Attention detection in driver simulator project #4

- INPG: L. Bonnaud, A. Caplier
- UCL: D. Trevisan, B. Macq
- Participants: A. Benoit, G. Chanel, P. Ngo, V. Levacic, C. Thillou
- Guest star: L. Lawson, Burak
Introduction
Redundancy / Equivalence

Graphical representation
- Screen

Sonic representation
- Speakers

EEG
- EEG interpreter
- EEG

ECG
- ECG interpreter
- ECG

Physiological sensors

Alarm (presentation task)

Loss of watchfulness

Dialog controller

Redundancy / Equivalence

EEG: electroencephalogram
ECG: electrocardiogram

Provided

To be developed

8/19/2005

Dialog controller

Screen

Speakers

EEG

ECG

Eye Blinking detector

Head motion detector

Mouth motion detector

Camera

EEG: electroencephalogram
ECG: electrocardiogram

Provided

To be developed

8/19/2005
Global architecture Project #4 Workshop eINTERFACE'05

all the boxes and arrows in red (dot lines) weren't implemented in this Workshop eINTERFACE'05 due to lack of time or non real-time devices used.

all the boxes in red filled in by yellow (specific draw) are OpenInterface components.
5 Challenges

• Driver Simulator
• Attention detection
  – Biological signals – Stress detection
  – Video-based information – Fatigue detection
• Fusion
• Fission
• Integration
  – Distributed architecture
  – OpenInterface
Driver Simulator

Stop moving your head, look at the road

8/19/2005
Driver Simulator

- TORCS - GPL program well built with source code well structured (C++ and OpneGl)
- Force Feedback with controlled level of wheel vibration
- Message (color)
- Button click (user’s interaction)
- Multi-thread server
- The network protocol used is TCP/IP. We used a “GPL” library called Openthalreads to allow threads access global variables with a Mutual Exception implementation
Attention Detection

Video-based system
  • Fatigue detection
    – Eyes
    – Yawn
    – Head movement
Attention Detection

- Biological-based system (stress detection)
- ECG and GSR
- 3 situations:
  - rest / relaxation
  - Stress stimuli while reading
    - Hand clapping
    - Light in eyes
    - Answering simple question
    - Telephone call, “your dead” (killer game)
  - Playing with the driving simulator (difficult tracks)
- GSR acquisition and analysis can be integrated in real time
- New experiments for detecting relax situation
Fusion

Video image

Face-based detection
  Head, eyelid and yawning movement

Data Fusion
  Bayesian Network

Fatigue?

No

Yes

Activate Data Fission component

Contextual Information
  Physical fitness
  Sleep history
  Time of day
  Temperature

Bayesian Network for detecting Fatigue State

8/19/2005
Fusion

Contextual information

Prior probability

video-based information


8/19/2005
Fission

Data fission responsibility is to collect the data from data fusion and to generate a XML message that is sent to the driver simulator.

<table>
<thead>
<tr>
<th>Fatigue range</th>
<th>[0,33]</th>
<th>[33,66]</th>
<th>[66,100]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message</td>
<td>« »</td>
<td>« Tired »</td>
<td>« A sleep »</td>
</tr>
<tr>
<td>Message color</td>
<td>« »</td>
<td>« Green »</td>
<td>« Red »</td>
</tr>
<tr>
<td>Shaking power</td>
<td>« 0 »</td>
<td>« 0 »</td>
<td>« 100 »</td>
</tr>
</tbody>
</table>
Global architecture Project #4

TCP/IP Socket

LAN

Dialog controller

Sonic feedback

Haptic feedback

Screen

Loudspeaker

Vibrating wheel

TCP/IP Socket

Fission (redundancy, equivalence)

Fusion (redundancy, equivalence)

High GSR?

Local Peak?

GSR analysis

GSR

Using Matlab under Windows

Using TORCS driving simulator under Windows

High GSR?

Local Peak?

High heart rate variation?

High frequency?

Both eyes closed?

Significant rotation?

Yawn?

GSR analysis

ECG analysis

Eyes blinks detector

Head rotation estimation

Mouth opening detector

Video Stream

Using Matlab under LINUX

all the boxes and arrows in red (dot lines) weren't implemented in this Workshop eINTERFACE'05 due to lack of time or non real-time devices used.

all the boxes in red filled in by yellow (specific draw) are OpenInterface components.
Each component is registered into OpenInterface Platform using the Component Interface Description Language (CIDL) and described in XML. The registered components' properties are retrieved by the Graphic Editor (Java).
Future Works

• Integrate biological signals for fatigue detection

• Usability tests to assess interface interactions.

• Improve the Bayesian Network to take account more specialized information about head orientation once this information is available in the head detection code.

• Transform the face detection component into 3 OpenInterface components.
Conclusion

- 1st goal: real time distributed system based on video data, integrated under OpenInterface, for driver attention level analysis with feed-back to the user

- 2nd goal: multimodal system taking into account the biological signals - Stress
TEAM
DEMO

Have Fun and come to play during our Demo session!
Questions?