Efficient Algorithm and Embedded Multicore Implementation for ECG Analysis in Multi-lead Electrocardiogram Records

Ben Abdallah Abderazek, Yasuyoshi Haga, Kenichi Kuroda
Graduate School of Computer Science and Engineering,
The University of Aizu, Aizu-Wakamatsu
Japan
E-mail: benab@u-aizu.ac.jp
Contents

• Background

• Contributions
  – Period-Peak Detection (PPD) Algorithm
  – Multicore Implementation for ECG Analysis in Multi-lead Electrocardiogram Records

• Results

• Conclusion
Contents

• Background

• Contributions
  – Period-Peak Detection (PPD) Algorithm
  – Multicore Implementation for ECG Analysis in Multi-lead Electrocardiogram Records

• Results

• Conclusion
Background

• Electrocardiography is a well known method for heart diagnosis
  – Used as one of major diagnosis for conventional health monitoring

• Electrocardiography main processing challenges arise from:
  – High computational demand for processing huge amount of data under:
    • Strict time constraints
    • Relatively high sampling frequency
    • Life critical conditions
Background

• Most ECG systems use Pan-Tompkins approach based on QRS complex
  – Usage of R-peak as a reference point
  – Accurate detection of R-peak is a must
    • R-peak detection might be inaccurate

• Traditional techniques may fail in detecting serious heart problems
Contents

• Background

• Contributions
  – Period-Peak Detection (PPD) Algorithm
  – Multicore Implementation for ECG Analysis in Multi-lead Electrocardiogram Records

• Results

• Conclusion
Contributions

I. Efficient Period-Peak detection Algorithm for ECG analysis in Multi-lead electrocardiogram records.

II. Multicore implementation for ECG analysis/processing in Multi-lead Electrocardiogram records.
Contents

• Background
• Contributions
  – Period-Peak Detection (PPD) Algorithm
  – Multicore Implementation for ECG Analysis in Multi-lead Electrocardiogram Records
• Results
• Conclusion
PPD Algorithm

- ACF-based approach

**Period detection**

- Reading data
- Derivation
- Autocorrelation
- Find interval

**Peaks detection**

- Extraction of max point
- Discrimination
- Store results
PPD Algorithm

- Based on autocorrelation approach

Period detection

Peaks detection

1. Reading data
2. Derivation
3. Autocorrelation
4. Find interval

Extraction of max point
Discrimination
Store results
PPD Algorithm

- Based on autocorrelation approach

Period detection

Reading data
Derivation
Autocorrelation
Find interval

Extraction of max point
Discrimination
Store results

Peaks detection
PPD Algorithm - Derivation

- Increases signal peaks $\rightarrow$ clear peaks
- Performance overhead is negligible
- Not indispensable

$$\frac{\partial y}{\partial t}(t) \approx \frac{y[n+1] - y[n]}{(n+1) - n}$$

$$= y[n+1] - y[n]$$

\[ y[n] \text{: current sampling data (filtered ECG signals) } \]

\[ t, n \text{: current time (step) } \]
Filtered data

Derivative of the ECG signal

Signal peaks P, Q, R, S, T, and U

Derivative amplifying R peaks

ICPPW-2010, San Diego, Sept. 13, 2010
PPD Algorithm - Autocorrelation

- Measures the degree of association between values in a series separated by some lags
- Periodicity analysis of signals

\[ R_y[L] = \sum_{n=0}^{N} y[n] \times y[n - L] \]

- \( R_y \) : autocorrelation value
- \( y[n] \) : filtered ECG signals
- \( N \) : the number of times needed for the calculations to get the period
- \( L \) : the number of lags of the autocorrelation (\( 0 \leq L < N \))
PPD Algorithm - Autocorrelation

Derivative of the ECG signal

AC of the derivative characterized by significant periodic peaks having the same value as the period of the ECG signal.
ACF calculation example (1/6)

• Assumption
  - The ECG signals are 9 samples.

<table>
<thead>
<tr>
<th>Time $t$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal $y[n]$</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

$$R_y[L] = \sum_{n=0}^{8} y[n] \times y[n-L]$$

When $n-L < 0$, $y$ is 0
ACF calculation example (2/6)

\[ R_y[0] = \sum_{n=0}^{8} y[n] \times y[n-0] = \sum_{n=0}^{8} y[n] \times y[n] \]

Signal \( y[n] \)

\[
\begin{array}{cccccccc}
0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\
\times & \times & \times & \times & \times & \times & \times & \times & \times \\
\end{array}
\]

\[
\begin{array}{cccccccc}
0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\
\end{array}
\]

Calculation

\[ R_y[0] = 15 \]
ACF calculation example (3/6)

$$R_y[1] = \sum_{n=0}^{8} y[n] \times y[n-1]$$

Zero (n-L < 0)
Calculation
No calculation (n > 8)

$$R_y[1] = 6$$
ACF calculation example (4/6)

\[ R_y[2] = \sum_{n=0}^{8} y[n] \times y[n-2] \]

<table>
<thead>
<tr>
<th>y[n]</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>y[n]</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>×</td>
<td>×</td>
</tr>
</tbody>
</table>

\[ R_y[2] = 4 \]

Zero (n-L < 0)
Calculation
No calculation (n > 8)
$$R_y[3] = \sum_{n=0}^{8} y[n] \times y[n-3]$$

**ACF calculation example (5/6)**

**Calculation**

$$R_y[3] = 10$$

**Zero**

$$n-L < 0$$

**No calculation**

$$n > 8$$
ACF calculation example (6/6)

• Results

\[ y[n] \]

\[
\begin{array}{cccccccc}
0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\
\end{array}
\]

\[
\begin{array}{cccccccc}
L & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
R_y & 15 & 6 & 4 & 10 & 4 & 2 & 5 & 2 & 0 \\
\end{array}
\]
ACF calculation example (6/6)

- Results

\[
\begin{array}{cccccccccc}
y[n] & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\
\hline
L & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
R_y & 15 & 6 & 4 & 10 & 4 & 2 & 5 & 2 & 0 \\
\end{array}
\]

Period

Period
ACF calculation example (6/6)

• Results

\[
\begin{array}{cccccccc}
0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \\
L & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
R_y & 15 & 6 & 4 & 10 & 4 & 2 & 5 & 2 & 0 \\
\end{array}
\]

Every 3 samples are periodic
PPD Algorithm – Period detection

Find maximum value

Reduce negative value

Peak detection from ACF result

Find base points

Sort base points

Calculate interval

Renew next start index
Find maximum value

Reduce negative value

Peak detection from ACF result

Find base points

Sort base points

Calculate interval

Renew next start index

used to determine a threshold.
Period detection - Find maximum value

In this program, ACF_STEP is 256

max_value = _input_values[0]
max_value_index = 0
i = 1

i < ACF_STEP

Yes

_input_values[i] > max_value

Yes

max_value = _input_values[i]
max_value_index = i

No

i++

No

end

start
Period detection - Reduce negative value

- Find maximum value
- Reduce negative value
- Peak detection from ACF result
- Find base points
- Sort base points
- Calculate interval
- Renew next start index
Contents

• Background

• Contributions
  – Period-Peak Detection (PPD) Algorithm
  – Multicore Implementation for ECG Analysis in Multi-lead Electrocardiogram Records

• Results

• Conclusion
System Architecture

12 leads
(extensible 15 leads and any more)

(a) 信号読み込み
(b) フィルタリング
(c) 分析
(d) 表示

図 2. システムアーキテクチャ
Contents

• Background

• Contributions
  – Period-Peak Detection (PPD) Algorithm
  – Parallel processing of ECG on multicore SoC

• Results

• Conclusion
### Complexity

<table>
<thead>
<tr>
<th>System Model</th>
<th>Logic utilization</th>
<th>Total block memory bits</th>
<th>Speed (MHz)</th>
<th>Power (mW)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Combinational ALUTs</td>
<td>Memory ALUTs</td>
<td>Dedicated logic registers</td>
<td>Total</td>
</tr>
<tr>
<td>1-lead</td>
<td>9,769 (9%)</td>
<td>16 (&lt;1%)</td>
<td>11,669 (10%)</td>
<td>14%</td>
</tr>
<tr>
<td>2-lead</td>
<td>17,169 (15%)</td>
<td>32 (&lt;1%)</td>
<td>21,297 (19%)</td>
<td>26%</td>
</tr>
<tr>
<td>3-lead</td>
<td>24,592 (22%)</td>
<td>48 (&lt;1%)</td>
<td>30,947 (27%)</td>
<td>38%</td>
</tr>
<tr>
<td>4-lead</td>
<td>32,047 (28%)</td>
<td>64 (&lt;1%)</td>
<td>40,566 (36%)</td>
<td>50%</td>
</tr>
</tbody>
</table>

(Target: Stratix III DSP board EP3SL150F1152C2)
System performance

- Execution time < Sampling time
  - Sufficient processing speed

### Execution Time (seconds)

<table>
<thead>
<tr>
<th>Database No.</th>
<th>Phase of Period-Peaks Detection algorithm</th>
<th>Reading data</th>
<th>Derivation</th>
<th>Autocorrelation</th>
<th>Find interval</th>
<th>Extraction</th>
<th>Discrimination</th>
<th>Store result</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>16265</td>
<td></td>
<td>3.608</td>
<td>0.008</td>
<td>2.087</td>
<td>0.028</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>5.812</td>
</tr>
<tr>
<td>16273</td>
<td></td>
<td>2.300</td>
<td>0.009</td>
<td>2.387</td>
<td>0.032</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>4.919</td>
</tr>
<tr>
<td>16420</td>
<td></td>
<td>3.363</td>
<td>0.008</td>
<td>2.088</td>
<td>0.028</td>
<td>0.011</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>5.903</td>
</tr>
<tr>
<td>16773</td>
<td></td>
<td>2.671</td>
<td>0.011</td>
<td>2.985</td>
<td>0.040</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>5.700</td>
</tr>
<tr>
<td>16786</td>
<td></td>
<td>1.586</td>
<td>0.011</td>
<td>2.984</td>
<td>0.040</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>4.659</td>
</tr>
<tr>
<td>17052</td>
<td></td>
<td>3.225</td>
<td>0.010</td>
<td>2.688</td>
<td>0.037</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>6.035</td>
</tr>
<tr>
<td>18177</td>
<td></td>
<td>2.890</td>
<td>0.009</td>
<td>2.386</td>
<td>0.032</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>5.362</td>
</tr>
<tr>
<td>18184</td>
<td></td>
<td>1.368</td>
<td>0.010</td>
<td>2.683</td>
<td>0.036</td>
<td>0.007</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>4.164</td>
</tr>
</tbody>
</table>

Sample data: Heart signal of 10 seconds (1,280 samples)
Sample type: MIT-BIH Normal Sinus Rhythm
Performance

- Execution time < Sampling time
  - Sufficient processing speed

<table>
<thead>
<tr>
<th>Database No.</th>
<th>Reading data</th>
<th>Derivation</th>
<th>Autocorrelation</th>
<th>Find interval</th>
<th>Extraction</th>
<th>Discrimination</th>
<th>Store result</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>16265</td>
<td>3.608</td>
<td>0.008</td>
<td>2.087</td>
<td>0.028</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>5.812</td>
</tr>
<tr>
<td>16273</td>
<td>2.300</td>
<td>0.009</td>
<td>2.387</td>
<td>0.032</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>4.919</td>
</tr>
<tr>
<td>16420</td>
<td>3.363</td>
<td>0.008</td>
<td>2.088</td>
<td>0.028</td>
<td>0.011</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>5.903</td>
</tr>
<tr>
<td>16773</td>
<td>2.671</td>
<td>0.011</td>
<td>2.985</td>
<td>0.040</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>5.700</td>
</tr>
<tr>
<td>16786</td>
<td>1.586</td>
<td>0.011</td>
<td>2.984</td>
<td>0.040</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>4.659</td>
</tr>
<tr>
<td>17052</td>
<td>3.225</td>
<td>0.010</td>
<td>2.688</td>
<td>0.037</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>6.035</td>
</tr>
<tr>
<td>18177</td>
<td>2.890</td>
<td>0.009</td>
<td>2.386</td>
<td>0.032</td>
<td>0.010</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>5.362</td>
</tr>
<tr>
<td>18184</td>
<td>1.368</td>
<td>0.010</td>
<td>2.683</td>
<td>0.036</td>
<td>0.007</td>
<td>0.001</td>
<td>&lt; 0.001</td>
<td>4.164</td>
</tr>
</tbody>
</table>

Sample data: Heart signal of 10 seconds (1,280 samples)
Sample type: MIT-BIH Normal Sinus Rhythm
Performance

- Algorithm has average 69% accuracy.
  - In typical normal waveform

<table>
<thead>
<tr>
<th>Database No.</th>
<th>Detected RR Interval (# of interval)</th>
<th>Failed Detection (# of interval)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16265</td>
<td>14</td>
<td>7 (50%)</td>
</tr>
<tr>
<td>16273</td>
<td>13</td>
<td>3 (23%)</td>
</tr>
<tr>
<td>16420</td>
<td>14</td>
<td>5 (36%)</td>
</tr>
<tr>
<td>16773</td>
<td>10</td>
<td>1 (10%)</td>
</tr>
<tr>
<td>16786</td>
<td>10</td>
<td>3 (30%)</td>
</tr>
<tr>
<td>17052</td>
<td>9</td>
<td>2 (22%)</td>
</tr>
<tr>
<td>18177</td>
<td>15</td>
<td>5 (33%)</td>
</tr>
<tr>
<td>18184</td>
<td>8</td>
<td>3 (38%)</td>
</tr>
</tbody>
</table>

Failing average: 31%

Sample data: Heart signal of 10 seconds (1,280 samples)  
Sample type: MIT-BIH Normal Sinus Rhythm
Peaks value

Peak point (place)

Calculated period

RR interval

The University of Aizu
Embedded Multicore System Research
-- AIZU-I BANSOM Project --
Author: Yasuyoshi Haga
Last Update Date: Jan 22, 2010.

******** Period-Peaks Detection Processing Report ********
Sample Data: MIT-BIH Normal Sinus Rhythm Database No.16255 from PhysioBank Filter lag=0.195s

*** Start of Processing

Range of processing: 0.000s - 2.000s
Interval: 0.602s [0.000s - 0.602s]
PP-P=0.281s, PPV=0.095mV
PP-P=0.375s, PPV=-0.825mV
PP-P=0.422s, PPV=1.469mV
SP-P=0.469s, SPV=-0.776mV
TP-P=0.578s, TPV=0.039mV
UP-P=0.070s, UPV=0.045mV
RR Interval: 0.422s [0.000s - 0.422s]

Interval: 0.609s [0.602s - 1.211s]
PP-P=0.883s, PPV=0.104mV
PP-P=0.984s, PPV=-0.816mV
PP-P=1.023s, PPV=1.457mV
SP-P=1.070s, SPV=-0.772mV
TP-P=1.186s, TPV=0.052mV
UP-P=0.580s, UPV=0.030mV
RR Interval: 0.602s [0.422s - 1.023s]
Conclusion

- Period-Peak Detection (PPD) Algorithm
- Scalable Multicore Implementation for ECG Analysis in Multi-lead Records
- 69% accuracy.
- 50% of total FPGA logics. for 4-lead system.