Pendocs

Please insert in the following the information that is relevant to your project

Deliverable Title	Report on Gesture Format. State of the Art. Partners'propositions
Research Direction number	RD3.3
Deliverable Number	D.RD3.3.1
Edited by:	INPG
Approved by	
Nature of Deliverable	Report
Distribution[1] (as by Technical Annex)	CO
Contractual Delivery Date (DD/MM/YYYY)	31 / OCT / 2005
Actual Delivery Date (DD/MM/YYYY)	3 / NOV / 2005
Authors	Editor Luciani Annie (INPG) , RD33 leader Authors INPG: A. Luciani, Matthieu Evrard, Damien Couroussé, Annie Luciani, Nicolas Castagné UNEXE: Ian Summers, Alan Brady PERCRO: Carlo Alberto Avizzano, Franco Tecchia, Mirko Raspolli, Marcello Carrozzino DIST: Gualtiero Volpe and Barbara Mazzarino SPCL: M. Wanderley UPS/UM1: B. Bardy HFRL : T. Stoffregen DEI: G. De Poli, F. Avanzini, A. Rodà, L. Mion, D'Incà, C. Trestino, D. Pirrò
Abstract	This report presents the state of the art in "gesture format" and preliminary propositions of the partners in the specification of a common format. It includes (1) presentation of the cardinal points in research on gesture data, (2) state of the art in existing gesture format (3) presentation of the work of each partner, (4) synthesis of the needs.
Keywords	Gesture format, motion, motion capture, signal, signal processing, expressive gesture.

[1]Please indicate the dissemination level for deliverables using one of the following codes:
PU = Public
PP = Restricted to other programme participants (including the Commission Services).
RE = Restricted to a group specified by the consortium (including the Commission Services).
CO = Confidential, only for members of the consortium (including the Commission Services).

ENACTIVE

"ENACTIVE INTERFACES"

Project IST-2004-002114-ENACTIVE
RD3.3/D.RD33.1 /DOC/2005

Edited by: A. Luciani, INPG

Nature of Deliverable: Report
Distribution: CO
Contractual Delivery Date: 31st October 2005

State of the Art on current interaction paradigms based on action and vision

(D.RD3.3.1)

Actual Delivery Date:
3rd November 2004

Abstract: This report presents the state of the art in "gesture format" and preliminary propositions of the partners in the specification of a common format. It includes (1) presentation of the cardinal points in research on gesture data, (2) state of the art in existing gesture format (3) presentation of the work of each partner, (4) synthesis of the needs.

Keywords: Gesture format, motion, motion capture, signal, signal processing, expressive gesture**.**

A.Status Sheet

DOCUMENT TITLE: State of the Art on current interaction paradigls based on vision and action

ISSUE	REVISION	DATE	CHAPTER - PAGE REVISED
1	0	1srt November 2005	n.a.

B. Executive Summary

Documents provided by the participants

Title	Authors and institutions	Reference
Introduction Reason of the interest in gesture format Synthetic view of cardinal points : gesture generation, gesture acquisition (performance), gesture synthesis, gesture processing.	A. Luciani, INPG	EI_D_RD33.1_INPG1_051101
State of the Art on Gesture Data Format in Motion Capture systems	Matthieu Evrard, Damien Couroussé, Annie Luciani, Nicolas Castagné	EI_D_RD33.1_INPG2_051025.doc
Gesture Data Format in the interactive simulation context of INPG	Annie Luciani, Claude Cadoz, Nicolas Castagné, Matthieu Evrard, Damien Couroussé, Jean-Loup Florens INPG	EI_D_RD33_INPG3_051101.doc
State of the Art on Gesture Format – Contribution from UNEXE	Ian Summers, Alan Brady UNEXE	EI_D_RD33 1_UNEXE_051028.doc
PERCRO – A new device for hand motion capture	PERCRO	EI_D_RD33.1_PERCRO1_051028.doc
Haptic Desktop characteristics	PERCRO	EI_D_RD33.1_PERCRO1_051028.doc
Extraction of gestural features with the EyesWeb Platform	Gualtiero Volpe and Barbara Mazzarino DIST	EI_D_RD33.1_DIST.doc
Gestures analysis in musical performance (SPCL, UPS, HFRL))	M. Wanderley (SPCL), B. Bardy(UPS/UM1), T. Stoffregen (HFRL	EI_D_RD33_1_UPS_SPCL_HRFL_051021.doc
Towards a multi-layer architecture for multi-modal rendering of expressive actions	DEI: G. De Poli, F. Avanzini, A. Rodà, L. Mion, D'Incà, C. Trestino, D. Pirrò INPG: A. Luciani, N. Castagne	ENACTIVE05_DEI_INPG.doc

Table of contents

1 Introduction

Reason of the interest in gesture format
Synthetic view of cardinal points : gesture generation, gesture acquisition (performance), gesture synthesis, gesture processing.
by Annie Luciani, October 2005

Gesture, motricity, haptic perception are deciding factors in the way we act on and we perceive our environment. During the last few years, many research centres have been working on gesture control, movement analysis or synthesis, in domains as various as surgery, aeronautics, multimedia, cognitive sciences, artistic creation and generally speaking in every interactive systems.

Gestures and motions data appear more and more as a "hub" in person – systems interaction processes and in the design and the production of computerized artifacts (computer sounds and music, computer graphics and animation, multimedia events, computer games, etc.).

As such, data concerning movement and gestures often need to be exchanged. More and more, 3D images synthesis are made in a context on which they have to communicate or they have to be correlated to Sound synthesis processes, as in interactive multimedia installations, and most of them wish to share similar processes of gestural control.

Consequently, understood as a generic type of data and processes, gestures and motions could be common to several and for the moment disjoined activities, leading to support exchanges and to share systems, data and experiences.

However, there are several approaches that have to render more clear and more explicit, in order to have a precise representation of what it is developed and of what type of difficulties "collaborative work" will rise.
In other words, a part of the work is to draw what are "the cardinal points" of the research around the question of "gesture-motion-action, etc".

We identified four "cardinal points":

1. Gesture Command System Paradigm (Figure 1)
2. Gesture Signal Processing Paradigm (Figure 2
3. Gesture Generating Process Paradigm (Figure 3)
4. Gesture Signal Reconstruction Paradigm (Figure 4)

On each figure, the grey circles indicate where "gestural signals or "motions signals" as considered as gestures signals can be identified. The grey squares identify the signals that are not usually considered as "gesture data".

1.1 Control-Command / system paradigm

The control – command paradigm is illustrated in figure 1. It leads to develop :
• New sensors and actuators
• New way of control – command of virtual instruments and/or virtual objects
• New real-time computer architecture for performance

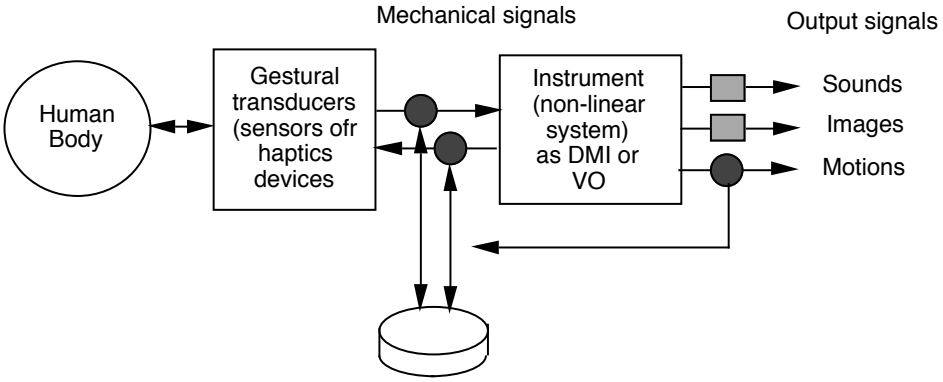


Figure 1. Generic scheme of "Gestural Command System Paradigm"

From the point of view of a standardization of gestural data inputs/outputs, it can be decomposed in two main approaches :

a. The "pure mapping concept" that does not implement force feedback and physically-based gestural control
b. The "and extended mapping concept" and the "ergotic concept, that implement force feedback and physically-based gestural control.

There are two main differences between both:

(1) in the first "pure mapping concept, the gesture signals are conventional positions (or their derivatives) although in the second, they can be also variables as forces.
(2) Force feedback introduces the need of higher frequency sampling: from some Hz for pure mapping concept to some KHz on case of force or force feedback.

1.2 Signal processing paradigm

The following figure schematizes the "signal processing paradigm". There is two different cases that canbe developed from signal processing: signal processing in itself (filters, etc.), and extraction of high level festures from signals.

Signal processing differs from the "command signal paradigm" in two points:
• In case of command-control paradigm, the controlled system is built to introduce a non-linear transformation between its inputs and its outputs. The exemplary case is that of musical instrument which transforms mechanical signals in sounds, the first being non-centered displacements at a low frequency bandwidth (from some Hz to some KHz, the second being centered-to-zero deformations at a high frequency bandwidth (from 10 KHz to 40 KHz).
• In case of signal processing paradigm, the system (such as filters, correlation systems, etc.) has not to transform the inputs data. The outputs are of the same nature than the inputs. In motion analysis, there are filtering technics (to reduce noises of motion capture), other motion data (velocities from positions, slopes, direction changing, etc.). Improved methods extract other data as evolution of the center of gravity of the body, evolution of contours (silhouette), etc.

Within the network, we can observe that the research on the motion analysis is not sufficiently clear identified and visible even though it represents a force of the network and one of the two bottlenecks to overcome.

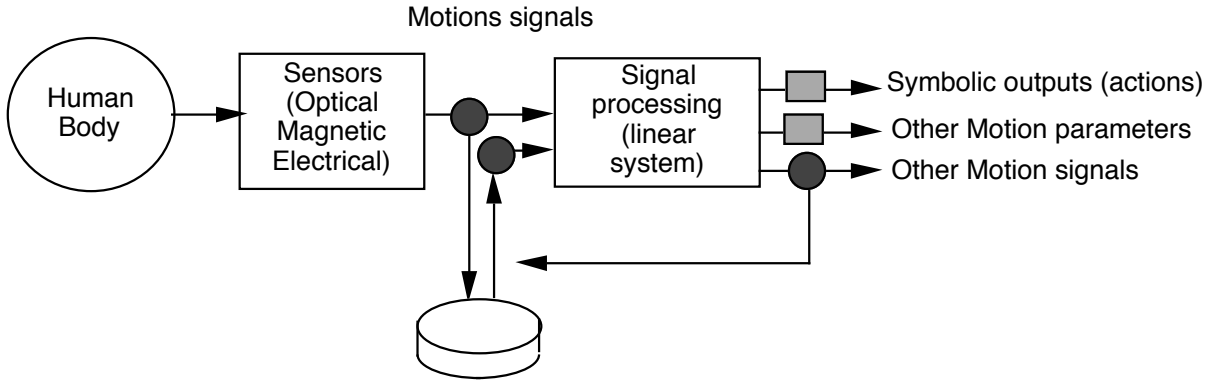


Figure 2. Generic scheme of "Gestural Signal Processing Paradigm"

Beyond signal processing and analysis, another field could be the identification of the parameters of the system which could be able to produce the motion. It implies to develop very complex identification and inverse methods and it is not represented within the network.

High level features extraction aims at to extract from the signals some symbolic features, by help of a predefined external knowledge. That is the aim of "gesture recognition: gesture being here understood at the higher symbolic level, and thus being synonymous of "actions". A brief state of the art in gesture recognition is made in D4b.1 and it is not the concern of this study on gesture format.

1.3 Generative process paradigm

As we noticed that gestures could be outputs of evolving system (humans being one of them), all synthetic processes that produces motions as temporal evolution functions can be considered as generative gestural process (Figure 3). The systems can be obviously controlled by stored gestural signals. Among them, there are cinematic computer animation and physically-based modeling and simulation.

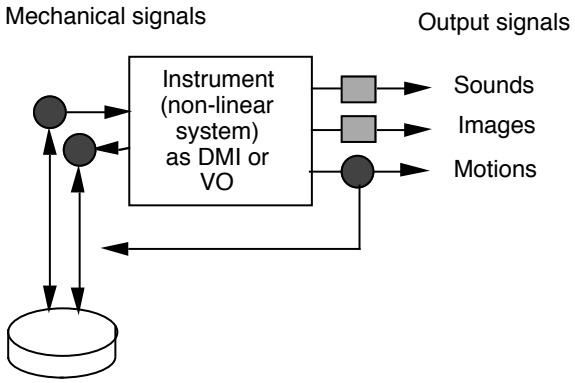


Figure 3. Generic scheme of "Gestural Signal Generative Paradigm"

We can observe than the simulation of the human body (by way of biomechanical models or not) produce exactly the same type of data than body motion capture.

1.4 Signal reconstruction paradigm

To be complete, a way to produce gestures that is complementary with the generative process paradigm presented in the previous paragraph is the signal reconstruction paradigm. It is the inverse operation of signal processing.

How can we produce motions (or other sensory signals)
• from high level (actions),
• from middle level (motions parameters)
• from other non gestural temporal signals as sounds or images? (Figure 4)

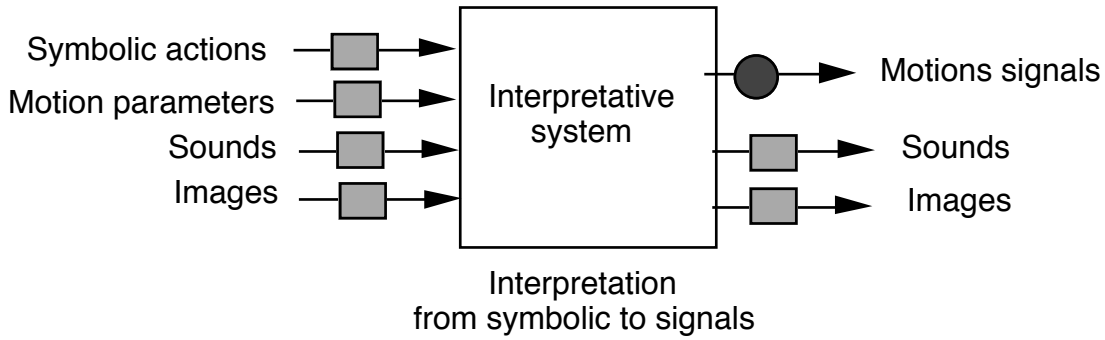


Figure 3. Generic scheme of "Gestural Signal Reconstruction Paradigm"

The first (symbolic actions) addresses the question of "How symbolic actions can be transformed in motions", that is models of motricity control (motricity controled by intentionality, (motricity controled by the goal, etc.).
Such reserach axis is not represented in RD33.

The second addresses the questions of :
• How qualities (softness, etc.) can be introduced in motions?
• What are the parameters of the systems to be controlled to reach a certain quality of the motions,of the sounds, of the optical flow?
Such research axis is driven by DIST and DEI in RD33.

The third (sounds and optical flow to motions) is in itself ambiguous:
• Whether the process is simple: for example, we can call gestures the macroscopic evolution of a musical event or of a visual animation. Thus, it refers to signal processing and signal features extraction.
• Whether it refers to complex inverse techniques of identification of the object manipulated and its system of control. It is impossible without knowing one of them. In any cases, it is very complex.
Such rsearch direction is not represented in RD33.

1.5 Conclusions

To examine if it could be possible to define a common gesture format,we have to examine immplementations of each paradigms when they exist.
Works of INPG, UNEXE, PERCRO, SPCL MacGill, DEI refers to the first "Gesture Command System Paradigm"
Works of DIST and DEI refer to the second "Gesture Signal Processing Paradigm".
Works of INPG, (and UNIGE, EPFL that are not in involved in RD3.3) refer to the third "Gesture Generating Process Paradigm"
Works of DIST and DEI refer to the second point of the forth "Gesture Signal Reconstruction Paradigm".

2 State of the Art on Gesture Data Format in Motion Capture systems

Matthieu Evrard, Damien Couroussé, Annie Luciani, Nicolas Castagné
INPG – October 2005

2.1 Introduction

The use of motion capture for motion analysis and synthesis has begun in the late 1970's and is nowadays wide spread. Motion capture is the recording of human body movement (or other movement) for immediate or delayed analysis and playback. The information captured can be as general as the simple position of the body in space or as complex as the deformations of the face and muscle masses. Motion capture for computer character animation involves the mapping of human motion onto the motion of a computer character. The mapping can be direct, such as human arm motion controlling a character's arm motion, or indirect, such as human hand and finger patterns controlling a character's skin colour or emotional state. The most current systems that capture the movement are optical or magnetic systems.
In order to communicate between a motion capture system and an application that uses those data (for character animation or any kind of data processing), or between two applications, many file formats have been created. This document draws an overview of the most current exchange file formats in motion capture.

2.2 BVA and BVH file format

2.2.1 General description

BVH means Biovision Hierarchical Data. This format is one of the most currently used today. At the basis, Biovision, a motion capture company, has developed it. The BVA format, also developed by Biovision is the BVH primal form. This format is mainly used for the animation of humanoid structures and gives a standardised definition of them. The format is largely and successfully spread in animation community certainly because of its simple specifications. However, some lack can be notice especially the definition of the skeleton in its rest position (basis pose), which seems to be incomplete and the lack of indication dedicated to the representation of the segments. BVA and BVH files are ASCII files.
In a BVA file, sample coming from each segment of the skeleton are stock in a more or less raw manner, without taking into account the structure of the animated form. For this reason Biovision has then developed the BVH format.
A BVH file is divided into two sections. The head section describes the structure (hierarchy, parent links) and the basis pose of the skeleton. The following section contains the data concerning the movement. The space, in this type of representation, is defined as a direct orthonormed system where the Y-axis is the vertical one. So, the segments of the skeleton are generally aligned following this axis in its rest position.

2.2.2 Example of a BVA file

```
Segment:   Hips
Frames:    2
Frame Time: 0.033333
XTRAN   YTRAN   ZTRAN   XROT    YROT    ZROT    XSCALE  YSCALE  ZSCALE
INCHES  INCHES  INCHES  DEGREES DEGREES DEGREES INCHES  INCHES  INCHES
 8.03    35.01   88.36   14.78  -164.35 -3.41    5.21    5.21    5.21
 7.81    35.10   86.47   12.94  -166.97 -3.78    5.21    5.21    5.21
Segment:   Chest
Frames:    2
Frame Time: 0.033333
XTRAN   YTRAN   ZTRAN   XROT    YROT    ZROT    XSCALE  YSCALE  ZSCALE
INCHES  INCHES  INCHES  DEGREES DEGREES DEGREES INCHES  INCHES  INCHES
 8.33    40.04   89.69  -27.24   175.94 -2.88    18.65   18.65   18.65
 8.15    40.16   87.63  -31.12   175.58 -4.08    18.65   18.65   18.65
Segment:   Neck
```

```
Frames:    2
Frame Time: 0.033333
XTRAN   YTRAN   ZTRAN   XROT    YROT    ZROT    XSCALE  YSCALE  ZSCALE
INCHES  INCHES  INCHES  DEGREES DEGREES DEGREES INCHES  INCHES  INCHES
 9.16    56.60   81.15  -69.21   159.37 -27.46   5.45    5.45    5.45
 9.28    56.09   78.00  -72.40   153.61 -33.72   5.45    5.45    5.45
Segment:   Head
Frames:    2
Frame Time: 0.033333
XTRAN   YTRAN   ZTRAN   XROT    YROT    ZROT    XSCALE  YSCALE  ZSCALE
INCHES  INCHES  INCHES  DEGREES DEGREES DEGREES INCHES  INCHES  INCHES
 10.05   58.32   76.05  -29.04  -178.51 -8.97    3.87    3.87    3.87
 10.20   57.46   72.80  -32.77  -179.46 -9.60    3.87    3.87    3.87
```

2.2.3 Descriptive section of the hierarchical structure in BVH file

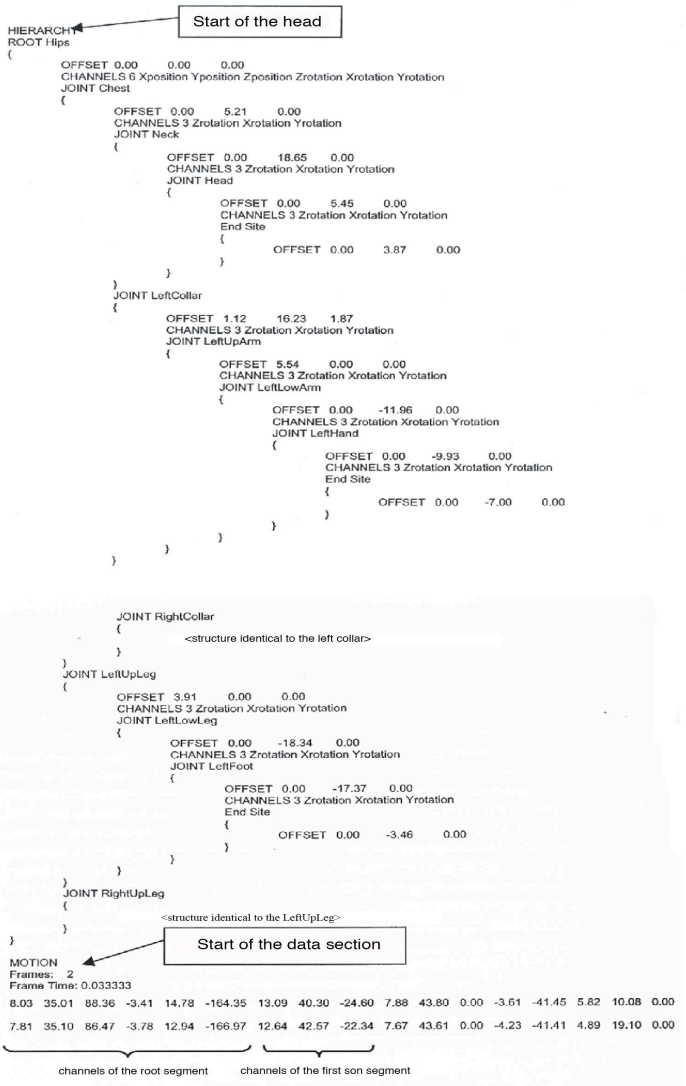
The form to animate is defined in a recursive manner: each segment is defined by its own parameters and by its parentage with an other segment. The head section start with the key word HIERARCHY.
The structure is described in this section. The first element described is the root element indicated by the ROOT keyword. The structure is defined from this first element. The links between the different segments are modelled by a father relation or by a son relation. New hierarchies can be added to the file by adding new ROOT keywords.
Information associated with each nodes of the hierarchy is:

- The offset (OFFSET keyword) gives the position relatively to origin for the root segment and to the parent position for the other segments.
- A number that indicate the number of channels and the name of each channel follows the CHANNEL keyword. A channel carries a one-dimensional signal (position or orientation of the segment relatively to its parent). The root segment generally has six channels (three for the position and three for the rotations) while the other segments generally have 3 channels (the 3 rotations because the joins are fixed in relative translation). Nevertheless it is possible to add channels depending on the user wishes.
- JOINT has the same function than ROOT but is used for the declaration of a son segment. Every son segment of a given father has to be declared in the father structure.
- The “End Site” token closes a hierarchy and means that a segment is a leaf of it. The OFFSET of this ending segment is finally given.

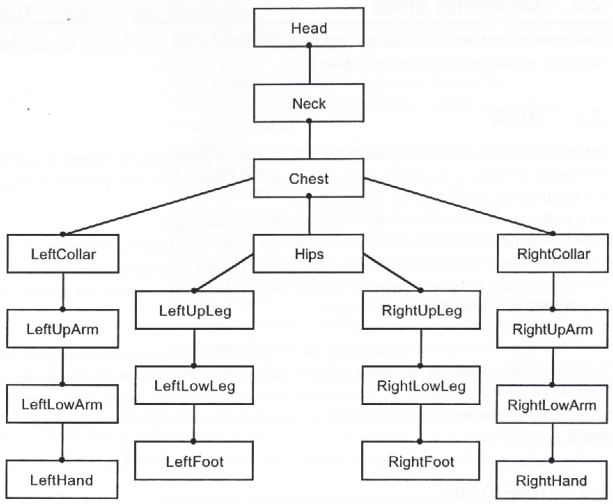
2.2.4 2.4. The data section

The section that contains the movement data starts by the key word MOTION. The next two lines successively give the number of frame and the sampling period. The data of each frame are then disposed one after one on the same line. We change of line for each new frame.
Here is an example of a BVH file

2.2.5 Example of a BVH file



The structure defined in this example (which is the basis structure for the BVH format) can be schematised in be following way:



Many problems are inherent to the BVH format. The most remarkable is that orientation and position of each segment are not absolute. It can lead to a huge cost in term of processing to obtain the position and the orientation for a segment at the leaf of the hierarchy.
Another problem is the lack of calibration for data such as the scale with which the offsets are measured… It becomes very hard to treat several files for which the capture process was not the same (done in the same conditions).

2.3 ASK/SDL file format

The format is a variant of the BVH file format developed by Biovision. Le ASK file (Alias Skeleton) only contain information concerning the skeleton and, as a result, does not contain any information about the channels or the movement. The offset coordinates are absolute unlike the BVH in which they are relative.
The SDL file associated to the ASK file contain the data of the movement but it can contain many other information concerning the scene than the very samples of the movement.

2.4 MNM file format

This file format allows renaming the segments of a BVH file: a name defined by the user is associated to the predefined label in the BVH file.

2.5 AOA file format

2.5.1 General description

Adaptative Optics is a company dedicated to the creation of hardware support for the motion capture. This ASCII file format simply describes the captors and their position at each sampling period. It is divided into to sections.
The first section (the head section) contain two lines:

- A commentary line without any peculiar starting character.
- A line with three token : frames = <number of frames> [space] nLimmarkers = <number of marker on each sample> [space] hz = <sampling frequency>

Inside the data section, the XYZ positions are given. One line is equal to one marker and once a sample is over a new one starts. No name is associated with the values. Therefore the structure is implicit.

2.5.2 Example of an AOA file

```
AOA file written by LambSoft Pro Motion
frames = 2 nummarkers = 10 hz = 30
  131.1524   163.0935    71.6937
  124.6823    71.3264    76.5239
  121.2981   102.4109    75.0063
  149.2998   181.1166   132.8877
  134.7889   192.6069   116.5414
  122.5840   190.7085    89.0098
  149.5519   239.0187   104.2023
  131.1980   121.0528   101.0101
  160.2013    95.1255   134.9703
  137.7529    47.9096    96.5317
  131.2017   163.1528    71.7291
  124.6200    71.3926    76.5561
  121.2042   102.5331    75.0017
  149.3542   181.1748   132.9477
  134.8884   192.6407   116.6287
  122.3916   190.7327    88.9545
  149.5145   239.0789   104.2528
  131.3174   121.1239   101.0545
  160.1843    95.1614   135.0377
  137.7782    47.9176    96.5733
```

2.6 The ASF/AMC file format

2.6.1 General description

This format was developed by Acclaim, a video game company. Once entered in the public domain it has been used by Oxford Metrics (Vicon Motion Capture Systems).
The Acclaim format is composed of two different files, one for the skeleton and the other one for the movement. The separation between these two types has been done because the same skeleton is often used for numerous distinct movements. The file containing the skeleton description in the ASF file (Acclaim Skeleton File) and the file containing the movement data is the AMC file (Acclaim Motion Capture data).

2.6.2 The ASF file

The starting point of the motion capture is the basis position of the skeleton. Each file has only one root segment, therefore only one structure. Each segment carries information useful for its geometric representation as well as information concerning the dynamic of the character. The same information than the other format on this type is present (id, name, length, rotation axes, degree of freedom…) but information about the interval of the possible movement for each rotation (from the basis position) is added.

2.6.3 Example of an ASF file

```
#
# Comment line
##
 #
:version 1.10
:name BioSkeleton
:units
   mass 1.0
   length 1.0
   angle deg
:documentation
   Example of an Acclaim skeleton
   To be used with "Walk.amc"
:root
   axis XYZ
   order TX TY TZ RZ RY RX
   position 0.0 0.0 0.0
   orientation 0.0 0.0 0.0
:bonedata
begin
   id 1
   name hips
   direction 0.000000 1.000000 0.000000
   length 0.000000
   axis 0.00000 0.00000 0.00000 XYZ
   dof rx ry rz
   limits (-180.0 180.0)
      (-180.0 180.0)
      (-180.0 180.0)
end
begin
   id 2
   name hips1
   direction 0.000000 1.000000 0.000000
   length  4.310000
   axis 0.000 0.000 0.000 XYZ
end

identical declaration for join 3 to join 24

:hierarchy
begin
   root hips
   hips hips1 hips2 hips3
   hips1 chest
```

```
   chest chest1 chest2 chest3
   chest1 neck
   neck head
   chest2 leftcollar
   leftcollar leftuparm
   leftuparm leftlowarm
   leftlowarm lefthand
   chest3 rightcollar
   rightcollar rightuparm
   rightuparm rightlowarm
   rightlowarm righthand
   hips2 leftupleg
   leftupleg leftlowleg
   leftlowleg leftfoot
   hips3 rightupleg
   rightupleg rightlowleg
   rightlowleg rightfoot
end
```

2.6.4 The AMC file format

This file contains the data concerning the movement of the skeleton. The samples are given one after one. A sample is presented on several lines (one for a segment and all of its degree of freedom).

2.6.5 Example of an AMC file associated with the ASF file above

```
# Space for comments
##
 #
:FULLY-SPECIFIED
:DEGREES
1
root -1.244205 36.710186 -1.148101 0.958161 4.190043 -18.282991
hips 0.000000 0.000000 0.000000
chest 15.511776 -2.804996 -0.725314
neck 48.559605 0.000000 0.014236
head -38.332661 1.462782 -1.753684
leftcollar 0.000000 15.958783 0.921166
leftuparm -10.319685 -15.040003 63.091194
leftlowarm -27.769176 -15.856658 8.187016
lefthand 2.601753 -0.217064 -5.543770
rightcollar 0.000000 -8.470076 2.895008
rightuparm 6.496142 9.551583 -57.854118
rightlowarm -26.983490 11.338276 -5.716377
righthand -6.387745 -1.258509 5.876069
leftupleg 23.412262 -5.325913 12.099395
leftlowleg -6.933442 -6.276054 -1.363996
leftfoot -1.877641 4.455667 -6.275022
rightupleg 20.698696 3.189690 -8.377244
rightlowleg 3.445840 -6.717122 2.046032
rightfoot -8.162314 0.687809 9.000264
2
root -0.227361 37.620358 1.672587 0.204373 -4.264866 -12.155879
hips 0.000000 0.000000 0.000000
chest 14.747641 2.858763 -1.345236
neck 44.651531 0.000000 -0.099206
head -38.546989 0.678145 -4.633668
leftcollar 0.000000 7.233337 -5.791124
leftuparm 9.928153 -50.015823 25.218475
leftlowarm -40.443512 -0.566324 0.482702
lefthand 6.011584 -0.216811 4.576208
rightcollar 0.000000 -1.936009 5.471129
rightuparm 3.926107 32.418419 -26.396805
rightlowarm -43.958717 3.548671 -3.415734
righthand -4.901258 -0.112565 0.681468
leftupleg 11.932759 0.406248 -1.921313
leftlowleg 13.698170 5.503362 2.643481
leftfoot -16.237123 2.755839 -7.182952
rightupleg 13.767217 0.331739 -1.353482
rightlowleg 22.576195 -7.388037 -3.537788
rightfoot -19.946142 2.525145 8.668705
```

2.7 The BRD file format

2.7.1 General Description

The format is uniquely used by the motion capture system Ascension Technology “Flock of Birds” developed by LambSoft. It allowed stocking the data coming from a magnetic system. The first number is the sample number; the first sample in the file is sample 0.The second number is the marker number. The first marker is marker 0. The next token is "P", this indicates that the next three values on the line are position values for the marker.

After the position values the token "Q" appears. This indicates that the next 4 values are quaternions. The last two values on each line are optional. These are the time stamp values for each marker. These numbers are identical to the values returned by the Unix system call gettimeofday() (The system call "_ftime()" under Windows does a similar thing except it returns milliseconds). The first number is in seconds; the second number is in microseconds. It is possible for the time stamp values of one marker to be different from the time stamp values for a second marker in the same sample.

2.7.2 Example of a BRD file

```
0   0 P     55.9477     32.7718     -5.2410  Q      0.7569      0.3548     -0.4984      0.2299 0
000000
0   1 P     52.5374     26.9382      9.3167  Q     -0.6740     -0.4260     -0.5366      0.2762 0
000000
1 14 P    43.0703    48.2499   -39.1160  Q     0.8875    -0.4361     0.1333     0.0660 0 16666
1 15 P    49.5336    30.1453     4.2165  Q    -0.7198    -0.3039    -0.5944     0.1904 0 16666
2 0  P    57.7803    34.8796    -4.7169  Q     0.7569     0.3548    -0.4984     0.2299 0 33332
2 1  P    53.8193    29.2250     9.4211  Q    -0.6740    -0.4260    -0.5366     0.2762 0 33332
```

2.8 The HTR and GTR file formats

2.8.1 General description

The HTR format (Hierarchical Translation Rotation) has been developed as a native format for the skeleton of the Motion Analysis software. It has been created as an alternative to the BVH format to make up for its main drawbacks. A HTR variant exist which is called the GTR format (Global Translation Rotation) and is the same format less the structural information. This format has known few uses.
The HTR format is divided into 4 sections: head, hierarchy and name of the segments, basis position and movement data.
The head section marked by the [HEADER] token contains many data lines including information carried by the BVH format plus:

- The number of segment
- The disposition of the eulerian angles
- The calibration unities
- The rotation unities
- The axes following which the gravity is exerted
- The axis flowing which the segments of the skeleton are aligned
- The global scale factor

One of the interests of this format is that the hierarchy [SegmentNames&hierarchy] is given separately of the characteristic of each segment [BasePosition] what simplifies the reading. Moreover, the movement data are not indicated by a key word but by the name of the concerned segment.

2.8.2 Example of a HTR file

```
[Header]                    #      Header keywords are followed by a single value
FileType       htr          # single word string
DataType       HTRS         # Hierarchical translations followed by rotations and Scale
```

```
FileVersion    1              # integer
NumSegments    18             # integer
NumFrames      2              # integer
DataFrameRate  30             # integer
EulerRotationOrder     ZYX    # one word string
CalibrationUnits       mm     # one word string
RotationUnits  Degrees        # one word string
GlobalAxisofGravity    Y      # character, X or Y or Z
BoneLengthAxis Y
[SegmentNames&Hierarchy]
#CHILD  PARENT
LOWERTORSO     GLOBAL
UPPERTORSO     LOWERTORSO
NECK           UPPERTORSO
HEAD           NECK
RSHOULDER      UPPERTORSO
RUPPERARM      RSHOULDER
RLOWARM RUPPERARM
RHAND          RLOWARM
LSHOULDER      UPPERTORSO
LUPPERARM      LSHOULDER
LLOWARM LUPPERARM
LHAND          LLOWARM
RTHIGH         LOWERTORSO
RLOWLEG RTHIGH
RFOOT          RLOWLEG
LTHIGH         LOWERTORSO
LLOWLEG LTHIGH
LFOOT          LLOWLEG
[BasePosition]
#SegmentName   Tx, Ty, Tz, Rx, Ry, Rz, BoneLength
LOWERTORSO     0.00      0.00      0.00      0.00      0.00      0.00    200.00
UPPERTORSO     0.00    200.00      0.00     -1.38      0.00      0.35    286.95
NECK           0.00    286.95      0.00      2.90     -0.08      3.20    101.66
HEAD           0.00    101.66      0.00     -1.53     -0.09     -3.55    174.00
RSHOULDER      -10.21    252.02     -0.84      1.85     -1.36     98.76    137.50
RUPPERARM      0.00    137.50      0.00      3.22      0.48     13.42    279.07
RLOWARM 0.00    279.07      0.00     -2.42     -1.40    -15.48    222.64
RHAND          0.00    222.64      0.00     -2.59     -0.32     -6.98     90.00
LSHOULDER      9.79    251.90     -0.84     -5.30      1.36    -98.63    132.79
LUPPERARM      0.00    132.79      0.00     13.46      1.21    -13.67    295.17
LLOWARM 0.00    295.17      0.00     -6.60      2.65     18.04    222.81
LHAND          0.00    222.81      0.00     -1.49      0.10      3.78     90.00
RTHIGH         -96.49    -31.41     26.89     -6.15      0.00    176.17    379.17
RLOWLEG 0.00    379.17      0.00      4.86     -0.14      1.34    394.60
RFOOT          0.00    394.60      0.00     71.40     -0.06      2.48    160.00
LTHIGH         107.90    -45.36      2.84     -4.81      0.00   -178.69    362.85
LLOWLEG 0.00    362.85      0.00      5.06     -0.03      0.30    398.36
LFOOT          0.00    398.36      0.00     69.87     -0.01     -1.61    160.00
[LOWERTORSO]
       1    263.72    816.20  -2874.77     18.03     -7.70    -10.34      1.00
       2    264.42    812.41  -2740.34     19.81    -13.46    -11.93      1.00
[UPPERTORSO]
       1      0.00      0.00      0.00      8.33    -17.38      8.59      1.00
       2      0.00      0.00      0.00      8.71     -6.14      8.64      1.00

Same declaration for marker 3 to 18

[EndOfFile]
```

2.9 The TRC file format

2.9.1 General description

The TRC file format is another file format from Motion Analysis. It contains not only the raw data from the full body motion capture system they developed but also the output data coming from their face tracker. The output data of the face tracker are 2D data. So to allow merging 2D and 3D data the Z-coordinate value is set to 0 in the 2D case. The positions are given in absolute coordinate. Each maker has a name and is only referenced by its in the format.

2.9.2 Example of a TCR file

```
PathFileType   3       (X/Y/Z) Example.trc
DataRate  CameraRate NumFrames  NumMarkers  Units
30 30 2 31 mm
Frame# Time    bridge  clowlip cuplip  lbrow1  lbrow2  lbrow3  lhead   llaugh  llcheek llowjaw
       llowlip lmouth  lsocket lucheek lupjaw  luplip  nose    rbrow1  rbrow2  rbrow3  rhead
       rlaugh  rlcheek rlowjaw rlowlip rmouth  rsocket rucheek rupjaw  ruplip  tophead
 X1 Y1 Z1  X2 Y2 Z2  X3 Y3 Z3  X4 Y4 Z4  X5 Y5 Z5  X6 Y6 Z6  X7 Y7 Z7  X8 Y8 Z8  X9 Y9 Z9  X10
Y10 Z10  X11 Y11 Z11  X12 Y12 Z12  X13 Y13 Z13  X14 Y14 Z14  X15 Y15 Z15  X16 Y16 Z16  X17 Y17
Z17  X18 Y18 Z18  X19 Y19 Z19  X20 Y20 Z20  X21 Y21 Z21  X22 Y22 Z22  X23 Y23 Z23  X24 Y24 Z24
X25 Y25 Z25  X26 Y26 Z26  X27 Y27 Z27  X28 Y28 Z28  X29 Y29 Z29  X30 Y30 Z30  X31 Y31 Z31

1       0.033333  131.1524  163.0935   71.6937  124.6823   71.3264   76.5239  121.2981  102.4109
75.0063  149.2998  181.1166  132.8877  134.7889  192.6069  116.5414  122.5840  190.7085   89.0098
149.5519  239.0187  104.2023  131.1980  121.0528  101.0101  160.2013   95.1255  134.9703  137.7529
47.9096   96.5317  128.6441   74.4367   94.9948  137.1264   85.1827  106.0074  139.9505  145.9818
111.2604  150.6080  130.9824  134.3215  204.5324   77.3407  144.1150  125.4540   99.9015   93.0702
111.0689  124.5950   72.5558  158.4736  184.0863   17.1546  139.9273  193.7812   32.1008  125.5029
190.9568   57.3250  148.4869  240.4077   55.7514  130.7053  120.1502   48.0696  161.9311   89.6753
21.3688  140.7662   47.0322   54.0298  129.6595   71.2903   57.0318  137.0737   83.6635   44.6577
143.2580  143.2805   37.2008  157.8477  128.4386   17.5263  200.7793   81.9230    9.8090  125.3299
99.0036  56.1530 131.1743 209.7377  71.5724
2       0.066667  131.2017  163.1528   71.7291  124.6200   71.3926   76.5561  121.2042  102.5331
75.0017  149.3542  181.1748  132.9477  134.8884  192.6407  116.6287  122.3916  190.7327   88.9545
149.5145  239.0789  104.2528  131.3174  121.1239  101.0545  160.1843   95.1614  135.0377  137.7782
47.9176   96.5733  128.6558   74.4725   95.0168  136.8899   85.2499  105.9204  139.9980  146.0101
111.2198  150.5929  131.0192  134.3598  204.5233   77.3376  144.1167  125.4860  100.0971   93.2541
111.0917  124.6370   72.5714  158.4663  184.0818   17.2422  139.9324  193.8225   32.1557  125.5762
190.9936   57.3842  148.5566  240.4861   55.8336  130.7453  120.2003   48.1136  161.9240   89.7217
21.3911  140.4146   47.1341   53.9443  129.6737   71.3267   57.0585  137.0995   83.7806   44.6918
143.3501  143.4694   37.4252  157.7448  128.5400   17.5307  200.9130   81.9462    9.9033  125.3576
99.0523  56.1991 131.2355 209.8056  71.6000
```

2.10 The CSM file format

2.10.1 General description

The CSM format is an optical tracking format that is used by Character Studio (an animation and skinning plug-in for 3D Studio MAX) for importing marker data. For CSM files to be compatible with Character Studio, they must use names that match the Character Studio setup and they must have an appropriate number of markers in specific locations on the actor. The CSM format itself is capable of holding any kind of marker data, but by being in the CSM (Character Studio Marker) format it is assumed that it also adheres to the name and marker configuration required by Character Studio. The motion data are given in the implicit order of the declaration of the joints.

2.10.2 Example of a CSM file

```
$comments
This is a Character Studio 2.0 CSM File

$firstframe 1

$lastframe 3

$spinelinks 3

$rate 60

$order
C7 CLAV LANK LBHD LBWT LELB LFHD LFIN LFWT LKNE LSHO LTOE LUPA LWRA LWRB RANK RBHD RBWT RELB
RFHD RFIN RFWT RKNE RSHO RTHI RTOE RWRA RWRB STRN T10

$points
1       980.6   -2365.8 1541.3  1030.6  -2239.9 1492.1  967.3   -2427.5 181.8   936.2   -2330.1
        1672.9  939.4   -2359.1 1109.3  782.9   -2263.3 1175.7  959.0   -2196.9 1753.8  762.3   -
2135.1  862.6   949.3   -2187.2 1082.1  934.2   -2343.2 583.1   870.4   -2246.7 1525.4  1031.7  -
2339.7  83.1    814.6   -2218.7 1318.8  799.4   -2132.3 993.7   729.2   -2230.2 966.5   1110.1  -
2060.1  197.3   1066.8  -2351.6 1670.5  1114.0  -2391.4 1116.4  1290.5  -2574.6 1396.5  1105.7  -
2231.2  1740.1  1217.5  -2369.8 1149.5  1159.8  -2233.6 1093.8  1138.8  -2119.2 605.9   1136.6  -
2342.8  1564.9  1081.3  -2126.7 775.0   1065.2  -1944.5 112.2   1276.8  -2398.1 1271.2  1294.0  -
2490.1  1183.6  1044.2  -2199.9 1395.9  988.1   -2404.3 1417.4
```

```
2       979.8   -2358.9 1540.1  1029.4  -2232.4 1489.5  966.5   -2426.7 181.8   936.2   -2322.0
        1671.1  940.7   -2351.6 1107.3  783.9   -2249.8 1173.7  959.4   -2189.0 1752.6  767.9   -
2113.2  863.4   948.9   -2177.9 1080.7  934.4   -2338.9 583.3   871.7   -2242.9 1523.4  1031.7  -
2339.5  82.9    814.6   -2207.6 1317.8  803.3   -2113.8 994.5   732.4   -2209.8 964.1   1108.3  -
2039.1  197.5   1067.2  -2343.2 1669.3  1115.0  -2382.1 1115.0  1292.6  -2563.1 1400.3  1106.5  -
2223.7  1738.5  1212.0  -2367.2 1155.5  1162.2  -2222.1 1093.8  1140.2  -2119.4 601.9   1135.4  -
2334.3  1563.9  1079.9  -2109.1 774.4   1064.5  -1921.3 116.0   1278.4  -2396.2 1273.8  1284.5  -
2483.4  1186.2  1043.6  -2192.5 1394.0  987.5   -2397.7 1415.2
```

2.11 The National Institute of Health C3D file format

2.11.1 General description

Many of the motion capture companies are often linked to the biomechanics research. The systems are then used to assess the performances of an athlete or the needs of a physically handicapped person. The needs of researchers, often supplied by more than one society, lead to the definition of a common format, the C3D format. This format has been built following this philosophy, so it has tried to carry the most complete amount of information useful for the biomechanics research. The features below was considered as necessities:

- The possibility to stock analogical data (i.e. directly coming from the measure instrument) and three-dimensional data (obtained by the information processing).
- The possibility to stock information on the material, which have been used (position marker, force captors), on the recording process (sampling rate, date, type of examination…), on the subject itself (name, age, physical parameters…).
- The possibility to add new data to the ones already recorded.
- The file was a binary file unlike most of motion capture file format, which often are ASCII files.

Therefore, the C3D format is a standard, which can be put aside because of its ability to stock a large amount of data types. This format is used by Adtech, ANZ, BTIS, C-Motion, Charnwood, Innovision Systems, Kaydera Inc., Lambsoft, Motion Analysis Corporation, Motion Lab Systems, National Institutes of Health, Oxford Metrics, Peak Performance Technologies, Qualisys, RUN technologies, Vicon Motion Systems…
The C3D format contains to principal data type: position data (3D coordinates after processing) and what the C3D calls analogical data, which actually are the raw data. The 3D coordinates are stocked into successive samples of XYZ coordinates with information on accuracy and specification of the captors. Each sample of a numerical data coming from a processing is linked to its source. So it becomes very easy to correct some numerical values or to access to the source data.

2.11.2 The header section

The header of the file is typically a 256 words of 2 bytes structure:

Byte	Typical value	Description
1	0x5002 hex	Byte 1 : number of the first recording parameter (typically 02) Byte 2 : keyword (50 hex) indicating a
2	-	Number of points recorded
3	-	Number of analogical chanels by sample.
4	1	Number of the first sample
5	-	Number of the last sample
7-8	-	Scale factor (a negative factor means no scale)
9	-	DATA_STAKT : number indicating the start of the recording
10	-	Number of analogical sample in a sampling interval for the 3D data
11-12	-	Sampling rate for the 3D data

2.11.3 Parameter record

The location of the first parameter record is given inside the header. The first 4 bytes of the parameter record contain the following header:

Byte	Typical value	Description
1	0x00 hex	parameter file indicator
2	0x00 hex	parameter file indicator
3	-	Number of parameter record that follow
		83 + processor type
4	85	type 1 - PC DOS
		type 2 = DEC (VAX, PDP-11)
		type 3 = MIPS (SGI, SUN)

The parameters are stock just after the header. They are organized into groups, each parameter belong to only on group. Each parameter and each group are defined by the same header format.
18 events can also be defined in a C3D file. They are instances used to define, for instance, the beginning or the end of a ground contact. The event units can content one to four labels, its execution moment relatively to the first sample, etc.

2.11.4 Recording of the data

Then follow the parameters themselves. They can appear inside the file whatever the order can be. The order just has to follow the one defined by the headers. The 3D data and the analogical data are written in the file as sequential fields:

Analogical data for the field number 1
Analogical data for the field number 2
3D data for the field number 2
Analogical data for the field number N
3D data for the field number N

2.11.5 3D position data format

- Integer data: The data are stored in 4 words, the X, Y and Z coordinates in the first 3 words. The fourth word contains information on the captor used for the measure and on the measure error after interpolation.
- Floating point data: The data are also stored in 4 words, which are dedicated to the same things than in the integer case.

2.12 A standardisation of the humanoid structures: the MPEG-4 standard

The amount of proposed format, the needs of the users, the future development associated with the object-oriented way of coding in the multimedia community lead to the definition of a new file standard for the following application:
- Digital television
- Interactive graphics applications (synthetic content)
- Interactive multimedia (World Wide Web, distribution of and access to content)

Inside this standard, a common structure for modelling the human skeleton and animating it has been proposed.
The data transmission is made in two steps:

- Humanoid structure initialisation before its animation. The data transmitted here will remain unchanged during the running of the application: structure of the skeleton, information on the surface texture, etc. The skeleton structure is defined as follow:
    - 6 degrees of freedom for the whole animated form
    - 62 degrees of freedom for the own movement of each skeleton.
    - The hand structure is defined apart and has 25 degrees of freedom. This is an optional structure, which can be omitted for low data rate transmission.

  The skeleton is defined with 59 segments or 29 segments without the hands.
- Update of the structure during the animation.

Many optimisations are possible through this format like the reduction of the number of byte used for each degree of freedom, the maximum being 4 bytes. Priority rules for data transmission have been also proposed. In this case the most important parts of the body should be actualized in priority. So, the priority order proposed is (from the most important to the less important):

- The global position and orientation of the skeleton
- The orientation angle having a great visual importance for the animation: shoulders, knee, elbow, leg, and some peculiar joins of the hand.
- The other values

2.13 Conclusion

Most of the file formats presented are largely used in today motion capture based applications. For instance, the most popular animation tools (3DS Max, Maya, Lightwave…) all support the importation of BVH files. The case is the same for C3D file, which is largely a standard in biomechanics research. Some files are more oriented for one system like the CSM file of Character Studio.
The main limitations for real time uses is that most of the available formats presented her are ASCII formats (except C3D file format). It can be a limitation for applications that need to run in real-time and that need to consume data, which is stocked in a hard drive. If we look at the case of multimedia for instance, sound or movie are coded into binary files what allows the reader software a fast access to the data.

In addition, most of those file formats are dependant of the application:

- The C3D is really biomechanical data centred. The presence of analogical data (raw data in C3D language), the link with the 3D data, information about captors, 3D orientation (what do we do with 2D or 1D data?) , are features that can't really be considered as generic. Nevertheless this is the only binary format, it is widely spread in biomechanical research community, and it is also supplied by the most known motion capture systems. Kaydera's FILMBOX gesture editor also allow import/export of C3D files what link C3D file with the Computer Graphics community in a certain sens. Moreover it is constructed to be flexible (ability to add new data structure) and very well documented (the web site is really well shaped and very accurate).
- BVH, HTR, ASF/AMC, MNM files are very skeleton dependant what make their use very oriented to character animation.
- AOA is quite less oriented but only considered 3D coordinate.
- TCR could allow taking into account data of different dimensionalities but is sadly an ASCII file.
- BRD is oriented by the magnetic technology of Ascencion Technology Corporation.
- CSM is dedicated to the Character Studio software

2.14 Sources and bibliography

Motion Capture Systems

• Optical systems :

  - Adaptive Optics Associated : http://www.aoainc.com
  - Mikromak GmbH : http://www.mikromak.com
  - Motion Analysis Corporation : http://www.motionanalysis.com
  - Vicon Motion Systems : http://www.vicon.com
- Magnetic systems :
  - Ascencion Technology Corporation : http://www.ascension-tech.com
  - Euclid Research : http://www.euclidres.com
  - Data Glove iReality : http://www.genreality.com
  - Polhemus : http://www.polhemus.com

File formats

[AOA] http://www.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/AOA.html
[ASF] http://www.darwin3d.com/gamedev/acclaim.zip
[BVH] http://www.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/BVH.html
[BRD]http://www.dcs.shef.ac.uk/~mikem/fileformats/brd.html
[C3D] http://www.c3d.org
[CSM]http://www.dcs.shef.ac.uk/~mikem/fileformats/csm.html
[HTR] http://www.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/HTR.html

[AP89] Apple Computer, Inc. « Audio Interchange File Format : « AIFF ». A standard for sampled sound files Version 1.3 ». January 4, 1989.

General documents

[HFP00] Herda, Fua, Plänkers, Boulic, Thalmann « Skeleton-Based Motion Capture for Robust Reconstruction of Human Motion », EPFL, Lausanne, Switzerland, janvier 2000

[ME99] Menache. « Understanding Motion Capture for computer animation and video games », Morgan Kauffmann Ed., August 1999.

[MM00] Meredith, Maddock. « Motion capture file formats explained. » Département of computer Science, University of Sheffield, 2000

[MOR85] Morrison. « EA IFF 85 - Standard for interchange format files » document de spécification Electronic Arts, janvier 1985

[SW84] M. Smyth, A. Wing. « The psychology of Human movement. » Académie Press, 1984

[Vba00] Susan Van Baerle. Motion Editing : Principles and Practice. 2000 Game Development Conference Proceedings. March 20-24, 2000. San Jose, CA.

3 Gesture Studies in the interactive simulation context of INPG

Annie Luciani, Claude Cadoz, Nicolas Castagné, Damien Couroussé, Jean-Loup Florens
INPG, October 2005

Gesture studies is a basic scientific research direction at ICA – ACROE.
The central paradigm is the "Instrumental Interaction" and the "Instrumental Communication", that is a generalization of what it happens in musical performance.
The implementation of such paradigm is the context of the computer tools supposes :
• The development of adequat gestural devices
• The development of a generic real time simulator of physically-based models
• All the contextual tools necessary to design physically-based models able to produce sounds and visual motions under the gestural control through gestural devices
• All the contextual tools necessary to manage the signals

3.1 Gestural devices

Gesture devices concept has been driven by analysis of gestures, including functional analysis (fonctions of the gesture), structural analysis (structure of gestures), quantitative analysis (spatial and temporal ranges and resolutions)) and type of data.

Functional analysis leads to develop a functional categorization in three basic functions:
• Ergotic function: to be implemented in a computer context, this function needs the implementation of sensors, actuators and closed-loop interaction between both.
• Semiotic function: to be implemented in a computer context, this function needs only sensors
• Epistemic function: to be implemented in a computer context, this function needs only sensors

Structural analysis leads to put the accent on one properties of gesture that is it versatility.
Conversely than sounds and images, the organization of gestures is highly variables and needs to be described a high flexibility in the organization of the instruments and of the produces signals. From sliders, data glove, data to 1D, 2D, 3D, 3D set of force feedback devices. The instruments (hardware and software) have to be highly reconfigurable.

Quantitative data and type of variables
Gestures have quantitative properties that are different than sounds and visual motion. They require to be sampled form some Hz to some KHz, according to the type of variables.
In case of "free gesture, (semiotic function and sensors only) as well epistemic function (actuators only), the frequency bandwidth could be of some Hz (50-400Hz)
In case of "ergotic interaction" (coupled actuators – sensors), the frequency bandwidth could be of some Hz (400Hz – 4KHz). In this case, two variables are circulating in the system: extensives variables (as positions) and intensive variables (as forces).

Consequently, instruments for gesture are developped following two tracks :
1. Panoply of complementary devices:
• Panoply of force feedback devices, each of them corresponding to a basic case of morphology and being reconfigurable,
• Panoply of non force feedback systems as pure sensors.
2. Computer hardware and software tools to manage such versatility.
The concept of "real-virtual connector" has been developed to dispatch each input and/or output on the simulator inputs / outputs.
The number of axis is a – priori not limited. It depends only on the number of tracks of the AD – DA converters. Currently, it is of 16 inputs and outputs tracks, freely reconfigurable.

3.2 Generic real time simulator of physically-based particle models

Physical models are the most generic to guaranty the use of ergotic interaction.
Several types of physical models exist. The choice made by INPG as considered as the most adapted to manage external device, physical models for sounds, large physical models for visual motions is the basic particle-interaction formalism.
By simulating physically-based particle models, the simulator combine physical inputs and outputs for gestural devices, deformations computation for the sounds and deformations and displacements for the visual motions.
The basic data managed by such simulator are:
• The frequency sampling rates for each type of physical phenomena (gestural, acoustical and visual). This leads to a multifrequency simulation. The basic choice is of three frequency sampling rates
• Forces and positions (no angles and no momentum).
• Forces and positions can be 0D (as in the sound control and synthesis for deformations), 1D, 2D or 3D space.

3.3 Interactive modeler to design physically-based particle models

Interactive design tools have been implemented to design physically-based particle models and to manage them.
• Interactive design for acoustical deformation
• Interactive design for 3D movements and objects
• Interactive connection with the gestural devices
• Interactive sound and images rendering.
• Library of basic modules
That is the CORDIS-ANIMA modeling system.

3.4 Tools necessary to manage the signals

Surrounding the modeler-simulator systems and working with them, tools are developed to manage:
• the modeling process: library of models, of parameters, etc...
• the observation of the simulation: positions and forces sensors), file formats to store the behaviors (gestures, sounds, visual motion) and of the evolution of the virtual object (parameter evolution).

3.5 Complete architecture

The complete architecture is drawn in the following figure.

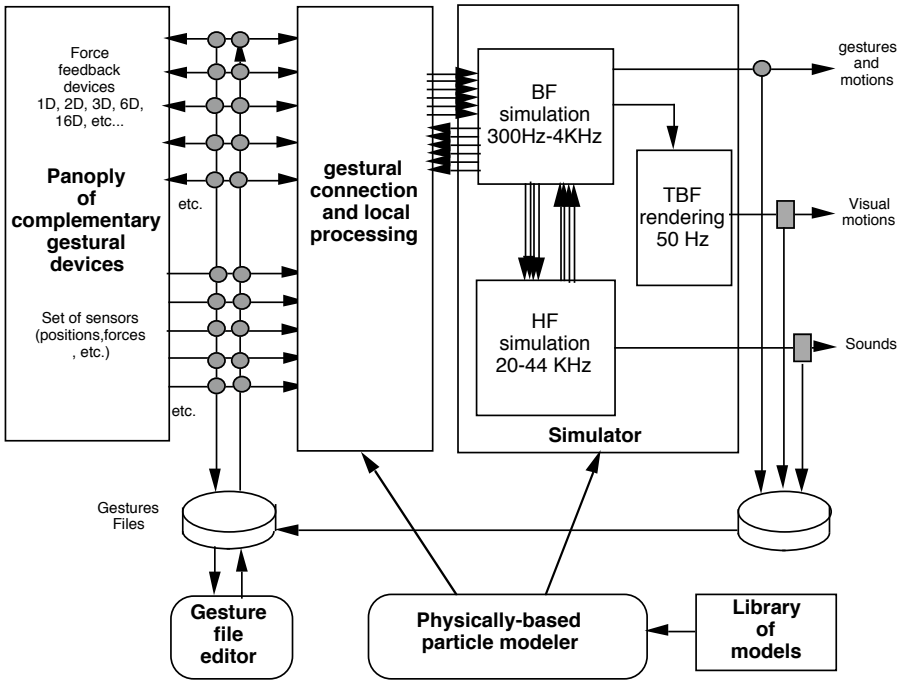
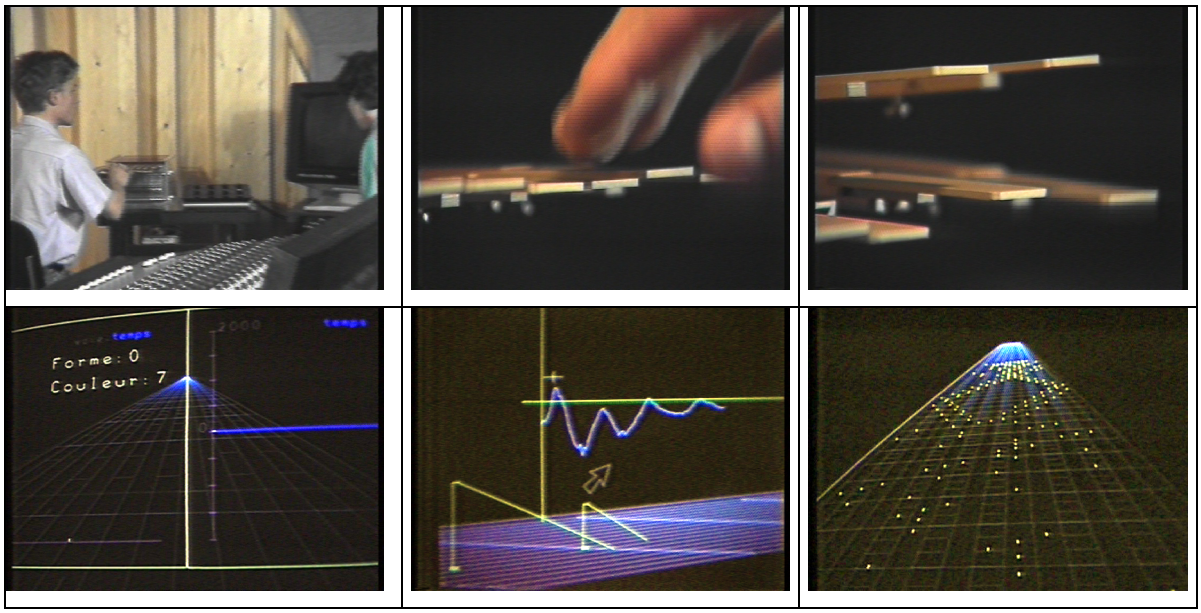


Figure 1. General INPG architecture for the implementation of instrumental multisensorial interaction and communication

3.6 About gestures

Several previous works have been produced in gesture storing, coding, representation and processing in instrumental interaction. The following images illustrate previous work on these topics.



(a) and (b) Capturing gestures from force feedback keyboard. (c) Replaying stored gestures. (d) Gesture representation framework. (e) Gesture analysis. (f) Gesture representation (the 3D score).

3.7 INPG References on gesture studies

CADOZ (C) & RAMSTEIN (C), "Capture, Representation and Composition of the Instrumental Gesture", International Computer Music Conference - Glasgow 1990.

RAMSTEIN (C), "Analyse, représentation et traitement du geste instrumental", Thèse pour obtenir le grade de Docteur de l'I.N.P.G., Spécialité Informatique, décembre 1991

GIBET (S) & FLORENS (JL), "Instrumental gesture modeling by identification with time varying mechanical models", International Computer Music Conference - Cologne 1988.

GIBET (S), "Codage, représentation et traitement du geste instrumental", Thèse de Docteur de l'INPG, Spécialité Electronique, - Grenoble 1987

CADOZ (C), LISOWSKI (L) & FLORENS (JL), "Modular Feedback Keyboard", International Computer Music Conference - Glasgow 1990.

CADOZ (C), LISOWSKI (L), FLORENS (JL). Clavier rétroactif modulaire et actionneur modulaire plat - Brevet Français. Déposé le 13 / 10 / 88. France N°88 14064, US n° 07/420 242 - Europe n° 0 365 441 - Titulaire : ACROE

LISOWSKI (L) & CADOZ (C), "Conception, optimization and realisation of extra flat DC linear motors", Proc. of 4th International Conference on Electrical Machines and Drives - IEEE - Sept 89

LUCIANI (A), CADOZ (C), FLORENS (JL), "The CRM device : a force feedback gestural transducer to real-time computer animation", Displays , Vol. 15 Number 3 - 1994 - Butterworth-Heinemann, Oxford OX2 8DP UK, pp. 149-155.

Evrard, Matthieu, *Le canal gestuel : contrôle d'objets multisensoriels simulés par modèle physique*, Application : spécification et implémentation des entrées et sorties geste dans MIMESIS, research Master report, 2004/06, ACROE-ICA, Grenoble (2004) 82 pages

Couroussé, D. La capture de mouvement: format des fichiers d'échange et procédés de traitement. Master Recherche 2. Signal Image Parole. Grenoble, 2002.

4 A basic gesture coding format for VR multisensory applications

Annie Luciani, Claude Cadoz, Nicolas Castagné, Matthieu Evrard, Damien Couroussé, Jean-Loup Florens

(Paper proposed to GRAPP 2005 Conference)

4.1 INTRODUCTION

Gesture, motricity, haptic perception are deciding factors in the way we act on and we perceive our environment. During the last few years, many research centers have been working on gesture control, movement analysis or synthesis, in domains as various as surgery, aeronautics, multimedia, artistic creation and generally speaking in every interactive systems. Nowadays, more and more, applications are developed in opened and versatile contexts. More and more, 3D images synthesis are made in a context on which they have to communicate or they have to be correlated to sound synthesis processes, as in interactive multimedia installations, and most of them wish to share similar processes of gestural control.

Thus, we observe that there is a need to have at disposal of developers and users in Computer graphics, Computer Music and sounds, interactive systems and versatile VR platforms, a low level format allowing the exchange of such motion control data between different contexts of multimodal interactions, multimodal synthesis and/or simulations. Several formats already exist in motion control data, but today, no one really have been considered as to be shared and to be used in other applications than of its origin. We want to point out here the need of establishing a minimal common description of gesture or motion data, able to code all their necessary feature independently of the context in which they have been produced and they will be used.

4.2 Actions, MOVEMENTS, GESTURES

4.2.1 Action / Movement / Signal

Therefore, in front of the diversity of ambiguous terms, let's start by some definitions.

One can affirm without too much possible contest, that *movement* (or *motion*) is the moving in space of a part or of the totality of a system (a human being, an object…). It should be distinguished from *action*, which is the result of the movement. Following [SW84], action is the task result achieved or to be achieved "to drink a glass of water". The action can be performed by means of several different movements. And finally the movement can be characterized by properties, such as soft, vivid, etc.

If actions can be described at a higher symbolic level by language for example of by means of event-based representations, movements as a spatio-temporal phenomena need to be represented as temporal evolution, i.e. by signals. Quality of motions addresses whether some properties that are readable on the temporal signal as velocities, acceleration, transients, but also properties that are not explicitly readable as softness, heaviness, etc. and that could be either represented in the system that produces the movement or extracted from the signal.

Several works are related to action encoding and description. Some of them are based on languages, as developed in temporal logic programming or in AI domain. Differently than action, movement observation and description supposes to hold up the temporal evolution of the concerned variables (positions, shapes, colors, sounds parameters, etc...) explicitly in time. They have to be based to temporal signal representations. From signals, motion processing may aim to extract either quality of the motion (expressive gestures understood as extraction of parameters of expressiveness in acquired or observed motions) and/or the high level event it performed, (action recognition or gesture recognition understood as action inference from the motion performance).

4.2.2 Movement and Gesture

Another evidence concerning Movement or Motion, is that it refers to the evolution produced by a physical system whatever it is, real human body, real mechanical objects equipped with sensors, virtual objects (avatars or objects), etc. We can speak about the movement of the human body, or the movement of a leaf or the movement of sounding source, etc. Movement is here considered as an output of an evolving system.
Differently, when one speaks about "gesture", beyond the fact that it is usually related to human movements, it addresses the property that such movements can be used as "a command" to control another system: we are *applying* a gesture on an object to control the motion of such object. Thus "gesture" is a motion used as an input signal.
Under this enlightening, motions or gestures signals could be consider in a unified way as "temporal signals" (Figure 1):
• as outputs of an evolving system (for example any mechanical system),
• as inputs able to be used as command signals of another evolving dynamic system[1].

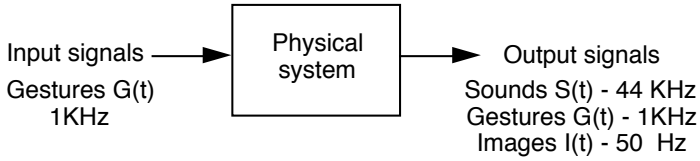


Figure 1: Gestures as input and outputs of physical systems

Consequently, more than discuss about the differences between motions and gestures, their similarity as being indifferently outputs or inputs of evolving systems leads to clarify objectively the differences between in one hand these type of signals and in other hand visual and acoustic signals.
We will called in the following these signals, whatever their origin are (object or human motions), "gestures signals", by putting the accent on the fact that we will be able to use them, not only as outputs to be processed, but also as inputs. In this vision, we could speak on the gesture of a leaf, meaning that the motion of the leaf could be used as an input to control another evolving system (gestural, visual or acoustical).

4.2.3 Gesture and Control

As presented in the previous paragraph, gestures can be inputs (as well as outputs) of evolving systems, one of these systems being able the human body.
Consequently, they can be common to several and various situations. They can be considered as the common features of several and various situations, and they can be a privileged way for their cooperation: animation control, animation performance, musical control and performance, multisensory control and performance such as in VR or in multimedia arts.

We illustrate some basic cases of gesture signals and gestural control in figure 2. The figure illustrate some basic cases. The grey circles indicate at what place we can consider signals as being homogeneous as control gestures. The grey squares show what it is not usually being considered as gestural signals.
In the case of gestural control of real or digital musical instruments, the gesture signals could be these sensed directly on the human body (fingers, hands, arms, whole body) (figure 2, grey circles 1 and 2) or the motions produced by an object manipulated by humans (hammer, sliders, sticks, etc...) (figure 2, grey circle 3).

In the case of gestural control of 2D or 3D visual motions, the gesture signals could be these sensed directly on the human body (figure 2, grey circles 4 and 5), or the motions produced by an object

[1] The concept of "command applied to a system" supposes that this system introduces necessarily a non-linear transformation. Differently, a signal processing system does not introduce such type of non-linearity.

manipulated by humans (Sticks, force feedback devices, etc.) (figure 2, grey circles 6 and 7). Conversely than in musical instruments, in which the output "sound" is not usually considered as a "gesture", the 3D motions provided by a 3D virtual objects (Figure 3, grey circle 7) are totally of the same nature of the 3D motions produced by the human body in body motion capture, and as them, they can be applied as input control of another system (3D object or musical instrument) (Figure 2, grey circles 4, 5, 6). Consequently, a 3D virtual object produces signals that can be considered as homogeneous of input control gestures.
In the case of gestures that produce formal and symbolic data as by acting on a computer keyboard, the gestural signals (Figure 3, grey circles 8 and 9) are transformed in outputs (words, sentences) that are not usually considered as gestural signals or data.

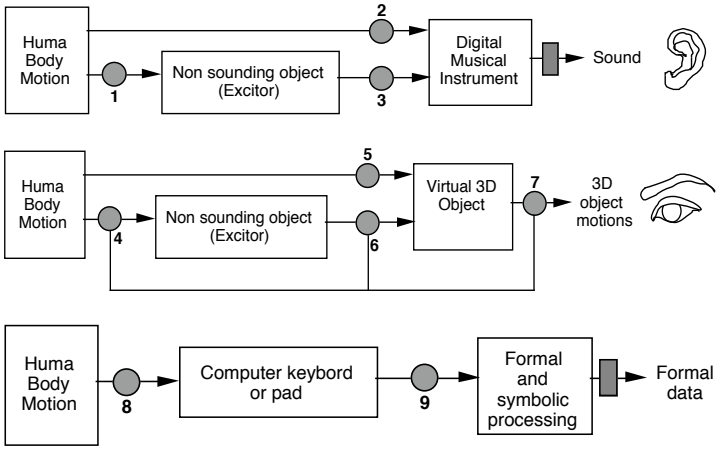


Figure 2: Various cases of gestural control:
Up: Gestural control of digital musical instrument - Middle: Gestural control of a 3D virtual object - Down: Gestural control of formal and symbolic data
The grey circles (resp. the grey squares) represent signals that are homogeneous (resp. non homogeneous) to gesture signals.

4.3 Characterization of gestural signals

Such signals, whatever they are (objects or human motion signals), used as outputs or inputs of evolving systems, present particular properties that allow to distinguish them among others temporal signals (aero-acoustical signals or visual signals), and that lead to the definition of a own format.

4.3.1 Morphological versatility

One of the first evidence is the morphological versatility of gestures. If images and sounds can be displayed in predefined environments (displays of a given resolution or 3D Caves of a given size for the images, Stereo or quadriphonic rendering for the sounds), the structure and the morphology of the gestures are always changing according to the tasks and the manipulated tools. To take into account such versatility, we propose to structure gestures according two complementary features: geometrical and structural dimensionalities.

Geometrical dimensionality refers to the dimensionality of the space in which the gesture is evolving.
Piano or Clarinet keys are pushed or closed according to a 1D finger motion. The control of the sound, and generally the parameter tuning (for example the value of an elasticity or the amplitude of a deformation), can be made through devices that evolve in a 1D non oriented space (set of sliders, set of knobs, etc...) and that can be described by a scalar or set of scalars.
In cartoon animation or in scrap-paper animation or animated painting under the camera, the space is reduced to a plane. The gestures and the motions evolve in a 2D space (Figure 3, g), described on two orthogonal oriented axis.

When we manipulate an object (real or virtual), the dimensionality of the space is obviously 3D, i.e. the descriptions needing three orthogonal oriented axis (figure 3, e, f, h).
This means that the geometrical dimensionality of a gesture can vary a lot: from a pure scalar or a set of pure scalars as in manipulation of sets of sliders or keys (figure 3c and 3d), to geometrical 1D (figure 3a and 3b), 2D (figure 3g), 3D (Figure 3f), 6D oriented vectors and/or tensors (Figure 3h).

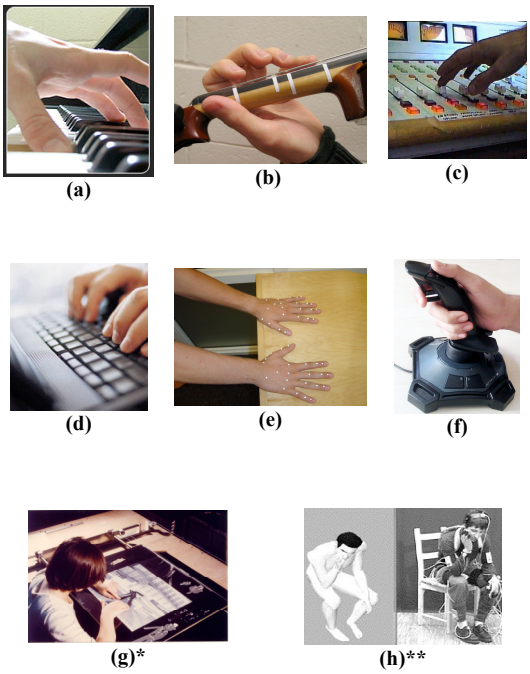


Figure 3: Versatility of the gestural morphology
(*) with the courtesy of A. Luciani
(**) EPFL image (VR Lab)

For a given geometrical dimensionality, the number of degrees of freedom (DoF) can vary. We call the axis of variation the structural dimensionality.
When we are acting on a keyboard of n keys (a piano keyboard, a computer keyboard, a set of buttons), the performed gesture can be considered in two ways - and similarly the n-keys produced signals:

- whether as n independent systems, gestures and signals.
- Whether as n degrees of freedom of 1D motions.

In the human body motion, the geometrical dimensionality is 3 (all the motions of the body can be described in a 3D oriented Euclidian space) and the number of axis of variability (the number of degrees of freedom) is more than 200 in the real body and is of sixteen if the motion is sensed by a motion capture systems with 16 sensors.
In the modelling of a bowed string, the two dimensions of the deformations are usually decoupled, and the system can be considered as two superposed 1D gestures (to press the string, to bow the string), thus as a 2DoF of 1D system.

4.3.2 Quantitative Ranges

Beside the two previous qualitative properties (number of space axis and number of DoF), gestural signals are characterized by specific spatial and temporal quantitative features.
A first quantitative feature that allow to distinguish gestures (and control motions) signals among others (aero-mechanical signals, visual motions) is the frequency bandwidth ranges (Figure 6):

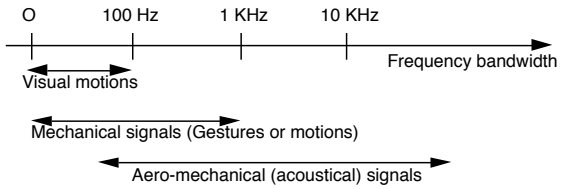


Figure 4. Temporal range of sensory signals

Although the three zones of the figure 4 are overlapped, they point out a useful categorization: visualizing motions requires a sampling rate up to 100 Hz: manipulating an object with force feedback requires sampling rate from some Hz to some KHz; recording sounds requires sampling rate from 20KHz to 40 KHz. The gestures signals are at the middle range.

Conversely, the audio signals are small deformations, centered to 0 and less than some millimeters. The mechanical and visual motions are usually non-centered large deformations and displacements (from centimeters to meters).

Consequently:

(1) To render compatible computer graphics with other fields as in computer multisensory or multimodal contexts (for example, physically-based simulation, VR models including sounds, etc.), we have to manage three range of frequency sampling rate: about 25-100 Hz for the visualization process (VLF: very low frequency), about 1-4KHz (LF, Low frequency) for the motions computations processes, about some 20 KHz – 40 KHz for audio rendering (HF: High frequency).

(2) The properties of the gesture and motions signals position them at the middle range: Spatially, it is similar to visual motion but it needs higher frequency rate. Temporally, it needs lower frequency rate than the sound but it runs at higher non-centered spatial range.

(3) This leads such signals to be the reference signals in multisensory VR simulation platforms. Visual Motions can be produced from them by very simple under-sampling. Sounds will be produced by a non-linear control-simulation process with transform BF gestural command signals in HF signals as in sounding sources or musical instruments.

4.3.3 Type of variables

As motions and gestures are produced by physical systems, and used to control physical systems the data could be of two different types: Extensive variables, as variables that derivates from spatial information (positions, velocities), and intensives variables as forces, torques, etc. Conversely, we may notice that visual data and acoustical are only of extensive ones (positions and/or displacements).
Indeed, in natural situations, when gestures are used in an object manipulation, physical energy is exchanged between the two interacting bodies (for example object and human). Such interactive dynamic systems have to be represented whether by explicit correlation between extensive and intensive variables as in Newtonian formalism or by implicit correlation as in energy formalisms.
After recording data from such dynamic system, and in absence of model of the system, we need to have all the extensive variables as well the intensive ones to reconstruct the system. This means that the data to be stored could be heterogeneous, extensive and/or intensive.
When the data produced by a system (real or virtual) are reused after recording, it is easy to reintroduced such data in every system (Computer Graphics models, VR platforms, Physically-based simulators) as “generators” of positions, “generators” of forces, etc.

4.4 Analysis of existing motion file formats

As said before, several motion file format have been developed [ME99] [MM00], no one really have been considered as to be shared and to be used in other applications than of its origin. In this paragraph, we position the most known of them faced to the properties of the gesture signals we elicited in the

previous paragraphs: versatility of the geometrical and structural dimensionality, spatial and temporal requirements, multiplicity of the types of data.

Most of file formats containing information about gestural signal come from motion capture community (motion capture firms or users of this technology like animation software for instance):
• The BVA, the BVH, the ASK and the SDL file formats [BVH] were developed by Biovision, a firm providing optical motion capture systems.
• The HTR and TRC file formats [HTR] was developed by the firm Motion Analysis.
• The AOA file format [AOL] was developed by Adaptive Optics, a firm that creates hardware support for motion capture.
• The ASF/AMC file formats [ASF], first developed by Acclaim, a video game firm, belong to the public domain are now used by Oxford Metrics (Vicon motion capture Systems).
• The BRD file format [BDR] was defined by Lambsoft, for the "Flock of Birds" magnetic motion capture system.
• The CSM [CSM] file format of 3D Studio Max and Character Studio is used to describe a character movement.
• The C3D file format of the National Institute of Health was developed to exchange data about athlete or physically handicapped person movements between every actors of a research project. It has been adopted by firm like Oxford metrics (Vicon Motion System) for their marker data.

Context dependence
Most of those formats contain data dealing with the situation in which the gestural signals have been produced. Such data can be separate into two types: (1) Motion data that infers on the signal data, and (2) Context data that gives information about the context of the data acquisition.
The motion data are necessary for reconstructing the gestural signals which are usually given in relative coordinates. Usually, this type of data is organized by a hierarchy, inherited of the hierarchy in the cinematic representation of 3D tree-hierarchical object. Typically, describing the movement of a humanoïd, according to the tree-structure of the cinematic chain of the body has made its proof in domains such as motion planning or character animation [HFP00]. However, such organization is adapted to specific structure of object as human body, rigid-articulated robot, etc.) and it cannot be useful for others motions as the motions of deformable non tree-articulated objects and mechanisms. Such hierarchy information can be found in BVH, ASF/AMC or HTR file formats. In the CSM file format, the motion is basically represented by the movement of a set of 3D points in absolute coordinate but, for CSM files to be compatible with Character Studio, names of those points must match the Character Studio setup and their number is determined. The only format in which the motion data are free of hierarchy structure is C3D format. It is a basic format composed of both the raw camera information during the data acquisition phase of laboratory test and other motion data derived from them by a reconstruction process. The context data are typically data concerning the capture system parameters (number of sensors, captors used to determine the movement at a certain moment, sensitivity of sensors, etc.). The C3D file format is the most representative example. Each set of motion data or measurements data is associated to the description of the measurement context that can be very complex and that allow this format to be very general and adapted to an experimental context.

Predefined dimensionality
In the usual application of these formats, the dimensionality of the data is fixed: 3D points in AOA or CSM, “7D points” in BRD (3D coordinate + 4 numbers for a quaternion). In BVH, ASF/AMC or HTR file format, the dimension of each join can be declared but it is usually omitted due to the hierarchical description of the movement, the omission of a dimension corresponding to a cinematic constraint that relies to the hierarchy.
C3D format stores conventional 3D positional information but any data of the measurement sensors (hearth rate, EMG, etc.), and thus it could be extended to 2D data. However, this extension is limited by very specific conditions of measurements: 2D information can be recorded by specifying one of the coordinates of the point and calculating the other two from the observer data. This allows the C3D file format to be used by systems that support single camera measurements – thus a camera might provide positional information for the Y and Z planes while constraining the X motion within a single fixed

plane". Thus, C3D file format can be used to describe a set of 1D or 2D movement but it is not an inherent property and to do so the producer and the consumer of such a file format have to agree on the dimensionality of the set of point.
The predefinition of the dimensionality of the stored data becomes a real problem when we will have to consider coexistence of different dimensionality, as we show that the versatility of the dimensionality is a core property of the gestures.

Type of variable and Data format coding
Except C3D file, every file is written in ASCII. This is a real problem for some applications as those based on force-feedback gestural interaction that needs the gestural data be sampled at some Khz. Thus, it becomes preferable to encode such an amount of data in a binary format (like for music or movies) for real-time data processing.
Mainly designed to store 3D position-like data, none of those file formats plan the possibility of encoding several type of variables. We showed previously that, as being representative of an interaction process with a manipulated object, the data needed to represent gestures could be extensive variables (as positions) and/or intensive variables (as forces).

In the following, we will try to determine the minimal set of information that should be include in a basic generic file encoding gesture.

4.5 Requirements for a generic low level gesture file format

On the basis of the previous analysis of gesture, in which we called all the motions that can be used as a command signals of an evolving system, we can draw out some requirements for obtaining a generic Low Level Gesture File Format. These requirements we propose are still under work. Comments and suggestions will be welcome.

4.5.1 Minimal information in a basic gestural file

First of all, a gesture signal is described as a sampled signal at the necessary rate of Shanon sampling. We indicated previously that this sampling rate is from some Hz to some KHz. Consequently, the file has to include the value of this sampling and the gesture data contained in the file have to store all the samples at this rate. Within this Shanon framework of conservation of information, we propose a low level gesture file format in which the gestural signals are organized in four levels at maximum: Tracks – Channels – Units – Scenes.

Before describing in detail the proposed format, we gives basic definitions of the four levels:
• Gestural Tracks: externals devices as gestural sensors are connected to computers via A-D converters. The track i contains the monodimensional scalar $a_i(t)$ corresponding to each A-D track, sampled at the Shanon rate a(t)
• Gestural Channels: A channel is composed of the several tracks supporting the geometrical dimensionality. A channel can be 1D0 (a pure scalar), 1Dx (a vector on x axis), 1Dy, 1Dz, 2Dxy, 2Dyz, 2Dzy, 3Dxyz. The nature of the variables of the channel can be "extensive variable (EV)" (positions P, velocities V, Accelerations G, Angles A) or (exclusively or) "intensives variables (IV)", i.e. homegeneous to forces F.
• Gestural Units: A unit is composed of several channels in which the signals exhibit dynamic correlations. The motions of all the sensed points of a human body in a motion capture process are dynamically correlated. The motion of all the points of hands or fingers are also dynamically correlated. This means that, this information has to be conserved in order to avoid its undesired breaking in a next stage of signal processing for example. This means that the only information we conserve is that some signals are correlated, and not the way in which they are correlated as done in several formats proposed by motion capture.
• Gestural Scenes: a scene is composed of several units that are not – or can be considered as not-dynamically linked.

This basic format allows us to describe heterogeneous gesture control situation and to consider the gestural systems (sensors and force feedback devices) as a workspace in which several systems can be used, organized and reorganized. Let take the example of an heterogeneous VR scene composed of:

- One musician playing with a small keyboard composed of 8 piano-like keys
- Another one who is controlling the spatial position of a sound source via a 3D stick
- Another one who is controlling the orientation of the light source via a 2D force sensor (a force pad)
- A dancer equipped with a motion capture system of 16 * 3D points.
- A juggler who is manipulating a 6D force feedback ball to play with a 6D virtual racket

Such scene will be described as following:

Gestural Channels:

Eight Mono dimensional 1Dz – EV(P) Channels "PianoKeys": PK1 [EV(P), 1Dz], PK 2 [EV(P), 1Dz], ...PK 8 [EV(P), 1Dz].
One 3Dxyz channel "Soundsource": SS [EV(P), 3Dxyz]
One 2Dxy Channels "LightSource": LS [IV(F), 2Dxy]
Eighteen 3Dxyz "DancerPoints": DP1 [EV(P), 3Dxyz], ..., DP16 [EV(P), 3Dxyz]
One 3Dxyz "Ball1": BL1 [EV(P), 3Dxyz]
One 3Drqf "Ball2": BL2 [EV(A), 3Drqf]

Gestural Units

Unit "Pianist gesture" composed of eight 1Dz channels: {PK1, PK2 ... PK8}
Unit "StickSource" composed of one 3Dxyz channel: {SS}
Unit "Light" composed of one 2Dxy channel {LS1}
Unit "Dancer" composed of eighteen 3Dxyz channels {DP1, ... DP16}
Unit "Juggler" composed of two 3D channel {BL1,BL2}

4.5.2 Implementation of the gesture file

This paragraph describes the current implementation of the concepts developed previously. In this implementation, choices have been done in the list of possible variables to be coded. They can be extended in future improvements.

Among various type of file data coding standard, the IFF standard (Interchange File Format) [MOR85] for binary files is largely used by several other formats as AIFF or WAVE for sound files, AVI for movie files, ILBM or GIFF for picture files… There are quite some reasons of using such a standard:

- It can be easily read by different applications, which have, a priori, nothing in common. Thus, it allows diffusion and uses as large as possible and the simplified import of similar formats (i.e. which respect the IFF standard).
- In such files, every data is encapsulated in a particular section (a chunk). Sections that are not recognized by a program are simply left aside.
- Encapsulation of data has the advantage of creating a file whose length is known, unlimited and for which the length of every chunk is also known. The file format can easily evolve by adding new chunk that will be ignored by application using older versions of this file.
- The IFF standard belongs to the public domain.

4.5.3 The signal structure

The signal is given *frame* by *frame*. A *frame* contains every coordinates of every sensed points at a given instant. So trajectories of each point are interlaced. A frame is therefore made of a set of number. Each of this number is the value of a track at a given time. The order of each track in a frame is implicitly given by the declaration order explain in the following paragraph.

4.5.4 The chunk sections

In a binary file of IFF norm, data are necessary encapsulated in structures called *chunks*. A chunk must start with 4 bytes id and a 4 bytes integer that give the size in byte of the chunk (the size does not include the id and this integer). Moreover, a chunk cannot have an odd number of data bytes. In the proposed file format, chunks are:

- Header IFF chunk
- Version chunk
- Scene chunk
- Unit chunks: for each unit chunk the channels chunks belonging to the unit follow
- Frame chunk

There is no necessity to have an explicit track chunk, the tracks being implicitly declared in the channel chunk. In the following parts, the chunk sections will be described in a pseudo-C language. The following binary types are defined as follow:

- CHAR : a byte
- USHORT : a 2 bytes unsigned integer
- ULONG : a 4 bytes unsigned integer
- FLOAT32 : a 4 bytes floating point number (IEEE 754 standard)
- FLOAT64 : a 8 bytes floating point number (IEEE 754 standard)

The header chunk

The header chunk always has to have the same structure in an IFF based file format. It must start by the keyword 'FORM', followed a 4 bytes number indicating the size of the file (the 4 bytes 'FORM' and this number are not taking into account), and a 4 bytes keyword giving the type of the file (here it will be 'GST '). Here is the pseudo-C description:

```
Typedef struct {
       CHAR[4] chunkId = 'FORM';
       ULONG          fileSize;
       CHAR[4] fileType = 'GST ';
}headerChunk
```

The version chunk

This chunk gives the version number and the subversion number of the gesture file. The current version describe here is the version 0.1. This chunk is very important for the evolution of the gesture file because it will give implicit information to the parser software on which kind of chunk follows what will allow adding new type of information in the future.

```
Typedef struct {
       CHAR[4] chunkId = 'VERS';
       ULONG         chunkSize;
       USHORT  versionNum /* currently equal to 0*/;
       USHORT  subVersionNum /* currently equal to 1*/;
}versionChunk
```

The scene chunk

This chunk contains information about the gestural scene encoded in the file. Note that the file contains only one scene. Indeed we have seen that a scene encompass unit that are uncorrelated. So two scenes can always been merged into one.

```
Typedef struct {
       CHAR[4] chunkId = 'SCEN';
       ULONG         chunkSize;
       USHORT  sceneNameLength;
       CHAR*   sceneName;
       ULONG   nbFrame;
       FLOAT64 frequency;
       USHORT  dataType;
       FLOAT64 scale;
       ULONG   blockSize;
}sceneChunk
```

scale is a scale factor to apply on the signal in the case when its type is ULONG. This value is equal to 1 if data are of types FLOAT32 and FLOAT 64. *dataType* gives the type of data of the signal. The current version support three type of data : 1 = FLOAT32, 2 = FLOAT64, 3 = ULONG.

blockSize gives the size of a block like in AIFF format. A block is a set of consecutive bytes inside the file. Indeed, certain real-time applications need to align data into fixed size blocks. By default, the size of a block corresponds to the size of a *frame*.

The unit chunk
This chunk just contains the name of the unit. The declaration order of the unit implicitly gives the id of each unit :

```
Typedef struct {
       CHAR[4] chunkId = 'UNIT';
       ULONG  chunkSize;
       USHORT  unitNameLength;
       CHAR*   unitName;
}unitChunk
```

The channel chunk
This chunk contains information on a gestural channel. This channel belong to the last unit declared.

```
Typedef struct {
       CHAR[4] chunkId = 'CHAN';
       ULONG  chunkSize;
       USHORT  channelNameLength;
       CHAR*   channelName;
       USHORT dimension;
       USHORT type;
}channelChunk
```

dimension is to identify: 1 = pure scalar, 2 = vector on 1Dx (resp. 3 = 1Dy and 4 = 1Dz), 5 = vector on 2Dxy (resp. 6 = 2Dyz and 7 = 2Dzx), 8 = vector on 3Dxyz.
type is chosen in a list of type of variables: {position = 0, force = 1}

The frame chunk
This chunk contains the signal itself that is presented into successive frames. Therefore there is *nbFrame*size_of_a frame*size_of_data_type* bytes in this chunk if we don't count the id and the integer that give its size (where *size_of_a_frame* is calculated by adding the dimension of every channels and *size_of_data_type* is the size of each value in the signal.

```
Typedef struct {
       CHAR[4] chunkId = 'FRAME';
       ULONG         chunkSize = nbFrame*size_of_a frame*size_of_data_type;
       (ULONG or FLOAT32 or FLOAT64) [ size_of_a frame ] [ nbFrame]  frames;
}frameChunk
```

4.6 Conclusion

While gesture device, especially haptic devices, develop, while applications communicate more and more though gesture-like data, the definition of a standard gesture format, or of standard gesture formats, can be seen as a major need for the near future. To do this, we extracted the specific properties of the motion signal among other temporal signals (sounds and images): its morphological versatility decomposed in geometrical and structural ones, its spatial and temporal ranges and the variety of type of data. We showed that these properties are able to explain why gestures can be considered as at "a hub place" in multisensory situations and processes and how they can be shared by various applications. This lead to the definition of a basic file format adapted to encode all these minimal and necessary features. We used such format in physically-based multisensory simulator for synchroneous haptic interaction with motion and sound synthesis in hard and soft real-time, as well in separated physically-based motions and sound synthesis simulators. Experiences by connecting both through such gesture file are in progress in the laboratory. Finally, transcoding such format in others (frame-based to trajectory based for example), as well as improving motion editing tools [Vba00] with specific gesture compositing is an easy task. Such tools can easily be shared and developed in various contexts. This format can be easily merge to other format as AIFF for the sound [AP89], leading to multimodal encoding.

4.7 References

Web sites of motion Capture firms and formats
[AOA] http://www.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/AOA.html
[ASF] http://www.darwin3d.com/gamedev/acclaim.zip
[BVH] http://www.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/BVH.html
[BRD] http://www.dcs.shef.ac.uk/~mikem/fileformats/brd.html
[C3D] http://www.c3d.org
[CSM] http://www.dcs.shef.ac.uk/~mikem/fileformats/csm.html
[HTR] http://www.cs.wisc.edu/graphics/Courses/cs-838-1999/Jeff/HTR.html

[AP89] Apple Computer, Inc. « Audio Interchange File Format : « AIFF ». A standard for sampled sound files Version 1.3 ». January 4, 1989.
[HFP00] Herda, Fua, Plänkers, Boulic, Thalmann « Skeleton-Based Motion Capture for Robust Reconstruction of Human Motion », EPFL, Lausanne, Switzerland, janvier 2000
[ME99] Menache. « Understanding Motion Capture for computer animation and video games », Morgan Kauffmann Ed., August 1999.
[MM00] Meredith, Maddock. « Motion capture file formats explained. » Département of computer Science, University of Sheffield, 2000
[MOR85] Morrison. « EA IFF 85 - Standard for interchange format files » document de spécification Electronic Arts, janvier 1985
[SW84] M. Smyth, A. Wing. « The psychology of Human movement. » Académie Press, 1984
[Vba00]Susan Van Baerle. Motion Editing : Principles and Practice. 2000 Game Development Conference Proceedings. March 20-24, 2000. San Jose, CA.

5 Tactile devices and data (UNEXE)

Ian Summers and Alan Brady
UNEXE
October 2005

5.1 Introduction

UNEXE has developed a system for tactile stimulation. In the context of the action/perception loop in a virtual scenario, the UNEXE system delivers tactile stimuli to the fingertip(s) in response to user movements in the workspace. Stimuli are delivered by an array of contactors on the fingertip(s) whose vibration waveform is under software control. The intention is to produce virtual touch sensations – edges and corners of objects, surface textures and contact area – by producing an appropriate spatiotemporal variation of mechanical disturbance over the skin. An array of this type does not aim to reproduce the topology of "real" surfaces: It aims to produce an appropriate excitation pattern over the various populations of mechanoreceptors in the skin.

5.2 Stimulator hardware

Figure 1(a) shows an example of a recent design – an array with 25 contactors over 1 cm^2 on the fingertip. This may be used to stimulate a single fingertip or can be configured as part of a 125-contactor, 5-digit stimulator for one hand. The contactor array, in the centre of the top surface, is driven by piezoelectric bimorphs (which appear black in the picture). Figure 1(b) shows an outline diagram of an array proposed for the HAPTEX project. The finger is represented by the cylinder towards the top of the picture. Note that the mechanism is placed around the back of the finger so that there is minimum interference to manipulation.

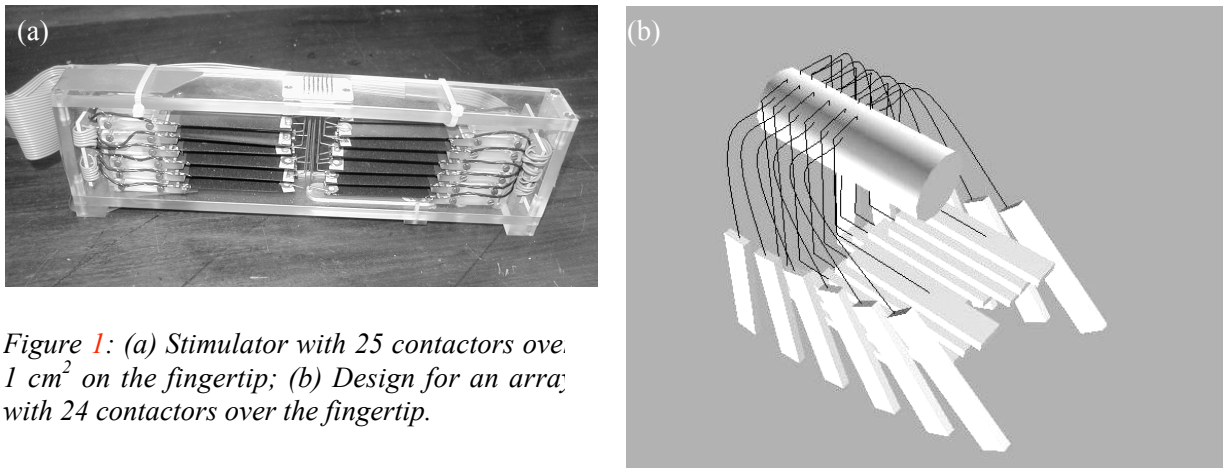


Figure 1: (a) Stimulator with 25 contactors ove 1 cm^2 on the fingertip; (b) Design for an array with 24 contactors over the fingertip.

The spatial resolution required for the contactor array is related to the density of mechanoreceptors in the skin, which is on the order of 1 mm^{-2} on the fingertip, or to the spatial acuity on the fingertip, which is around 1 mm [A]. In a previous investigation [B], using a 100-contactor array with a spatial resolution of 1 mm on the fingertip [C], we have shown that it is difficult to discriminate between moving vibratory stimuli presented at resolutions of 1 mm or 2 mm. This suggests that a 1-mm pitch array may offer little advantage

over a 2-mm pitch array in some contexts. The preferred contactor spacing for the UNEXE system has been chosen to be around 2 mm.

5.3 Electromechanical design

In order to evoke "realistic" touch sensations an array must operate over most of the tactile frequency range of, say, 10 to 500 Hz. To produce “comfortable” sensation levels requires a few microns at frequencies around 250 Hz and a few tens of microns at frequencies around 50 Hz.

Electromechanical design may be facilitated by a mathematical model of the piezoelectric cantilever and the mechanical load presented by the skin. For example see Figure 2(a), which indicates that the first and second resonant modes are separated by an antiresonance. Comparative experimental results are shown in Figure 2(b). It can be seen that there is good agreement in terms of the main features of the response. In the examples shown, the antiresonance is inconveniently placed and must be moved to a higher frequency by redesign.

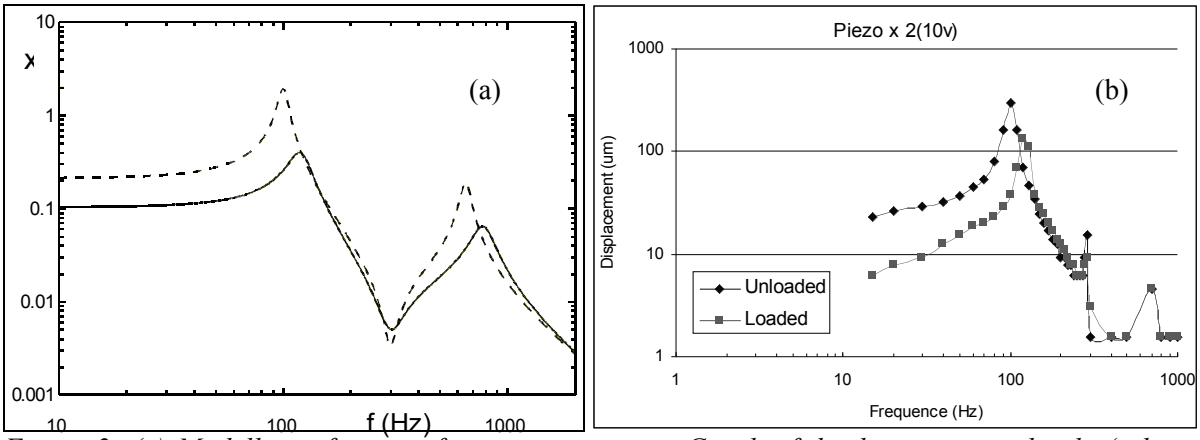


Figure 2: (a) Modelling of system frequency response. Graph of displacement amplitude (arbitrary units) vs frequency. The dashed line represents the predicted system response with no load, and the full line represents the predicted system response when loaded by the skin. The frequencies of the resonant modes are determined by the dimensions of the piezoelectric bimorph. (b) Experimental results for comparison. The additional peak at around 300 Hz is due to a resonance in the mechanical linkage between cantilever and the contactor.

5.4 Stimulus design

A significant problem for the operation of an array stimulator is the need to specify multiple parallel waveforms – there is essentially infinite choice within the system bandwidth. In an attempt to provide a user-friendly solution, in the UNEXE system each waveform is constrained to be a specified mixture of 40 Hz and 320 Hz sinewaves, i.e., the output is a superposition of a spatiotemporal distribution of vibration at 40 Hz and a spatiotemporal distribution of vibration at 320 Hz.

There are various populations of touch receptors in the skin [A]. The 40 Hz output is intended to stimulate primarily non-pacinian receptors and the 320 Hz output is intended to stimulate primarily pacinian receptors. This scheme was first proposed by Bernstein [D] in the context of single-channel vibrotactile stimulation. The two-frequency system may be considered as analogous to a 3-colour video display – the stimulator produces a sequence of frames in two tactile “colours”. Psycho-physics experiments have been performed to compare the perception of moving stimuli at the two different

stimulation frequencies [C]. Data were obtained on the masking of 40 Hz or 320 Hz stimuli by a uniform vibrating background at 40 Hz or 320 Hz. The two different stimulation frequencies produce different results, suggesting that different receptor populations have been targeted as intended. For stimuli at 40 Hz and 320 Hz to have the same subjective intensity, stimulus amplitude at 40 Hz must be around 10 times greater [C].

5.5 Software design (tactile rendering) and data format

The user explore a virtual tactile surface which is specified in software. The drive signal to each point in the stimulator array is specified in terms of an amplitude at 40 Hz and an amplitude at 320 Hz, and these amplitudes are in turn specified by the interaction between the virtual surface and the exploratory movements of the user. A major problem for the software designer is the lack of information about the "real" tactile stimuli which are produced by the exploration of real objects. At the present state of knowledge, it is possible to suggest designs of software for tactile rendering (for example, see Figure 3), but it is not possible to be certain that these are the most appropriate solutions.

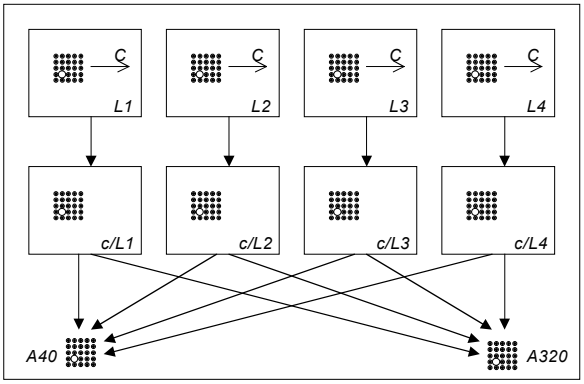


Figure 3: A simple tactile rendering scheme. The surface is described in terms of amplitude distributions at length scales L1, L2, etc.. These may be considered as amplitude distributions in frequency ranges c/L1, c/L2, etc., where c is the speed of exploration. For each point in the array, e.g. the white dot indicated, the amplitudes may be combined by appropriate filter functions to produce a drive amplitude A40 at 40 Hz and a drive amplitude A320 at 320 Hz.

For a stimulator array with 2 mm spacing, we suggest the tactile surface should be specified on a resolution ("pixel size") of 1 mm, with effective feature widths of ≥ 2 mm. (In practice, however, the tactile surface may be distorted when it is used to render a 3D shape, and so the 1 mm resolution may not be maintained.)

The tactile rendering algorithm must deliver amplitude values *A40* and *A320* for each channel of the tactile stimulator at an update rate of around 50 Hz. (It is counter-productive to update the stimulator drive signals at a higher rate, since this would only disrupt the sinusoidal drive signals.) Four-bit data is adequate to specify each of these amplitudes. However, in practice it is easier to use 8-bit words to specify the 4-bit data, and in a serial interface it is necessary to transmit an 8-bit address code along with each 8-bit word. Thus each amplitude is specified by 16 bits, each channel requires 1600 bits s^{-1} and a 25-contactor array requires 40 k bits s^{-1}.

Figure 4 shows a design for the stimulator drive electronics. The drive signal in each channel is a mixture of signals from two sinewave generators (40 Hz and 320 Hz). The mixture is specified by digitally controlled attenuators (2-bit attenuators are shown for convenience but, as explained above, 4-bit systems are specified in practice).

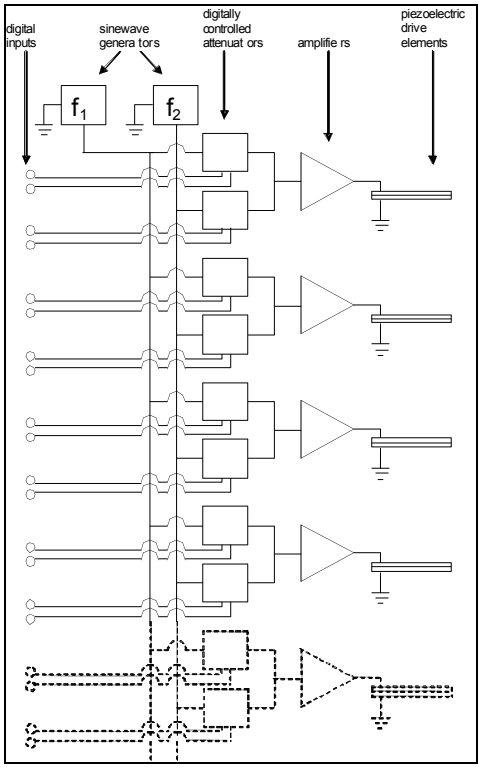


Figure 4: Schematic diagram of the drive electronics for a stimulator array.

5.6 Relevance to the creative context

In the creative or artistic context, the gesture often relies on fine motor control which is informed by tactile feedback. For example, consider the guitarist whose movements on the fingerboard are informed by tactile cues about the positions of the strings and the frets. The UNEXE tactile stimulator is intended to provide the accurate tactile feedback which is necessary for a virtual implementation of such activities.

5.7 Technical specification of the UNEXE system

Number of channels: 24 or 25 per finger
Spatial resolution on the finger: 2 mm
Working bandwidth: 20 – 500 Hz
Maximum output amplitude: 100 _m
Spatial resolution of the virtual workspace: 1 mm
Input data: Data frames with 32 bits per channel, i.e., 800 bits per frame for 25 channels
Input data rate: Frame rate around 50 Hz, i.e., 40 kbits s^{-1} for 25 channels
Input connectivity: USB
Output data: NOT APPLICABLE

5.8 References

[A] K.O. Johnson, T. Yoshioke, F. Vega-Bermudez, “Tactile functions of mechanoreceptive afferents innervating the hand," J. Clin. Neurophysiol. 17, (2000), pp. 539–558.
[B] I.R. Summers, C.M. Chanter, A.L. Southall, and A.C. Brady, “Results from a Tactile Array on the Fingertip,” Proc. Eurohaptics 2001, Birmingham (2001) pp. 26-28.
[C] I.R. Summers, and C.M. Chanter, “A broadband tactile array on the fingertip,” J. Acoust. Soc. Amer. 112, (2002) pp. 2118-2126.
[D] L.E. Bernstein, S.P. Eberhardt, and M.E. Demorest. Single-channel vibrotactile supplements to visual perception of intonation and stress. *J. Acoust. Soc. Am*., 85: 397-405, 1989.

6 Device based on flexible goniometric sensors patented by PERCRO Laboratory

6.1 Abstract

At the PERCRO laboratory innovative Flexible Goniometric Sensor has been developed in order to realize a sensorized Data Glove for the acquisition of the posture of the human hand.
The device is characterized by a low cost and rugged construction and requires no calibration before use. Indeed, the sensors used are purely goniometric so they are not sensible to dimensions of the hand of the user. In this article this new technology, for which a patent has been filed, is described.

6.2 Introduction

Datagloves are widely used in many applications, such as virtual reality, telerobotics, and biomechanics. Essentially, a dataglove is a glove fitted with sensors to measure the relative angular displacement of the joints of the hand. There are several types of dataglove commercially available. All of these serve the purpose for which they were originally intended (e.g. data input, object manipulation, computer game accessories), but they are either too delicate, too expensive and they lack the required accuracy and comprehensiveness. [1], [2].
We propose a device characterized by a low cost and rugged construction requiring no calibration before use. Calibration is necessary when the dataglove is equipped with sensors measuring the deformations of the glove itself, including those due to the variations of length occurring each time a different user wears the glove. Actually, people have different length, thickness of their fingers and size of the palm of their hands, so it is necessary a calibration/normalization procedure to match the sensor output spans with the specific user range of motion. For these kinds of gloves, quantitative assessment of rigid range of motion (ROM) is required and a measuring procedure must be done ([4], [5], [6]). Differently, the PERCRO dataglove (Fig.10.1) is equipped with purely goniometric sensors measuring the relative angular displacement between two phalanxes, independently of the specific deformed elastic line of each sensor. A patent application has been filed for the new goniometric sensor and the dataglove in December, 30, 2002 [3].

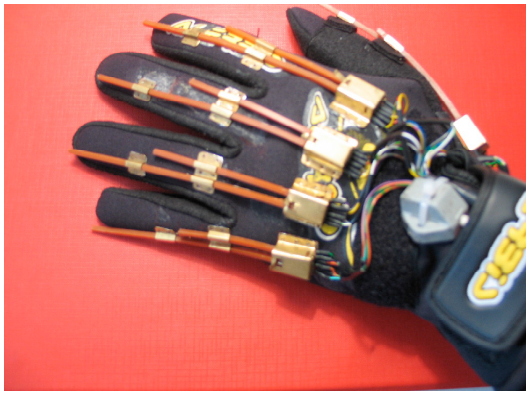


Figure 10.1: The PERCRO dataglove

6.3 The goniometric sensor

The working principle of the sensor relies on the fact that a flexible beam having a deformed elastic line lying on a plane has the property that the longitudinal elongation of the fibers depends linearly on their curvature and their distance from the neutral axis of the beam.
If we impose a relative bending angle $d\_$ to a beam element of length dx (Fig.10.2), the length of the neutral fiber of the beam remains unchanged while the fiber positioned at a distance of e from the neutral axis, changes its length by a quantity of:

(1) $$dl = e \cdot d\vartheta$$

Integrating this variation along the entire length *L* of the beam, in case of constant *e,* we obtain:

$$(2)\quad \Delta l = \int_0^L dl = \int_0^L e \cdot d\vartheta = e \cdot \int_0^L d\vartheta = e \cdot \Delta\vartheta$$

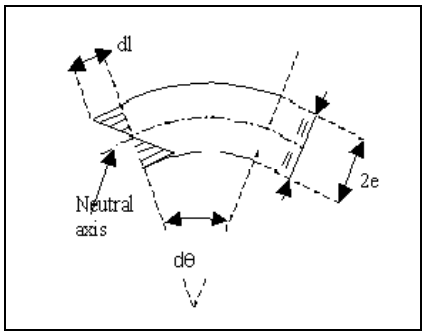


Figure 10.2: Working principle

Therefore, the total elongation of the fiber is a function of the angle between the two endpoints of the flexible beam and it is independent from its specific elastic line.
In particular, if we impose an omega deformation to the beam having a zero angle between the two end points, we have a null total elongation of the eccentric fiber, because of the compensation between portions with opposite positive and negative curvature.
If we make a longitudinal hole in correspondence of the eccentric fiber and we put inside it a free moving axially rigid wire, having its end fixed to one endpoint of the beam, we get a linear displacement of the other end of the wire that is proportional to the rotational displacement. This quantity can be transduced with a low-cost linear Hall Effect Sensor measuring the intensity of a magnetic field produced by a magnet attached to the movable end of the wire.
In Fig. (10.3) is reported the chosen implementation of the said working principle. The goniometric sensor is composed by only four parts:

- a commercial cylindrical permanent magnet;
- a commercial miniaturized Hall Effect sensor with a built-in signal amplifier;
- a multiwire flexible steel cable;
- a flexible thin beam made of plastics with a square cross section and a longitudinal hole. The beam ends with a bulb hosting the magnet and the Hall Effect sensor.

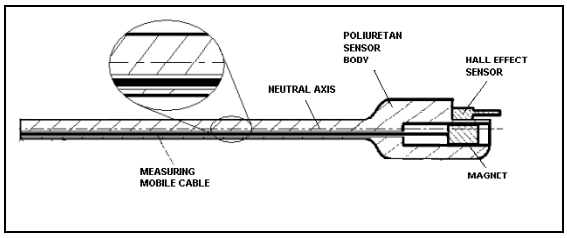
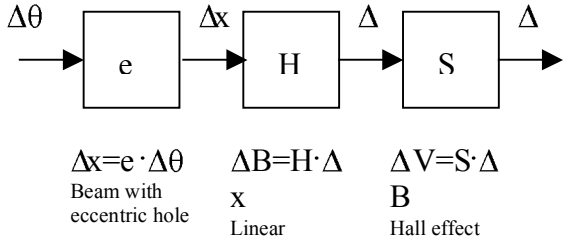


Figure 10.3: Schematic diagram of the goniometric sensor.

The block-diagram of the measuring system is shown in the following figure:



where:

- e = eccentricity
- H = magnetic field variation per unit of linear displacement
- S = hall effect sensor sensibility

The relation governing the sensing system is:

(3) $$\Delta V = S \cdot H \cdot e \cdot \Delta\vartheta = K_{\vartheta(rad)} \cdot \Delta\vartheta$$

where:

(4) $$K_{\vartheta(rad)} = S \cdot H \cdot e$$

Inverting the relation (3) we obtain:

(5) $$\Delta\vartheta = \frac{1}{S \cdot H \cdot e} \cdot \Delta V = K_{V(rad)} \cdot \Delta V,$$

where:

(6) $$K_{V(rad)} = \frac{1}{K_{\vartheta(rad)}}$$

$K_{V(rad)}$ is the slope coefficient of the theoretical characteristic of the sensor.

An experimental setup has been built in order to measure the real characteristic curve of a set of 50 sensors. For each sensor, a linear interpolation of the experimental data has been derived and the relative slope coefficient (A1) has been calculated. The mean value of A1 for the said set of sensors resulted:

(7) $$A1_m = 52.8[\frac{\deg}{V}]$$

The relative deviation of the theoretical estimation with respect to the experimental one is:

(8) $$E_{A1} = \frac{K_{V(\deg)} - A1_m}{K_{V(\deg)}} \cdot 100 = 2.3\%,$$

The said error is due to the manufacturing tolerances on flexible beam and on the characteristic parameters of the used commercial components (sensitivity of hall effect sensor, coercive field of the magnet, etc.). The tests of the goniometric sensor have shown an hysteretic behaviour in the angle-voltage characteristic, mainly due to the backlash between the hole and the steel cable. Indeed, in the transition between two curvatures of different sign on the same portion of the beam, the cable will lean against the two opposite generatrices of the hole causing a variation of the eccentricity of the cable with respect to the neutral fibre. Therefore, the eccentricity is not still constant but is a function of the curvilinear abscissa *x* (i.e. *e(x)*). As a consequence, the total length elongation of the fibre is slightly dependent on the particular elastic line. This effect can be reduced using a better technology for the construction of the beam.

The resulting performances of the sensor are the following:

- Range of measure: (0-180) degrees
- Resolution: 0.1 degrees (depending on electronics)
- Accuracy: 2 degrees
- Output span: 0-5V

Advantages of the sensor are:

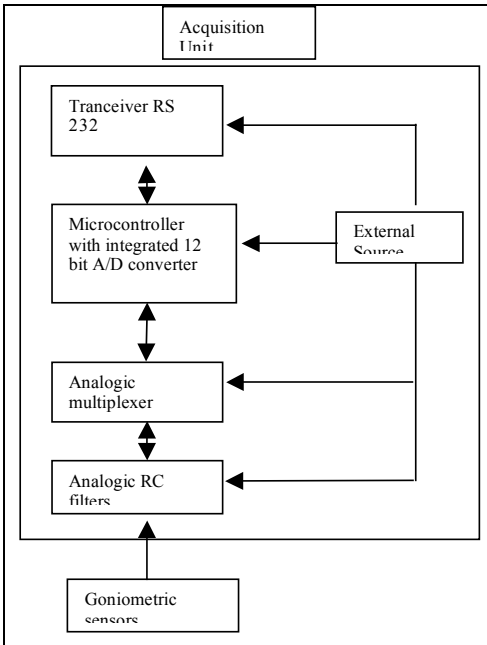
- due to its flexible body, the sensor can adjust to any external kinematics; in particular, it can be easily attached to the human body without exerting constraints;
- the production costs can be low because it is based on few commercial low-cost components and on one only non-commercial component that can be mass produced by technologies like injection molding;
- the relation Angle-Voltage is linear;
- calibration is "for life";
- the electronic acquisition unit is very simple because the commercially available Hall Effect Sensors have integrated signal amplifiers and no further conditioning is required;
- the construction is very rugged.

6.4 The dataglove

The dataglove is equipped at least with two sensors of different lengths for each finger; they measure the angular displacement of proximal and medial phalanxes with respect to the back of the hand. The difference between the two signals can be implemented via software in order to obtain the relative angular displacement between the two phalanxes. The flexo-extension of the distal phalanx with respect to the medial is considered equal to the one of the medial with respect to the proximal, so the said two sensors are enough to describe the flexo-extension of each finger.
The adduction-abduction movement of each finger is not measured, except for the thumb where a third sensor bends in a plane normal to the flexo-extension of the other two sensors.
The sensors are mounted on the glove with the bulbs fixed to the back of the hand, while the thin parts of the flexible beam are guided by plastic elements in order to make it bend with the fingers. The plastic guides allow free axial movement of the sensor; this way, no axial force occurs and the sensor can measure a pure angle.

6.5 Acquisition and Graphic representation

A dedicated electronic acquisition unit has been realized in order to acquire the analogic signals coming from the goniometric sensors, digitally convert and send them via serial line to the PC (Fig.10.4).



Taking into account the maximum frequency content of human voluntary finger movements, the refresh rate of the acquisition unit has been limited to 100 Hz. A program running on a commercial PC has been developed in order to communicate with the electronic unit and evaluate the angular displacements on the basis of the calibration parameters of each goniometric sensor, derived by the linear interpolation of the experimental data.
Finally, the system has been completed with a visualization module in order to animate in real time an avatar of the user's hand. The graphic representation of the hand (Fig.5) has been implemented in a virtual 3D environment, based on OpenGL high performance graphic libraries. 3D meshes realized with the most widely used modelling software can be imported and deformed applying skeletal motions, i.e. applying geometrical transformations to the point of the mesh associated to the corresponding point of the skeletal. The angular positions of the skeletal joints are associated to the evaluated angles.

6.6 Conclusions and future works

The results are promising. This kind of glove could have a wide commercial diffusion due to his low cost and robustness compared with his measuring qualities. As the glove stands at the moment, the presence of the electric cable for the connection with the electronic acquisition unit may still slightly constraint the normal movements of the wearer. The transition to wireless technology may eventually allow the development of a glove which is more transparent for the user.
Measurement errors occur in reconstructing the fingers` angles from the electric signals coming from the sensors; these errors are mainly due to:

- the backlash between the steel cable inside the sensor and the hole;
- the backlash between the flexible thin beam and the guides;
- the deformability of the tissue of glove.

An increased accuracy in motion capture of the hand movements could be done by:

- applying sensors for the adduction-abduction movements of the fingers;
- applying a sensor to the back of the hand in order to acquire its movements;
- developing a finite element model of the glove in order to estimate the relative movements between the fingers and the sensors` guides.

Future research will be aimed at reducing the overall dimensions of the sensor, also investigating similar goniometric sensors (i.e., substituting the steel cable with an incompressible fluid and the hall effect sensor with a pressure sensor).

Figure 10.5: Application in virtual graphic environment.

6.7 Summary technical sheet

PERCRO as developed a glove for the acquisition of hand movements. A first prototype was equipped with 12 sensors (2 each finger and 2 for the wrist). The bulbs of the sensors are all positioned on the back of the palm of the hand in order to realize a compact glove, simplify the cabling and render the glove more reliable (the cables are not subject to fatigue). The flexible bars need to slide in order to compensate the change of length so they are placed in guides built with metallic foils.

Advantages:

- no specific calibration required;
- low cost production;
- good precision (see sensor);
- Data Glove (due to the characteristics of the sensor) it is insensible to external factors;
- it is a real-time device;
- it has a rugged construction.

Characteristics:

- N° Degrees of freedom: 11
- Resolution: 0.5°
- Repeatability: 0.1°
- Weight: 70g
- Ergonomy: High
- Cost: about 1000$

Each angular sensors used in the Glove has the following characteristics:

- Range of measure: (0-180) degrees
- Resolution: 0.1 degrees (depending on electronics)
- Accuracy: 2 degrees
- Output span: 0-5V

The Glove outputs are electrical signals (one for each sensor). These signals are proportional to the angular position of the fingers. A specific software transforms these electrical signals in angular coordinates.

6.8 References

[1] H. Eglowstein: Reach out and touch your data, BYTE, 1990, 15 (7), p. 283-290

[2] D. J. Sturman, D. Zelter: A Survey of Glove-based Input, Media Lab, Massachussets Institute of Technology, Cambridge, Mass.

[3] PCT WO 2004/059249 A1 “Goniometric Sensor”, International Patent Application, Publication Date 15 July 2004, International Filing Date 30 December 2002.

[4] Laura Dipietro, Angelo M. Sabatini, Paolo Dario: "Evaluation of an instrumented glove for hand-movement acquisition", Journal of Rehabilitation Research and Development. Vol. 40 No. 2, March/April
2003. Pages 179190.

[5] Wise S., Gardner W., Sabelman E., Valainis E., Wong Y., Glass K., Drace J., “Evaluation of a fibre optic glove for semi-automated goniometric measurement”, Journal of Rehabilitation Research and Development. 1990; 27(4):411-24.

[6] Quam D., Williams G., Agnew J., Browne P.,”An experimental determination of human hand accuracy with a DataGlove”. Proceedings of the Human Factor Society 33rd Annual Meeting 1989; vol. 1 p. 315-19.

7 PERCRO – A novel system for the acquisition and the teaching of gestures

7.1 Abstract

This paper presents two applications developed for a novel multimodal device called Haptic Desktop System (HDS). The HDS is an integrated system which merges haptic functionalities and Video Display Terminal (VDT) systems into one. It has been designed to provide visual and haptic co-located perceptual information. The first application is the integration of the HDS with a handwriting recognition system. In the second application, the interface can assist the users in a drawing task and act as a virtual guide through its force feedback capabilities; the application can read various image formats and convert the main image features in output control trajectories for user interaction.

7.2 Introduction

Haptic Interfaces (HI) have been designed in order to interact with humans through different types of force feedback. Teaching robots how to perform a difficult task has been presented as an example of this interaction [1,2]; in this case, of course, the skill transfer is from the teacher/user to the robot. Preliminary experiments to invert this flow have been conducted by Mussa-Ivaldi and Patton,Yoshikawa, Sakuma, etc. Mussa-Ivaldi and Patton [3] presented a robot-aided therapist for stroke patients. The robot provides adjustable levels of guidance and assistance to facilitate the person's arm movement. The patient is guided in moving the robot end effector from an initial position towards a fixed number of points.

Yoshikawa [4,5] and Sakuma [6] presented a calligraphy transfer skill system. These Haptic Systems use information from the teacher's movements (position and force trajectories) stored locally or remotely to show to the student the proper force/position relation that leads to writing correctly a Japanese and Chinese character [6,7].

The main disadvantage of these systems is that the interaction is in some way restricted. The teacher/therapist decides the action to perform and the HI collaborates with the user to accomplish the task. In a more interactive system, the user should be free to decide the task to perform. These systems lead the user to reproduce a trajectory or generate a character by moving through a very well defined sequence of trajectories, which makes practically impossible to use these systems to teach how to sketch or draw. To this end, a more dynamic interaction between computer and human is necessary.

Up to now, few systems have been planned to teach drawing or sketching, in spite of the suggestions described in [8]. Reach-In [9] for example, integrates a 3D graphic system with a PHANToM device [10, 11]. The systems described above allow the users to interact with 3D environments; however the small number of contact points and the complexity of the images displayed may demand from operators a high level of skills to process all the information. In fact, such systems are used only by specialists (i.e. architects, engineers, designers, etc.) that can accept the technological access constraints. Furthermore, the need of special equipment to visualize the 3D images and provide haptic information increases considerably the cost of such systems and the workspace required to deploy these systems.

On the other hand, some solutions for integrating 2D systems have been proposed. Brederson proposed the Virtual Haptic Workbench [12], which integrates the PHANToM with a planar screen. The dimension of the workspace is attractively large, but the system requires a complex calibration procedure. Moreover, the complexity of the haptic interface does not match the requirements of the visualization system (2D).

A different solution is based on linear induction motors mounted under a desk that produce forces on a metal plate attached to the user's finger or to the end part of the tool [13, 14]. The disadvantage of this solution is that the operator view of the graphical interface can be obstructed by the operator's own arm, since the projector is not under the desk but it is above user's head.

The Haptic Desktop System (HDS) developed at PERCRO is our multimodal interface solution for the office environment [15]. In this paper we present two applications. In the first one, we have integrated the HDS with a commercial handwriting recognition system. In the second one, we have

developed a drawing/sketching assistant that helps the users develop motor skills for producing simple sketches and drawings.

In the second section of this paper, we will give a brief description of HDS's characteristics. In the third section, we will present the main features of HDS and their influence on applications. In the fourth and fifth sections we will describe the two applications. Finally, in the sixth and seventh section we will present the results and conclusions of this work.

7.3 Description of the System

The System used for the development of the applications presented in this paper is the Haptic Desktop System (HDS) [15]. HDS is an integrated system developed by PERCRO, which merges haptic functionalities and VDT systems into one. HDS in fact, integrates proprioceptive, visual, audio and haptic functionalities into a desk, minimizing the visual interference of the physical components. Haptic functionalities are generated by a parallel planar interface with two degrees-of-freedom, which has been mounted on the surface of a desk. The transparent haptic interface is made of a plastic material with low refraction index and it can be grasped directly with the finger or manipulated by means of a sensorized pen.

Fig. 11.1 Haptic Desktop system.

The integration of components of the HD system has been designed to offer the best ergonomics and user comfort to the operator. The desktop can be adjusted to different positions depending on the ergonomic needs of each user (as a standard, the inclination of the desktop plane has been fixed to 16^{o}).

The whole system is integrated within the work-plane of a desk: the computing unit, the power supply, the motors and the electronics for the control of the haptic interface have been placed under the desktop so that the desk plane is completely free and the operator has direct access only to the visual and haptic systems.

The graphical visualization is also integrated on the desk so that visual and haptic coordinate systems are coincident. Such features allow user tactile interaction (rendering forces and natural tactile stimuli) and visual interaction to be completely colocated and coherent.

The design of the hardware system is characterized by its compactness and low cost. The high integration achieved has allowed us to implement all control procedures and drivers within the hosting OS kernel (Windows 2K).

The haptic device has been designed for exerting a maximum continuous force of 4.0 N and a maximum peak force of 10.0 N.

7.4 System Features

The HDS, while providing all the interaction modes of traditional systems, adds the possibility of accessing the computer resources through the graphical cursor that coincides with the interaction point. This feature allows generating and rendering natural force stimuli that are completely coherent and co-located with respect to the visual information. In this way HDS reduces considerably the mental load required to the users during interaction operations [16]. By attaching a pen stylus, the writing and

sketching tasks (Fig. 11.2) become easier than using the conventional input devices (mouse, trackball, etc.).

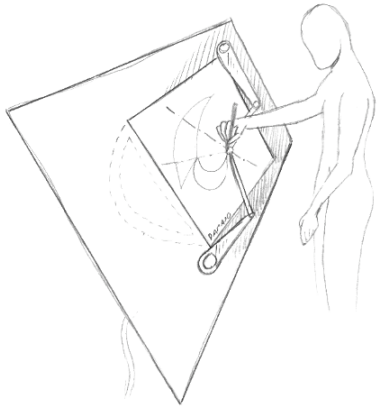


Fig. 11.2 Using the Haptic Desktop to draw

The principal characteristic of the device is the addition of operator force feedback co-located with the visual interaction focus to the interface vocabulary of the computer. This makes possible the creation of a new kind of interaction with the user that is now enabled to “touch” buttons, “feel” and “move” lines. In this new context, the haptic device replaces the use of compasses, squares and/or other drawing instruments.

Fig. 11.3 shows the user during the use of Haptic Desktop. Its implicit functionality as a pointing device permits interaction with the Windows GUI without a mouse or a keyboard. The action buttons are currently installed on a small box (represented in the user's right hand in the picture). They will eventually be moved to the polycarbonate end-effector.

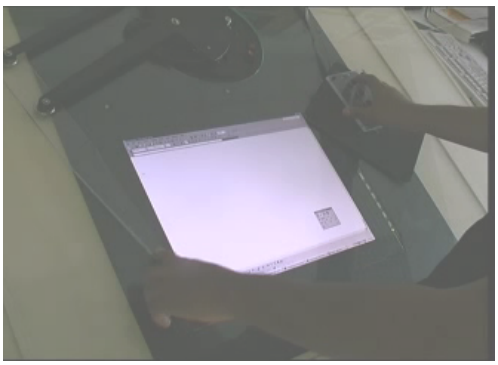


Fig. 11.3 User GUI interaction

7.5 Handwriting recognition system (HRS)

We have integrated the HDS with the Microsoft Office® handwriting recognition system. This is a feature of that allows the user to enter text in any component of the Office suite by writing instead of typing. The handwritten text is then converted into typed text. Additionally, for example in Word®, the input can be left as “ink object” handwriting. Handwriting recognition also integrates a drawing function; it is thus possible to include in Office documents hand-drawn sketches.

A handwriting recognition system enables the user to interact with to the HDS in a natural way. Using the HDS as a pointing device is possible to write a word in any place of the screen (Fig. 11.4). The HRS will then kick in and type the recognized word (Fig. 11.5).

V. A TRANSFER SKILL SYSTEM FOR TEACHING HOW TO DRAW TO UNSKILLED PERSONS.

The idea of exploiting the HDS as a tool capable of interacting dynamically with the human on the basis of the action he decides to perform has been introduced through the concept of Reactive Robots (RR) [17, 18].

RR technology enhances the capabilities of the HDS. Such systems can analyze the data exchanged between the user and the interface, then analyze and elaborate a movement strategy that dynamically complies with user's wills and a set of arbitrary paths to be taught that have been previously stored in the control.

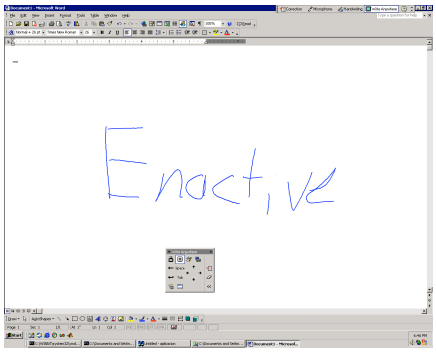


Fig. 11.4 Word to be recognized.

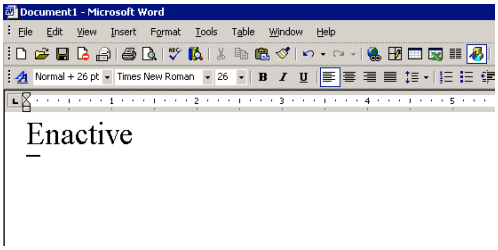


Fig. 11.5 Recognized Word.

Reactive Robots aims to be an interactive bi-directional skill transfer system that can emulate the presence of a human tutor, guiding the student/patient movements on the trajectories chosen by the user himself (Fig. 11.6).

Fig. 11.6 Reactive Robots as tutors of abilities

The use of an automated system in the skill transfer process has several advantages: the process could be repeated over and over with a high level of precision, time and location flexibility, the user can improve its performance without the assistance of a teacher/therapist and progress could be evaluated numerically through automated procedures.

The Reactive Robot system has a complex organization and its structure can be divided in different specialized subsystems that must be transparent to the user. Four main subsystems compose the architecture of a Reactive Robot: the user, an input/output device, the control system and the recognition system [18].

With the RR paradigm in mind, we have begun to develop an interactive application where the interface will teach to the users to sketch and act like a virtual assistant [19,20, 21]. Up to now we have developed two subsystems of our reactive robot: Control System and Input/Output device (HDS). These subsystems will subsequently be connected to the recognition system currently under development.

To test the subsystems, we have developed an application where the HDS assists the users in sketching certains simple drawings. We see this as the first step to develop a system that can teach drawing. In this application, the user can interact with the system through different means of feedback (visual, force, audio).

The application is composed of two sub applications. The first one transforms the principal features (edges) of an input image file (jpg, bmp, tiff, etc.) into trajectories for later HDS interpretation. This way is possible to create "trajectories" without any programming. The image is pre-processed to detect its edges using the Canny edge detector [22], then it is segmented using label connected components in a binary image [23], subsequently its components that are not more longer than a minimum predefined size are erased. The remaining components are converted to one pixel wide curves by the application of the morphological thin operator [24]. Finally, the coordinates of the curves are stored in a file to be used for the control system and the second sub application. The Fig. 11.7.a shows an example of original jpg image and Fig. 11.7.b shows the results of the image processing.

The second sub application is a GUI whose function is display the processed images and interacts with the user. The application is controlled through four buttons, the first two (top to bottom) act like the left and right mouse buttons, the third cleans the screen and the fourth toggles the force feedback. The user can move the HDS's end effector (HEE) freely in all its workspace until he presses the first button. On button press, a blue dot is drawn in the end effector position (visual feedback) and a collision detection algorithm looks for collisions between the HEE position and any trajectory inside a radius of 40 mm from the HEE center.

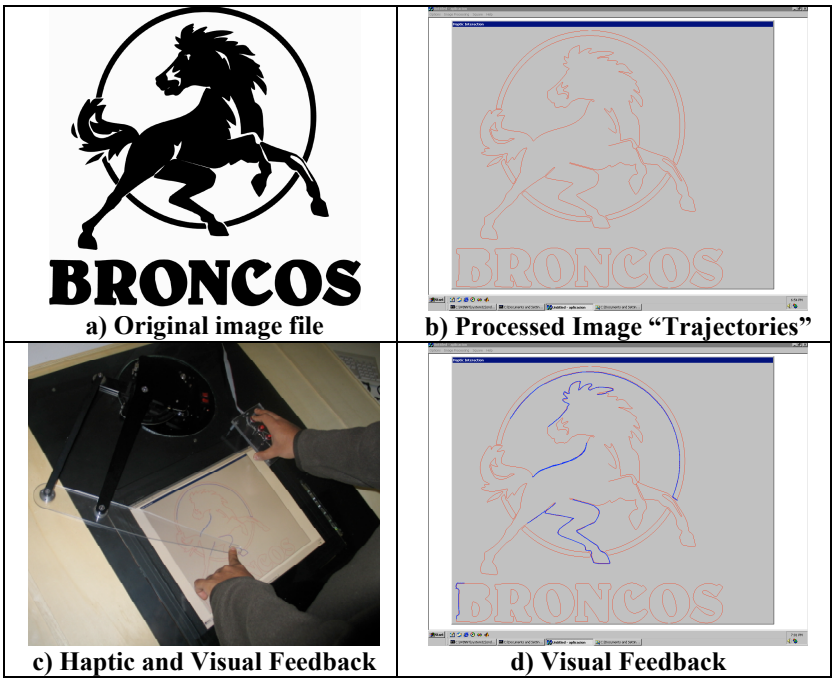
If a collision is detected, the HDS responds using force feedback to maintain the HEE constrained inside the desired trajectory. Force feedback is generated with the following control law:

$$F = -K_P \Delta - K_v \dot{\Delta} \qquad (1)$$

Where Δ is the distance between the HEE Position and its nearest point on the trajectory, and K_P and K_v should be small enough to allow pointer free movement along the trajectory while creating a force feedback outside the trajectory.

It is necessary to mention that when the HDS exerts a force of two Newtons or more, the user receives an auditory feedback.

Fig. 11.7.c shows a user interaction with the application: it is possible to see the coherence and co-location between haptic and graphical information. The visual feedback presented to the user is shown close up in Fig. 11.7.d.



a) Original image file | b) Processed Image "Trajectories"

c) Haptic and Visual Feedback | d) Visual Feedback

Fig. 10.7 The HDS live a virtual tutor.

7.6 Validating drawing skills

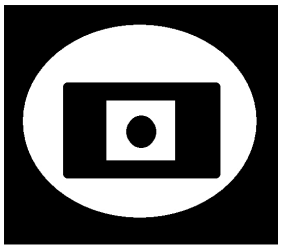
To check that the users learn sketching faster when they are provided with force feedback, we carried out the following tests:

A. Test image

The image used in the experiment is shown in the Fig. 8.b, which is based on the original image shown in the Fig. 8.a. The test image was chosen because it has the basic components of more complex designs (lines, curves, circles, squares, etc).

B. Procedure

We requested to 10 users to trace the test image seven times as precisely as possible in terms of position. Since this task should model free drawing, no restriction on completion time or effector velocity was imposed.



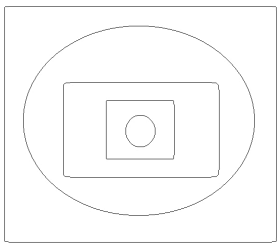


Fig. 11.8 Test Image. a) Original **b) Processed Image.**

The users were divided in two groups of 5 members. At the beginning of each experiment, the users had five minutes for drawing on the test image in order to become familiar with the HDS. The users of the first group were stimulated only with visual feedback throughout the experiment. The users of the second group were stimulated with visual feedback on every trial, and they received the force feedback only in the second, fourth and sixth trial.

We used the performance parameter described by (2) to measure if user learning of drawing skills was indeed accelerated. The P_{SD} parameter involves the time, velocity, distance and error (between the HEE Position and the point on the trajectory nearest to the HEE) when the user is drawing.

$$P_{SD} = \frac{\int_o^t |\Delta| dt}{L = \int_0^t v dt} \qquad (2)$$

C. Results

In the Fig. 9, we present the normalized sum of P_{SD} over the subjects of the two groups: the P_{SD} sum for trial 1 has been made equal to 1.

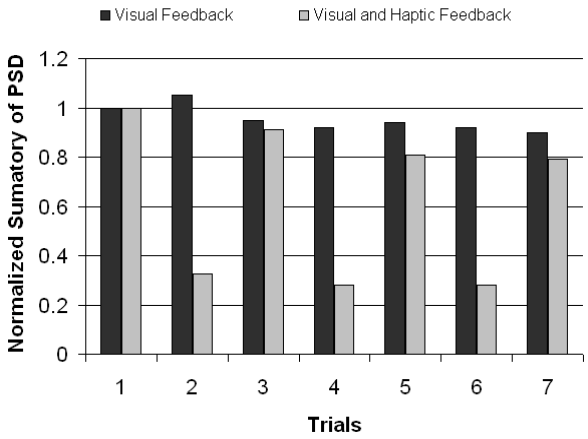


Fig. 11.9 Evolution of normalized sumatory of the P_{SD}.

We can obviously observe a strong difference between performances on trial 2, 4 and 6. What is more interesting, we can observe that for the Visual Feedback Only group, learning stops at Trial 4, while the Visual Feedback and Haptics group continues learning to Trail 7. Furthermore, at trial 7 the normalized sum of P_{SD} is lower for the Visual Feedback and Haptics group, which leads us to conclude (within the limits of a small scale experiment with a small number of subjects) that Haptic feedback enables learning for a longer time and leads to better results for the task at hand.

7.7 Conclusions and future work

In this paper, a novel multimodal device has been presented. The HDS integrates a computer and a haptic interface which replaces the conventional input devices (mouse, keyboard, etc.) for generating user experiences that lead the user to draw using the integration of different perceptual channels: haptic, visual, and audio.

We have presented two applications for the HDS: A handwriting recognition application and our first attempt to produce a drawing tutor. In order to develop a robust drawing tutor, the next step will be to integrate the recognition subsystem actually in development. The recognition subsystem is made more complex by the fact that is necessary to model and identify the observable data relevant to the teaching of drawing (i.e pressure, speed, inclinations, etc). For this motive we are building a pen with the capacity to measure some of the aforementioned variables.

7.8 Haptic Desktop characteristics Summary

The Haptic Desktop System allows both to acquire hand gesture (hand writing, drawing, painting) thanks to its position sensors and to teach gesture to the users thanks to its haptic system.
The haptic interface can generate on user's hand up to 3 N of continuous forces and 5 N of peak forces. The haptic system has two degrees of freedom (linear movements on the desktop) and it can generate a force in any direction leading on the desktop.
To detect device motion 1024cpr optical encoders with a 4X decoding were adopted. Such a choice leads to a spatial sensitivity of about 30um at workspace centre (pixel size is about 300um). Other characteristics of the device are:
Workspace: 430 x 320 mm
Reflected inertia: 0.07 Kg;
Typical Stiffness: 4 N/mm;
Position resolution: 0.01 mm.

7.9 References

[1] Kaneko K, Takashiki H., Tanie K., Komoriya K. *Bilateral control system for scaled teleoperation –improving dexterity for macro-micro teleoperation-*. Proceedings IEEE International Conference on Robotics and Automation, 1998.

[2] A. Kheddar, C. Tzafestas, P. Coiffett. *Parallel multi-robots long distance teleoperation*. International Conference on Advanced Robotics, 1997.

[3] Mussa-Ivaldi F.A., Patton J.L. *Robots can teach people how to move their arm*. Proceedings IEEE International Conference on Robotics and Automation, 2000.

[4] K. Henmi, T.Yoshikawa, H. Ueda. *Virtual Lesson and its application to virtual calligraphy system*. Proceedings IEEE International Conference on Robotics and Automation, 1998.

[5] Tsuneo Yoshikawa and Kazuyuki Henmi. *Human skill transfer using haptic virtual reality technology*, Preprints of the 6th International Symposium on Experimental Robotics, 1999.

[6] Masayasu, Sakuma, Tetsuya, Harada, *A System for japanese calligraphy lesson with force feedback on internet*. Proc. Of Dynamic Systems and Control Division (ASME), 1999.

[7] Teo C.L., Burdet E., Lim H.P. *A robotic teacher of chinese handwriting*. 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2002.

[8] MacLean K.E. *Designing with haptic feedback*. IEEE International Conference on Robotics and Automation, 2000.

[9] Reachin web site: www.reachin.se.

[10] Thomas, Massie and Salisbury, "*The PHANTOM haptic interface: a device for probing virtual objects*", In Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Chicago, IL, Nov. 1994.

[11] SenseAble Technologies Inc. 15 Constitution Way Woburn, MA 01801, USA, www.senseable.com.

[12] Bredson, Ikits, Johnson and Hansen, *"The visual haptic workbench"*, In Proceedings of Fifth PHANTOM Users Group Workshop '00, pp. 46-49, 2000.

[13] Pangaro G., Maynes-Aminzade D. and Hiroshi I. *"The actuated workbench: computer-controlled actuation in tabletop tangible interfaces"*, In Proceedings of ACM Symposium on User Interface Software and Technology '02, pp. 181-190, 2002.

[14] Noma, Yanagida and Tetsutani, *The proactive desk: a new force display system for a digital desk using a 2-DOF linear induction motor*. In Proceedings of the IEEE Virtual Reality 2003.

[15] Solis J., Marcheschi S., Portillo O., Raspolli M., Avizzano C.A., Bergamasco M. *The haptic desktop: a novel 2D multimodal device*. 13th IEEE International Workshop on Robot and Human Interactive Communication, 2004.

[16] Jansson G., Öström M. *The effects of co-location of visual and haptic space on judgments of form*. EuroHaptics 2004.

[17] Bergamasco M., Avizzano C.A. *Introduction to reactive robots*. Telematics Application Program Conference. 1998, Barcelona.

[18] Avizzano C.A. *PhD thesis: control systems of haptic interfaces*. Scuola Superiore Sant'Anna, Pisa, 2000.

[19] Avizzano C.A., Bergamasco M. *Haptic interfaces: a new interaction paradigms*. Proceedings of IROS. 1999.

[20] Solis, J., Avizzano C.A., Bergamasco M. *Teaching to write Japanese characters using a haptic interface*. 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2002.

[21] Yokokohji Y., Hollis R.L., Kanade T., Henmi K., Yoshikawa, T. *Toward machine mediated training of motor skills. Skill transfer from human to human via virtual environment*. IEEE International Workshop on Robot and Human Communication, 1996.

[22] Canny, J. *A computational approach to edge detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8. 1986.

[23] Haralick, R. and Shapiro L. *Computer and Robot Vision*. Volume I. Addison-Wesley, 1992.

[24] Pratt, W. *Digital Image Processing*. John Wiley & Sons, Inc. 1991.

8 Extraction of gestural features with the EyesWeb Platform (DIST)

Gualtiero Volpe and Barbara Mazzarino
DIST – October 2005

8.1 Theories underlying expressive gesture analysis

Analysis of gesture focuses on the information conveyed by human movement.
The information that gesture contains and conveys is often related to the affective, emotional domain. In this case gesture can be considered "expressive". Expressive gesture may convey what Cowie and colleagues (2001) call "implicit messages", or what Hashimoto (1997) calls KANSEI. In our approach expressive gesture is the instrument for communicating information we call "expressive content".
For better understanding the mechanisms of non-verbal communication and, in particular, to analyze *expressive gesture* a cross-fertilization among scientific and technical knowledge on one side, and arts and humanities on the other side is particularly indicated and useful. A deep investigation of the mechanisms of human-human communication is also needed.
From a cross-disciplinary perspective, research on expressive gesture descriptors can build on several bases, ranging from biomechanics, to psychology, to theories coming from performing arts. For example, in our work we considered theories from choreography like Rudolf Laban's Theory of Effort (Laban, 1947, 1963), theories from music and composition like Pierre Shaeffer's Sound Morphology (Shaeffer, 1977), works by psychologists on non-verbal communication in general (e.g., Argyle, 1980), on expressive cues in human full-body movement (e.g., Boone and Cunningham, 1998; Wallbott, 1980), on components involved in emotional responses to music (e.g., Scherer, 2003).
In more details, Wallbott reviewed a collection of works concerning movement features related to expressiveness and techniques to extract them (either manually or automatically). He classified these features by considering spatial, temporal, and spatio–temporal aspects; aspects related to the force of a movement, gestalt aspects, and categorical approaches.
Boone and Cunningham identified six expressive cues involved in the recognition of anger, fear, grief, and happiness. Such cues include frequency of upward arm movement, the duration of time arms are kept close to the body, the amount of muscle tension, the duration of time an individual leans forward, the number of directional changes in face and torso, and the number of tempo changes an individual makes in a given action sequence.
In his theory of Effort, Laban pointed out the dynamic nature of movement and the relationship among movement, space and time. Laban's approach is an attempt to describe, in formalized way, the characteristics of human movement without focusing on a particular kind of movement or dance expression. This is the main reason of the importance of this theory in the analysis of expressive gesture. Effort for Laban is a property of movement. From an engineering point of view we can consider it as a vector of parameters that identifies the quality of a movement performance. Such vector describes the quality of movement. Concretely, Theory of Effort is not concerned with, for example, degrees of rotation of a certain joint or the moment that has to be applied, instead it considers movement as a communication media and tries to extract parameters related to its expressive power. During a movement performance the vector describing the motion quality varies in effort space. Laban studies the possible paths followed by this vector and the intentions they can express. Therefore variations of effort during the movement performance should be studied. Laban indicates four components that generate what we call "effort space": *space, weight, time* and *fl*ow. Each component is measured on a bipolar scale, in this way every component of effort space can have values to indicates opposite quality. Laban's basic theory considers the first three factors to develop a description system for human movement. In this way we can identify eight possible combinations of the space, time and weight factors, corresponding to states that the movement can assume in its development. These eight combinations can be considered as the vertexes of a cube in the effort space whose axes are Space, Time, and Weight. The eight basic efforts and their qualities are summarized in Table 1.1.

Basic Effort	Space	Time	Weight
Pressing	*Direct*	*Sustained*	*Strong*
Flicking	*Flexible*	*Sudden*	*Light*
Punching	*Direct*	*Sudden*	*Strong*
Floating	*Flexible*	*Sustained*	*Light*
Wringing	*Flexible*	*Sustained*	*Strong*
Dabbing	*Direct*	*Sudden*	*Light*
Slashing	*Flexible*	*Sudden*	*Strong*
Gliding	*Direct*	*Sustained*	*Light*

Table 1.1: the eight basic effort and their qualities as described in (Laban, 1963)

The four Laban components can be described as follows:
a) Space: regarding space Laban identifies two different areas. The first is the Kinesphere, also known as *Personal Space*, that is the sphere of movement surrounding the dancer, centered on the Center of Gravity (CoG), and reached by normally extended limbs. The second is the *General Space*, described as the environment in which the act of movement is taking place, the area where the kinesphere is moving.
Using these two definitions the study of movement can be divided into two branches:
The movement performed inside the kinesphere
The movement of the kinesphere in the environment.

b) Time: Laban considers two aspects of time: an action can be sudden or sustained, which allows the binary description of the time component of effort space. Moreover, in a sequence of movements, each of them has a duration in time, the ratio of the duration of subsequent movements gives the time-rhythm, as in a music score.
c) Weight: It is a measure of how much strength and weight is present in a movement. For example, in pushing away a heavy object it will be necessary to use a strong weight, while in handling a delicate and light object, the weight component must be light.
d) Flow is the measure of how bound or free a movement, or a sequence of movements is.

Using these theories it is possible to identify features of motion both kinematical and expressive.
Expressive features are likely to be structured on several layers of complexity. In analysis of dance fragments with videocameras, for example, some features can be directly measured on the video frames. Others may need more elaborate processing: e.g., it may be needed to identify and separate expressive gestures in a movement sequence for computing features that are strictly related to single gestures (e.g., duration, directness).

For this reason, in the EU-IST project MEGA a conceptual framework for expressive gesture processing has been defined, structured on four layers.

Layer 1 (Physical Signals) includes algorithms for gathering data captured by sensors such as videocameras, microphones, on-body sensors (e.g., accelerometers), sensors of a robotic system, environmental sensors.

Layer 2 (Low-level features) extracts from the sensors data a collection of low-level features describing the gesture being performed. In case of dance, for example, features include kinematical measures (speed, acceleration of body parts), detected amount of motion, amount of body contraction/expansion. We will describe them in more details in the following.

Layer 3 (Mid-level features and maps) deals with two main issues in multimodal expressive gesture analysis: segmentation of the input stream (movement, music) in its composing gestures, and representation of such gestures in suitable spaces. Thus, the first problem here is to identify relevant segments in the input stream and associate to them the cues deemed important for expressive

communication. For example, in dance analysis a fragment of a performance might be segmented into a sequence of gestures where gesture's boundaries are detected by studying velocity and direction variations. Measurements performed on a gesture are translated to a vector that identifies it in a semantic space representing categories of semantic features related to emotion and expression. Sequences of gestures in space and time are therefore transformed in trajectories in such a semantic space. Trajectories can then be analyzed e.g., in order to find similarities among them and to group them in clusters. In Figure 3 an example of such process is shown: gestures are represented in a 2D space whose X axis represents Quantity of Motion (i.e., the amount of detected motion) while Y axis is Fluency.

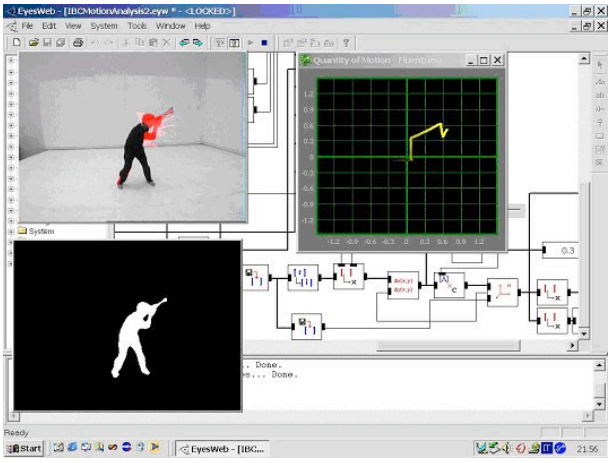


Figure 1.2. Gesture represented as trajectory in a 2D space (X axis: Quantity of Motion, Y axis: Fluency).

Layer 4 (Concepts and structures) is directly involved in data analysis and in extraction of high-level expressive information. In principle, it can be conceived as a conceptual network mapping the extracted features and gestures into (verbal) conceptual structures. For example, a dance performance can be analyzed in term of the performer's conveyed emotional intentions, e.g., the basic emotions anger, fear, grief, and joy. However, other outputs are also possible: for example, a structure can be envisaged describing the Laban's conceptual framework of gesture Effort, i.e., Laban's types of Effort such as "pushing", "gliding", etc. (Laban, 1947, 1963). Experiments can also be carried out aiming at modeling spectators' engagement.

8.2 Feature extraction

Starting form the framework described in the previously it is possible to organize gesture analysis in subsequent steps of increasing complexity starting from the physical sphere of the motion toward the affective sphere of information conveyed by gesture.
All the algorithms described in this paragraph are developed as software modules for the EyesWeb open platform (www.eyesweb.org) and, in particular, are included in the EyesWeb Expressive Gesture Processing Library.

8.2.1 Layer 1:

We start from a stream of noising data, and in this layer we apply synchronization and filtering of data. It is important to notice that different applications need different sensors, but the subsequent algorithms are usually the same. It is at this level that we customize the analysis to the application. There are different typologies of possible sensors supported by the EyesWeb platform, e.g. from serial, USB, or parallel interfaces. Some examples of input blocks are displayed in table 1.2.1.

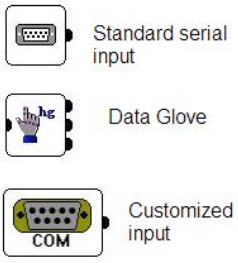
Networking protocols	Audio input	Video input	Other devices
			

Table 1.2.1 A screenshot of EyesWeb input modules.

The platform, obviously, provides tools for filtering data in the time or frequency domain.

8.2.2 Layer 2:

The data at this point are elaborated in order to detect kinematical information, and low lever motion features. Pointing to Laban Effort theory at this level we have two major analysis directions:

- The analysis of the personal space and
- The analysis of the general space.

8.2.2.1 GENERAL SPACE

For this analysis we look at the interaction between the subject and the environment, between different subjects, between different subjects and the environment, or between subjects and spectators.
In the analysis at this layer what we extract are either kinematical features of the movement of the kinesphere or just the kinematical features of the kinesphere's CoG of the subject, from an upper point of view. From these features, we perform a trajectory analysis of the subject given also information about the possible interactions between subjects.
For clarifying the underlying idea it is useful to focus the attention to the use of the stage. In a theatre representation it is important the use of the stage: if the actor, for example, speaks in front of the public or back to the stage (in general or with respect to another actor) the meaning of the words is differently interpreted by the spectator, the impact is different. We call these regions meaningful regions, i.e., regions on which a particular focus is placed.
Our general space analysis is based on a model considering a collection of discrete potential functions. The space is divided into active cells forming a grid. A point moving in the space is considered and tracked. Three main kinds of potential functions are considered: (i) potential functions **not** depending on the current position of the tracked point, (ii) potential function depending on the current position of the tracked point (see for example Fig. 1.3), (iii) potential functions depending on the definition of regions inside the space.
Using the potential function independent from the position we can model meaningful regions or objects (such as fixes scenery furniture).
Interaction between potentials is used to model interaction between (real or virtual) objects and subjects in the space.

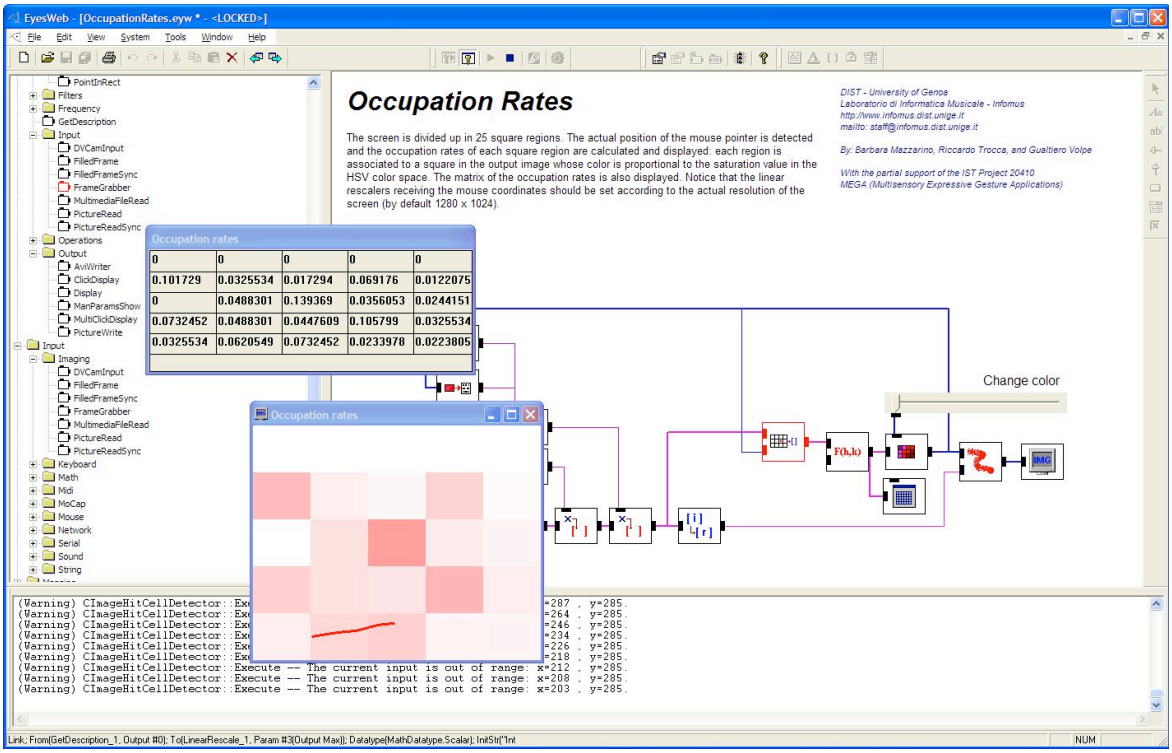
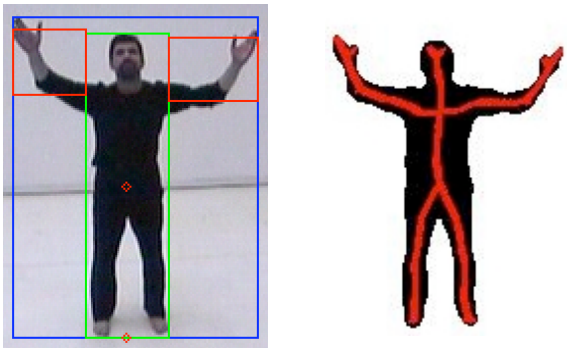


Figure 1.3. Occupation rates in a stage divided in actives cells.

8.2.2.2 PERSONAL SPACE

In this case we focus the attention on each subject.
Here we want to extract how a subject is moving in his/her kinesphere. From a general analysis we look at the full body, but for different applications we can concentrate on the movement of a limb or the head or something else. Figure 1 shows two examples of features.



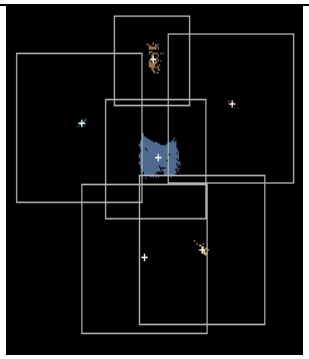
(a) (b)

Figure 1.4. Examples of low-level motion cues.

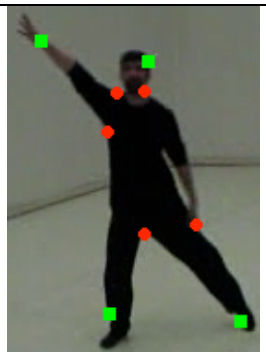
In the figure on the left some sub-regions of the body are individuated together with the body CoG. The temporal evolution of both sub-regions and the CoG can be analyzed. In the figure on the right the skeleton of the performer extracted in real-time is represented. Just for giving an overview about the principal algorithms included in the EyesWeb Expressive Gesture Processing Library for this layer some of them are described below.

Silhouette segmentation:
Using background subtraction techniques the silhouette of a subject can be extracted. To the Silhouette image statistical methods can be applied in order to identify the position of the CoG. As example

consider the image in Fig1 left part. For the identification of the CoG, the part of the blob containing the arms can be removed. It is possible to do this by evaluating the percentage of the area covered by arms with respect to the total body, without applying any marker or active sensor on the body. After the identification of the correct CoG, it is possible to identify the torso and limbs regions by looking at the projection on the axes.



(a)



(b)

Figure 1.5. The output of some EyesWeb modules for 2D and 3D tracking: (a) Color blob tracking; head, hands, and feet are tracked; the rectangles around each point represent the areas where the module will look for that point in the next frame. (b) Limbs' extreme points (squares) and convexity defects (circles): both are based on body contour processing. Note that in Figure 1.5b the left hand is labeled as a convexity defect. In fact, the left arm and hand lean on the body and the hand is the point of the body contour locally at the maximum distance from the body convex hull (i.e., it is a convexity defect).

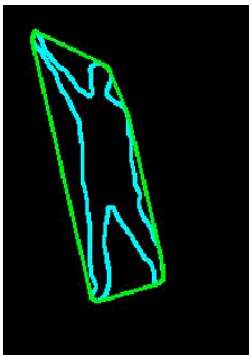
Silhouette shape orientation and representation:

Using simple image processing it is possible to evaluate the contour of the Silhouette. The representation of the body can also be not anthropomorphic: for example, it is possible to represent the Silhouette as an ellipse (fig. 1.6.a) and evaluate the orientation of the body looking at the axes. It is interesting also to use the convex hull (fig. 1.6 b) of the body. This representation maintains information about the vertex of the silhouette blob.

Combining the previous descriptions of the shape of the body it is possible to evaluate special points that allow us to estimate the position of the head, feet, hands and so on (see fig 1.5 (b)).

Another representation of the body, or part of it, available in the EyesWeb Gesture Processing Library is the skeleton (fig 1.4 b). Also this algorithm is used for localization of the extremity of the body.

(a)



(b)

Figura1.6 **(a) Ellipse (b) Convex hull**

Quantity of Motion

The QoM represents the amount of detected movement. It is based on the Silhouette Motion Images, and its evolution in time can be seen as a sequence of bell-shaped curves (motion bells)

In order to have a subject-independent measure a scaling of the SMI value is applied. Using different SMIs it is also possible to evaluate both internal and external motion.

The formula used for the QoM is the following:

$$QuantityOfMotion=Area(SMI[t, n])/Area(Silhouette[t])$$

Using this formula the measure is also independent from the distance from the camera (in a range depending on the resolution of the videocamera), and it is expressed in terms of fractions of the body area that moved.

Contraction Index:

This measure, ranging from 0 to 1, is related to the use of the kinesphere. The algorithm to compute the CI combines two different techniques dependent on the representation of the kinesphere:

- the individuation of the ellipse approximating the body silhouette and
- computations based on the bounding region.

The former is based on an analogy between image moments and mechanical moments: in this perspective, the three central moments of second order build the components of the inertial tensor of rotation of the silhouette around its center of gravity: this allows to compute the axes (corresponding to the main inertial axes of the silhouette) of an ellipse that can be considered as an approximation of the silhouette: eccentricity of such an ellipse is related to contraction/expansion. The second technique used to compute CI relates to the bounding region, i.e., the minimum rectangle surrounding the subject's body. The algorithm compares the area covered by this rectangle with the area actually covered by the silhouette. Intuitively, if the limbs are fully stretched and not lying along the body, this component of the CI will be low, while, if the limbs are kept tightly nearby the body, it will be high (near to 1).

8.2.2.3 TRAJECTORY ANALYSIS

We have collected in the trajectory analysis sub library all the methods for extracting features from trajectories in 2D space. These features can be related both to general space and to personal space analysis. Obviously it is necessary, for personal space analysis, to identify special points to track.

At the moment, the EyesWeb Expressive Gesture Processing Library contains several modules for 2D and 3D tracking: besides some method sketched above (e.g. body skeleton, sub-regions modules, modules for extracting convexity defects, modules for computing limbs' extreme points based on the body contour…), other available modules include a color blob tracker, Lucas-Kanade algorithm, modules for estimating future positions of tracked points (e.g., based on Kalman filtering).

From the selected trajectory features can be extracted. In this layer the features are related to the kinematics domain such as velocity and acceleration.

A meta-layer of analysis gathers and integrates the data coming from such modules, by applying suitable rules and heuristics, to provide an approximation of the current position of the tracked points. Similarly, 2D tracking information coming from each videocamera is integrated in order to provide 3D positions of the tracked points. Even if the tracking precision is lower than in commercial dedicated motion tracking systems, the advantages (e.g., markerless system) and the obtained data are reliable enough for expressive gestural control in several multimodal interactive applications and in tangible acoustic interfaces.

8.2.3 Layer3

This layer receives data from Layers 1 and 2 and has two main tasks: segmenting expressive gestures and representing them in a suitable way. Such a representation would be the same (or at least similar) for gestures in different channels, e.g., for expressive gestures in music and dance. Data from several different physical and virtual sensors are therefore likely to be integrated in order to perform such a step. Each gesture is characterized by the measures of the different features extracted in the previous step (e.g., speed, impulsiveness, directness, etc. for movement, loudness, roughness, tempo, etc. for music). Segmentation is a relevant problem at this level: the definition of expressive gesture does not help in finding precise boundaries.

Motion segmentation:
A straightforward way to individuate movement strokes and therefore to segment movement in motion and pause phases is to apply a threshold on the detected energy or amount of movement. As a first approximation, the QoM measure has been therefore used to perform such segmentation. In order to segment motion, a list of these motion bells has been extracted and their features (e.g., peak value and duration) computed. For this task an empirical threshold can be defined on the QoM: for example, according to a threshold that has been used in several applications, the dancer is considered to be moving if the area of his/her motion image (i.e., the QoM) is greater than 2.5% of the total area of the silhouette. Figure 1.7 shows motion bells after automated segmentation: a motion bell characterizes each motion phase.

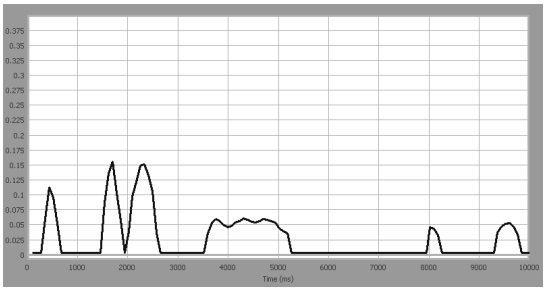


Figure 1.7: motion segmentation

Fluency and Impulsiveness
They are related to Laban's Flow and Time axes.
Fluency can be estimated starting from an analysis of the temporal sequence of motion bells. A dance fragment performed with frequent stops and restarts (i.e., characterized by a high number of short pause and motion phases) will result less fluent than the same movement performed in a continuous, "harmonic" way (i.e., with a few long motion phases). The hesitating, bounded performance will be characterized by a higher percentage of accelerations and decelerations in the time unit (due to the frequent stops and restarts), a parameter that has been demonstrated of relevant importance in motion flow evaluation.
A first measure of impulsiveness can be obtained from the shape of a motion bell. In fact, since QoM is directly related to the amount of detected movement, a short motion bell having a high peak value will be the result of an impulsive movement (i.e., a movement in which speed rapidly moves from a value near or equal to zero, to a peak and back to zero). On the other hand, a sustained, continuous movement will show a motion bell characterized by a relatively long time period in which the QoM values have little fluctuations around the average value (i.e., speed is more or less constant during the movement).
Is possible evaluate also the Directness Index (DI), calculated as the ratio between the length of the straight trajectory connecting the first and the last point of a motion trajectory and the sum of the lengths of each segment constituting the trajectory.

Gesture Representation
Referring to Camurri et al. a possibility for representing gesture consists in producing a symbolic description of the analysed sequence of movements. For example, depending on the Contraction Index a motion phase can be seen as a contraction phase (if the value of CI at the end of the phase is higher than the one at the beginning) or as an expansion phase. It is therefore possible to obtain a description like the following one:

Contraction(Start_Frame,Stop_Frame,Initial_Value_CI, Final_Value_CI, other cues…)
Expansion(Start_Frame,Stop_Frame,initial_Value_CI, Final_Value_CI, other cues…)

Another possibility is to build a representation in terms of points or trajectories in multidimensional semantic spaces, i.e., spaces whose axes are expressive cues having a relevant influence with respect to the conveyed expressive content.

Posture recognition

Methods for posture recognition are included in this layer. In the Gesture Processing Library there are different modules and patches for this task, using pattern matching or Support Vector Machines (see figure 1.8).

Another method, robust enough to be employed in real-time performances, is based on Hu moments (Hu, 1962), a set of seven moments, which are translation, scale and rotation invariant, and have been widely used in computer vision for shape discrimination.
The algorithm employs a nearest-neighbour technique. For each considered (normalised) posture Hu moments are calculated and stored in a matrix. During each pause phase, Hu moments are calculated on the incoming (normalised) silhouette. Euclidean distances are computed between the Hu moments of the silhouette in the current frame and the Hu moments of each candidate posture.

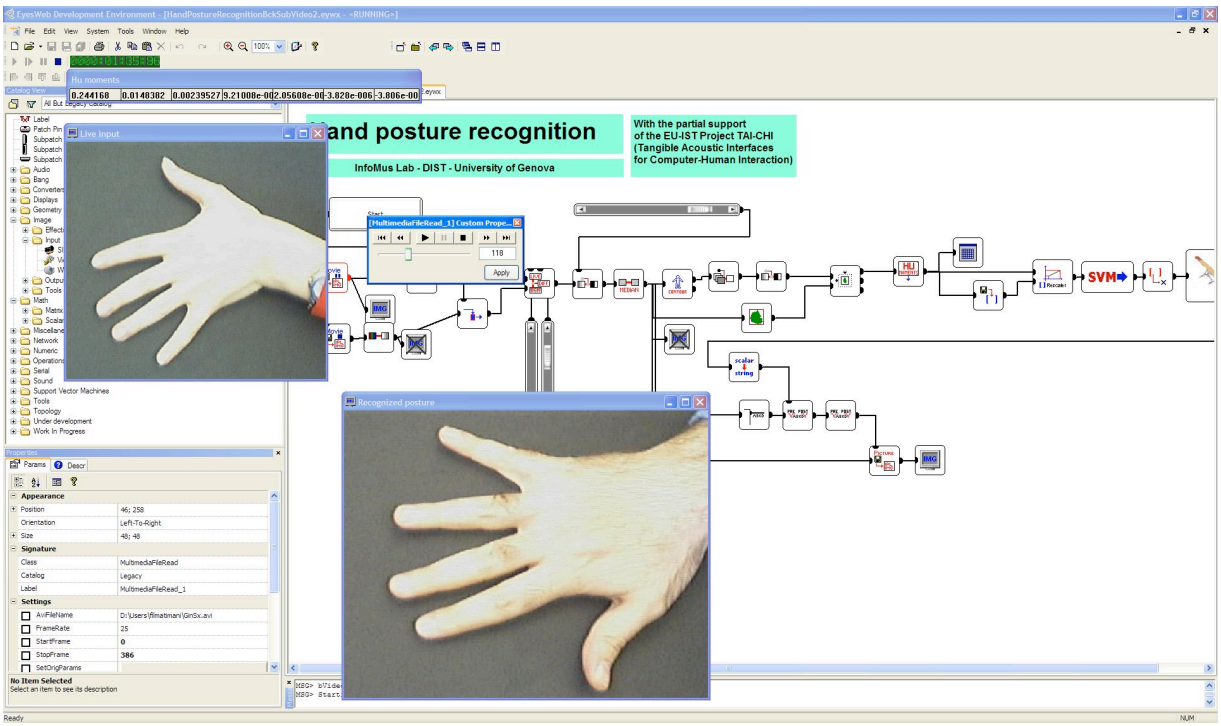


Figure 1.8 Gesture recognition algorithm using Hu moments and Support Vector Machines. The first image (top left site) is the input frame from the camera. The second image (bottom centre site) is the recognize posture.

8.2.4 Layer 4

Layer 4 collects inputs mainly from Layers 2 and 3 and is responsible to extract high-level expressive content form expressive gestures. It can be organized as a conceptual network mapping the extracted features and gestures into (verbal) conceptual structures. Several different outputs are possible: for example, a structure could be envisaged describing the Laban's conceptual framework of gesture Effort, i.e., Laban's types of Effort such as "pushing", "gliding", etc. Several different machine learning techniques can be used for building such structure, ranging from statistical techniques like multiple regression and generalized linear techniques, to fuzzy logics or probabilistic reasoning systems such as Bayesian networks, to various kinds of neural networks (e.g., classical back-propagation networks, Kohonen networks), support vector machines, decision trees.

8.3 References

Argyle M. (1980), “Bodily Communication”, Methuen & Co Ltd, London.

Camurri A., Mazzarino B., Ricchetti M., Timmers R., Volpe G. (2004a), “Multimodal analysis of expressive gesture in music and dance performances”, in A. Camurri, G. Volpe (Eds.), Gesture-based Communication in Human-Computer Interaction, LNAI 2915, Springer Verlag.

Camurri A., Mazzarino B., Volpe G. (2004b), “Expressive interfaces”, Cognition, Technology & Work, 6(1): 15-22, Springer-Verlag.

Camurri A., Mazzarino B., Menocci S., Rocca E., Vallone I., Volpe G. (2004c), “Expressive gesture and multimodal interactive systems”, in Proc. AISB 2004 Convention: Motion, Emotion and Cognition, Leeds, UK.

Camurri A., De Poli G., Friberg A., Leman M., Volpe G. (2004d), “The MEGA project: analysis and synthesis of multisensory expressive gesture in performing art application”, submitted.

Camurri A., Coletta P., Massari A., Mazzarino B., Peri M., Ricchetti M., Ricci A., Volpe G. (2004e) “Toward real-time multimodal processing: EyesWeb 4.0”, in Proc. AISB 2004 Convention: Motion, Emotion and Cognition, Leeds.

Camurri A., Lagerlöf I., Volpe G. (2003a) “Recognizing Emotion from Dance Movement: Comparison of Spectator Recognition and Automated Techniques”, International Journal of Human-Computer Studies, 59(1-2): 213-225, Elsevier Science.

Camurri A., Mazzarino B., Volpe G., Morasso P., Priano F., Re C. (2003b) “Application of multimedia techniques in the physical rehabilitation of Parkinson's patients”, Journal of Visualization and Computer Animation, 14(5): 269-278, Wiley.

Camurri A., Hashimoto S., Ricchetti M., Trocca R., Suzuki K., Volpe G. (2000) “EyesWeb – Toward Gesture and Affect Recognition in Interactive Dance and Music Systems” Computer Music Journal, 24(1): 57-69, MIT Press.

A. Camurri and G. Volpe, editors. Gesture-Based Communication in Human-Computer Interaction, 5th International Gesture Workshop, GW 2003, Genova, Italy, April 15-17, 2003, Selected Revised Papers, volume 2915 of Lecture Notes in Computer Science. Springer, 2004.

Cowie R., Douglas-Cowie E., Tsapatsoulis N., Votsis G., Kollias S., Fellenz W., Taylor J. (2001), Emotion Recognition in Human-Computer Interaction. IEEE Signal Processing Magazine, 1.

C. Brockmann and H. Müller. Remote vision-based multi-type gesture interaction. In A. Camurri and G. Volpe, editors, 5th International Gesture Workshop, GW 2003, Genova, Italy, pages 198–209. Springer-Verlag Heidelberg, April 2003.

Dahl S., Friberg A. (2004) “Expressiveness of musician’s body movements in performances on marimba” in A. Camurri, G. Volpe (Eds.), Gesture-based Communication in Human-Computer Interaction, LNAI 2915, Springer Verlag.

Hashimoto S. (1997), “KANSEI as the Third Target of Information Processing and Related Topics in Japan”, in Camurri A. (Ed.) “Proceedings of the International Workshop on KANSEI: The technology of emotion”, AIMI (Italian Computer Music Association) and DIST-University of Genova, 101-104.

Kurtenbach G., Hulteen E. (1990), “Gestures in Human Computer Communication”, in Brenda Laurel (Ed.) The Art and Science of Interface Design, Addison-Wesley, 309-317.

Picard R. (1997), “Affective Computing”, Cambridge, MA, MIT Press

Pollick F.E. (2004), “The Features People Use to Recognize Human Movement Style”, in A. Camurri, G. Volpe (Eds.), Gesture-based Communication in Human-Computer Interaction, LNAI 2915, Springer Verlag.

Laban R. (1963), “Modern Educational Dance”, Macdonald & Evans Ltd., London.

Rowe R. (2001), “Machine Musicianship”, Cambridge MA: MIT Press.

Rowe R. (1993), “Interactive music systems: Machine listening and composition”, Cambridge MA: MIT Press.

Wallbott H.G. (1980), “The measurement of Human Expressions”, in Walbunga von Rallfer-Engel (Ed.) Aspects of communications, 203-228.

G. McGlaun, F. Althoff, M. Lang, and G. Rigoll. Robust video-based recognition of dynamic head gestures in various domains – comparing a rule-based and a stochastic approach. In A. Camurri and G. Volpe, editors, 5th International Gesture Workshop, GW 2003, Genova, Italy, pages 180–197. Springer-Verlag Heidelberg, April 2003.

G. S. Schmidt and D. H. House. Model-based motion filtering for improving arm gesture recognition performance. In A. Camurri and G. Volpe, editors, 5th International Gesture Workshop, GW 2003, Genova, Italy, pages 210–230. Springer-Verlag Heidelberg, April 2003.

Anne-Marie Burns, Barbara Mazzarino. Finger Tracking Methods Using EyesWeb. Lecture Notes in Artificial Intelligence, Springer-Verlag 2005 in press.

9 Gesture Extracted features and Musical control (DEI, INPG)

Paper proposed to the Enactive Conference 2005 under the title
"Towards a multi-layer architecture for multi-modal rendering of expressive actions"
DEI: G. De Poli, F. Avanzini A. Roda, L. Mion, G. D'Incà, C. Trestino, D. Pirro
INPG: A. Luciani, N. Castagné.

9.1 Abstract

Expressive content has multiple facets that can be conveyed by music, gesture, actions. Different application scenarios can require different metaphors for expressiveness control. In order to meet the requirements for flexible representation, we propose a multi-layer architecture structured into three main levels of abstraction. At the top (user level) there is a semantic description, which is adapted to specific user requirements and conceptualization. At the other end are low-level features that describe parameters strictly related to the rendering model. In between these two extremes, we propose an intermediate layer that provides a description shared by the various high-level representations on one side, and that can be instantiated to the various low-level rendering models on the other side. In order to provide a common representation of different expressive semantics and different modalities, we propose a physically-inspired description specifically suited for expressive actions.

9.2 Introduction

The concept of expression is common to different modalities: one can speak of expression in speech, in music, in movement, in dance, in touch, and for each of these context the word expression can assume different meanings; this is the reason why expression is an ill-defined concept. In some contexts expression refers to gestures that sound natural (human-like), as opposed to mechanical gestures. As an example, see [11], [9], [10], [3], [4] for musical gestures and [1], [12] for movements.

In other contexts, expression refers to different qualities of natural actions, meaning with this that gestures can be performed following different expressive intentions which can be related to sensorial or affective characteristics. As an example, see [18], [13] for musical gesture, and [15], [16] for movements. These works have shown that this level of expression has a strong impact on non verbal communication, and have led to interesting multimedia applications and to the development of new types of human-computer interfaces.

In this paper we will stick to this latter meaning of expression, therefore when speaking of expression we refer to the deviations from a natural performance of a gesture or action. In section 2 and 3 we will discuss the expressive content in actions from a multimodal perspective. In a rendering system, different application scenarios require different metaphors for expressiveness control. On the other hand, achieving coherence in a multimodal rendering context requires an integrated representation. In order to meet the requirements for flexible and unified representation, we propose in section 4 a multi-layer architecture which comprises three main levels of abstraction. In order to provide a shared representation of different expressive semantics and different modalities, we propose a physically-inspired description which is well suited to represent expressive actions. Some examples and applications are presented in section 5.

9.3 Multimodal perception and rendering

Looking at how multi-modal information is combined, two general strategies can be identified: the first is to maximize information delivered from the different sensory modalities (sensory combination),

while the second is to reduce the variance in the sensory estimate in order to increase its reliability (sensory integration). Sensory combination describes interactions between sensory signals that are not redundant: they may be in different units, coordinate systems, or about complementary aspects of the same property. By contrast sensory integration describes interactions between redundant signals.
Disambiguation and cooperation are examples for these two interactions: if a single modality is not enough to come up with a robust estimate, information from several modalities can be combined. For example, for object recognition different modalities complement each other with the effect of increasing the information content.

The amount of cross-modal integration depends on the features to be evaluated or the tasks to be accomplished. The modality precision (or appropriateness) hypothesis is often cited when trying to explain which modality dominates under what circumstances. The hypothesis states that discrepancies are always resolved in favor of the more precise or more appropriate modality. In spatial tasks, for example, the visual modality usually dominates, because it is the most precise at determining spatial information. For temporal judgments, however, the situation is reversed and audition, being the more appropriate modality, usually dominates over vision. In texture perception haptics dominates on other modalities, and so on.

When we deal with multimodal rendering of expression in actions, we are interested not only in a fusion at the perceptual level, but also in the modeling and representation level. The architecture of the system should be specifically designed for this purpose, taking into account this problem. Normally a combination of different models, one for each modality, is used. These models map directly intended expression on low level parameters of the rendering system. We believe that a proper definition of a common metaphoric level is a fundamental step for the development of effective multimodal expression rendering strategies.

9.4 Expression in different modalities

A second point to be addressed when looking for a better definition of expression is the wide range of expressive gestures that are studied in the literature. Roughly, we can identify studies on three level of gestures: single gestures (see [6], [7]), simple patternbased gestures (see e.g. [5], [8], [1]), and structured gestures (see [13], [18] for musical gesture, and [15], [17] for movement). We can think about analogies between music and movement with reference to these three levels of structural complexity. By single gestures we intend single tones for music or simple movements like arm rotation. These single gestures represent the smallest non structured actions, which combined together form simple patterns. Single patterns in music can be represented by scales or repetition of single tones, while example of basic patterns in movement are a subject walking or turning. Highly structured gestures in music are performances of scores, while in movement we can think about a choreography. This classification yields interesting analogies between the different structures of gestures in music and dance, and provides a path to a common representation of different expressive semantics.

The literature on expressiveness analysis and rendering exhibits an evident lack of research on the haptic modality with respect to the visual and audio modalities. This circumstance can be explained by observing that the haptic modality does not present a range of structured messages as wide as for audio and video (e.g., music or speech, and dance, respectively). In fact, due to the very nature of haptic perception, haptic displays are strictly personal and are not suitable for communicating information to an audience. This is why just a very few kinds of structured haptic languages have been developed along the history. The haptic modality is indeed hugely important in instrument playing for controlling the expressive content conveyed by other modalities, as shown for example by the haptic interaction between a player and a violin, which quality affects deeply the expressive content in the sound. On the contrary tactile-kinesthetic perception, despite its importance in the whole multisensory system, does not seem to convey expressivity back to the player [31].

9.5 An architecture for multi-modal expressive rendering

In order to meet the requirements for flexible representation of expressiveness in different application scenarios, we propose a multi-layer architecture which comprises three main levels of abstraction. At the top there is a semantic description, which stays at the user level and is adapted to a specific representation: for example, it should be possible to use a categorical approach (with affective or sensorial labels) or a dimensional approach (i.e. the valence-arousal space) [36].

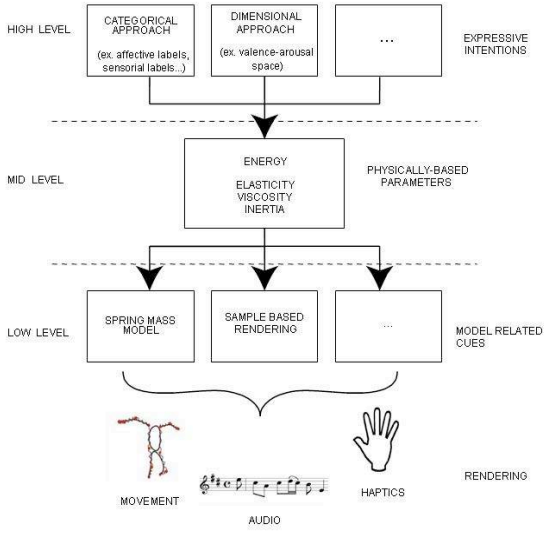


Figure 1: Multi-layer architecture

At the other end are low-level features that describe parameters strictly related to the rendering models. Various categories of models can be used to implement this last level. Sticking to the musical example at this level, signal-based sound synthesis models are adapted to represent note onset, duration, intensity, decay, etc. As depicted by Cadoz *et al.* in [32], physical models can be adapted to render timbre characteristics, interaction properties (collision, friction), dynamic properties as transients (attacks), evolution of decays (laws of damping), memory effects (hysteretic effects), energetic consistency between multisensory phenomena, etc.

Physical-modeling techniques have been investigated for years and have proven their effectiveness in rendering rich and organic sounds [2]. Among such rendering techniques, one of the models that are best suited for controlling expressiveness is made of a network of masses and interactions [32]. Basic physical parameters of the masses and interactions (damping, inertia, stiffness, etc.) determine the behavior of the model. A change in parameters affects the audio rendering, and especially its expressive content.

In between these two extremes, an intermediate layer provides a description that can be shared by the various high-level representations on one side, and can be instantiated to the various low-level rendering models on the other side. In order to provide a common representation of different expressive semantics and different modalities, we propose a physically-based description. For the definition of the intermediate level we need the different modalities to converge towards a common description. In this case, we want this description of the actions (movements, objects and so on) to be based on a physical metaphor. This choice arises from the fact that expressive contents are conveyed by gestures, which are essentially physical events. Therefore, direct or indirect reference to human physical behavior can be a common denominator to all the multi-modal expressive actions and yield a suitable representation.

Using a single model for generating the various categories of phenomena allows to enhance energetic coherency among phenomena [30]. Furthermore, such a physically-based mid-level description is

shifted towards the "source side", which is better suited for multi-modal rendering. This amounts to making a shift from existing rendering techniques which are derived from perceptual criteria (at the 'receiver side') and are therefore referred to a specific modality or medium (e.g., music).

The main effort needed at this point is to define a suitable space for this physical metaphor-based description. We have a set of dimensions which describe actions by metaphors. This space must be described by mid-level features, which provide the overall characteristics of the action. As an example, consider a pianist or a dancer who wants to communicate, during a performance, an intention labeled as "soft" (in a categorical approach). Each performer will translate this intention into modifications of his action in order to render it softer, e.g. by taking into account the attack time of single events (such as notes or steps). The actions will therefore be more "elastic" or "weightless". These and other overall properties (like "inertia" or "viscosity"), together with "energy" (used as a scale factor), will be taken into account to define the mid-level description. Citing Castagné, "though users are not commonly confronted in an intellectual manner with the notions of inertia, damping, physical interaction etc., all these notions can be intuitively apprehended through our body and our every-day life" [34].

This kind of multi-layered approach is exemplified in figure 1.

9.6 Experiments on expression mapping

Previous experiments conducted at CSC/DEI in Padova led to interesting results on automatic detection of expression for different types of gestures.
These studies showed that the expressive content of a performance can be changed, both at the symbolic and signal levels. Psychophysical studies were also conducted in order to construct mappings between acoustic features of sound events and the characteristics of the physical event that has originated the sound in order to achieve an expressive control of everyday sound synthesis.

9.6.1 Mid-level feature extraction from simple musical gestures

Several experiments on analysis of expression on simple pattern-based musical gestures have been previously carried out. In [5] short sequences of repeated notes recorded with a MIDI piano were investigated, while [19] reports upon an experiment on expression detection on audio data from professional recordings of violin and flute (single repeated notes and short scales). In both works, the choice of the adjectives describing the expressive intention has been considered as an important step for the success of the experiments. In [5], the choice of adjectives has been based on theories of Imberty [20] and Laban [21]. Laban believed that the expressive content of every physical movement is mainly related to the way of performing it, and it is due to the variation of four basic factors: time, space, weight and flow. The authors defined as *basic efforts* the eight combinations of two values (quick/sustained, flexible/direct and strong/light) associated with the first three factors. Each combination gives rise to a specific expressive gesture to which is associated an adjective, as an example a slashing movement is characterized by a strong weight, quick time and flexible space (i.e., a curved line).

It was supposed that sensorial adjectives could be more adequate for an experiment on musical improvisations, since they suggest a more direct relation between the expressive intention and the musical gestures. Starting from Laban theory of expressive movement, the set of adjectives for our experiments was derived by analyzing each of the eight combinations of the values *high* and *low* assigned to articulation, intensity and tempo (velocity). Both value series [quick/sustained, flexible/direct and strong/light] and [articulation, intensity and tempo] have a physical base and can be related to the concepts of energy, inertia, elasticity and viscosity.

Factor analysis on the results of a perceptual test indicated that the sonological parameters tempo and intensity are very important in perceiving the expression of this pattern-based musical gestures. Also,

results of a perceptual test showed that listeners can recognize performer's expressions even when very few musical means are used.

Results of analysis were used to tune machine learning algorithms, to verify their suitability for automatic detection of expression. As an example, we used Bayesian networks and a set of HMMs able to give as output the probability that the input performance was played according to an expressive intention [22]. High classification ratings confirmed that automatic extraction of expression from simple pattern-based musical gestures can be performed with a mid-level description.

9.6.2 Mid-level feature extraction from complex musical gestures

In [27] we showed that a musical performance with a defined expressive intention can be automatically generated by modifying a natural performance of the same musical score. This requires a computational model to control parameters such as amplitude envelope, tempo variation (e.g. accelerando, ritardando, rubato), intensity variation (e.g. crescendo, decrescendo), articulation (e.g. legato, staccato), by means of a set of profiles. A family of curves, which presents a given dynamic evolution, is associated to every expressive intention. Fig.2 shows an example of curves for the control of amplitude envelopes. These curves present strict analogies with motor gestures, as already highlighted by various experimental results (see [28], [29], [10] among others) and the concepts of inertia, elasticity, viscosity and energy can be therefore easily related to them.

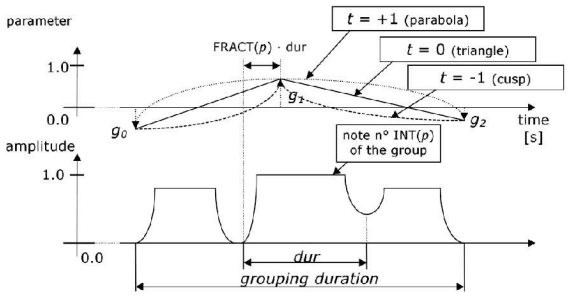


Figure 2: Curves to control the amplitude envelope of a group of notes.

9.7 Mid-to low-level mappings

As already mentioned, the main open issue for the realization of the multi-layer architecture proposed in this paper is the definition of mappings from the intermediate, shared representation and the low-level features that describe parameters strictly related to the rendering models. In this section we analyze two relevant examples of such mappings.

9.7.1 Ecological mapping

Many studies in ecological acoustics address the issue of the mapping between acoustic features of sound events and the characteristics of the physical event that has originated the sound [23]. As an example, it has been found that the material of a struck object can be reliably recognized from the corresponding impact sound. In previous studies we have developed a library of physically-based sound models based on modal synthesis [24], that allow simulation of many typologies of such "everyday sounds" and specifically contact sounds (impact, friction, bouncing, breaking, rolling, and so on). Using these models we have conducted a number of psychophysical studies in order to construct mappings between the "ecological level", e.g. object material, or hardness of collision, or viscosity in motion, and the low-level physical parameters of the sound models (see e.g. [25] and [26]). Such an "ecological-to-physical" mapping can be straightforwardly incorporated into the multi-layer architecture that we propose in this paper, where the ecological level corresponds to the mid-level

physically-based parameters which maps to the low-level parameters of the modal synthesis models. In this way we realize expressive control of everyday sound synthesis.

9.7.2 Physically-based expression rendering

In [35] Cadoz demonstrated that physical modeling is suited not only for sound synthesis but also for the synthesis of musical gesture and musical macroevolution. As explained in that paper, one can obtain a succession of sound events rather than isolated sounds by assembling both high and low frequency mass-interaction physical models into a complex structure. The low frequency structure then stands for a modelling and simulation of instrumental gesture.

In this process, low frequency models are slightly perturbed in a natural manner through feedback from the sound models. Therefore the generated sound events present convincing short-term evolutions, expressiveness and musicality, such as changes in a rhythm or in the timbre of successive musical events – somehow resembling the way a musician would behave.

In motion control and modelling, physically-based particle models can be used to simulate a human body, not as a realistic biomechanical model, but rather as an abstract minimal representation that allows access to the control of the quality of dance motions as they are thought and experienced by dancers during the performance and by teachers [1]: motors of motion, propagation, external resistance, full momentum transfers, etc. This minimal model produces the quality of the motion in a “natural way of performance and thinking” (figure 3 left). In a similar way, Luciani used this type of model in [33] to control the expressive evolution in visual animation as shown in figure 3 right).

Thus, by implementing the middle level of figure 1 through mass-interaction models that stand for musical gesture generators, and by controlling the physical parameters of these models through outputs of the first semantic level, it becomes possible to control the quality of the “instrumental gesture”. The instrumental gesture model will then generate accordingly musical events that have some expressive content, and will be mapped onto the last audio rendering level.

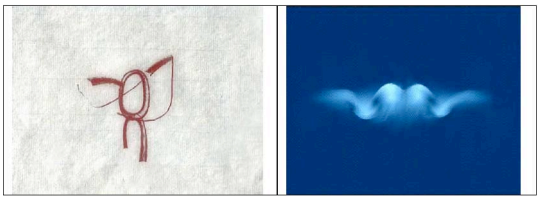


Figure 3. Physically-based particle model for dance and animation

9.8 Expression rendering systems

In this section we show some concrete examples of instantiation of the proposed architecture, with reference to the models described in previous section.
Our studies on music performances [13] have shown that the expressive content of a performance can be changed, both at the symbolic and signal levels. Models able to apply morphing among performances with different expressive content were investigated, adapting the audio expressive character to the user desires.

Figure 4. The expressive movements of a dancer control a musical performance

The input of the expressiveness models are composed of a musical score and a description of a neutral musical performance. Depending on the expressive intention desired by the user, the expressiveness model acts on the symbolic level, computing the deviations of all musical cues involved in the transformation. The rendering can be performed by a MIDI synthesizer and/or by driving an audio processing engine. As an example, we can deduce a desired position in the energy–velocity space from analysis and processing of the movement of a dancer in a multimodal setting (fig. 4), and then use this space position as a control input to the expressive content and the interaction between the dancer and the final music performance [15].

On the other side, recent studies at INPG have showed that dynamic models are suitable for the production of natural motions (fig. 3). By designing his own dynamic model, the user has a high level motion control to modify the quality of such dynamically generated movement.

9.9 Conclusions

In this paper we have proposed a multi-layer architecture which comprises three main levels of abstraction: a semantic description at the top provides the user-level layer and can be adapted to specific user requirements and conceptualization; low-level features at the other end describe parameters strictly related to the rendering model; in between these two extremes, we proposed a physically-inspired description, which is particularly suited to expressive actions and provide a common representation of different expressive semantics and different modalities.

We have proposed direct or indirect reference to human physical behaviour, as a common denominator to multi-modal expressive actions that allows to enhance energetic coherency among phenomena. Furthermore, such a mid-level description is shifted towards the ‘source side’, which makes it suited for multi-modal rendering applications.

Although users are not necessarily familiar with the concepts of inertia, damping, physical interaction etc., all these notions can be intuitively learned through every-day interaction and experience. This amounts to making a shift from existing rendering techniques which are derived from perceptual criteria (at the ‘receiver side’) and are therefore referred to a specific modality/medium (e.g., music).

9.10 References

[1] C.M. Hsieh, A. Luciani, ``Physically-based particle modeling for dance verbs'', *Proc of the Graphicon Conference 2005*, Novosibirsk, Russia, 2005.

[2] N. Castagné, C. Cadoz, ``GENESIS: A Friendly Musician-Oriented Environment for Mass-Interaction Physical Modeling'', *International Computer Music Conference -ICMC 2002* – Goteborg – pp. 330-337, 2002.

[3] B. Repp, ``Patterns of expressive timing in performances of a Beethoven minuet by nineteen famous pianists", *Journal of Acoustical Society of America*, vol. 88, pp. 622-641, 1990.

[4] B. Repp, ``Diversity and commonality in music performance: an analysis of timing microstructure in Schumann's 'Traumerei'", *Journal of Acoustical Society of America*, vol. 92, pp. 2546-2568, 1992.

[5] F. Bonini, A. Rodà, ``Expressive content analysis of musical gesture: an experiment on piano improvisation", *Workshop on Current Research Directions in Computer Music*, Barcelona, 2001.

[6] M. Melucci, N. Orio, N. Gambalunga, ``An Evaluation Study on Music Perception for Content-based Information Retrieval", *Proc. Of International Computer Music Conference*, Berlin, Germany, pp. 162-165, 2000.

[7] E. Cambouropoulos, ``The Local Boundary Detection Model (LBDM) and its Application in the Study of Expressive Timing", *Proceedings of the International Computer Music Conference (ICMC 2001),* 17-22 September, Havana, Cuba, 2001.

[8] L. Mion, ``Application of Bayesian Networks to automatic recognition of expressive content of piano improvisations", in *Proceedings of the SMAC03 Stockholm Music Acoustics Conference*, Stockholm, Sweden, pp. 557-560, 2003.

[9] N. P. Todd, ``Model of expressive timing in tonal music", *Music Perception*, vol. 3, pp. 33-58, 1985.

[10] N. P. Todd, ``The dynamics of dynamics: a model of musical expression", *Journal of the Acoustical Society of America*, 91, pp. 3540-3550.

[11] A. Friberg, L. Frydèn, L. Bodin, J. Sundberg ``Performance Rules for Computer-Controlled Contemporary Keyboard Music", *Computer Music Journal*, 15(2): 49-55, 1991.

[12] D. Chi, M. Costa, L. Zhao, N. Badler, ``The EMOTE Model for Effort and Shape", In *Proceedings of SIGGRAPH00*, pp. 173-182, July 2000.

[13] S. Canazza, G. De Poli, C. Drioli, A. Rodà, A. Vidolin ``Modeling and Control of Expressiveness in Music Performance", *The Proceedings of the IEEE*, vol. 92(4), pp. 286-701, 2004.

[14] R. Bresin, ``Artificial neural networks based models for automatic performance of musical scores", *Journal of New Music Research*, 27(3):239–270, 1998.

[15] A. Camurri, G. De Poli, M. Leman, G. Volpe, ”Communicating Expressiveness and Affect in Multimodal Interactive Systems”, *IEEE Multimedia*, vol. 12, n. 1, pp. 43-53, 2005.

[16] S. Hashimoto, ``KANSEI as the Third Target of Information Processing and Related Topics in Japan", in Camurri A. (ed.): *Proceedings of the International Workshop on KANSEI: The technology of emotion*, AIMI (Italian Computer Music Association) and DIST-University of Genova, 101-104, 1997.

[17] K. Suzuki, S. Hashimoto, ``Robotic interface for embodied interaction via dance and musical performance", In G. Johannsen (Guest Editor), *The Proceedings of the IEEE*, Special Issue on Engineering and Music, 92, pp. 656–671, 2004.

[18] R. Bresin, A. Friberg, ``Emotional coloring of computer controlled music performance", *Computer Music Journal*, vol. 24, no. 4, pp. 44–62, 2000.

[19] L. Mion, G. D'Incà, ``An investigation over violin and flute expressive performances in the affective and sensorial domains", *Sound and Music Computing Conference (SMC 05)*, Salerno, Italy, 2005 *(submitted)*.

[20] M. Imberty, *Les ecritures du temps*, Dunod, Paris, 1981.

[21] R. Laban, F.C. Lawrence, *Effort: Economy in Body Movement*, Plays, Inc., Boston, 1974.

[22] D. Cirotteau, G. De Poli, L. Mion, A. Vidolin, and P. Zanon, "Recognition of musical gestures in known pieces and in improvisations", In A. Camurri, G. Volpe (eds.) *Gesture Based Communication in Human-Computer Interaction*, Berlin: Springer Verlag, pp. 497-508, 2004.

[23] W. W. Gaver, ``What in the world do we hear? An ecological approach to auditory event perception", *Ecological Psychology*, 5(1):1 29, 1993.

[24] F. Avanzini, M. Rath, D. Rocchesso, and L. Ottaviani, ``Low-level sound models: resonators, interactions, surface textures", In D. Rocchesso and F. Fontana, editors, *The Sounding Object*, pages 137-172. Mondo Estremo, Firenze, 2003.

[25] L. Ottaviani, D. Rocchesso, F. Fontana, F. Avanzini, ``Size, shape, and material properties of sound models", In D. Rocchesso and F. Fontana, editors, *The Sounding Object*, pages 95-110. Mondo Estremo, Firenze, 2003.

[26] F. Avanzini, D. Rocchesso, S. Serafin, ``Friction sounds for sensory substitution", *Proc. Int. Conf. Auditory Display (ICAD04)*, Sydney, July 2004.

[27] Canazza S., De Poli G., Di Sanzo G., Vidolin A. ``A model to add expressiveness to automatic musical performance", In *Proc. of International Computer Music Conference,* Ann Arbour, pp. 163-169, 1998.

[28] Clynes, M. ``Sentography: dynamic forms of communication of emotion and qualities", *Computers in Biology & Medicine*, Vol, 3: 119-130, 1973.

[29] Sundberg J, Friberg A. ``Stopping locomotion and stopping a piece of music: Comparing locomotion and music performance", *Proceedings of the Nordic Acoustic Meeting Helsinki 1996*, 351-358, 1996.

[30] A. Luciani, "Dynamics as a common criterion to enhance the sense of Presence in Virtual environments". *Proceedings of "Presence Conference 2004"*. Oct. 2004. Valencia. Spain.

[31] A. Luciani, J.L. Florens, N. Castagné. "From Action to Sound: a Challenging Perspective for Haptics", *Proceedings of WHC Conference 2005*.

[32] C. Cadoz , A. Luciani, J.L. Florens: "CORDISANIMA: a Modeling and Simulation System for Sound and Image Synthesis-The General Formalism", *Computer Music Journal*, Vol. 17-1, MIT Press, 1993.

[33] A. Luciani, "Mémoires vives". Artwork. Creation mondiale. *Rencontres Internationales Informatique et Création Artistique*. Grenoble 2000.

[34] N. Castagné, C. Cadoz : "A Goals-Based Review of Physical Modelling" -*Proc. of the International Computer Music Conference ICMC05* -Barcelona, Spain, 2005.

[35] C. Cadoz, "The Physical Model as Metaphor for Musical Creation. pico..TERA, a Piece Entirely Generated by a Physical Model", *Proc. of the International Computer Music Conference ICMC02,*
Sweden, 2002.

[36] P. Juslin and J. SLoboda (eds.), *Music and emotion: Theory and research*, Oxford Univ. Press, 2001

10 Gestures analysis in musical performance (SPCL, UPS, HFRL)

M. Wanderley (SPCL), B. Bardy(UPS/UM1), T. Stoffregen (HFRL)
May 2005

SPCL (M. Wanderley), UPS/UM1 (B. Bardy), HFRL (T. Stoffregen) contribute to RD3.3 with a joint project on music-gesture interaction. The project is detailed below. The principal applicant is Marcelo Wanderley (SPCL). Bardy and Stoffregen are co-applicants. The project was conceptually thought during a 3-day visit of Bardy and Stoffregen at SPCL in May 2005.

10.1 Study 1. Perception of musical performances from optical kinematics

Humans (and other animals) can readily perceive a wide variety of complex physical dynamics from optical displays that preserve only the kinematic properties of events. Examples include gender, psychological states, such as mood or the intention to deceive (Runeson & Frykholm, 1983), and properties of the animal-environment system, such as support surface deformability (Stoffregen & Flynn, 1994) and affordances (Stoffregen et al., 1999). In this study, we seek to understand the extent to which optical kinematics preserve information about complex aspects of musical performance. Point light displays have been used by Davidson (1993) to verify that musical expressive intention is conveyed through movements. In this work we are interested in the perception of instrumental performance from the point of view of a musician's gestures, both effective (those performed to generate sounds, such as blowing, bowing, plucking a string, etc.) and accompanist (e.g. torso movements in pianists). We will study what kinds of information are preserved in kinematic displays, such as information about performer expertise, gender, musical style, and even the recognition of a specific performer. Funding permitting, functional data analysis techniques will be used to perform statistical tests on the timeseries data obtained (see McAdams et al., 2004).

10.2 Design

Generation of kinematic displays for expert and novice performers, same instrument (clarinet).
Three experts (optimally five), three novices (idem), five reps of each performer (but with different excerpts for each rep). Each excerpt to be about 20 s in duration. So: each performer does 5 reps, with different excerpts on each rep, and all performers do the same 5 excerpts. Do both sitting and standing (5 reps each, the same 5 excerpts for sitting and standing). Yield: 5 sitting plus 5 standing for each of 6-10 performers = 60-100 stimuli.
Tapes to be viewed by both expert (minimum 8) and novice (minimum 8) musicians. Allow S to make judgment before the end of each 20 s excerpt, use “discontinuation time” as a dependent variable.

- Exp 1. Viewers asked to state whether the performer is expert or novice
- Exp 2. Viewers asked to give some rating of degree of expertise (e.g., 1-10 scale).
- Exp 3. Viewers (different subjects, probably) asked to categorize the tapes by performer, that is “select the 10 tapes that were performed by person A, the 10 by person B, etc.” (here, S is told that there are 10 performers), or “group these by performer” (here, the perceived number of performers is a dependent variable).
- Exp 4. Delete markers from the instruments (and from the arms?) and again ask about expertise
- Exp 5. Give the viewer 3-d control over the image, and use their actual manipulation as a dependent variable (task is to judge expertise).
- Exp 6. Viewers have 3-d control, look at the same tapes as in Exp. 5, but make a different judgment (e.g., performer gender); do the viewer’s 3-d control actions differ for different judgment tasks?

One approach is to make a single set of stimuli, then ask people to make several different types of judgments about these stimuli, in separate experiments.

10.3 Study 2 Postural dynamics in music performance

To maintain an upright stance, the body's center of mass must be kept above the feet. Stance however is rarely maintained for its own sake; that is, humans are almost always engaged simultaneously in supra-postural (e.g., playing an instrument) as well as postural activity (standing upright). In one point of view, postural states behave like attractors in the postural space, and changes between states behave like self-organized, non-equilibrium phase transitions between attractors (see Bardy et al., 2002, for a review). In this study, we will investigate how human postural dynamics (stable postural states) are constrained by musical performance. Values and stability of postural patterns (as well as their changes) during musical performance will be analyzed. Using musical examples, we will seek to validate previous results by Bardy and colleagues on nonmusical activities. The data captured to make tapes for the first project will be used again in this study (standing and seating, 5 experts, 5 novices, 5 repetitions each). No new data should be necessary.

11 Conclusions

Annie Luciani

Gestures studies addresse a wide diversity of the approaches:
• Real time instrumental simulation,
• Development of new gestural devices,
• Motions capture and Motion analysis

In the following, we summarize the data used and produced by each type of process developped by the partners and that have to be analysed to go toward a definition of gesture formats.

INPG haptic devices, simulation and gestural data

- Geometrical Dimensionality: Scalar (0D), 1Dx(y, z), 2Dxy (yz, zx), 3Dxy
- Number of degrees of freedom: 1 to 64
- Type of data: Positions and Forces (no angle, no torque)
- Frequency rate: 300Hz – 4 KHz
- Simulation number coding: Float 64 (Float double)
- Simulation Outputs coding: Float 32, Float 64 (Float double)
- Input number coding for gesture file: all
- Gesture file coding: Float 32, Float 64 (Float double), Integer 32 with scaling factor (double 64 bits), Integer 64
- Device inputs/outputs: AD-DA

UNEXE tactile device

- Dimensionality of each channel: Scalar
- Number of channels: 24 or 25 per finger
- Spatial resolution on the finger: 2 mm
- Working bandwidth: 20 – 500 Hz
- Maximum output amplitude: 100 _m
- Spatial resolution of the virtual workspace: 1 mm
- Input data: Data frames with 32 bits per channel, i.e., 800 bits per frame for 25 channels
- Input data rate: Frame rate around 50 Hz, i.e., 40 kbits s^{-1} for 25 channels
- Input connectivity: USB
- Output data: NOT APPLICABLE

PERCRO Hand Glove

- Number of tracks: 12 sensors (2 each finger and 2 for the wrist)
- N° Degrees of freedom: 11
- Resolution: 0.5°
- Type of data: angle
- Range of measure: (0-180) degrees
- Resolution: 0.1 degrees (depending on electronics)
- Accuracy: 2 degrees
- Output span: 0-5V

The Glove outputs are electrical signals (one for each sensor). These signals are proportional to the angular position of the fingers. A specific software transforms these electrical signals in angular coordinates.

PERCRO Haptic Desktop

- Number of degrees of freedom: 2
- Type of data: Positions and Forces
- Spatial encoding: 1024cpr optical encoders with a 4X decoding
- Spatial sensitivity: 30um at workspace centre (pixel size is about 300um).
- Range of data: 3 N of continuous forces and 5 N of peak forces.

- Workspace: 430 x 320 mm
- Reflected inertia: 0.07 Kg;
- Typical Stiffness: 4 N/mm;
- Position resolution: 0.01 mm.

DIST Motion analysis platform

Layer 1: Inputs:
- Audio: Microphones, Asio devices, MIDI inputs
- Video: DV, Cameras
- Other devices: serial inputs, data glove, Customized inputs

Layer 2 Outputs from Layer 1
- Center of Gravity: 3D point
- Silhouette segmentation data: 2D and 3D tracking: 2D and 3D points
- silhouette contour: set of 2D (3D) points (shape)
- Skeleton extraction: Hierarchical set of 2D (3D) points (shape)
- Quantity of Motion: set of scalars (each set corresponds to fraction of the body)
- Contraction Index: scalar

Trajectory analysis
- 2D (3D) velocities
- 2D (3D) accelarations

Layer 3 Outputs from Layer 2:
- Motion segmentation: pause detection (starting point – end point)
- Fluency and impulsiveness: scalar (% of accelerations and deceelerations, Directness index)
- Gesture representation: symbolic label (parameters)
- Posture recognition: Hu moments (matrix)

Layer 4: high level content

In progress
- Motion quality inference as pushing, gliding etc.
- Etc.

As shown in the state of the art in motion capture, several format exist. However they are dedicated to specific applications (3D solid hierarchical body representations, ACSII coding), that are not sufficiently versatile to cover all the data produced ans used here.

Among the existing format that could be a basis for a future gesturla format:

• The C3F format seems general, including scalars information as well as geometricla ones. It is of an heavy structure. Aditionnal information is required.

• The INPG format could be adequate for low leval information, including all the data descibed here, excepted some of those of layers 3 and 4 of DIST motion analysis.

Future work of RD3.3 will go deeper in each of them, in cooperation with RD1.4 and with other Enactive requirements(hardware input/output standardization).