Gesture recognition and dynamics of an articulated movement

Bertrand SIMON

Student at ENS Lyon, intern at Inria Sophia-Antipolis, Stars team

July 10, 2012
### Table of contents

1. OpenNi functioning

2. Recognition of gestures
   - Preprocessing
   - Analysis
   - Application: digit recognizer

3. Medical Application
   - Context
   - First exercise: Repeated chair stand test
   - Second exercise: Walking

4. Conclusion
Where are we?

1. OpenNi functioning
2. Recognition of gestures
3. Medical Application
4. Conclusion
OpenNi providing data

Information transmitted

- Position and orientation of 15 joints (head, shoulders, hands, knees...), and reliability (0, 0.5 or 1).
- A parametrable smoothing is performed.
- Most robust data: torso.
- Least robust data: knees and elbows, and feet or hand which don’t move and touch something (arm-rest or floor).
Where are we?

1. OpenNi functioning

2. Recognition of gestures
   - Preprocessing
   - Analysis
   - Application: digit recognizer

3. Medical Application

4. Conclusion
How can we process data in order to get proper results?

- Choice of the coordinate system: Cartesian coordinate system (fixed or mobile) based on the position and the orientation of the torso. Computing the speed:

\[
\text{speed}[i] := (\text{pos}[i + 1] - \text{pos}[i - 1])/dt
\]
How can we process data in order to get proper results?

- Choice of the coordinate system: Cartesian coordinate system (fixed or mobile) based on the position and the orientation of the torso. Computing the speed:
  \[
  speed[i] := (pos[i + 1] - pos[i - 1]) / dt
  \]

- Reduction of the noise:
  - median of the last 10 points
  - then mean of the last 5 points

Effect of the filter on a speed signal
Comparison of 2 gestures

- Analysis philosophy: direct curve studies (we don’t use Machine Learning or Markov chains)
  - traditional coordinate system is not appropriate to compare gestures (speed and time don’t matter enough)
  - use of the phase space: speed and position stand equal and the time is not useful any more: the problem is then geometric.

- Method:
  - record the trajectory in the phase space of one or more joints of each gesture.
  - on a video: every X frame, compare the last Y frames to the gestures recorded and returns the closest and the distance to it.

- Difficulties:
  - When does a gesture begin and end?
  - How can we compare two curves in the phase space?
Comparison of 2 gestures

- Analysis philosophy: direct curve studies (we don’t use Machine Learning or Markov chains)
  - traditional coordinate system is not appropriate to compare gestures (speed and time don’t matter enough)
  - use of the phase space: speed and position stand equal and the time is not useful any more: the problem is then geometric.

- Method:
  - record the trajectory in the phase space of one or more joints of each gesture.
  - on a video: every X frame, compare the last Y frames to the gestures recorded and returns the closest and the distance to it.

- Difficulties:
  - When does a gesture begin and end?
  - How can we compare two curves in the phase space?
Comparison of 2 gestures

• Analysis philosophy: direct curve studies (we don’t use Machine Learning or Markov chains)
  ▶ traditional coordinate system is not appropriate to compare gestures (speed and time don’t matter enough)
  ▶ use of the phase space: speed and position stand equal and the time is not useful any more: the problem is then geometric.

• Method:
  ▶ record the trajectory in the phase space of one or more joints of each gesture.
  ▶ on a video: every X frame, compare the last Y frames to the gestures recorded and returns the closest and the distance to it.

• Difficulties:
  ▶ When does a gesture begin and end?
  ▶ How can we compare two curves in the phase space?
3 comparison methods of two sets $s$ and $t$

- Distance between two points $i$ and $j$ concerning $n$ joints:

$$d(i,j) = \sqrt{\sum_{(k:\text{joint}), (x:\text{coord})} \left( (x_{ki} - x_{kj})^2 + (x'_{ki} - x'_{kj})^2 \right)}$$

- Dynamic Time Warping (DTW)
- Hausdorff distance (DMAX)
- Mean distance (DMOY)
3 comparison methods of two sets $s$ and $t$

- Distance between two points $i$ and $j$ concerning $n$ joints
- **Dynamic Time Warping (DTW):**
  - Idea: Compare two curves of which the beginning and end must match. Each point of a curve is matched with at least one of the other. Result: minimum sum of distances between each couple.
  - Implementation: Use of dynamic programming, where $M[i,j]$ is the distance between $s[0...i]$ and $t[0...j]$

$$M[i,j] = d(i,j) + \min(M[i-1,j-1], M[i,j-1], M[i-1,j])$$

- Hausdorff distance (DMAX)
- Mean distance (DMOY)
3 comparison methods of two sets \( s \) and \( t \)

- Distance between two points \( i \) and \( j \) concerning \( n \) joints
- **Dynamic Time Warping (DTW)**
- **Hausdorff distance (DMAX):**
  - *Idea:* Compute the maximum distance between a point and the other curve.
  - *Implementation:*
    
    $$ \max \left( \max_{i \in s} \left( \min_{j \in t} d(i, j) \right), \max_{i \in t} \left( \min_{j \in s} d(i, j) \right) \right) $$

- **Mean distance (DMOY)
3 comparison methods of two sets $s$ and $t$

- Distance between two points $i$ and $j$ concerning $n$ joints
- **Dynamic Time Warping (DTW)**
- **Hausdorff distance (DMAX)**
- **Mean distance (DMOY):**
  - Idea: *Same than Hausdorff distance but less sensible to local phenomena: computation of the mean distance between a point and the other curve.*
  - Implementation:
    $$\text{mean}\left(\left\{\min_{j \in t} d(i,j) \mid i \in s\right\} \cup \left\{\min_{j \in s} d(i,j) \mid i \in t\right\}\right)$$
3 comparison methods of two sets $s$ and $t$

- Distance between two points $i$ and $j$ concerning $n$ joints
- **Dynamic Time Warping** (DTW)
- **Hausdorff distance** (DMAX)
- **Mean distance** (DMOY)

Example (Distances taken in account on two curves)

**Fig 1:** Dynamic Time Warping. **Fig 2:** Hausdorff distance. **Fig 3** Mean distance.
Digit Recognizer

<table>
<thead>
<tr>
<th>DMOY</th>
<th>DMAX</th>
<th>DTW</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.30</td>
<td>0.77</td>
<td>24.19</td>
</tr>
<tr>
<td>-9.42</td>
<td>-1.38</td>
<td>-6.56</td>
</tr>
<tr>
<td>12.98</td>
<td>0.70</td>
<td>8.97</td>
</tr>
<tr>
<td>1.87</td>
<td>0.86</td>
<td>2.30</td>
</tr>
<tr>
<td>5.40</td>
<td>0.99</td>
<td>6.94</td>
</tr>
<tr>
<td>4.63</td>
<td>0.70</td>
<td>7.24</td>
</tr>
<tr>
<td>14.39</td>
<td>0.71</td>
<td>20.30</td>
</tr>
<tr>
<td>10.92</td>
<td>0.88</td>
<td>9.96</td>
</tr>
<tr>
<td>12.46</td>
<td>1.08</td>
<td>23.80</td>
</tr>
<tr>
<td>13.36</td>
<td>0.97</td>
<td>8.55</td>
</tr>
</tbody>
</table>

*From left to right: $y = f(x)$. $x' = f(x)$. $y' = f(y)$. $z' = f(z)$.*

The red 3 is performed and the 2 others digits are sample records.
The results for each distance between the 3 and the digits are on the left.

**Analysis:**
- Remark: the blue digits are not smoothed: the difference is quite obvious.
- Centering and normalisation have been performed.
- DTW and DMOY are much more discriminant.
- DTW is very sensitive to the delimitation of the gesture.
From left to right: $y = f(x)$.
$x' = f(x)$. $y' = f(y)$. $z' = f(z)$.

The red 3 is performed and the 2 others digits are sample records.
The results for each distance between the 3 and the digits are
on the left.

Analysis:
- Remark: the blue digits are not smoothed: the difference is quite obvious.
- Centering and normalisation have been performed.
- DTW and DMOY are much more discriminant.
- DTW is very sensitive to the delimitation of the gesture.
Where are we?

1. OpenNi functioning
2. Recognition of gestures
3. Medical Application
   - Context
   - First exercise: Repeated chair stand test
   - Second exercise: Walking
4. Conclusion
Is Kinect skeleton precise enough to perform medical applications?

- Scenario: Alzheimer and healthy elder people performing precise exercises and recorded by a Kinect.
- Purpose: Analyse skeleton data to detect particularities of Alzheimer disease.
- Method:
  - No dataset is available, so I reproduced what patients performed in videos.
  - Criteria can’t be evaluated properly.
  - The patient moves quite much: use of an absolute coordinate system (axes determined at the beginning).
Is Kinect skeleton precise enough to perform medical applications?

- **Scenario:** Alzheimer and healthy elder people performing precise exercises and recorded by a Kinect.
- **Purpose:** Analyse skeleton data to detect particularities of Alzheimer disease.
- **Method:**
  - No dataset is available, so I reproduced what patients performed in videos.
  - Criteria can’t be evaluated properly.
  - The patient moves quite much: use of an absolute coordinate system (axes determined at the beginning).

Assessment room
Is Kinect skeleton precise enough to perform medical applications?

- Scenario: Alzheimer and healthy elder people performing precise exercises and recorded by a Kinect.
- Purpose: Analyse skeleton data to detect particularities of Alzheimer disease.
- Method:
  - No dataset is available, so I reproduced what patients performed in videos.
  - Criteria can’t be evaluated properly.
  - The patient moves quite much: use of an absolute coordinate system (axes determined at the beginning).

Assessment room
Repeated chair stand

Purposes

- Measuring the time between the transfers.
- Analyse the evolution of the orientation of the torso and detect a fall at the end of the sitting action.
- Analyse the trajectory of the shoulders.
Purposes

- Measuring the time between the transfers.

Method employed

Compute the distances $d_1$, $d_2$ between the head and the feet in the $(xy)$ plan. When the speed of the torso is low, the position is stable: according to $d_1,d_2$, the patient is either standing or sitting.

Problem: depends on the behaviour between transfers.

Example
Purposes

- Analyse the evolution of the orientation of the torso and detect a fall at the end of the sitting action.

Method employed

- Compute the speed of the angle $\alpha$ between the torso and the vertical in the $(yz)$ plan and detect peaks.

- Fall detection: a positive peak follow a negative peak and is 1.5 higher, and the person get sit.

_Idea: a significant difference in amplitude means an uncontrolled movement at the end._

Alpha angle
**Purposes**

- Analyse the evolution of the orientation of the torso and detect a fall at the end of the sitting action.

**Graphic results: A person sits down, then stands up**

The peaks detected are the red curves.

During the first two peaks: forward leaning, then getting straight sat.
Values: left: 0.8; right: 1.8.

During the last two peaks: forward leaning, then getting straight stand.
Possible studies: quantify the flattening of the first peak. *(Idea: the patient can’t leave the chair faster).*

Computation of the maximum angle of the torso.

From top to bottom: $\frac{d\alpha}{dt}$, $d_1$ and $d_2$, $\alpha$ (depending on the time).
Left: normal action. Right: difficulties and fall.
Purposes

- Analyse the trajectory of the shoulders.

Method employed

- Study of the trajectory of the shoulder between two stable positions (sat or stand) in the (yz) plan (cf. fig 2 and 3).
- Compute an estimation of the curvature ($d/AB$ : (cf. fig 1)).

Example (Two people sit down (2) and stand up (3))

- Red curves: difficulties in movement.
- Purple curves: fluent movement.
- Results:
  - fig 2: 33% and 54%.
  - fig 3: 35% and 67%.
Purposes

- Measuring the time between the transfers.
- Analyse the evolution of the orientation of the torso and detect a fall at the end of the sitting action.
- Analyse the trajectory of the shoulders.

Parameters to adjust

- axes of the coordinate system
- minimum height of a standing person
- maximum height of a sitting person
- maximum speed of a stable position
- property of a peak (amplitude and time thresholds)
- minimum rate of a fall
- minimum curvature
Walking

Purposes

Compute the step length, the stride frequency and the walking speed.

Problem: How to know when the person changes foot?

Example (Angle of view)
Purposes

Compute the step length, the stride frequency and the walking speed.
Problem: How to know when the person changes foot?

Method employed

- Compute the $z$ position and speed of each foot.
- Trace curves of two parameters in $\{\text{time}, z_{\text{right}}, z'_{\text{right}}, z_{\text{left}}, z'_{\text{left}}\}$.
- Best results obtained: study the speed of the feet in function of the time.
- The delimiters of a step:
  - beginning: a foot moves fast enough.
  - end: this foot slows down enough, or the other foot moves faster. The movement must last enough.
- Idea: Openni tracks more the ankle than the foot: the joint moves while the foot has stopped. So it's easier to study the feet comparatively. But the last step is biased.
Purposes

Compute the step length, the stride frequency and the walking speed.
Problem: How to know when the person changes foot?

Method employed

- Compute the $z$ position and speed of each foot.
- Trace curves of two parameters in $\{time, z_{right}, z'_{right}, z_{left}, z'_{left}\}$.
- Best results obtained: study the speed of the feet in function of the time.
- The delimiters of a step:
  - beginning: a foot moves fast enough;
  - end: this foot slows down enough, or the other foot moves faster. The movement must last enough.

  Idea: Openni tracks more the ankle than the foot: the joint moves while the foot has stopped. So it’s easier to study the feet comparatively. But the last step is biased.
Purposes

Compute the step length, the stride frequency and the walking speed.
Problem: How to know when the person changes foot?

Method employed

- Compute the $z$ position and speed of each foot.
- Trace curves of two parameters in $\{time, z_{right}, z'_{right}, z_{left}, z'_{left}\}$.
- Best results obtained: study the speed of the feet in function of the time.
- The delimiters of a step:
  - beginning: a foot moves fast enough.
  - end: this foot slows down enough, or the other foot moves faster. The movement must last enough.
- Idea: Openni tracks more the ankle than the foot: the joint moves while the foot has stopped. So it’s easier to study the feet comparatively. But the last step is biased.
Purposes

Compute the step length, the stride frequency and the walking speed.
Problem: How to know when the person changes foot?

Method employed

- Compute the $z$ position and speed of each foot.
- Trace curves of two parameters in $\{time, z_{right}, z'_{right}, z_{left}, z'_{left}\}$.
- Best results obtained: study the speed of the feet in function of the time.
- The delimiters of a step:
  - beginning: a foot moves fast enough.
  - end: this foot slows down enough, or the other foot moves faster. The movement must last enough.
- Idea: Openni tracks more the ankle than the foot: the joint moves while the foot has stopped. So it’s easier to study the feet comparatively. But the last step is biased.
Purposes

Compute the step length, the stride frequency and the walking speed.  
Problem: How to know when the person changes foot?

Method employed

- Compute the $z$ position and speed of each foot.
- Trace curves of two parameters in $\{time, z_{right}, z'_{right}, z_{left}, z'_{left}\}$.
- Best results obtained: study the speed of the feet in function of the time.
- The delimiters of a step:
  - beginning: a foot moves fast enough.
  - end: this foot slows down enough, or the other foot moves faster. The movement must last enough.
- Idea: OpenNi tracks more the ankle than the foot: the joint moves while the foot has stopped. So it’s easier to study the feet comparatively. But the last step is biased.
Purposes

Compute the step length, the stride frequency and the walking speed.
Problem: How to know when the person changes foot?

Method employed

- Compute the $z$ position and speed of each foot.
- Trace curves of two parameters in $\{\text{time, } z_{\text{right}}, z'_{\text{right}}, z_{\text{left}}, z'_{\text{left}}\}$.
- Best results obtained: study the speed of the feet in function of the time.
- The delimiters of a step:
  - beginning: a foot moves fast enough.
  - end: this foot slows down enough, or the other foot moves faster. The movement must last enough.
- Idea: Openni tracks more the ankle than the foot: the joint moves while the foot has stopped. So it’s easier to study the feet comparatively. But the last step is biased.
Purposes
Compute the step length, the stride frequency and the walking speed.
Problem: How to know when the person changes foot?

Method employed
- Compute the $z$ position and speed of each foot.
- Trace curves of two parameters in $\{\text{time}, z_{\text{right}}, z'_{\text{right}}, z_{\text{left}}, z'_{\text{left}}\}$.
- Best results obtained: study the speed of the feet in function of the time.
- The delimiters of a step:
  - beginning: a foot moves fast enough.
  - end: this foot slows down enough, or the other foot moves faster. The movement must last enough.
- **Idea:** *Openni tracks more the ankle than the foot: the joint moves while the foot has stopped. So it’s easier to study the feet comparatively. But the last step is biased.*
Purposes

Compute the step length, the stride frequency and the walking speed.
Problem: How to know when the person changes foot?

Graphic results: A person walking backward then forward

Green (and yellow) curves: left foot. Blue (and purple) curves: right foot.
Fig 1: Speed-time curves. Fig 2: position-time curves. Fig 3: Left in function of right feet positions.
Purposes

Compute the step length, the stride frequency and the walking speed.
Problem: How to know when the person changes foot?

Example (detection of the steps)
Purposes

Compute the step length, the stride frequency and the walking speed.
Problem: How to know when the person changes foot?

Parameters to adjust

- axes of the coordinate system
- property of a step peak (amplitude and time thresholds)
Where are we?

1. OpenNi functioning
2. Recognition of gestures
3. Medical Application
4. Conclusion
How to ease the Kinect process?

- Avoid very fast moves (because of smoothing).
- Place the Kinect not farther than 4-5 meters and so that it could see both legs most of the time.
- Ease the detection of the axes (note the angle of the Kinect for instance)
- Ease the detection of the standing/sitting positions.

General ideas for future studies of such exercises

- Relevant quantities:
  - position and speed of concerned joints (except the least robust data).
  - orientation of the torso.
- Space of studies:
  - phase space doesn’t seem adapted
  - the evolution and comparison of related data in function of time does.
- Useful patterns:
  - peaks and comparison between neighbors ones.
  - Geometric features (curvatures, maxima...).
### Conclusion

#### How to ease the Kinect process?
- Avoid very fast moves (because of smoothing).
- Place the Kinect not farther than 4-5 meters and so that it could see both legs most of the time.
- Ease the detection of the axes (note the angle of the Kinect for instance)
- Ease the detection of the standing/sitting positions.

#### General ideas for future studies of such exercises
- **Relevant quantities:**
  - position and speed of concerned joints (except the least robust data).
  - orientation of the torso.
- **Space of studies:**
  - phase space doesn’t seem adapted
  - the evolution and comparison of related data in function of time does.
- **Useful patterns:**
  - peaks and comparison between neighbors ones.
  - Geometric features (curvatures, maxima...).
Any remarks, questions?

I will try to answer...