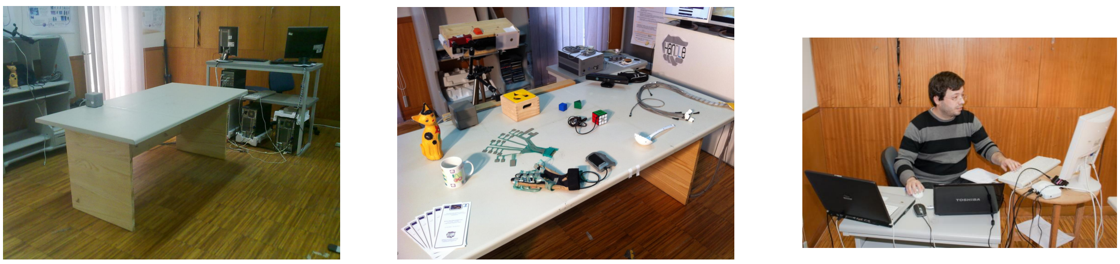
Chapter 4 - Recording human manipulation and exploration movements

Chapter 4

Recording Human manipulation and exploration movements

One of the most popular approaches for endowing robotic systems with human-like capabilities (e.g. dexterous manipulation, locomotion, and social interaction) is to learn those strategies from human demonstrations. Extensive overviews are presented in [Argall et al., 2009] and [Billard et al., 2008]. The research works provided in this PhD thesis replicate human-like capabilities for dexterous in-hand manipulation and haptic exploration of objects/surfaces.

Due to the mechanical and functional complexity of the human hand, the recording of human dexterous manipulation and exploration movements is a challenging task. Typically, a multi-modal approach (e.g. external RGB cameras, depth sensors, data gloves, motion trackers, and tactile arrays) is followed to capture all the diverse movements and contact interactions between the human hand and the objects [Faria et al., 2012]. However, care should be taken to integrate such a high number of devices simultaneously in order not to constrain the natural movements and sensing capabilities (e.g. touch feedback) of the human hand. A complementary approach consists of using instrumented objects instead of over-instrumenting the human hand (e.g. [Matsuo et al., 2009], [Lobo



(a) (b) (c)

Figure 4.1: Overview of the experimental area at the *Artificial Perception for Intelligent Systems and Robots* AP4ISR laboratory: a) table. b) data acquisition devices and objects. c) data acquisition operator controlling the central computer.

et al., 2011]).

This chapter presents the experimental area of *Artificial Perception for Intelligent Systems and Robots* AP4ISR laboratory, describing the data acquisition devices (section 4.2) used to record manipulation tasks. The data acquisition software tools and data acquisition architecture developed for this purpose, during the PhD studies reported by this thesis, are also described (see section 4.1).

Several datasets of manipulation tasks recorded during the PhD studies are presented in section 4.3. The contribution of these datasets to robotics research is also reported. Multiple software tools were developed to promote and ease the integration and annotation of datasets (section 4.4).

4.1 Experimental area and data acquisition architecture

Figure 4.1 shows an overview of the experimental area of of *Artificial Perception for Intelligent Systems and Robots* AP4ISR laboratory and the data acquisition devices available to record data regarding human demonstrations of manipulation tasks.

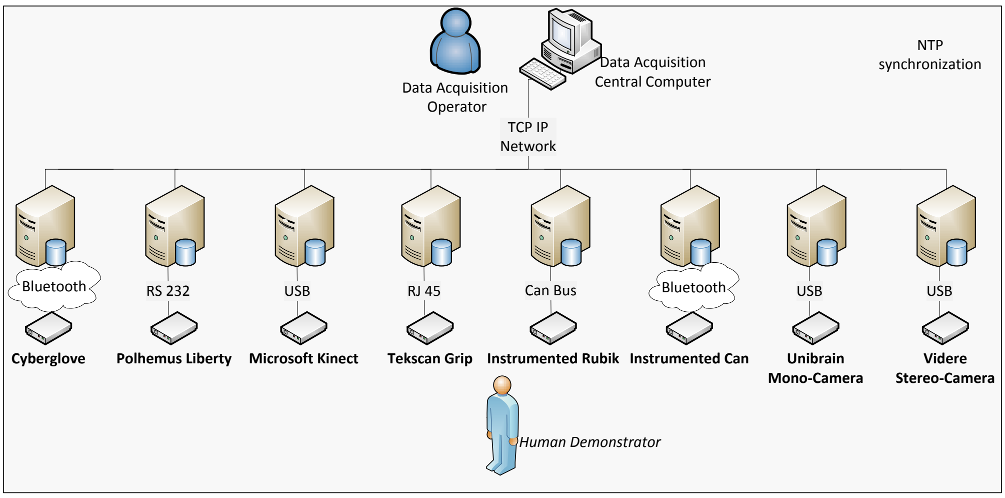


Figure 4.2: Representation of the data acquisition architecture implemented in the experimental area of *Artificial Perception for Intelligent Systems and Robots* AP4ISR laboratory.

A distributed data acquisition architecture was implemented on the experimental area. This approach was followed due to the high number of data acquisition devices that can be used simultaneously in a single session. The data acquisition architecture is described in Figure 4.2.

In this data acquisition architecture, each computer can be connected to one or more data acquisition devices. For simplicity in Figure 4.2, only one data acquisition device is connected to each computer. All the computers involved in a data acquisition session are connected by a TCP/IP computer network. One of the computers of this architecture works as server and the remaining ones (computers having data acquisition devices connected to them) work as clients. Software clients for each of the data acquisition devices (RGB cameras, motion trackers, data gloves, tactile sensors, instrumented objects) were developed, as well as software for the server computer.

The main concept behind this approach is to have the server software coordinating all the data acquisition sessions. At the beginning of the data acquisition session, the server software is responsible for acknowledging the connection requests made by the clients software. The server builds a list of all the clients connected to it. Each client software manages both the data acquisition from the respective device, as well as the storage of the data.

The structure of the dataset is based on XML files. Each data acquisition device stores the data using a specific and predetermined structure of XML files. This allows easy integration of the datasets by other software applications and the transfer of data between partners. The structure of the XML files for each data acquisition device is detailed in the technical report *Protocol for the corpus of sensed grasp and handling data: storage of multi-modal datasets* [HANDLE-UC, 2009] expanded upon during the PhD studies, under the scope of the *HANDLE* project. The document is available online (url: `http://www.rmartins.net/phd-docs/tr02/`).

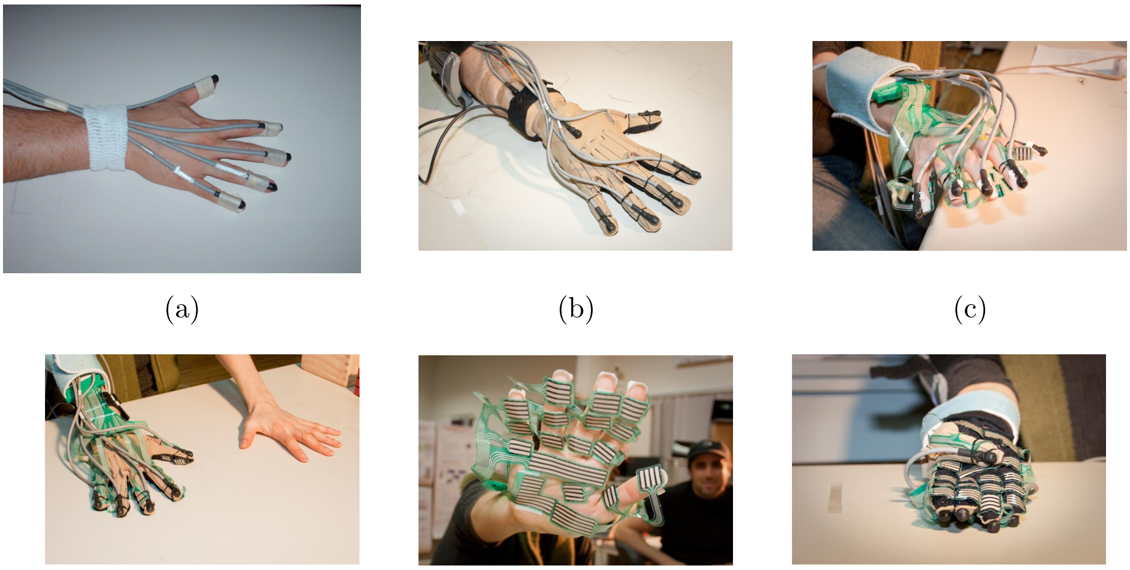
All the data acquired from the different devices is time-stamped. To enable a common temporal reference between the different computers involved in the data acquisition session, the Network Time Protocol (NTP) is used for clock synchronization of the different computers. NTP is a free, widely available protocol designed to synchronize the clocks of computers over a network. The steps involved on setting-up the NTP synchronization are detailed in the technical report *Distributed synchronization of multi-modal data acquisition devices using NTP (network time protocol)* [Martins, 2010] examined during the PhD studies under the scope of the *HANDLE* project. This tutorial is available online (url: `http://www.rmartins.net/phd-docs/tr03/`).

4.2 Data acquisition devices

4.2.1 *Cybersystems Cyberglove II*

Data glove systems are devices designed to acquire data about movements of the hand: specifically, the level of flexure of the joints of fingers, palm, and wrist. A survey by

[Dipietro et al., 2008] provides a detailed description of the historical evolution of the research (technologies, main features, and materials) in the field of data glove development. The type of glove, number of sensors and their position on the glove, as well as the glove material differ among the various commercial devices available on the market. One of the most popular high-end models is the *Cybersystems CyberGlove II*.



(a) (b) (c)

(d) (e) (f)

Figure 4.3: Human hand instrumented using different data acquisition devices (*Cybersystems CyberGlove II*, *Polhemus Liberty*, *Tekscan Grip* system).

The *Cybersystems CyberGlove II* [CyberGloveII, 2008] (CyberGlove Systems LLC, San Jose, CA, U.S.) is a wireless data glove. *Cybersystems CyberGlove II* is equipped with 22 piezo resistive bend sensors. Each glove's finger (index, middle, ring, and little) has two bend sensors (located in the metacarpophalangeal (MCP) and proximal interphalangeal joint regions), an abduction/adduction sensor (located in the MCP region), and an additional sensor in each finger (index, middle, ring, and little) to measure the distal interphalangeal joint flexure. The *Cybersystems CyberGlove II* also has sensors to measure the thumb crossover, palm arch, wrist flexure, and abduction/adduction.

The *Cybersystems CyberGlove II* is powered by a battery attached to the forearm. A wireless module (Bluetooth) transmits the data acquired by the glove to a host computer. The glove has mounting provisions in the wrist region for motion tracking sensors. However, it can be integrated easily with other types of sensing devices such as tactile sensing arrays, or motion trackers on wrist and fingertips, as described in Figure 4.3.

4.2.2 *Polhemus Liberty*

Most of the data gloves available on the market only provide information about the level of flexion/bending of the fingers and wrist joints. However, some applications also require the determination of the global position and orientation of the hand (wrist) in space (reach-to-grasp trajectories, manipulation trajectories), as well as the relative location of the joints of the hand or the fingertips, palm global orientation, and position. This type of data can be provided by motion tracking devices. These devices can be used in conjunction with the data gloves (Figure 4.3).

The *Polhemus Liberty* system [Polhemus-Liberty, 2008] (Polhemus, Colchester, VT, U.S.) is an electromagnetic motion tracking device used to record the pose (6DOF) (position and orientation) of magnetic sensors within a restricted area. The main components of this device are the *system central unit*, the sensors, and the *source*.

The *system control unit* contains the hardware and software responsible for the algorithms supporting the emission and sensing of magnetic fields and interpretation of the sensed fields. This unit is responsible for determining the position and orientation of each sensor. The unit communicates with a computer through a *RS-232* or *USB* connection. The *source* of the *Polhemus Liberty* device is responsible for the emission of the magnetic field. The source defines the inertial reference frame for sensor measurements.

Each sensor is connected to the *system control unit* by a cable. The position and orientation of the sensors is determined by considering the magnetic field sensed by each sensor. The *Polhemus Liberty* can have up to eight sensors connected to the *system control unit*, of which the position and orientation can be updated up to 240 times per second.

This type of motion tracker has a limited operational range around the magnetic field source, typically between 0.5 and 2 m. However, there are no requirements for direct line of sight between the sensors and the magnetic source. This device requires that the environment, where the data will be acquired, does not have metallic materials near or between the magnetic field source and the magnetic sensors. This requirement ensures that the metallic materials do not distort the emitted magnetic field, thus avoiding the introduction of large errors in the measurements.

4.2.3 *Tekscan Grip*

The devices presented previously were used to determine the position, orientation, and level of flexure of the hand or specific segments of the hand. However, analysis of the force/pressure applied by the hand during the execution of grasping and manipulation tasks is also critical. This analysis will be important to determine the temporal profile of the pressure/forces applied by the human hand (fingertips and palm) while performing different types of grasps. This analysis can also provide information about the fingers that are more active in different types of grasps and about the sequence of segments of the hand that come into contact with the object.

The tactile sensing device *TekScan Grip* [Tekscan-Grip, 2010] (Tekscan Inc, Boston, MA, U.S.) is a system specifically designed to acquire the pressure applied by the different regions of the human hand (fingers, thumb, and palm) during the execution of tasks that require grasping movements. The device consists of a flexible thin film (0.1 mm) with embedded pressure sensors and electronics connecting the sensors to a data acquisition module (*TekScan VersaTek*) attached to the wrist region. The *TekScan VersaTek* module acquires data at up to 850 Hz and provides a *USB* connection to transfer the acquired data to a computer.

The *TekScan Grip* system is composed of five segments which can be attached to the ventral regions of the fingers, thumb, and palm of the human hand (total of 361 sensing elements). The sensing regions have a spatial resolution of about 6.2 sensing elements per square centimeter, and each element can sense pressures up to 50 psi (344.7 KPa). This device can be used in either the right or left hand.

The *TekScan Grip* system can be attached directly to the human hand or to a data glove (e.g. *Cybersystems CyberGlove II*), as demonstrated in Figure 4.3.

The performance, characteristics, and calibration methods of the *TekScan Grip* system were tested extensively and documented in the technical report *Experimental evaluation and calibration protocol of Tekscan Grip system* [Martins, 2012b] available online (url: `http://www.rmartins.net/phd-docs/tr05/`).

4.2.4 Instrumented objects

The previous sections described the sensors/devices that can be attached to the human hand. These devices give a human hand-centred perspective.

However, it can be useful to acquire data related to the movement/pose of the object during the task being performed; this is the object-centred approach. This type of data can be obtained by integrating external sensors on the object (attaching motion trackers or tactile arrays), or making custom-designed objects with those types of sensors embedded in the design (e.g. instrumented Rubik cube and instrumented sensing can). The object-centred approach does not constrain the natural movements of the human hand.

Instrumented Rubik cube

The instrumented Rubik cube has the same dimensions and colors as a standard Rubik cube, as presented in Figure 4.4a. The instrumented Rubik cube [HANDLE-SHADOW,

(a) (b) (c)

Figure 4.4: Instrumented objects. a) Rubik cube. b) sensing soda can. c) *Nintendo Wiimote*

.

2009] (Shadow Robot Company, London, UK) has 54 tactile sensing regions (six in each side of the cube) that provide non-calibrated 12-bit-resolution tactile data. Each side of the instrumented Rubik cube is equipped with a three-axis accelerometer, providing acceleration data with eight-bit resolution. The instrumented Rubik cube is powered by a cabled external source and has a wired CAN-BUS communication interface. A technical report *Installing controller area network (CAN-Bus) drivers and compiling code on Ubuntu* was published and is available online [Martins, 2013] (url: `http://www.rmartins.net/phd-docs/tr04/`).

Instrumented sensing can

The instrumented sensing can [HANDLE-SHADOW, 2010] (Shadow Robot Company, London, UK) is designed to mimic a standard 330 ml soda can (Figure 4.4b). It has a total of 40 (4×10) independent tactile sensing regions distributed using ten sensing vertical panels. Each panel contains four tactile sensing elements (eight-bit resolution per sensing element) and has a three-axis accelerometer attached to it (ten-bit resolution per channel).

The instrumented sensing can is portable. It is powered by an internal set of four AAA batteries, and it has a Bluetooth communication interface.

Nintendo Wiimote

The *Nintendo Wiimote* [Nintendo, 2006] (Nintendo Co. Ltd, Kyoto, Japan) is a useful instrumented object that can be employed during the data acquisition sessions (Figure 4.4c). The *Nintendo Wiimote* is a portable device (Bluetooth interface) of an appropriate size for handling tasks; it contains embedded sensors (three-axis accelerometer and infrared camera sensor) and several buttons.

The accelerometer provides information about the movements of the device. The infrared camera sensor has an integrated multi-object tracking engine that can determine the position (x,y) and size of up to four simultaneous infrared light sources [Lee, 2008]. The buttons can be used to acquire user inputs during the execution of the task.

4.2.5 *Microsoft Kinect*

The *Microsoft Kinect* [Microft-Kinect, 2009] (Microsoft Corporation, Redmond, WA, U.S.) is a device initially designed to be used by the gaming and entertainment industry as a human-computer interface [Smisek et al., 2013]. However, due to its versatility and low price, it became a revolutionary device in other areas such as robotics, physical rehabilitation and therapy, art installations, and do-it-yourself (DIY) homemade projects [Zhang, 2012].

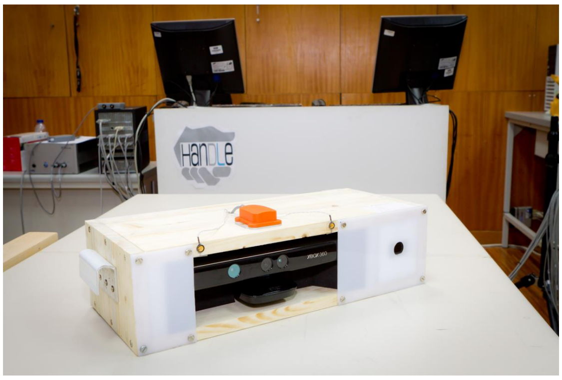


Figure 4.5: *Microsoft Kinect* integrated in a wood box.

The *Microsoft Kinect* device integrates different types of sensors (Figure 4.5). An infrared (IR) projector of light patterns and IR camera (structured light principles) provides a depth map (1280×1024) of the environment surrounding the device. The depth data is fused with the output of a monocular *RGB* camera (1280×1024), generating a *RGB-D* map (cloud points with color). The *Microsoft Kinect* also incorporates an array of four microphones. They can be used to implement voice recognition systems and speaker spatial tracking applications.

Several software development kits (SDK) that have been released take advantage of the features of the *Microsoft Kinect*: face recognition, human body tracking, and 3D reconstruction.

4.2.6 *Videre* camera

The *Videre STH-MD CS3* [Videre, 2006] is a digital stereo camera consisting of two 1.3 MP CMOS sensors mounted on a metallic (aluminium) package, as shown in Figure 4.6b. This camera has a fixed 9 cm baseline.

The stereo camera can acquire colour and monochromatic data at different frame rates (30 Hz for 640x480, 15 Hz for 1024x768 colour only, and 7.5 Hz for 1280x960). The camera is powered through a firewire cable, which is also used for data transmission and control (synchronization, exposure, gain, and colour balance).

(a)

(b)

Figure 4.6: RGB cameras: a) *Unibrain* camera. b) *Videre* camera.

4.2.7 *Unibrain* camera

The *Unibrain Fire-I Digital* [Unibrain, 2007] (Unibrain, Athens, Greece) is a firewire colour digital camera (CCD sensor), as presented in Figure 4.6a. This device can acquire video at different frame rates (30, 15, 7.5, 3.75 frames per second), different resolutions (640x480, 320x240, 160x120) and video coding modes (YUV, RGB 24-bit, monochromatic eight-bit).

The *Unibrain Fire-I Digital* can be powered by the firewire cable (when the camera is connected to a desktop computer) or through a DC adapter (when the camera is connected to a laptop computer). The camera is housed in a plastic polymer package.

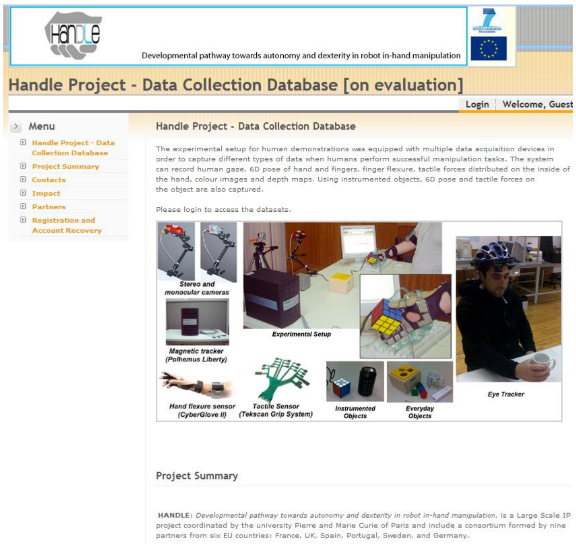


Figure 4.7: Homepage of the HANDLE project [HANDLE, 2009] online data repository.

4.3 Datasets

The datasets described in this section were recorded at the experimental area of *Artificial Perception for Intelligent Systems and Robots* AP4ISR laboratory during the PhD studies, under the scope of the HANDLE project. The datasets were acquired to fulfil the objectives and requests made by other partners. The publications and applications related to each of the datasets are listed in Table 4.1. All the datasets are available online at the HANDLE project web repository.

The participants were seated in a chair in front of the table presented in Figure 4.1. The data acquisition devices and objects used in each session are summarized in Table 4.1, and the configuration of the experimental area is demonstrated in the figures listed in the description of each dataset (section 4.3.1 to 4.3.8).

4.3.1 Dexterous manipulation of a laboratory pipette

The participants were instructed to move a liquid from a container to a second container, using a laboratory pipette. The experimental materials were dispensed as presented in Figure 4.8. The laboratory pipette was grasped and manipulated using only the dominant hand. Before starting each run, the pipette was placed on a pipette stand. The participants were instructed to place the pipette back on the stand or to leave the pipette directly on top of the table by the end of each run. A video demonstrating this task is available online `www.rmartins.net/phd-docs/ds01`.

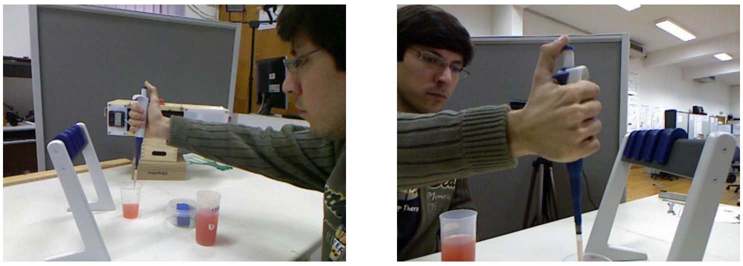
To stimulate and challenge the participants with unexpected situations, during some

Table 4.1: List of HANDLE datasets acquired during the PhD studies

Task	**Data Acquisition Devices**	**Participants**	**Runs**	**Application**
Dexterous manipulation of a laboratory pipette	2 *Microsoft Kinect*	12	240	final demonstration: HANDLE project
Thumb movement during manipulation tasks	1 *Microsoft Kinect* 7 *Polhemus Liberty* sensors 1 *Cyberglove II*	1	5	journal paper: HANDLE partner [Berglund et al., 2012]
Screwdriver in-hand rotation	1 *Microsoft Kinect* 1 *Cyberglove II* 1 *Tekscan Grip*	1	5	conference paper, PhD thesis: HANDLE partner [Cheng et al., 2012], [Cheng, 2013]
In-hand manipulation of toys	1 *Microsoft Kinect* 8 *Polhemus Liberty sensors* 1 *Videre* camera 1 *Tekscan Grip*	1	35	conference paper, PhD thesis: HANDLE partner [Cheng et al., 2012], [Cheng, 2013]
Grasp the Wii remote and press a button	1 *Cyberglove II* 1 *Unibrain* camera 6 *Polhemus Liberty* sensors 1 *Videre* camera 1 *Tekscan Grip*	3	18	conference papers: HANDLE partner [Hendrich et al., 2010], [Hendrich et al., 2012]
Fill a toy sorting box with objects	1 *Cyberglove II* 1 *Unibrain* camera 5 *Polhemus Liberty* sensors 1 *Videre* camera 1 *Tekscan Grip*	3	17	conference papers: HANDLE partner [Hendrich et al., 2010], [Hendrich et al., 2012]
Pick up a pen and write	1 *Unibrain* camera 8 *Polhemus Liberty* sensors 1 *Videre* camera 1 *Tekscan Grip*	3	10	conference papers: HANDLE partner [Hendrich et al., 2010], [Hendrich et al., 2012]
Pick an object and slide	1 Instrumented rubik cube 1 *Unibrain* camera 7 *Polhemus Liberty* sensors 1 *Videre* camera 1 *Tekscan Grip*	4	20	conference papers: HANDLE partner [Hendrich et al., 2010], [Hendrich et al., 2012]

runs, the manipulation of the pipette was done without using the thumb to press the pipette buttons. This variant of the protocol was used to record the ways in which the participants adapted the standard manipulation strategy to this constraint. The data acquired during the demonstrations was manually annotated (grasp and in-hand manipulation primitives) using the MATLAB tool developed for that purpose and presented in section 4.4.3.

The datasets were used by several partners of the HANDLE project to develop various applications concerning robotic learning by human demonstration. This approach was used to provide the SHADOW robotic platform with pipette manipulation skills. The results were demonstrated during the final review meeting of the HANDLE project. The HANDLE final demonstration was presented publicly in *Euronews Futuris*, disseminating science and technology. The video of the demonstration is available online at `http://www.youtube.com/watch?v=XSw5QVdzGW4`. Figure 4.9 summarizes some segments of the TV show. The datasets are available online at `http://www.rmartins.net/phd-docs/ds01/`.



(a) (b)

Figure 4.8: Participant demonstrating the task *Dexterous manipulation of a laboratory pipette.*

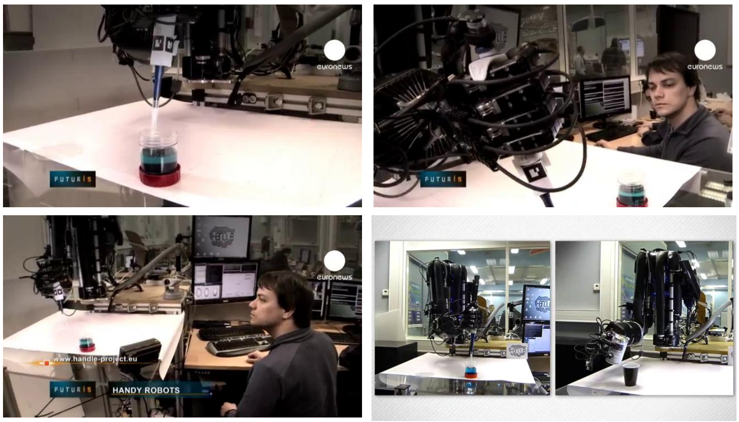
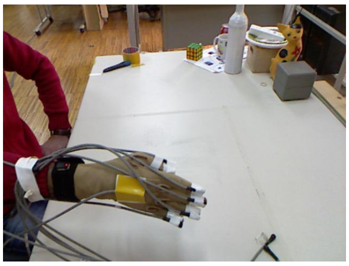


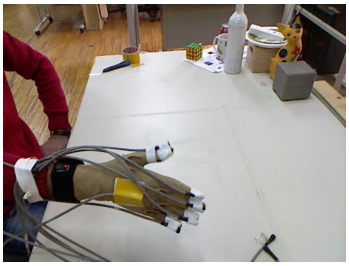
Figure 4.9: Final demonstration of HANDLE project presented during *Euronews* TV show *Futuris* (url: `https://www.youtube.com/watch?v=XSw5QVdzGW4`). The final demonstration consisted of the robotic dexterous manipulation of a laboratory pipette learned from the human demonstrations. The TV show also featured the instrumented Rubik cube data visualization tool presented in section 4.4.4 (url: `https://www.youtube.com/watch?v=KJybBorZjH0`).

4.3.2 Thumb movement during manipulation tasks

The participant started with the palm of the right hand flat on the table. The hand was lifted off the table, keeping the palm and index, middle, and little fingers straight (Figure 4.10). The thumb was then moved around. The participant went through all possible movements of the thumb several times, exploring all the degrees of freedom of that finger. A video demonstrating this task is available online at `www.rmartins.net/phd-docs/ds02/`.

The data recorded by the data glove, motion tracker, and camera was sent to the ORU partner (Orebro University - AASS Learning Systems Lab) of the HANDLE project. The data was used to train an algorithm that learns the mapping and correspondences between the kinematic structure of the human hand and Shadow robotic hand. The human thumb





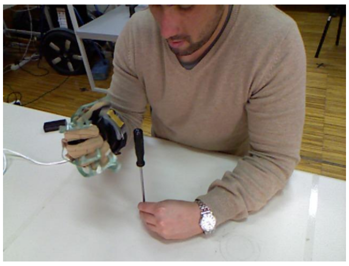
(a) (b)

Figure 4.10: Participant demonstrating the task *Thumb movement during manipulation tasks*.

has a complex mechanical structure, which is different from the kinematic structure of the robotic thumb. The data was used to improve the behaviour of the robot during robotic learning from human demonstrations or in tele-manipulation applications. The results of this work were published in an international journal [Berglund et al., 2012].

4.3.3 Screwdriver in-hand rotation

The participant is instructed to use the two hands to grasp a screw and use a screwdriver to insert and adjust the screw in a hole. The participant was seated in front of the table, where a screwdriver and screw were placed in their initial configurations (Figure 4.11).



(a) (b)

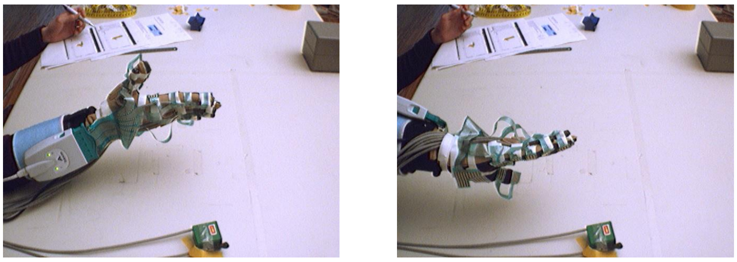
Figure 4.11: Participant demonstrating the task *Screwdriver in-hand rotation*.

The left hand was not instrumented, and it was used to grasp the screw, insert it in the hole, and keep it stable as long as the task progressed. The right hand was used to perform consecutive in-hand manipulation movements (re-grasps), using the screwdriver, to insert the screw. By the end of each run, the participant placed the screwdriver on the table (initial configuration). A video demonstrating this task is available online `www.rmartins.net/phd-docs/ds03/`.

The data was delivered to the UHAM partner (University of Hamburg - Technical Aspects of Multimodal Systems (TAMS) Lab) of the HANDLE project. The data was used to develop a software tool to automatically segment dexterous in-hand manipulation movements. The automatic segmentation was compared (benchmark) to the segmentation performed manually by a human operator. This work was published at an international conference [Cheng et al., 2012] and in a PhD thesis [Cheng, 2013].

4.3.4 In-hand manipulation of toys

The participant was instructed to grasp the toys (wood pieces) placed on top of the table and then perform several in-hand manipulation movements to rotate the toy around itself inside the hand. When the demonstration finished, the toy was placed back on the table.



(a) (b)

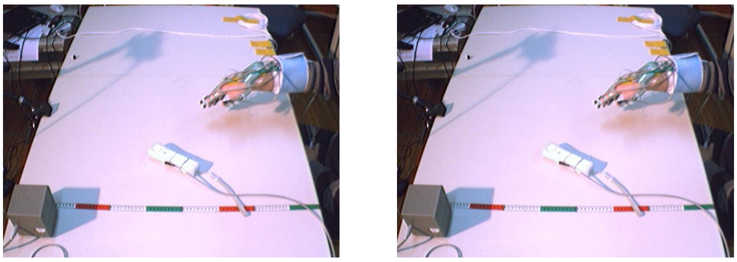
Figure 4.12: Participant demonstrating the task *In-hand manipulation of toys*.

The data was delivered to the UHAM partner (University of Hamburg - Technical Aspects of Multimodal Systems (TAMS) Lab) of HANDLE project. The objectives and applications of this data acquisition were the same as the objectives reported for the dataset described in section 4.3.4. A video demonstrating this task is available online `www.rmartins.net/phd-docs/ds04/`.

4.3.5 Grasp the Wii remote and press a button

The participant was seated in a chair in front of the table where the *Nintendo Wii-mote* was placed. Three different orientations of the *Nintendo Wii-mote* on the table were used as starting pose. As the data recording starts, the subject moves the right hand toward the *Nintendo Wii-mote*, grasping and picking-up the object (Figure 4.13). The subject performed some in-hand manipulation movements so that the *Nintendo Wii-mote* points toward the frontal camera recording the session. Participants used the thumb to press a button on the *Nintendo Wii-mote* and then put the *Nintendo Wii-mote* back on the table in the starting configuration.

This dataset was acquired and delivered to the UHAM partner (University of Hamburg - Technical Aspects of Multimodal Systems (TAMS) Lab) of the HANDLE project. The multi-modal dataset was used to develop a method to automatically segment the data



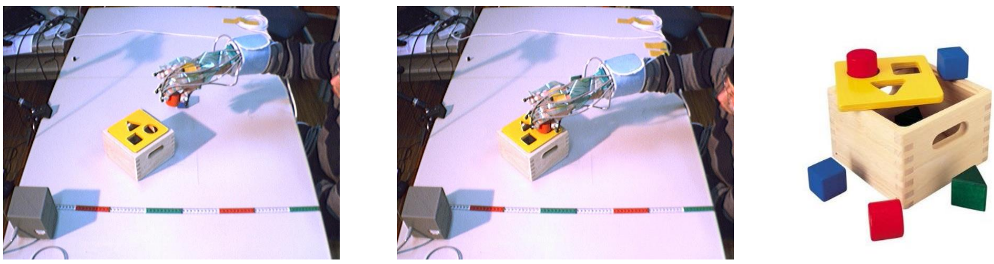
(a) (b)

Figure 4.13: Participant demonstrating the task *Grasp the Wii remote and press a button*.

from in-hand manipulation tasks. The primitives consist of a basic set of finger and hand movement patterns, which can be used to model manipulation tasks. The learned model was transferred to robotic platforms. The datasets were used in the experimental implementation of the work [Hendrich et al., 2010], which was published in the proceedings of an international conference. A video demonstrating this task is available online at `www.rmartins.net/phd-docs/ds05/`.

4.3.6 Fill a toy sorting box with objects

The participant was seated in front of a table where the toy sorting box and the toys were placed in their starting configurations, as shown in Figure 4.14.



(a) (b) (c)

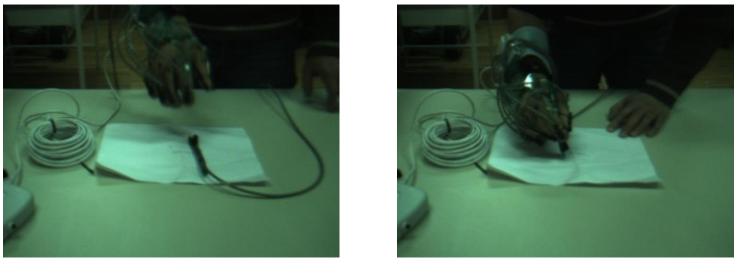
Figure 4.14: a) - b) Participant demonstrating the task *Fill a toy sorting box with objects*. c) Toy sorting box and toys (wood pieces).

The participant was instructed to use the right hand to consecutively grasp each of the toys displayed on the table and fit each of them in the corresponding hole of the box. The positions of the toys were adjusted using only in-hand manipulation movements during the transport movement between the table and the toy sorting box. Each run ended when all the toys were placed inside the box.

This dataset was acquired and delivered to the UHAM partner (University of Hamburg - Technical Aspects of Multimodal Systems (TAMS) Lab) of the HANDLE project. The motivation and applications of this dataset were the same as for the dataset described in section 4.3.5. A video demonstrating this task is available online at `www.rmartins.net/phd-docs/ds06/`.

4.3.7 Pick up a pen and write

The participant was seated in front of a table where a pen and a piece of paper were placed in their starting configurations (Figure 4.15). The participant manipulated a standard ball-point pen with typical active click mechanics on the back-end. The pen was placed in different starting poses. The pen was also instrumented with a motion tracking sensor.



(a) (b)

Figure 4.15: Participant demonstrating the task *Pick up a pen and write*.

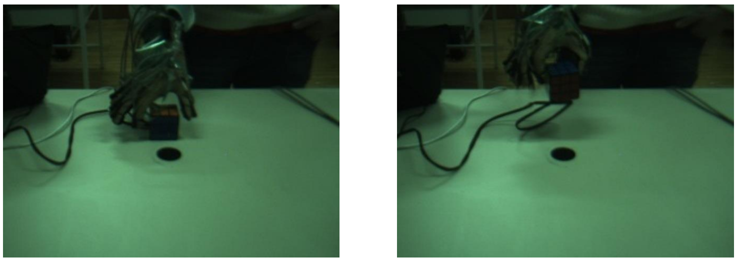
The participant started each run by picking up the pen. After grasping the pen, in-hand manipulation movements were performed to move the pen to a configuration suitable to activate the click mechanism with the thumb a few times. Then, the pen was in-hand manipulated to achieve a configuration for writing. The participant wrote *"HANDLE"* on a piece of paper. After that, the participant placed the pen back on the table in the starting configuration.

This dataset was acquired and delivered to the UHAM partner (University of Hamburg - Technical Aspects of Multimodal Systems (TAMS) Lab) of the HANDLE project. The motivation and applications of this dataset were the same as for the dataset described in section 4.3.5. A video demonstrating this task is available online at `www.rmartins.net/phd-docs/ds08/`.

4.3.8 Pick an object and slide

The participant was comfortably seated in front of the table where the objects were placed. The participants performed this task using two different objects: an instrumented Rubik cube and a box, as described in Figures 4.16 and 4.17, respectively.

The participant was instructed to grasp the object (box or instrumented Rubik cube) with the index finger and thumb. This precision grasp was performed not in the geometric

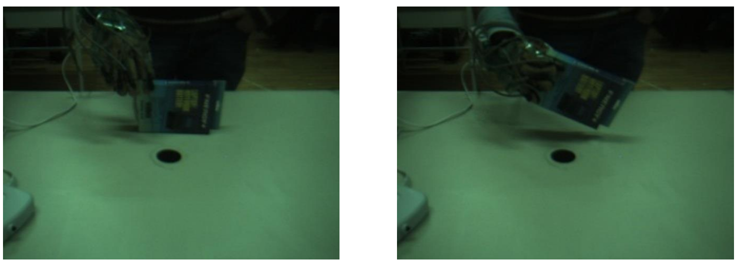


(a) (b)

Figure 4.16: Participant demonstrating the task *Pick an object and slide* with an instrumented Rubik cube.

center of the object, but near one end. The object was moved up using only the index and thumb, maintaining the orientation of the object. The index and thumb applied a pressure, which prevented the inertial momentum of the object from changing its pose. After reaching the maximum height of the movement, the participant reduced the pressure applied by the index and thumb fingers, allowing the object to slide and rotate around the index and thumb fingers, changing its orientation. The object was then moved back to its starting pose on the table.

This dataset was acquired and delivered to the UHAM partner (University of Hamburg - Technical Aspects of Multimodal Systems (TAMS) Lab) of the HANDLE project. The motivation and applications of this dataset were the same as for the dataset described in section 4.3.5. A video demonstrating this task is available online `www.rmartins.net/phd-docs/ds09/`.



(a) (b)

Figure 4.17: Participant demonstrating the task *Pick an object and slide* with box.

4.4 Software tools

4.4.1 Software clients for data acquisition devices

As described in section 4.1, each of the data acquisition devices has a dedicated software client. The software clients are responsible for the integration of the data acquisition devices in the distributed data acquisition architecture. They connect to a server software which coordinates the data acquisition session, and they receive the triggers to start and stop the data acquisition during each run.

Parameters such as data acquisition rate, image resolution, and calibration parameters are configured locally in each client prior to the start of the session. The data acquired by each device is formatted by the corresponding software client according to the XML format presented in the technical report *Protocol for the corpus of sensed grasp and handling data: storage of multi-modal datasets* [HANDLE-UC, 2009].

During the PhD studies reported in this thesis, I was responsible for the implementation of the software clients for the *Cybersystems Cyberglove II* [Martins, 2009b], instrumented Rubik cube [Martins, 2009c], instrumented sensing can [Martins, 2009d], *Tekscan Grip* system [Martins, 2009f], and an alternative software client for *Polhemus Liberty* [Martins, 2009e]. All software clients were developed using *C++* programming language for *Ubuntu* environment. The alternative software client for *Polhemus Liberty* was developed using *Python*.

The programming of these clients required the analysis of requisites for each of them and the understanding of the technologies: I/O communications (USB, RS-232, CAN-bus), TCP-IP socket communications, and XML data structure.

4.4.2 *importDatasetTB*: toolbox for integrating data in MATLAB

A MATLAB toolbox (*importDatasetTB*) was developed to promote the use of the datasets by the HANDLE partners, improve dissemination of the datasets, and facilitate the integration of the datasets in the research activities of the HANDLE partners (including this PhD study). The *importDatasetTB* MATLAB toolbox automatically imports the different types and configurations of datasets that were recorded using the XML storage scheme described in the technical report *Protocol for the corpus of sensed grasp and handling data: storage of multi-modal datasets* [HANDLE-UC, 2009].

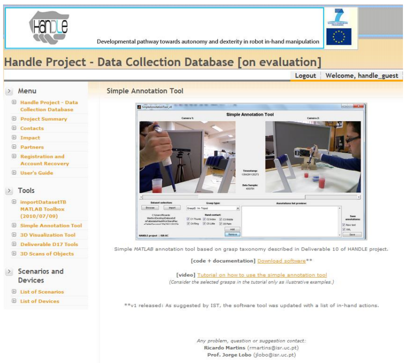
The toolbox allows an automatic and fast integration of large multi-modal datasets. Initially, the toolbox was compatible with datasets containing data structures of *Cybersystems Cyberglove II*, *Polhemus Liberty*, instrumented Rubik cube, instrumented sensing

Figure 4.18: Section of HANDLE project website [HANDLE, 2009] presenting and supporting the MATLAB toolbox *importDatasetTB*.

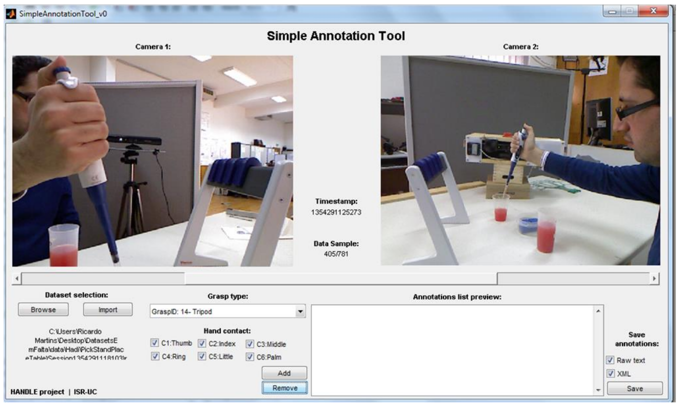
can, *Tekscan Grip* system, monocular RGB cameras, and stereo RGB cameras. Afterward, it was extended and made compatible with additional data acquisition devices of the *Artificial Perception for Intelligent Systems and Robots* AP4ISR laboratory: *Microsoft Kinect*, *Nintendo Wii-mote*, *NDI optotrak*, and *Xsens IMU*. The *importDatasetTB* MATLAB toolbox source code and documentation are described in the technical report *"importDatasetTB: a MATLAB toolbox to integrate multi-modal datasets"* [Martins, 2009g].

This MATLAB toolbox, *importDatasetTB*, was developed on top of a Java library (*jParserToolbox*) which was programmed from scratch for this tool. This Java library implements a XML *SAX* (Simple API for XML) parser for the XML dataset structures [HANDLE-UC, 2009]. This custom-designed Java library was used in the development of the MATLAB toolbox because MATLAB natively includes a Java Virtual Machine. Thus, the Java interpreter can be used via the MATLAB scripting shell. The *importDatasetTB* MATLAB toolbox imports datasets of hundreds of megabytes in a few seconds. This tool improved the low performance and bad memory management for large XML data structures of the XML parsing tools native with MATLAB; it is based in DOM (Document Object Model).

The source code and documentation with practical examples are available online at [Martins, 2009g] (url: `http://www.rmartins.net/phd-docs/st02/`).



(a)



(b)

Figure 4.19: a) Section of HANDLE project website [HANDLE, 2009] presenting and supporting the MATLAB annotation tool for multi-modal datasets. b) Snap shot of the graphical interface of the MATLAB annotation tool for multi-modal datasets.

4.4.3 Annotation tool for multi-modal human grasping datasets

The dataset *Dexterous manipulation of a laboratory pipette* presented in section 4.3.1 was annotated offline by several partners of the HANDLE project. To facilitate this task, an annotation tool was developed using MATLAB. The MATLAB annotation tool presents to the user a graphical interface displaying side-by-side two synchronized images from the two *Microsoft Kinect* RGB cameras. The user performing the annotation employs the two perspectives of the manipulation strategy to annotate the images with the types of grasps and in-hand manipulation actions performed by the participant. The dictionary of possible labels is presented on the graphical interface and follows the categories of manipulation movements given in section 3.4. The MATLAB exports the annotations in plain text or XML format.

The annotation tool is demonstrated in Figure 4.19a. The MATLAB code, documentation, and video tutorials are available online at [Martins, 2012c] (url: `http://www.rmartins.net/phd-docs/st09/`).

4.4.4 Instrumented Rubik cube: touch data visualization tool

An interactive and intuitive 3D visualization tool was developed to demonstrate the design and touch sensing capabilities of the instrumented Rubik cube (presented in section 4.2.4). The 3D model of a Rubik cube changes the color of each cell, as long as the corresponding sensing element is pressed on the real cube. The visualization tool was developed in *Ubuntu*, using the *C++ OpenGL-GLUT* library.

The 3D visualization tool was presented during the final demonstration of the HAN-

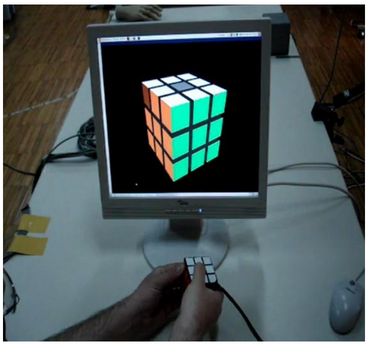


Figure 4.20: Demonstration of the instrumented Rubik cube visualization tool during a interactive session.

DLE project. The source code, tutorials for the installation of the *CAN-BUS* Linux drives, and *OpenGL-GLUT* libraries used on this Linux C++ software project are available on-line at [Martins, 2012a] (url: `http://www.rmartins.net/phd-docs/st08/`). A video demonstrating a typical live interaction with the instrumented Rubik cube is available as well [Martins, 2012a], and it is illustrated in Figure 4.20.

Bibliography

[Argall et al., 2009] Argall, B. D., Chernova, S., Veloso, M., and Browning, B. (2009). A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5):469 – 483.

[Berglund et al., 2012] Berglund, E., Iliev, B., Palm, R., Krug, R., Charusta, K., and Dimitrov, D. (2012). Mapping between different kinematic structures without absolute positioning during operation. *Electronics Letters*, 48(18):1110–1112.

[Billard et al., 2008] Billard, A., Calinon, S., Dillmann, R., and Schaal, S. (2008). Robot programming by demonstration. In *Springer Handbook of Robotics*, pages 1371–1394.

[Cheng, 2013] Cheng, G. (2013). *State-Action Gist based In-hand Manipulation Learning from Human Demonstration*. PhD thesis, Universitt Hamburg, Von-Melle-Park 3, 20146 Hamburg.

[Cheng et al., 2012] Cheng, G., Hendrich, N., and Zhang, J. (2012). Action gist based automatic segmentation for periodic in-hand manipulation movement learning. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 4768–4775.

[CyberGloveII, 2008] CyberGloveII (2008). Cyberglove ii - wireless instrumented glove. URL: `http://www.cyberglovesystems.com/cyberglove-ii/`.

[Dipietro et al., 2008] Dipietro, L., Sabatini, A., and Dario, P. (2008). A survey of glove-based systems and their applications. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 38(4):461 –482.

[Faria et al., 2012] Faria, D. R., Martins, R., Lobo, J., and Dias, J. (2012). Extracting data from human manipulation of objects towards improving autonomous robotic grasping. *Robotics and Autonomous Systems*, 60(3):396 – 410.

[HANDLE, 2009] HANDLE (2009). project handle - developmental pathway towards autonomy and dexterity in robot in-hand manipulation. URL: `http://www.handle-project.eu/`.

[HANDLE-SHADOW, 2009] HANDLE-SHADOW (2009). Instrumented rubik cube - augmented sensing object. *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation*. URL: `http://www.rmartins.net/phd-docs/irc/`.

[HANDLE-SHADOW, 2010] HANDLE-SHADOW (2010). The can sensing object. *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation*. URL:`http://www.rmartins.net/phd-docs/isc/`.

[HANDLE-UC, 2009] HANDLE-UC (2009). Protocol for the corpus of sensed grasp and handling data: storage of multi-modal datasets. *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation*. `http://www.rmartins.net/phd-docs/tr02/`.

[Hendrich et al., 2010] Hendrich, N., Klimentjew, D., and Zhang, J. (2010). Multi-sensor based segmentation of human manipulation tasks. In *Multisensor Fusion and Integration for Intelligent Systems (MFI), 2010 IEEE Conference on*, pages 223–229.

[Hendrich et al., 2012] Hendrich, N., Klimentjew, D., and Zhang, J. (2012). 5th international conference on cognitive systems, cogsys2012, vienna.

[Lee, 2008] Lee, J. (2008). Hacking the nintendo wii remote. *Pervasive Computing, IEEE*, 7(3):39 –45.

[Lobo et al., 2011] Lobo, J., Trindade, P., and Dias, J. (2011). Observing hand grasp type and contact points using hand distributed accelerometers and instrumented objects. In *Proc. of IEEE/ICRA 2011 Workshop on Autonomous Grasping, Shanghai, China.*

[Martins, 2009a] Martins, R. (2009a). Distributed data acquisition architecture: software client for cyberglove (data glove). *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation*. URL: `http://www.rmartins.net/phd-docs/st03/`.

[Martins, 2009b] Martins, R. (2009b). Distributed data acquisition architecture: software client for instrumented rubik cube (instrumented object). *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation*. URL: `http://www.rmartins.net/phd-docs/st04/`.

[Martins, 2009c] Martins, R. (2009c). Distributed data acquisition architecture: software client for instrumented sensing can (instrumented object). *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation*. URL: `http://www.rmartins.net/phd-docs/st05/`.

[Martins, 2009d] Martins, R. (2009d). Distributed data acquisition architecture: software client for polhemus liberty system (6d motion tracking). *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation.* URL: http://www.rmartins.net/phd-docs/st07/.

[Martins, 2009e] Martins, R. (2009e). Distributed data acquisition architecture: software client for tekscan grip system (tactile sensing array). *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation.* URL: http://www.rmartins.net/phd-docs/st06/.

[Martins, 2009f] Martins, R. (2009f). importdatasettb: a matlab toolbox to integrate multi-modal datasets. *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation.* URL: http://www.rmartins.net/phd-docs/st02/.

[Martins, 2010] Martins, R. (2010). Distributed synchronization of multi-modal data acquisition devices using ntp (network time protocol). *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation.* URL: http://www.rmartins.net/phd-docs/tr03/.

[Martins, 2012a] Martins, R. (2012a). 3d visualization tool of instrumented rubik cube touch data. *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation.* http://www.rmartins.net/phd-docs/st08/.

[Martins, 2012b] Martins, R. (2012b). Experimental evaluation and calibration protocol of tekscan grip system. *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation.* URL: http://www.rmartins.net/phd-docs/tr05/.

[Martins, 2012c] Martins, R. (2012c). Matlab annotation tool for the dataset "dexterous manipulation of a laboratory pipette". *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation.* URL:http://www.rmartins.net/phd-docs/st09/.

[Martins, 2013] Martins, R. (2013). Installing controller area network (can-bus) drivers and compiling code on ubuntu. *HANDLE FP7 project: Developmental pathway towards autonomy and dexterity in robot in-hand manipulation.* URL: http://www.rmartins.net/phd-docs/tr04/.

[Matsuo et al., 2009] Matsuo, K., Murakami, K., Hasegawa, T., Tahara, K., and Kurazume, R. (2009). Segmentation method of human manipulation task based on measurement of force imposed by a human hand on a grasped object. In *Intelligent Robots* *and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 1767–1772. IEEE.

[Microft-Kinect, 2009] Microft-Kinect (2009). Microsoft kinect - controller-free gaming and entertainment experience. URL: `https://dev.windows.com/en-us/kinect`.

[Nintendo, 2006] Nintendo (2006). Nintendo wii remote - the primary controller for nintendo wii console. URL: `http://www.nintendo.com/wii/console/controllers`.

[Polhemus-Liberty, 2008] Polhemus-Liberty (2008). Polhemus liberty - electromagnetic motion 6d tracker. URL: `http://www.polhemus.com/?page=Motion_Liberty`.

[Smisek et al., 2013] Smisek, J., Jancosek, M., and Pajdla, T. (2013). 3d with kinect. In Fossati, A., Gall, J., Grabner, H., Ren, X., and Konolige, K., editors, *Consumer Depth Cameras for Computer Vision*, Advances in Computer Vision and Pattern Recognition, pages 3–25. Springer London.

[Tekscan-Grip, 2010] Tekscan-Grip (2010). Tekscan-grip system - tactile grip force and pressure sensing. URL: `http://www.tekscan.com/grip-pressure-analysis-system`.

[Unibrain, 2007] Unibrain (2007). Unibrain fire-i digital camera. URL: `http://www.unibrain.com/products/fire-i-digital-camera/`.

[Videre, 2006] Videre (2006). Videre sth-md cs stereo camera. URL: `http://www.rmartins.net/phd-docs/vsc`.

[Zhang, 2012] Zhang, Z. (2012). Microsoft kinect sensor and its effect. *MultiMedia, IEEE*, 19(2):4–10.