td>2,022,521.48</td>
<td>4,738,961.77</td>
</tr>
<tr>
<td>Real time (sec)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNCALLED</td>
<td>20,628.57</td>
<td>2,794.76</td>
<td>1,544.68</td>
<td>285.42</td>
<td>2,138.91</td>
<td>8,794.30</td>
<td>19,409.71</td>
</tr>
<tr>
<td>Sigmap</td>
<td>6,725.26</td>
<td>3,222.32</td>
<td>2,067.02</td>
<td>1,167.08</td>
<td>2,398.83</td>
<td>158,904.69</td>
<td>361,443.88</td>
</tr>
<tr>
<td>RawHash</td>
<td>3,917.49</td>
<td>1,949.53</td>
<td>957.13</td>
<td>215.68</td>
<td>1,804.96</td>
<td>65,411.43</td>
<td>152,280.26</td>
</tr>
<tr>
<td>Peak memory (GB)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNCALLED</td>
<td>0.65</td>
<td>0.19</td>
<td>0.52</td>
<td>0.37</td>
<td>0.81</td>
<td>9.46</td>
<td>9.10</td>
</tr>
<tr>
<td>Sigmap</td>
<td>111.69</td>
<td>28.26</td>
<td>111.11</td>
<td>14.65</td>
<td>29.18</td>
<td>311.89</td>
<td>489.89</td>
</tr>
<tr>
<td>RawHash</td>
<td>4.13</td>
<td>4.20</td>
<td>4.16</td>
<td>4.37</td>
<td>11.75</td>
<td>52.21</td>
<td>55.31</td>
</tr>
</tbody>
</table>

The mapping step of RawHash is **significantly faster than Sigmap** for all genomes, and **faster than UNCALLED** for small genomes

RawHash requires **larger memory space** than UNCALLED
The mapping step of RawHash is **significantly faster than Sigmap** for all genomes, and **faster than UNCALLED for small genomes**
# Parameter Configurations

<table>
<thead>
<tr>
<th>Tool</th>
<th>Contamination</th>
<th>SARS-CoV-2</th>
<th>E. coli</th>
<th>Yeast</th>
<th>Green Algae</th>
<th>Human</th>
<th>Relative Abundance</th>
</tr>
</thead>
<tbody>
<tr>
<td>RawHash</td>
<td>-x viral -t 32</td>
<td>-x viral -t 32</td>
<td>-x sensitive -t 32</td>
<td>-x sensitive -t 32</td>
<td>-x fast -t 32</td>
<td>-x fast -t 32</td>
<td>-x fast -t 32</td>
</tr>
<tr>
<td>UNCALLED</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>map -t 32</td>
</tr>
<tr>
<td>Sigmoid</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-m -t 32</td>
</tr>
<tr>
<td>Minimap2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-x map-ont -t 32</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Preset (-x)</th>
<th>Corresponding parameters</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>viral</td>
<td>-e 5 -q 9 -l 3</td>
<td>Viral genomes</td>
</tr>
<tr>
<td>sensitive</td>
<td>-e 6 -q 9 -l 3</td>
<td>Small genomes (i.e., &lt; 50M bases)</td>
</tr>
<tr>
<td>fast</td>
<td>-e 7 -q 9 -l 3</td>
<td>Large genomes (i.e., &gt; 50M bases)</td>
</tr>
</tbody>
</table>
## Versions

<table>
<thead>
<tr>
<th>Tool</th>
<th>Version</th>
<th>Link to the Source Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>RawHash</td>
<td>0.9</td>
<td><a href="https://github.com/CMU-SAFARI/RawHash/tree/8042b1728e352a28fccc79c2ef80c8b631fe7bac">https://github.com/CMU-SAFARI/RawHash/tree/8042b1728e352a28fccc79c2ef80c8b631fe7bac</a></td>
</tr>
<tr>
<td>UNCALLED</td>
<td>2.2</td>
<td><a href="https://github.com/skovaka/UNCALLED/tree/74a5d4e5b5d02fb31d6e88926e8a0896dc3475cb">https://github.com/skovaka/UNCALLED/tree/74a5d4e5b5d02fb31d6e88926e8a0896dc3475cb</a></td>
</tr>
<tr>
<td>Sigmap</td>
<td>0.1</td>
<td><a href="https://github.com/haowenz/sigmap/tree/c9a40483264c9514587a36555b5af48d3f054f6f">https://github.com/haowenz/sigmap/tree/c9a40483264c9514587a36555b5af48d3f054f6f</a></td>
</tr>
</tbody>
</table>
RawHash
Enabling Fast and Accurate Real-Time Analysis of Raw Nanopore Signals for Large Genomes

Can Firtina
Nika Mansouri Ghiasi
Meryem Banu Cavlak
Joel Lindegger
Haiyu Mao
Gagandeep Singh
Onur Mutlu

Paper
Code