

MIRROR

IST-2000-28159

Mirror Neurons based Object Recognition

Deliverable Item 3.2 and 3.3 Biological data acquisition set-up and Data Collection and Analysis Software

Delivery Date: October 31, 2002

Classification: Public

Responsible Person: Prof. Giulio Sandini – DIST University of Genova

Partners Contributed: Matteo Brunettini (DIST); José Santos-Victor, Alexandre Bernardino and Manuel Lopes (IST)

Short Description: This deliverable describes the biological data acquisition setup and the software developed for data acquisition. The setup allows for the synchronous acquisition of stereo images, hand joint angles, hand position and finger pressure. This information will be used to develop methodologies conducting to the recognition of different gestures and in particular different types of grasping.



**Project funded by the European Community
under the “Information Society
Technologies” Programme (1998-2002)**

Content list

1. The Acquisition Software	2
1.1. Program Installation	2
1.2. The Main window	3
1.3. The Calibration Dialog.....	5
1.4. The Options Setup	6
1.5. The Data Windows	7
2. An example of program usage	8
2.1. Description	8
2.2. Setting up the program.....	8
2.3. Connection	8
2.4. Instruments Calibration	9
2.5. Data Recording	9
2.6. Data Analysis	9
3. The Output Data.....	11
3.1. Images.....	11
3.2. The Text file.....	11
3.3. Additional data.....	12
4. The processing.....	13
Appendix: An example of Matlab data importing function.....	14

1. The Acquisition Software

This software acquires visuomotor grasping data from an experimental set-up, in particular sequences of images and kinesthetic information during grasping actions.

It is written in C++ and compiled with *Microsoft Visual C++*, on *Windows 2000* platform.

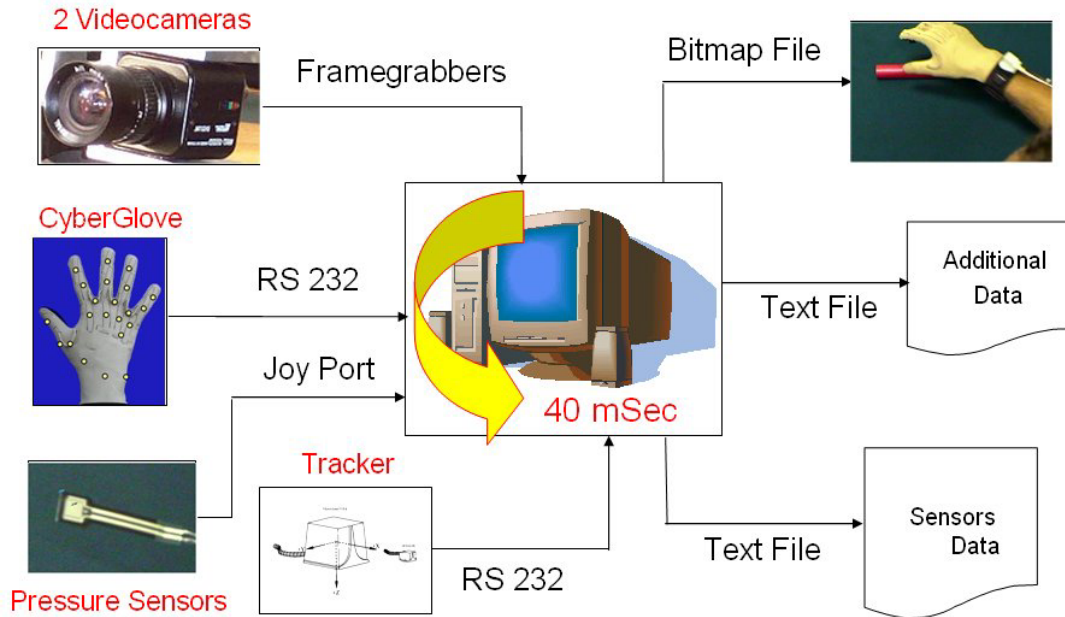


Figure 0 - The experimental setup and the acquisition software

Briefly, the setup is composed by:

- Two *Watec WAT202D* digital cameras with PAL standard (768x576 pixels, 25 Hz of frame rate, color) acquired by two *Picolo Industrial* frame grabbers
- A *CyberGlove* dataglove produced by *Immersion*, which consists of a glove mounting 22 sensors reading the hand joints angle.
- A *Flock of Birds* tracker (manufactured by *Ascension*) determines the position of a sensor in space.
- Two pressure sensors, to read the pressure applied by the thumb and the index onto object during grasping.

The program, called *GraspCapture.exe* consist of a main dialog window, that allows to connect to the peripherals and to record the data, and several utility windows used to control the grasping data collected in real time and to customize the program options.

1.1. Program Installation

To run the program it is required to install all the drivers need by the peripherals, such as:

- The **Picolo Industrial drivers** for the framegrabbers
- The **Immersion CyberGlove drivers and software** for the dataglove
- To setup from the Windows control panel, a **gaming peripheral** called *Pressure Sensor*, simply a joystick with two axes and no buttons.

Some dlls are also required to run the program:

- **Bird.dll**, necessary to the Flock of Bird tracker
- **A dll package**, the *Intel Image Processing Library (IPL)*, a component of the *Intel Performance Library Suite*.

These libraries can either be included in the system path or stored in the same directory as the executable file.

before connecting to the peripherals it is also required, to run the dataglove communication program, called *DeviceManager.exe*. This program is a part of the CyberGlove software suite.

When the program is executed for the first time, it creates a files called *GrabCapture.ini* that contains the program options and a directory called *swap* to store the temporary images swap file, both of them in the executable path. The swap file is created during the data record and it is erased at the end of the execution.

The program also creates a directory for each recording data set in the data path (that can be specified in the options window, see below) to store the output data.

1.2. The Main window

As shown in *figure 2*, the main window can be divided in the different parts: the lower area, containing the main controls, the central area, containing the images produced by the two video cameras and the upper area, containing the utility controls.

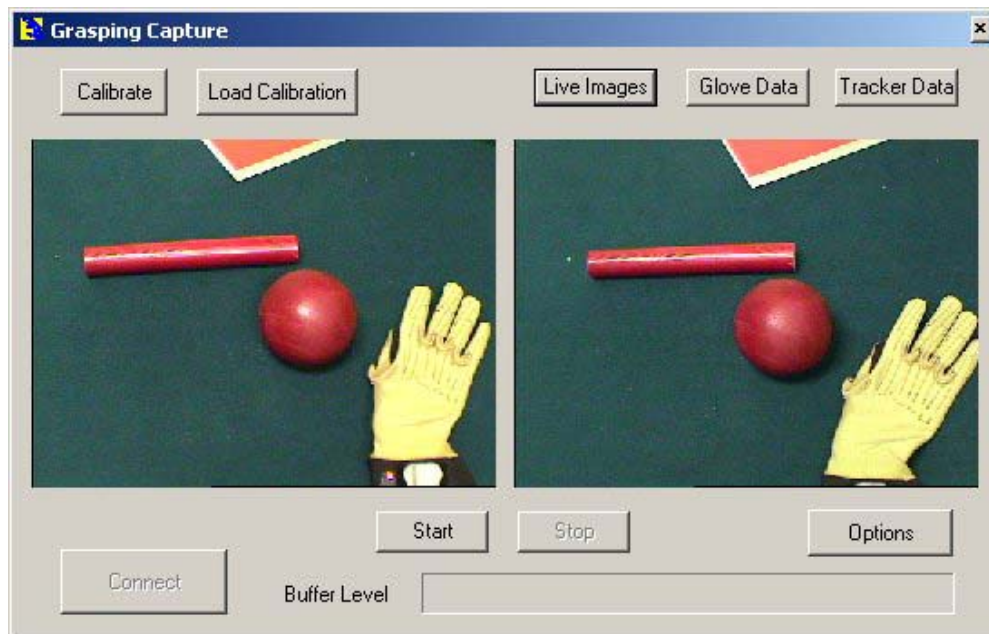


Figure 2 - The main window

The main controls are:

- The **Connect** button – starts the communications with all the peripherals included in the setup, performs initialization and prepares them for the acquisition.
- The **Start** and **Stop** buttons – control the main acquisition process.
- The **Options** button – opens the options dialog, where it is possible to modify the program options.

- A **progress bar** – this control shows the buffer level during recording; if it reaches the full level, it means that there is no more space in memory for new data, so these data will be lost and the entire recording set is compromised, a error message will be shown at the end of the recording.

The utility controls can be divided in two categories: on the left there are the calibration controls and on the right we have the data showing controls.

The calibration controls are:

- The **Calibrate** button – it starts the peripheral calibration, it is necessary to adapt the dataglove to the subject hand morphology; a new calibration is required for each different subject that wears the dataglove. Calibration data can be saved on disk and retrieved later.
- The **Load Calibration** button – by clicking this button is possible to recover a calibration file, and set the glove up with previously saved data.

The data showing controls allow checking the data captured in real time, so it is possible to see the data that the peripheral are acquiring live. During the recording process, those controls are not updated to speed the acquisition up. They are:

- The **Live Images** button – allows to show the images acquired by the two cameras in the central part of the main window.
- The **Glove Data** and **Tracker Data** buttons – open/close the two windows that show data collected by the dataglove, the pressure sensor and the tracker. To close these windows click the buttons again.

When the program starts, only the **Connect** and **Options** buttons are active. Once the connection has been established the other controls become active and the **Connect** button becomes inactive.

The connection takes a few seconds to complete, so when the **Connect** button is clicked the program freezes, as soon as this procedure is completed correctly, a message appears on the screen otherwise, if the connection cannot be established, an error message will warn the user about the problem.

Once the recording is stopped, the program starts to synchronize the data and write them to disk. When it is done, a message windows appears to inform you about where those data have been saved and ask if you want to see the debug info file. This file (*figure 3*) contains useful information about the recording process (described below) and it is customizable according two different levels of accuracy.

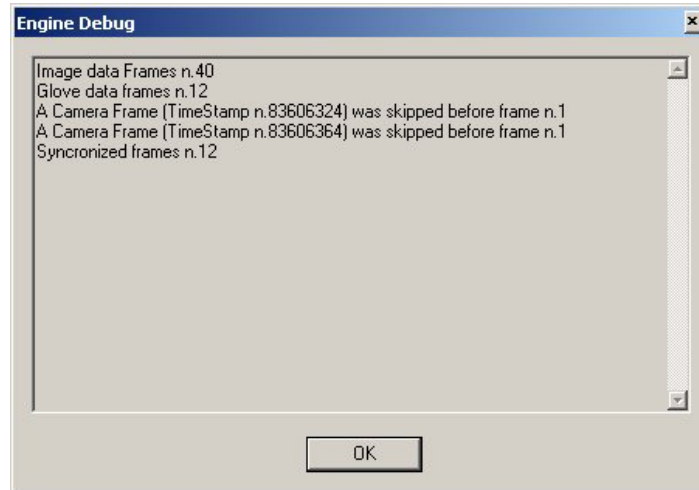


Figure 3 - The engine debug file preview

1.3. The Calibration Dialog

This window (*figure 4*) shows how to execute the calibration operations, required by the dataglove and the pressure sensors. The calibration itself, consists of several steps, which have to be executed in a particular sequence, according to the instruction show in the window.



Figure 4 – The calibration window

The main area of the dialog window contains the description of the instructions, while in the lower part there are the controls that allows, once all the instructions present in the current calibration step are satisfied, to go on with the next step or to quit the whole calibration. At the end of the calibration procedure, it is possible to save the calibration data on file.

1.4. The Options Setup

As shown in *figure 5*, this window contains all the controls useful to change the program parameters.

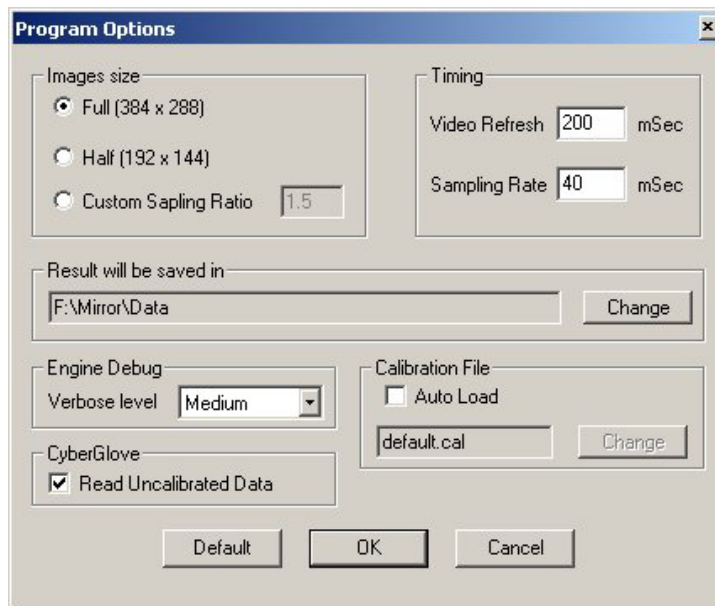


Figure 5 – The program options

These parameters are:

- The **image size** can be set to full (384 x 288 pixel) resolution, half resolution (192 x 144) or a custom subsampling rate defined. For instance, if a value of 1.5 is used for the subsampling rate, it means that the X and Y dimensions of the output image will be the original size divided by that number ($384/1.5 \times 288/1.5$ so it gives a size of 256 x 192 pixels).
- The **program timings**, the video refresh of the live video camera images and the sampling rate of the data. Please note that the maximum value for the sampling rate is 25 Hz corresponding to a timing of 40 mili-Seconds.
- **Where the data will be saved**; the specified path will contain a separate directory for each recording set.
- The **accuracy of information reported by the debug file**.
- Whether to read from peripherals **calibrated or uncalibrated data** and then write them to the output file.
- Whether to **automatically load a calibration file** and which file has to be loaded.

By pressing **OK**, the options are saved in the *GrabCapture.ini* file to be recovered at the next program starting. The **Default** button recovers the default values for all program parameters.

Once the program is connected with the peripherals, only a few options are active, so to change the unavailable parameters you must restart the program.

1.5. The Data Windows

There are two data analysis windows: one for the tracker data (*figure 6*) and one for the dataglove and pressure sensors data (*figure 7*).

Both of them show real time data measured by the peripherals, grouped by their typology.

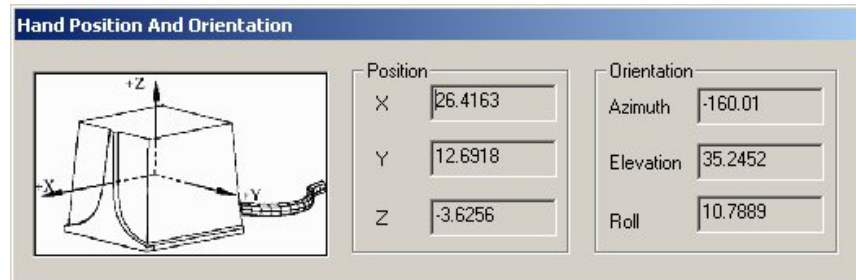


Figure 6 - The tracker data window

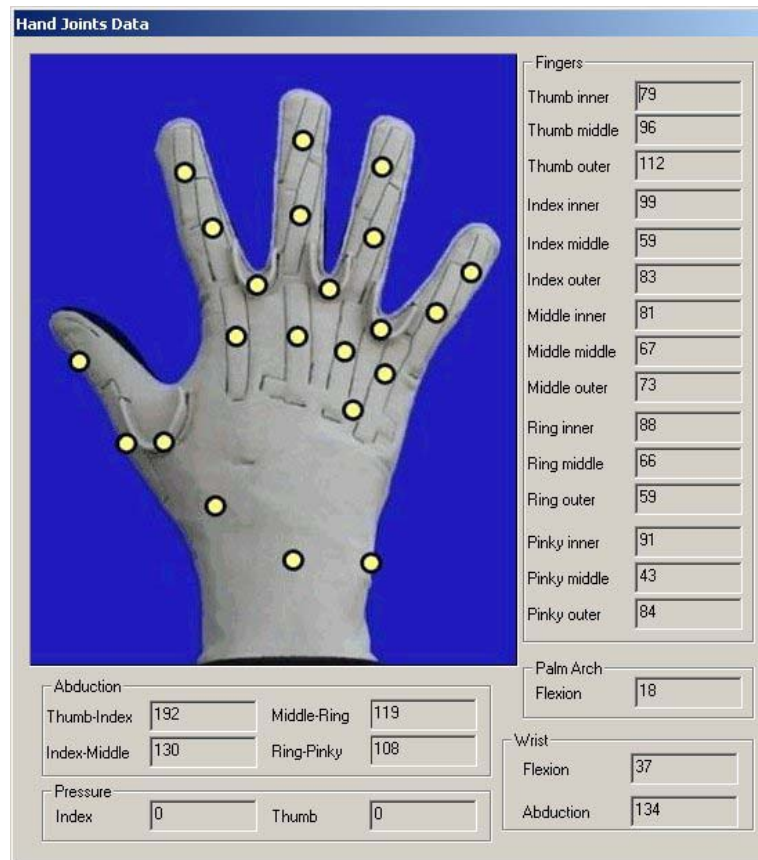


Figure 7 – The glove data window

2. An example of program usage

The purpose of this section is to explain the program usage by a practical example, starting from program installation and setup to the analysis of output data.

2.1. Description

As an example, we want to record the data of a spherical power grasp experiment. First of all, we need to install the program and set it up. Then we can calibrate the CyberGlove, and finally we can start the recording. Once the data has been recorded, it's possible to use the Matlab tool to analyze them.

2.2. Setting up the program

After the program has been started, press the **Options** button to open the option dialog window, then do the following:

- Choose the full image size (384 x 288 pixels) for the output images.
- Leave the default timing of 40 mSeconds for the sampling rate and of 200 mSec for the video refresh (whereas is critical to use 40 mSec of sampling rate to get good grasping data, you can change the video refresh as you prefer).
- Select the output data path, by pressing the **Change** button near the current path, and select the desired directory.
- Select the **medium** engine debug accuracy.
- Leave the **Read uncalibrated data** check box unselected, in order to write calibrated data (using the calibration information calculated with the calibration procedure described in *section 2.4*) on the output file.

Then press **OK** to confirm the options, these options will be stored in the **GrabCapture.ini** file and will be automatically loaded at the next program run.

2.3. Connection

Before starting the program, it is necessary to connect to the peripherals, starting the communication with them. Before doing that, switch them on in the following order:

- Turn the cameras on.
- Set the Flock of Birds tracker switch to the position **Fly** and check that the led is turned on.
- Execute the CyberGlove communication program called *Device Manager*.
- Then turn the CyberGlove main switch on. The communication program wakes the dataglove up and then it becomes ready to collect data.

After that, you can press the **Connect** button so the connection procedure can start. You must see the tracker led blinking 2 times and then the light should stay on.

When all operations have completed successfully, the program advice you that it is ready to record data and the **Start** button becomes active.

If everything has gone right you should see, by clicking the **Live Images** button now active, the two camera live images as well as the tracker and glove data, by clicking the other two similar buttons, **Glove Data** and **Tracker Data**.

Please note that, after the connection has been established most of the controls in the options windows, except those concerning the calibration, become inactive. This occurs because once the connection with the peripherals is established, it is not possible to modify the options.

2.4. Instruments Calibration

In order to calibrate the peripherals, press the **Calibrate** button at the left top part of the main window, which starts the calibration sequence. It is divided in three steps and to accomplish the calibration follow the instructions shown in the dialog. At the end of each step, press the **Continue** button. When you press that button keep your hand still in the required position until the program changes the instructions on the screen, so that the calibration data can be recorded correctly. To quit the calibration procedure, simply press the **Cancel** button at any time.

The three steps are:

- 1) **Dataglove sensors offset** - push your hand against a flat surface with all fingers parallel.
- 2) **Dataglove sensors gain** - use the calibration object, putting your hand in the mould, trying to fit it as much as possible.
- 3) **Pressure sensors** – keep your index and thumb lifted up without applying any pressure with them

When the calibration is done, you can save the calibration data and recover them later, by pressing the **Load Calibration** button.

2.5. Data Recording

Before starting to record data, check the correct frame size in the two camera images. When you are ready to start, press the **Start** button to begin the data acquisition. Please note that during the recording, live images, tracker and glove data are not updated on the screen, to reduce the computation overhead that could otherwise lead to loss of frames.

During recording, a progress bar shows the buffer filling status: in normal condition the buffer will not be filled up, indicating that no frames will be lost. However, if the indicator reaches the full level, the recording is compromised and an error message will be shown when you stop the recording.

When you are done, press the **Stop** button to stop the recording. Data synchronization starts and the output is written on disk. It may take a while and, when it is completed, you can see the engine debug file, saved in the same directory of the output data. The most important information reported by this file is about data synchronization: if some frames are lost, it will be reported here. If the program loses some frames at the beginning of the data stream, this is not a problem because the time required to start images acquisition is slightly different from that of the glove data. But once both procedures have been started, the data stream must be continuous and, at every duty cycle an image-glove data couple must be written on disk. If the debug file reports the loss of frames in the middle of the data streams, it means that the data of some duty cycle are lost and all the data of that recording set are practically unusable.

2.6. Data Analysis

A *Matlab* application (*figure 8*) can be used in order to analyze data acquired and recorded by the program as described in the last chapters.

Using this tool is possible to open the text file containing the data from tracker, dataglove and pressure sensors, and read all the recorded values; after that, it could help to do some data processing in order to compare kinetic data the information extracted from image processing.

In order to use it, you have to execute Matlab (from the version 6.0 and later), and move to the directory in which the application is stored, and run it. Select the directory that contains the images sequence and the *glovedata.csv* file. All the values written in the .csv file are stored in the memory, in form of a matrix, and you can perform data processing (it can plot, for example, the trajectory of the hand in the xy plane). If you choose a visual data analysis, it's possible to segment the hand on the images and so extract useful information, such as the motion of the center of mass, the speed of the hand or other geometrical information about the hand.

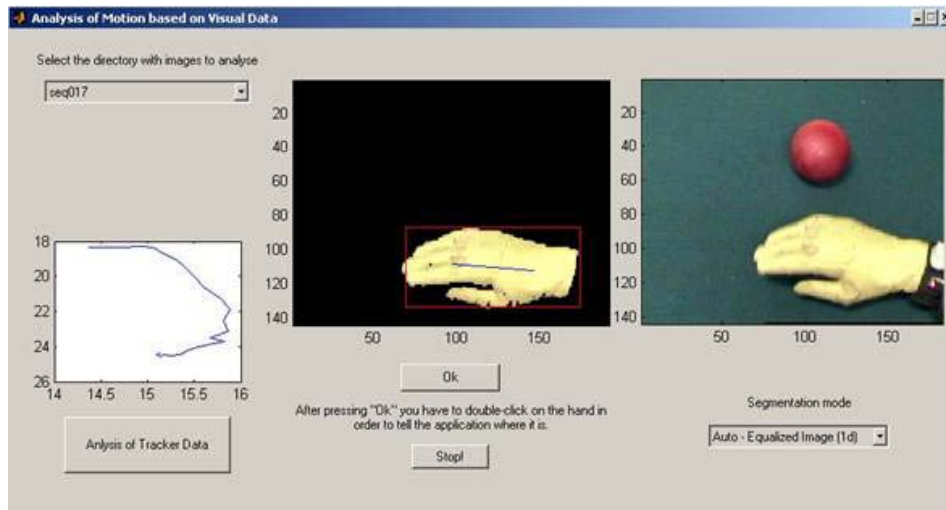


Figure 8 – The Matlab tool

3. The Output Data

The program produces two different kind of output: text files and images.

Images are common Windows Bitmap files up to 384 x 288 pixels of size and 24 bits per pixel of color depth, coded in RGB format.

There are two different text files: a file containing additional data, such as the number of frames recorded or if some frames are missing in the data stream, and the main data file, containing the recordings from every peripheral.

Each recording set is stored in a different directory named *seqNNN* where NNN is a progressive 3 digit number given at every different recording sequence. The path of all these directories can be customized by the options window. Please note that the amount of space required for the output data is large: at the standard frame rate of 25MHz (corresponding to 50 images per second) and with a resolution of 283 x 288 pixels, it needs more than 16250 Kbytes per second of disk spaces, this amount of data is double considering the swap file, created during recording time, but erased at the end of synchronization process.

3.1. Images

Each frame produces two images (*figure 9*) taken from the left and the right camera, saved as common windows bitmap files, named *LNNNN.bmp* and *RNNNN.bmp* where NNNN is the progressive number of the frame, and the first letter is "R" for right camera images and "L" for the left ones.



Figure 9 – An image couple

3.2. The Text file

This file contains all the numeric data produced by the tracker, the data glove and the pressure sensors. It is saved in the *Comma Separated Values* (.csv) windows format, a normal text file with commas used as separators for the different values in each row.

The first row contains the description of each row value.

There are 31 columns in the file: the first one is the frame number, the remaining columns correspond to the 30 different data produced by the peripherals, 22 from the data glove, 6 from the tracker and 2 from the pressure sensors.

The data glove measures the hand joint angles in radians and the values stored are:

- 3 angles for each finger (inner, middle and outer joint)
- 4 angles for finger abductions (thumb-index, index-middle, middle-ring and ring-pinky)
- 2 angles for wrist flexion and abduction
- 1 angle for the palm arch

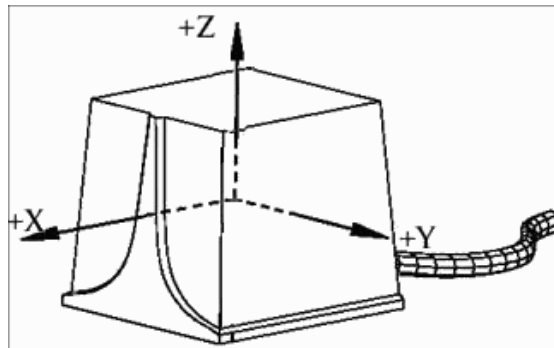


Figure 10 – The tracker reference frame

The tracker records the position of the hand, referred to a frame placed in a point corresponding to its magnetic field transmitter as shown in figure. It produces 6 values, 3 for the hand position in space (**X**, **Y** and **Z** Cartesian coordinates) and 3 for its orientation in space (**Azimuth**, **Elevation** and **Roll** using the Euler angles).

The pressure sensor produces a value in range 0 to 1 proportional to the pressure applied by the finger onto object. We have 2 pressure sensors: one for the index and one for the thumb.

The first column of the text file contains the frame number, which also identifies the related image file. Data recorded in a row that has the same number of an image couples are recorded in the same temporal interval of 40 mSeconds (using the standard sampling time).

3.3. Additional data

Additional data are useful to know if the recording processes and the synchronization phase have completed successfully and are stored in a text file, called *EngineDebug.txt*.

It can be customized with two levels of accuracy, medium and verbose, or excluded at all and it contains:

- The number of frames recorded by the cameras
- The number of frames recorded by the data glove, tracker and pressure sensor
- The number of synchronized frames
- The frames that have been skipped in the synchronization phase, because there isn't the corresponding frame in the image-numeric data couple.
- The total recording time (only in verbose version)
- The time stamps of every frame (only in verbose version)

4. The processing

At the current state of development, only very preliminary processing tests have been tried out, since most of the effort was directed to the design and implementation of the synchronized data acquisition setup.

Amongst other possibilities, in the future we will consider data processing tasks divided into Visual Perception and Visuo-Proprioception Association. Visual Perception has the purpose of hand localization and gesture recognition. Deliverable 2.3 already presents some work addressing these problems (stereopsis and color analysis for object segmentation, software for motion analysis, and statistical learning methods for hand posture classification). Although preliminary, the work on hand posture classification has achieved high classification rates (about 94%) on complex and varied data sets (see Fig. 11).

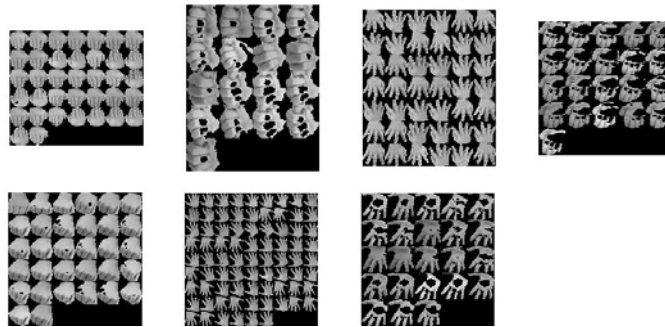


Figure 11 – Hand posture classification

Visuo-Proprioception Association has the purpose to create a map between visual data and data-glove information. This map is an important step towards the creation of Visuo-Motor maps since motor and proprioception information are tightly linked. The existence of a Visuo-Motor map together with a view-point transformation (Deliverable 2.3), will be of major importance for imitation tasks, allowing the projection of the visual information directly into the motor language.

The processing of visual, kinematic, position and contact/pressure information provided by the setup herein described will thus be the at the core of the project activities during the second year.

