UNIVERSITÀ DEGLI STUDI DI PADOVA

Sede Amministrativa: Università degli Studi di Padova

Sede Consorziata: Università degli Studi di Trento

Dipartimento di Dipartimento di Ingegneria civile, ambientale e meccanica

SCUOLA DI DOTTORATO DI RICERCA IN : Scienze Tecnologie e Misure Spaziali (STMS)

INDIRIZZO: Misure Meccaniche per l'Ingegneria e lo Spazio (MMIS)

CICLO: XXVI°

Analisi del moto umano mediante sistemi di visione: stima dei parametri cinematici e dinamici

Analysis of human motion with vision systems: kinematic and dynamic parameters estimation

Direttore della Scuola : Ch.mo Prof. Giampiero Naletto

Coordinatore d'indirizzo: Ch.mo Prof. Stefano Debei

Supervisore :Ch.mo Prof. Mariolino De Cecco

Dottorando : Nicolò Biasi

Abstract

This work presents a multicamera motion capture system able to digitize, measure and analyse the human motion. Key feature of this system is an easy wearable garment printed with a color coded pattern. The pattern of coloured markers allows simultaneous reconstruction of shape and motion of the subject. With the information gathered we can also estimate both kinematic and dynamic motion parameters.

In the framework of this research we developed algorithms to: design the color coded pattern, perform 3D shape reconstruction, estimate kinematic and dynamic motion parameters and calibrate the multi-camera system. We paid particular attention to estimate the uncertainty of the kinematics parameters, also comparing the results obtained with commercial systems.

The work presents also an overview of some real-world application in which the developed system has been used as measurement tool.

Sommario

In questo lavoro viene presentato un sistema mutlicamera per la misura e digitalizzazione del moto umano. Caratteristica peculiare di questo sistema è l'indumento che deve essere indossato dal soggetto del quale si vuole ricostruire il moto. Su tale indumento è stampato un pattern di marker colorati che permette simultaneamente una digitalizzazione della forma e del moto del soggetto. Con queste informazioni è possibile ottenere una misura dei parametri cinematici e dinamici del moto umano.

Nel corso della ricerca sono stati sviluppati algoritmi per: la realizzazione del pattern di marker colorati, la ricostruzione di forma, l'analisi dei parametri cinematici e dinamici del moto e la calibrazione del sistema stesso. Particolare attenzione è stata dedicata alla analisi dell'incertezza della misura dei parametri cinematici e la comparazione con sistemi commerciali.

Vengono inoltre presentati alcuni progetti in cui il sistema realizzato è stati utilizzato come strumento di misura.

Contents

1 Introduction **7**

2 State of the art **11**
2.1 Shape reconstruction and motion estimation 11
2.2 Dynamic parameters estimation 15
2.3 Multicamera calibration 27
2.3.1 Camera model 29

3 Algorithms proposed **33**
3.1 Feature extraction 33
3.1.1 Feature detection 33
3.1.2 Color recognition 43
3.1.3 Feature descriptor 46
3.2 Shape reconstruction 56
3.3 Multicamera calibration 58
3.4 Motion analysis 66
3.4.1 Motion segmentation 66
3.4.2 Kinematic parameters estimation 70
3.4.3 Dynamic parameters estimation 74
3.5 Uncertainty estimation and Vicon comparison 77
3.5.1 Uncertainty propagation with Monte Carlo 77
3.5.2 Comparison with Vicon 80

4 Experimental results **85**
4.1 Hardware setup 85
4.2 Uncertainty analysis and Vicon Comparison 90
4.3 Kinematic and Dynamic parameters estimation of a motion 99

5 GaMoCap applications **105**
5.1 Veritas Project 105
5.2 Memscon 111

6 Conclusions **117**

Bibliography **119**

1 Introduction

The analysis of the human motion is a highly active research area due to the significative number of potential applications in different fields: health, sports, robotics, engineering or entertainment industry, etc.

The aim of the analysis of human motion is to gather quantitative information relative to the mechanics of the motion of a subject during the execution of a certain task: i.e. measure the variables that describes the kinematic and dynamic of the human limbs during that task.

In particular in medical applications motion capture systems are used as a clinical tool to evaluate quantitatively the parameters of motion. These information are useful for many purposes: evaluate the rehabilitation state of a patient, evaluate the result of a chirurgical hip operation, gait analysis just to name a few. The systems employed have thus to be to able to capture the human motion with an high degree of accuracy.

Thanks to a diversified application in several fields the human motion analysis draw a lot of attention of the scientific community and business enterprises which developed hardware and software specialized for this purpose.

The results are motion capture systems, or briefly MoCap, able to digitize and analyse the motion in details in terms of kinematic and dynamic parameters.

In the last decade, several MoCap technologies have been proposed and commercial systems based on each of them have been realized. Technologies can be classified as: inertial, magnetic and optoelectronic.

Each technology has its advantages and disadvantages and founds a particular field of application. For example inertial systems are widely used in the entertainment industry due to their the good cost-accuracy ratio. While magnetic systems are applied to the study of the motion of small and articulated body parts (hands) since they do not suffer of the occlusion problem. However are optoelectronic system, and in particular vision based systems, which are obtaining the best results.

In fact, in the last years the vision based systems are having a great success thanks to their ability to achieve precise measurements without contact and recover an high number of information.

The increased consumption of electronic goods has drastically reduced the production costs of integrated chips and optoelectronic devices.

While MoCap systems technologies here mentioned are able to capture the motion, kinematic or dynamic analysis requires a kinematic and dynamic model of the subject.

In order to obtain accurate results, it is of crucial importance the development of a personalised model with accurate parameters definition. The measurement of these parameters is a time expensive operation that usually requires a skilled operator and the use of specific instrumentation.

One key aspect to measure kinematic and dynamic parameters is the analysis of the 3D shape of the subject.

The main limitation of the mentioned MoCap systems is that, while they are able to capture the motion of the subject, no one manages to simultaneously capture its shape. This is due to the limited number of sensors that can be used together, in order to avoid problems relates to the reciprocal interference.

The shape of the subject can be digitize with 3D scanners. The leading technology on the marker are based on optoelectronic systems like laser or cameras.

While these systems are very accurate ($< 1mm^3$) and able to acquire a high number of points ($> 1M$) they have two main drawbacks that limit their use: high cost and long acquisition time. The last one, in particular, can cause big errors in shape estimation, allowing a relative motion between the subject and the sensors and thus a bad reconstruction.

In the last years several optoelectronic systems have been commercialized on the market (Kinect, TOF) trying to give a solution to these problems by offering a simultaneous reconstruction of shape and motion of a scene. These systems produce a direct 3D measurement of the environment with good acquisition frequency. Despite these good properties they have not yet reached accuracies suitable for clinical applications.

In this work we are interested in the development of a new concept multicamera MoCap able to perform a kinematic and dynamic analysis of the human motion.

The system proposed tries to solve the limits of mocap systems and laser scanners proposing a simultaneous reconstruction of the human shape and motion. These information are exploited to measure kinematic and dynamic parameters of motion without the definition of a known a priory model.

To perform a simultaneous reconstruction of shape and motion we are interested in the design of a special garment with a high density marker per unit of surface. This high density is possible only with special marker unique descriptors.

In order to capture the motion without interfering with it, we decided to rely on a multicamera vision system solution. The marker-based approach assures accuracies suitable for clinical applications.

As almost all vision marker-based systems, our system will suffer of Soft Tissue Artefacts. These artefacts, caused by relative motion of the skin (and the marker attached to it) and the underling bone, will be analysed and compensated.

We also aim to develop a toolbox that performs kinetic and dynamic motion analysis with limited user interaction required.

For this reason kinetic analysis of motion will be performed without an a-prior known model using a data driven technique. The realized algorithm will segment automatically the 3D cloud of points in limbs and identify the joint position without further information.

Dynamic analysis will be performed solving the inverse dynamic problem: from motion to the forces causing it.

These problems require the knowledge of some dynamic parameters like: dimension, mass and inertia of the limbs. These information will be recovered automatically from the shape reconstructed with a parametric model.

Mass, inertia, length and accelerations of limbs are used to solve the equation of the dynamic and thus infer torques and forces at joint.

We pay particular attention to model the uncertainties involved in the phase of angle of relative rotation between limbs. The uncertainty is propagated in the overall procedure up to the angle estimated with a Monte Carlo method.

The results of kinematic analysis, with particular attention to angle at joint estimated with GaMoCap, are compared with the commercial mocap system Vicon. The comparison is performed analysing lower and upper limb motion.

A prototype of the system has been realized and used in laboratory and real conditions.

The present work is organized in 4 parts.

First part, Chapter 2, includes a state of the art review on motion analysis which allow the reader to better understand the mocap systems with a particular focus on vision based systems and related problems. In this part are also analysed the state of the art tools to obtain reliable dynamic human parameters.

The second part, Chapter 3, describe in detail the algorithms proposed. This core Chapter is articulated in 5 sections.

Section 3.1 presents the algorithms proposed for features extraction, pattern generation and system calibration. 3D Shape reconstruction is reported in Section 3.2.

Since we realized a new concept mocap system, in Section 3.3 we propose a personalized calibration toolbox.

In Section 3.4 the strategies of kinematic and dynamic analysis are described in detail. Section 3.5 is dedicated to the uncertainty estimation of angle at joints and comparison with Vicon system.

The third part, Chapter 4, focuses on the analysis of experimental results to prove the usefulness of the kinematic and dynamic algorithms of human motion proposed.

In the last part, Chapter 5, two application in which the system was used as a measurement tool are reported.

The final Chapter presents the conclusions and future developments of this works.

2 State of the art

2.1 Shape reconstruction and motion estimation

The quest for an accurate shape and motion reconstruction has developed several motion capture approaches tuned to the specific analysis requirements and operational constraints. Advantages and disadvantages of each technology have been analysed and discussed in several comprehensive surveys [75, 104, 76].

The human mocap systems are here classified by the type of sensors employed (magnetic, inertial, optoelectronic) and the possibility to obtain a dense or sparse reconstruction of the shape of the human body.

Magnetic MoCap

Magnetic motion capture systems utilize low-frequency magnetic fields transmitter sources placed on the body limbs to reconstruct their position and orientation [53]. The fixed part of the setup is composed of 3D sensors that provide the magnetic field estimation from which is possible to reconstruct orientation and distance from the transmitter. This technology does not suffer of occlusion related problems due to the fact that human body is transparent to the magnetic fields generated by the transmitter. The problems are related to the presence of ferromagnetic materials in the capturing volume or sensors cross talk [109] that severely degrade accuracy. For these reasons magnetic motion capture is mainly employed for the study of small and complex parts, especially hands [88].

The reconstruction format is sparse due to the reduced number of markers (transmitters) that can be applied. These system can thus reconstruct only the motion not the shape of the subject. Furthermore the human motion reconstruction is possible only through a knowledge of the kinematic model of the subject and the position of the transmitters on the limbs. These information can be obtained using an automatic calibration procedure[80]. However this procedure must be repeated each time the transmitter position changes with respect to the limb.

Inertial MoCap

Inertial systems rely on accelerometers and gyroscopes. Sensors attached to the subject limbs measure the linear acceleration and angular velocity. Through an integration process is possible to recover position and orientation of each limb. Each inertial sensor is attached to a limb and communicates wireless with an elaboration unit, Figure 2.1.1.

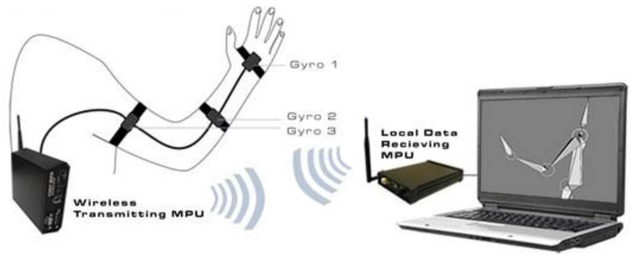


Figure 2.1.1: Inertial mocap. Wireless communication between the sensor attached to on the limbs and the central unit of elaboration

Commercial systems (like Xsense[1]and Animazoo[2]) find several applications, mainly thanks to the wide range of motion. The advantages of these systems are: wide capture areas, indoor as well as outdoor use, no lighting restriction, no occlusion related problem or missing markers and the miniaturize size of the sensor that does not interfere with the motion [87]. A disadvantage is the integration noise that leads to low frequency random walks that de facto limit the time span. This effect can be partially compensated by using magnetic field sensors that sense heading for attitude [55] or a GPS-aided Inertial Navigation System for position compensation. As in magnetic systems, human motion reconstruction is possible only through a knowledge of the kinematic model of the subject captured and the relative position of sensor-limbs must be computed through a calibration process.

Both magnetic and inertial systems allow to reconstruct only kinetic parameters of the motion of the subject but not its shape due to the reduced number of markers (transmitters/sensors) that can be practically applied.

Optoelectronic MoCap

Vision-based mocap systems have a high potential thanks to the great improvements achieved in video sensing and image processing. Vision based motion capture can be divided into two categories: marker and markerless. In both these groups the motion of the subject is reconstructed by analysing the sequence of images captured by cameras.

Marker based mocap systems recover human motion by triangulating a set of markers (passive or active) placed on the subject limbs. In this category it is possible to identify many successful commercial products: Vicon, Qualisys, WordViz.

Both categories suffer of the problem of self-occlusion: limbs of the analysed subject interfere with the measure partially hiding the body to the cameras.

In marker-based systems the self-occlusion problem is related to the fact that to reconstruct the 3D position of the marker this must be directly visible by at least two cameras. This is not assured especially for complex motions, in which the marker is hidden by the limbs of the user causing missing detection. This can be a problem if a

[1] http://www.xsens.com/

[2] http://www.animazoo.com/

significant portion of markers are missing for a long period of time, especially for systems with a low number of markers (< 100).

A further classification of marker based systems can be performed on the basis of markers typology: passive or active.

In passive marker based systems the most commonly used marker is constructed of small spheres, Figure 3.5.5, covered with a highly reflective material. These markers are associated with active cameras that emit infrared light, Figure 3.5.3. The marker detection process in images captured is straightforward since it consists in a simple threshold of the intensity of the image. This low computational complexity in image elaboration is traduced into high achievable capturing frequency. Main drawbacks of these systems are: marker ambiguity, swapping and positioning.

Marker ambiguity arises from the fact that there is no distinction between two markers because both appear identical in all images. Since it is not possible to actuate an error-free automatic marker identification, it is required the intervention of a trained human operator that assists the process of markers labelling. This operation has to be repeated if the marker is not reconstructed for a self-occlusion problem.

This ambiguity limits also the maximum number of markers that can be employed in a single subject without introducing too many markers swap. Markers swap appears during the reconstruction of the trajectory of a marker in an acquisition. If two markers are placed very close to each other it is possible that the two are confused and the label assigned swapped. This leads to a bad motion reconstruction. The number of markers that can be employed in this case is not sufficient to have a reconstruction of the shape of the body of the subject.

The illumination of the environment is another important element. For passive marker systems, bad light conditions are represented by the presence of reflective material in the scene that interfere with marker identification or can create artefacts that lead to wrong 3D reconstruction.

The setup of these systems can be complex and time consuming since markers must be placed one by one. In clinical applications the markers position must be chosen carefully to obtain good results. These particular positions, called *bony landmark,* correspond to points in which the distance between skin and underlying bone is minimum, [48, 7]. These landmarks must be identified by a trained operator and are not always directly accessible. In order to obtain valid kinematic measurements the marker positioning must follow also a detailed procedure operator in order to reduce the variability of results [42].

Active marker systems address specifically the problem of marker ambiguity employing spherical flashing light emitting diodes (LEDs) as markers. These markers communicate their identities encoded with pulses of light [94]. The light pulses encoded identities solve the marker labelling procedure.

A different solutions to light pulses was proposed by [115] which solved markers ambiguity relying on markers with different colours. [57] proposed instead a binary coding of the markers.

However the number of marker identifiers that can be generated with these systems is reduced and not suitable to reconstruct the shape of the subject. These systems found some applications in simple gait analysis, [59], and motion capture (Qualisys, PhaseSpace).

In recent years, markerless vision systems have proven to be a valid alternative to marker-based technologies [78, 24, 23] providing a direct and dense surface reconstruction of the subject.

Markerless algorithms are based on silhouettes: using a high number of cameras all around the workspace, images can be processed to extract 2D profile of the subject; these represent a set of visual cones. The intersection of these cones is a volumetric representation of the subject, [10]. By using kinematics and dynamics models it is possible to estimate kinematic and dynamic parameters of motion by fitting the models to the estimated surfaces. Accuracy and reliability are critical points. In particular, since the resulting shape can be very complex, the method can generate protrusions, and it is not able to handle cavities. For these reasons hybrid systems were introduced, [5]. These systems combine markerless (silhouette) and marker (natural texture of the subject) strategies to improve accuracy. The main drawback is that the system requires a customized room (uniform background). Despite their great simplification in the measurement procedure the reduced accuracy still remains an issue since natural landmarks are less accurate than their artificial counterparts. The complexity of human kinematics modelling also introduces uncertainties.

Another interesting approach is to turn the capture system inside-out, as proposed by [91] placing the cameras on the subject limbs. Despite the innovative approach and the improved capturing space, the cameras on the limbs can obstacle free movements. However camera miniaturization will partially solve this problem. One important interference factor is represented by the limbs motion that can induce camera rotations that can cause an accuracy deterioration. Also this system requires the definition of the kinematic model of the subject. Being the cameras worn by the subject this system is able to reconstruct only the subject motion but not his shape.

Since November 2010, a low cost vision system based on IR lighting patterns projection came on the market, the Kinect. Human posture is obtained by means of a classificator used directly on images [92] while human motion is estimated by means of the points reconstructed in the 3D domain.

A complete shape reconstruction of the subject is possible only using multiple monocular views from one sensor, [103]. [79] proposed a different solution based on more than one sensor but with the limit of an asynchronous acquisition due to pattern interference of different sensors, [98]. The problem with both approaches is the registration of different scansions.

Kinect is however an excellent trade-off between accuracy and cost. Despite the high number of points reconstructed and a fast segmentation of body limbs, this system provides low accuracy for clinical applications. Hybrid approaches have been used to improve the accuracy of this system, especially for hand movements [114].

Soft tissue artefacts

Because of the direct observation of soft tissues motion, i.e. subject skin, all the above motion capture systems suffer from soft tissue artefacts, or STAs, that can severely degrade accuracy.

Inaccuracies introduced by STAs are much larger than stereophotogrammetric ones and are both systematic and random in nature. The pattern of the artefact is task and subject dependent which prevents an experimental generalization and compensation. The artefacts are highly correlated in time and frequency content with the acquired motion, thus avoiding filtering techniques to be effective.

STAs are therefore considered the bounding error source in evaluating the skeletal motion by optoelectronic systems recording markers on the skin.

The STAs effects have been investigated by several authors, [17, 86, 63] and addressed in two different ways: avoiding the interference or minimizing it. It is possible to completely avoid STAs but only with invasive techniques like radio-opaque markers implanted on the bone [66] or bones pin [60, 97].

On the other way it is possible to minimize the effect of STA with numerical techniques [25] or the Point Cluster Technique (PCT), [3, 2]. The PCT considers a large number of markers uniformly distributed on the body segments, each with an assigned arbitrary mass that can be varied from sample to sample. The eigenvalues of the inertia matrix are calculated and its variations minimized assigning a proper mass to each point in an optimization procedure. In this way the noisy points are weighted less than the stable ones. The eigenvectors of the same matrix provide the reference system for each body.

2.2 Dynamic parameters estimation

The study of the human motion is subjected to the measurement of kinematic and dynamic variables. While the kinematic studies how the body moves, the dynamic is concerned about the study of the causes of the motion, i.e. forces and torques.

The general term given to the forces that cause the movement is *kinetics*. The kinetics take into consideration both internal (muscle activity, ligaments, or the friction in the muscles and joints) and external forces (muscle activity, ligaments, or the friction in the muscles and joints).

The knowledge of the patterns of the forces is an useful information in many fields: evaluate different chirurgical reconstruction techniques [62], comparison of rehabilitation exercises [68] or athletes performances [50].

The measure of forces is possible with two approaches: direct or indirect. The first rely on the direct measurement of the forces that the muscle generates. The second method allows an estimation of the forces on the basis of the inverse dynamic analysis of motion.

In the direct approach transducer are implanted within the the limbs to study and and measure the elongation of the tendon of the muscle in analysis, [49, 58]. A classification of transducers employed can be performed on the basis of the physic principles on which

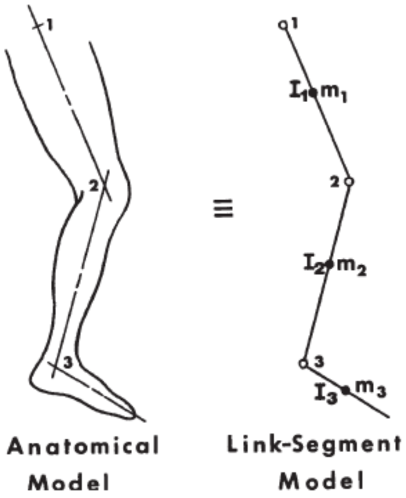


Figure 2.2.1: From anatomical to *link-segment* modelling.

they are based: the variation of the electrical resistance of liquid metal strain gauge (LMSG), [14], the deformation of strain gauges mounted on a rectangular buckle frames, [38], the displacement of a steel tube covered with Teflon coupled to a Hall generator, [41].

Main disadvantages with these method are that: only invasive devices are applicable, these devices must be compliant enough to measure the tissue tension without interfering with movement, their implant is limited in time because the development of fibrous healing reactions and a calibration is required in order to convert the output signal into a strain or a force, [85]. Since these techniques are highly invasive have found applications only in animal experiments.

The indirect approach of measurement of the forces instead is based on the inverse solution of the equation of the dynamics. With a full kinematic description of the body in analysis, accurate anthropometric parameters and external forces measurement, it is possible to calculate the joint reaction forces and muscle moments. The process by which the reaction forces and muscle moments are calculated is *called link-segment modelling*, or LSM, [107].

The LSM replace an anatomical model with a simpler model in which joints are replaced by hinge joints and segments are replaced by masses and moments of inertia located at each segment's centre of mass, Figure 2.2.1.

This model rely on some assumptions:

- each segment has a constant mass located at its Centre Of Mass, or COM
- the location of each segment's COM remains fixed respect the reference system of the segment during the movement
- the joints are considered to be hinge joints

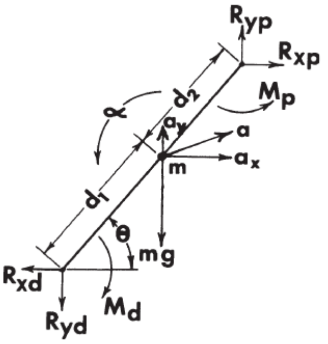


Figure 2.2.2: Free-body diagram of a segment, showing reaction and gravitational forces, moments of force, and linear and angular accelerations.

- the moment of inertia of each segment about its mass centre is constant during the movement
- the length of each segment remains constant during the movement

With this model each body segments acts independently under the influence of external and internal forces can be analysed separately respect to the others. Considering the movement of a segment on a plane, Figure 2.2.2, is possible to define a system of equations of *free-body diagram* of a single segment as follow:

$$\begin{cases} \sum F_x = ma_x \\ \sum F_y = ma_y \\ \sum M = I_0\alpha \end{cases} \tag{2.2.1}$$

A full description of kinematic parameters of the body can be obtained with the motion capture system, described in previous Section 2.1. While inertial motion capture systems give a direct measurement of the linear and rotational acceleration of the limbs, magnetic and optoelectronic give an indirect measure of the acceleration. An estimation of the acceleration of each limb can be calculated with a double integration of the results of displacement. In presence of noise this operation is quite tricky. In this work we solve this problem relying on an interest property of the Wavelets.

External reaction forces, ground reaction forces, can be measured with special sensors called force platforms. These platforms are constituted by a base under which are mounted 3 or more load cells. Elaborating these data is possible to calculate the centre of pressure and the component of the forces in three directions.

Anthropometric parameters estimation

Anthropometric parameters that are required to calculate the forces are: mass of the limb in analysis, location of Centre Of Mass or COM, and the moments of inertia of body segments.

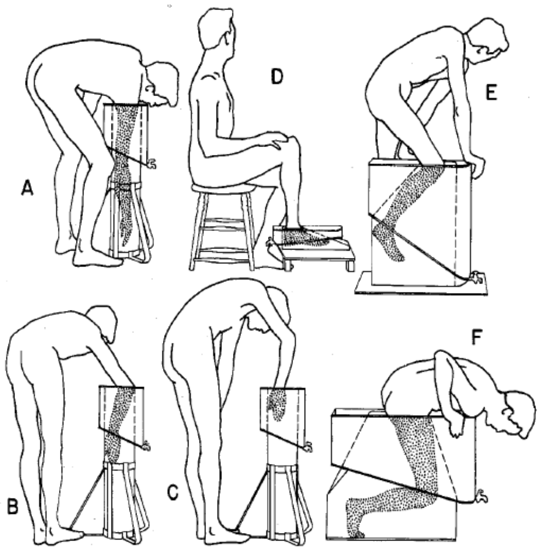


Figure 2.2.3: Setup of volume calculation by immersion realized by New York University. Determination of the arm volume.

These measures have been determined with various methods. In particular the mass definition of limbs was based on the idea that exist a relation between human body size and its mass distribution properties. The definition of these model was based on measure on cadavers and comparison with living,[29, 21, 74]. The idea is to obtain a direct measure of segment volumes in conjunction with density tables. [44] was the first to introduce two techniques estimate the volumes based on non invasive techniques: volume immersion and computational method.

Volume immersion technique consists in the determination of how much water is displaced by the submerged segment. A typical setup of volume calculation by immersion is shown in Figure 2.2.3.

The computational method consists in the approximation of the limb in analysis with a frustum of a cone. In order to obtain a better approximation of the volume the limbs can be subsequently subdivided into segments of equal height.

[32] proposed also the volume calculation of a segment based polar planimeter photogrammetric method, [105]. The reliability of this technique was proven by [47].

Main limitation of these studies is represented by the fact that many of the subjects used in the studies were biased towards a specific gender or population type, an thus the model of mass distribution may not be valid for different gender or population.

More recently measurements on living were performed by[112] using gamma mass scanning. The problem presented by this approach is that is based on using of electromagnetic radiations that can cause health issues to the patients and also requires high cost for the setup.

[22] presented an experimental technique to measure of the mass of a distal segment

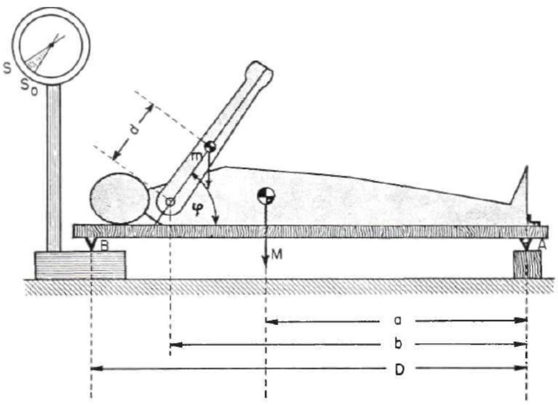


Figure 2.2.4: Reaction method for segment mass calculation.

called the reaction-board method. The subject is placed on a board supported by two knife edges one on a fixed base, the other on the platform of a weighing scale, Figure 2.2.4. The desired segment is lifted to a defined and repeatable positions and the mass of the limb can be calculated with the formula:

$$M = \frac{(S_M - S_o)D}{d(1 - cos\psi)} \tag{2.2.2}$$

where: S_M is the measured reaction force for a given position, S_o is the measured reaction force a 0 position, D is the distance between the two knife edges, d is the distance between from the segment COM to the proximal joint, ψ is the angle between the segment and the horizontal.

While this method can be performed in vivo the procedure is quite slow and allow to retrieve only the mass of the segment and not its moment of inertia.

The density of the body was estimated by [22] as a function of the Ponderal Index (PI) defined as $PI = h/w^{1/3}$, where h is the height of the subject and w is its weight. The density function proposed is:

$$d = 0.69 + 0.9PI \tag{2.2.3}$$

Since the density of the upper limb is quite different from lower some studies were performed on cadavers, [44, 29], to correlate whole-body density to segment density. These studies produced the graph reported in Figure 2.2.5.

The calculation of the COM for each limb was performed on cadaver by [44] simply determining the centre of balance of each segment. The method proposed by [107] is to calculate the centre of mass in vivo segmenting the profile cross-sectional area of the limb in analysis into n sections, Figure 2.2.6. The COM is calculated as the weighted mean of each segment with the formula:

$$x = \frac{1}{M}\sum_{i=1}^{n} m_i x_i$$

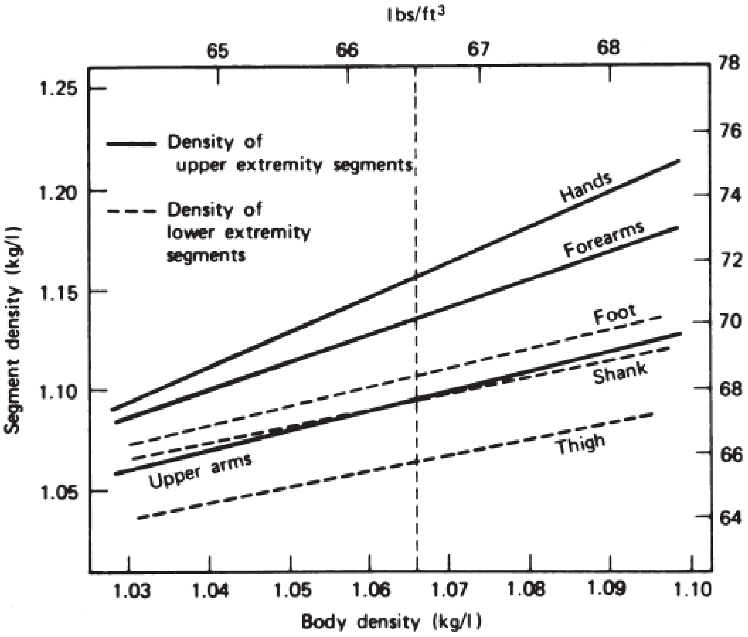


Figure 2.2.5: Density of upper and lower limb segments as a function of average body density.

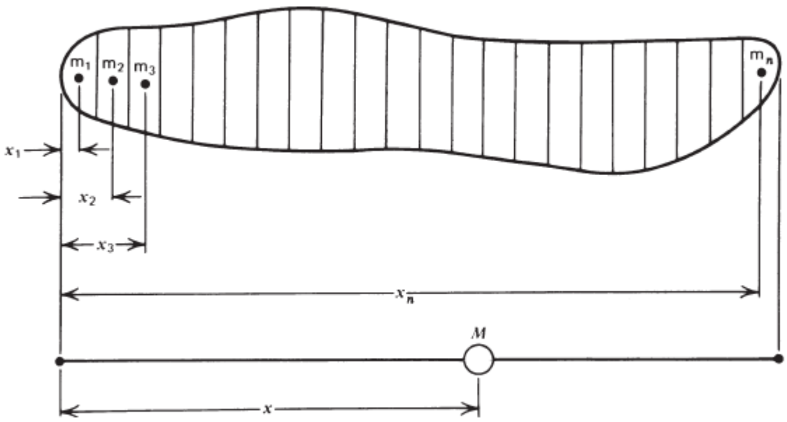


Figure 2.2.6: Calculation of the COM by sectioning the limb in n segments.

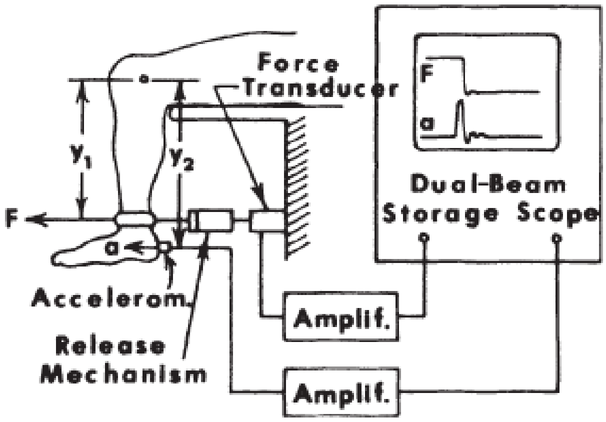


Figure 2.2.7: Quick release technique setup.

[22] proposed also a method to compute the mass moment of inertia called: *quick release.* This method rely on the well known Newton's second law: $M = I\alpha$. If it is possible to apply a moment M to the limb in analysis and measure the angular acceleration α, I can be directly calculated. The setup of the quick release technique is shown in Figure 2.2.7.

A different approach was used by [111] which adopted a geometric approach. The body of the subject is split into 40 geometric solids which are specified by 95 anthropometric measurement and density is considered uniform in each segment. The main disadvantage with this technique is that is time consuming since can take 40 minutes for each subject. Recently this approach was refined by [40] which used a more complex geometric model with take into account also the different density of bones and lungs.

With the digitize techniques possible with 3D laser scanners the accuracy of volume estimation increased. Their use in body inertial parameter estimation is however inhibited by the cost and the time to obtain a complete scan, [103].

Also structured light techniques are stimulating new method to estimated the volume of the body segments, [84, 20]. These system are faster and cheaper than 3D laser scanners but with lower accuracies.

While these system give better results in terms of accuracy of volume estimated, relative motion between subject-sensor, due breathing or involuntary movements, remains an issue.

Superquadrics and anthropometric parameters estimation

In the analysis of the literature of different methods to recover the anthropometric inertia parameters of human limbs from 3D data, superquadrics suggested a compact solution to the problem.

After a brief description of the formulation of superquadrics and some of their application in the field of object modelling and recognition we report the formulas to analytically calculate the moments of inertia of the superquadrics. In Subsection 3.4.3 we propose a method to estimate inertial parameters of the human limbs.

Superquadrics, firstly introduced by [6], is a natural an extension of quadratic surfaces. A superquadrics is defined as the solution of the implicit equation:

$$f(x,y,z) = \left(\left(\frac{x}{a_1} \right)^{\frac{2}{\varepsilon_1}} + \left(\frac{y}{a_2} \right)^{\frac{2}{\varepsilon_2}} \right)^{\frac{\varepsilon_2}{\varepsilon_1}} + \left(\frac{z}{a_3} \right)^{\frac{2}{\varepsilon_2}} \tag{2.2.4}$$

Scaling factors along the coordinate axes x, y and z are respectively controlled by parameters a_1, a_2 and a_3.

Parameters ε_1 and ε_2 instead determine respectively the shape of the superquadric in a plane perpendicular and parallel to the (x, y) plane, Figure 2.2.8.

Looking at the implicit formulation of the superquadrics, Equation 2.2.4, is possible to recognize the ellipsoid equation enriched with two the two shape parameters ε_1 and ε_2.

Aside the implicit formulation there is a parametric formulation, in which superquadrics are represented in spherical coordinates η and μ as follows:

$$S(\mu,\eta) = \begin{bmatrix} a_1 \cos^{\varepsilon_1}(\eta) \cos^{\varepsilon_2}(\mu) \\ a_2 \cos^{\varepsilon_1}(\eta) \sin^{\varepsilon_2}(\mu) \\ a_3 \sin^{\varepsilon_1}(\eta) \end{bmatrix}, \quad \begin{array}{c} -\frac{\pi}{2} \leq \eta \leq \frac{\pi}{2} \\ -\pi \leq \mu \leq \pi \end{array} \tag{2.2.5}$$

While the parametric formulation, Equation 2.2.5, is particularly suitable for sampling and rendering, the implicit formulation, Equation 2.2.4, gives a direct information on the position of a 3D points respect the superquadric surface. In particular considering the superquadric defined by the implicit equation $f(x, y, z)$ and the point P with the coordinates (x, y, z) we have that:

- if $f(x, y, z) = 1 \rightarrow$ the point P lies on the surface of the superquadric
- if $f(x, y, z) < 1 \rightarrow$ the point P is inside the superquadric
- if $f(x, y, z) > 1 \rightarrow$ the point P is outside the superquadric

This formulation is also called $inside - outside$ function, [93].

Superquadrics are convenient since with a compact model defined by only 5 parameters $(a_1, a_2, a_3, \varepsilon_1, \varepsilon_2)$ allows to handle a variety of shapes, Figure 2.2.8.

These properties were exploited principally to create compact model of 3D object and to recover object position in space.

[61, 18] used superquadrics to build a model of a generic 3D objects from range data. The modelling procedure of the object adopted are respectively based on the region growing principle and the split and merge approach.

[8, 56] performed 3D object detection with a geometric model description using approximated Superquadrics and RANSAC. While Superquadric fitting procedure is fast and reliable the data segmentation phase is quite time consuming.

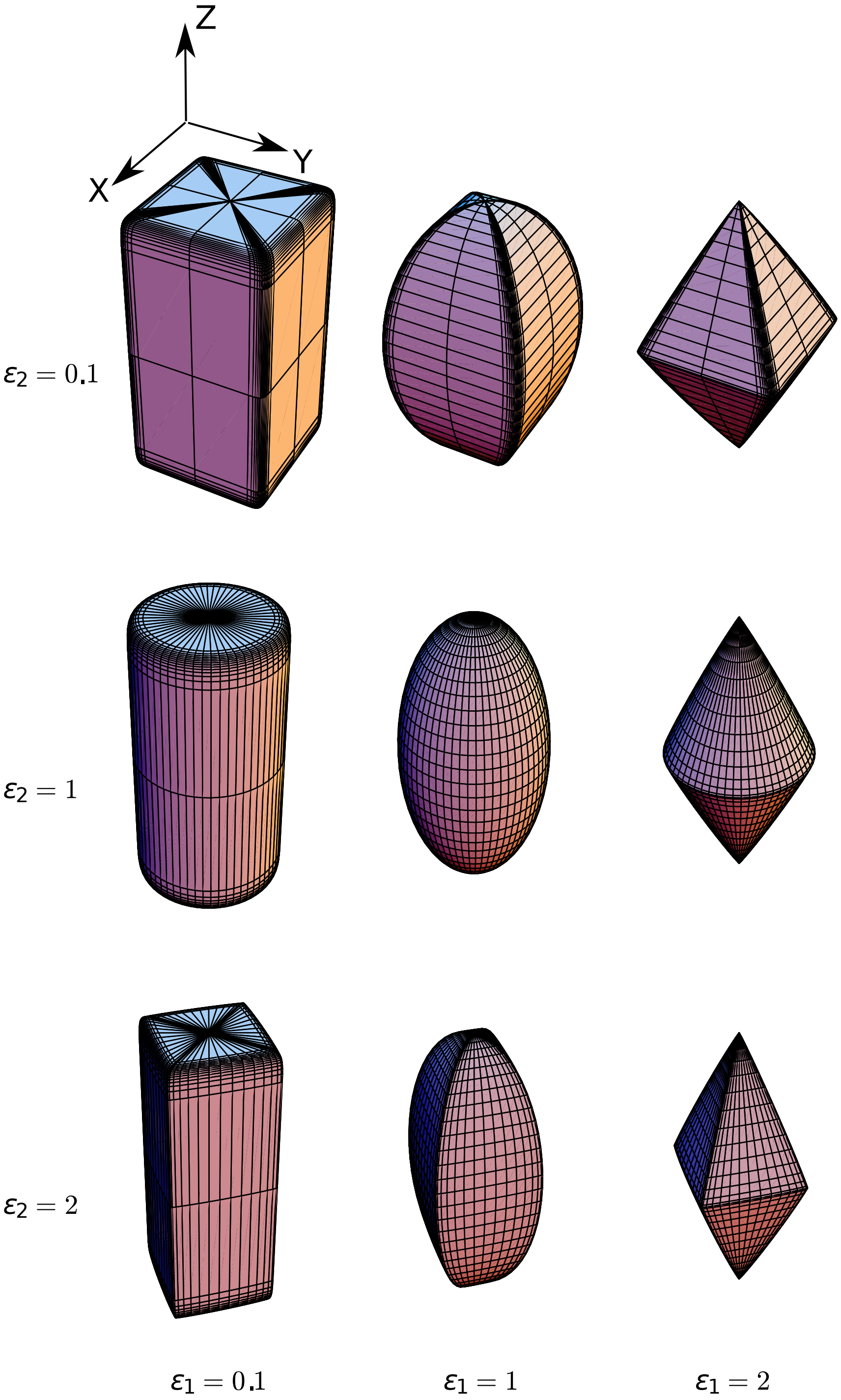


Figure 2.2.8: Different superquadrics that can be obtained with different values of exponents ε_1 and ε_2. Parameters a_1 , a_2 and a_3 are kept constant.

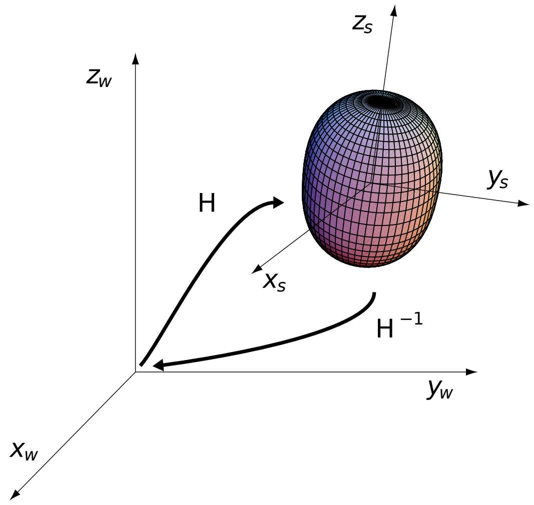


Figure 2.2.9: Superquadric defined with a general position and orientation in space.

Superquadrics were also used to reconstruct 3D human models from body scan data, [108]. A whole scansion of a human body, obtained with laser scanners, is processed to obtain a coarse segmentation of the points cloud into limbs with a topology-based decomposition. This segmentation is then refined with a consistency measure algorithm then each segment is fitted with a superquadric. While this method prove the utility of superquadrics in 3D human shape retrieval problems, segmentation of 3D data with complex postures of the subject remains an issue.

All these works rely on a cost function based on the implicit formula of the superquadrics. [93] firstly proposed a cost function for superquadric parameters calculation. The cost function is build on the implicit equation of superquadric as follows:

$$F^{\varepsilon_1}(x, y, z) - 1 \tag{2.2.6}$$

where F is the implicit equation of the superquadric. The exponent ε_1 ensures that, despite of the current value ε_1 of the superquadrics, points at same distance from the superquadrics have the same value of $F(x)$.

[93] proposed also a cost function to further reduce the error metric dependence on the ratio of the size parameters a_1, a_2 and a_3.

The equation is the follow:

$$\sqrt{a_1 \cdot a_2 \cdot a_3} \cdot F_2^{\varepsilon_1}(x, y, z) - 1 \tag{2.2.7}$$

The cost functions here reported refers to a superquadric defined in a *local*, or superquadric, centred coordinate system, Figure 2.2.9.

A superquadric in a general position requires 6 additional parameters to express the rotation and translation of the superquadric relative to the centre of the world coordinate system. In Figure 2.2.9 the position and orientation of the superquadrics is defined by an homogeneous transformation H to transform the 3D points expressed in the superquadric coordinate system $[x_s, y_s, z_s, 1]$ into the world coordinates $[x_w, y_w, z_w, 1]$.

Another interesting feature of the superquadrics is that the volume and moment of inertia of the superquadrics can be obtained analytically with the formula proposed by [52].

The volume of a generic superquadric can be computed with the formula:

$$V = 2a_1a_2a_3\varepsilon_1\varepsilon_2 B\left(\frac{\varepsilon_1}{2}+1, \varepsilon_1\right) B\left(\frac{\varepsilon_2}{2}, \frac{\varepsilon_2}{2}\right) \tag{2.2.8}$$

Where $B\left(x, y\right)$ is an integral defined as:

$$B\left(x, y\right) = 2\int_0^{\frac{\pi}{2}} \sin^{2x-1}\phi \cdot \cos^{2y-1}\phi d\phi \tag{2.2.9}$$

The moments of inertia of a superquadrics about the x, y and z axes can be calculated as follows:

$$I_{xx} = \frac{1}{2}a_1a_2a_3\varepsilon_1\varepsilon_2\left(a_2^2 B\left(\frac{3}{2}\varepsilon_2, \frac{1}{2}\varepsilon_2\right) B\left(\frac{1}{2}\varepsilon_1, 2\varepsilon_1+1\right) + 4a_3^2 B\left(\frac{1}{2}\varepsilon_2, \frac{1}{2}\varepsilon_2+1\right) B\left(\frac{3}{2}\varepsilon_1, \varepsilon_1+1\right)\right) \tag{2.2.10}$$

$$I_{yy} = \frac{1}{2}a_1a_2a_3\varepsilon_1\varepsilon_2\left(a_1^2 B\left(\frac{3}{2}\varepsilon_2, \frac{1}{2}\varepsilon_2\right) B\left(\frac{1}{2}\varepsilon_1, 2\varepsilon_1+1\right) + 4a_3^2 B\left(\frac{1}{2}\varepsilon_2, \frac{1}{2}\varepsilon_2+1\right) B\left(\frac{3}{2}\varepsilon_1, \varepsilon_1+1\right)\right) \tag{2.2.11}$$

$$I_{zz} = \frac{1}{2}a_1a_2a_3\varepsilon_1\varepsilon_2\left(a_1^2+a_2^2\right) B\left(\frac{3}{2}\varepsilon_2, \frac{1}{2}\varepsilon_2\right) B\left(\frac{1}{2}\varepsilon_1, 2\varepsilon_1+1\right) \tag{2.2.12}$$

Assuming a constant density of the superquadric with these formulas is possible to calculate relative the mass and the inertia of the superquadric.

Wavelets properties and derivative calculation

Wavelet transforms are a high performance signal processing technique which have been employed with success in many field of applications: image compression, de-noising, analysis of analytical signal. One of the last field of application is derivative calculation. The simplest method for derivative approximation is with finite differences, but with real signal we have a rapid degradation of signal-to-noise ratio (SNR) .

[89] proposed a general approach to derivative calculation using wavelet transform. Their paper shows how that wavelet transforms can be regarded as a smoothing and a differentiation process, and the order of differentiation is determined by the property of the wavelet function. Here we analyse this property.

A wavelet is a function ψ of zero average:

$$\int_{-\infty}^{+\infty} \psi(t)dt = 0$$

This function is dilated and translated respectively with the parameters s and u:

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right)$$

The wavelet transform of a function f with scale s and translation u can be computed by the correlation of f with the wavelet:

$$Wf(s,u) = \int_{-\infty}^{+\infty} f(t)\frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right)dt$$

One of crucial property of wavelets, for differentiation purposes, is the vanishing moments. The wavelet $\psi(t)$ is said to have n vanishing moments when:

$$\int_{-\infty}^{+\infty} t^k\psi(t)dt = 0 \qquad 0 \leq k < n$$

A wavelet with n vanishing moments has the property to be orthogonal to polynomial of degree $n-1$, i.e., it can be used to suppress polynomial of degree $n-1$ through convolution.

As stated by [71] a wavelet function ψ can be written as the n-th order derivative of a function $\theta(t)$ as follows:

$$\psi(t) = (-1)^n \frac{d^n\theta(t)}{dt^n} \tag{2.2.13}$$

where $\theta(t)$ is a function with fast decay and whose integral is a non-zero constant.

The wavelet transform is defined as the projection of a function onto the wavelet. The wavelet transformation of a generic f function can be written as:

$$Wf(u,s) = f \star \bar{\psi}_s(u) \qquad with \qquad \bar{\psi}_s(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{-t}{s}\right) \tag{2.2.14}$$

Deriving from 2.2.13 $\bar{\psi}_s(t) = s^n\frac{d^n\bar{\theta}(t)}{dt^n}$. Commuting the convolution and the differentiation operator we obtain:

$$Wf(u,s) = s^n f \star \frac{d^n\bar{\theta}(u)}{dt^n} = s^n\frac{d^n(f\star\bar{\theta})}{dt^n}(u) \tag{2.2.15}$$

2.3 Multicamera calibration

The calibration of the multicamera system is an essential step to obtain precise 3D reconstruction and reliable results of motion estimation. Multicamera calibration is the procedure to find the parameters of the cameras model that describe the relationship: between 3D world and 2D images observed by a camera (calibration of intrinsic parameters) and cameras reference systems (calibration of extrinsic parameters). Typical calibration procedure consists of two steps: control points localization and the camera parameters calculation.

[113] proposed a user-friendly camera calibration method based on a known 2D planar object.

The object taken as reference (or calibration tool) is a chessboard printed on a plane with well known dimensions in which control points are corners of squares. Synchronous images are captured by the cameras with the chessboard in different position and orientations, Figure 2.3.1. Corner positions are then extracted and the cameras parameters are computed. On the basis of this procedure, in these years, have been developed several single camera [12] and multicamera calibration software [102]. The chessboard calibration pattern have been used because of the sub-pixel precision that can be achieved in the points detection, [65].

Main limitation with this calibration pattern is that all the control points are equal to the others (corner of black and white squares). The position and orientation of the calibration pattern can be recovered from an image only if all the control points are visible (full visibility of the pattern). A partial visibility of the grid in a image results in a useless image for calibration purpose, Figure 2.3.1.

The full visibility requirement affects both calibration of distortion and extrinsic parameters.

Indeed a good estimation of distortion parameters can be obtained with a high number of control points at image borders, since there image distortions are stronger, Section 2.3.1. Recover control points at image borders (and ensure full visibility of the pattern) is a delicate operation that can be time expensive if must be repeated for a medium multicamera system (~10 cameras).

Extrinsic parameters estimation require that the calibration pattern to be fully seen by two or more cameras. If the full visibility of the chessboard in two cameras is not assured it is not possible to recover the relative orientation between cameras. Figure 2.3.1, and thus calibrate the system.

The full-visibility problem was addressed in many papers that have developed calibration techniques based on different calibration pattern.

Common idea is to use control points that have distinctive properties. [34] uses markers with bi-tonal planar patterns containing a unique ID number encoded with robust digital techniques of checksums, Figure 2.3.2.

[83] uses an arbitrary textured planar pattern as calibration tool. The pattern is automatically detected in the frames where it is visible and used to simultaneously recover geometric and photometric camera calibration parameters. Despite flexibility

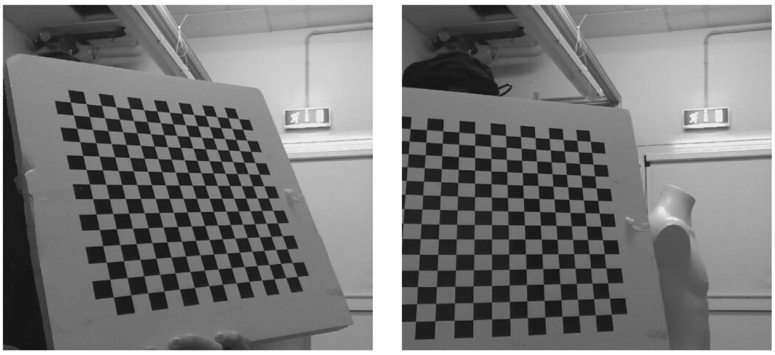


Figure 2.3.1: Sample images for camera calibration with [12] calibration toolbox. Visibility of the pattern must be ensured to allow the position reconstruction. Left: Valid images, the chessboard is entire visible. Right: Invalid images, the chessboard is partially out of image plane.

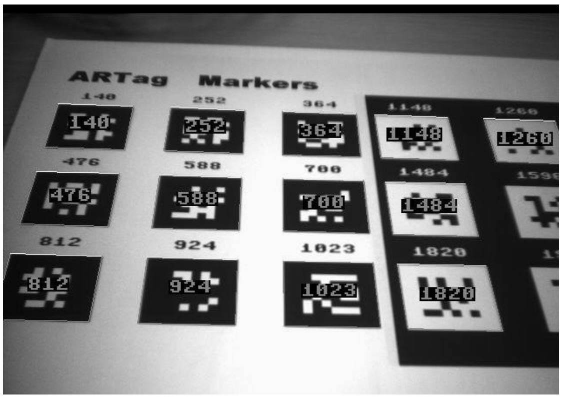


Figure 2.3.2: Bi-tonal markers used for multicamera calibration by [34]. Overlapped with each marker the distinctive numeric ID.

and illumination invariance in this algorithm the image distortions are not computed.

[82] developed a calibration algorithm based on the tracking of a random dotted pattern able to deal with different lighting conditions and partial occlusion of the grid. The main drawback is that the algorithm is only invariant to scale and rotation of the pattern in image plane but not to out of plane.

Another approach proposed by, [73, 67, 99, 96], is the self-calibrating approach. These systems does not assume the knowledge on known shape objects but uses only natural landmarks. Main disadvantage of these system is that the reconstruction is possible only up to a scale factor.

2.3.1 Camera model

Shape reconstruction and motion estimation have to deal with the mathematical and geometrical description of the image formation and the formulation of the multiple view geometry. In this section will be introduced and described the camera model employed.

From a mathematical point of view, a camera can be seen as a mapping between the 3D world (object space) and a 2D image. This mapping can be expressed with different models, mathematically formalized with matrices. In [45] are presented and analysed different camera models.

In this section we will describe the simplest camera model: the pinhole model and we will generalize it to the finite projective camera.

The pinhole camera model is defined with a centre of projection $\boldsymbol{C}$ to be the origin of a Euclidean coordinate system, and a plane $Z = f$, which is called the *image plane* or *focal plane*, Figure 2.3.3.

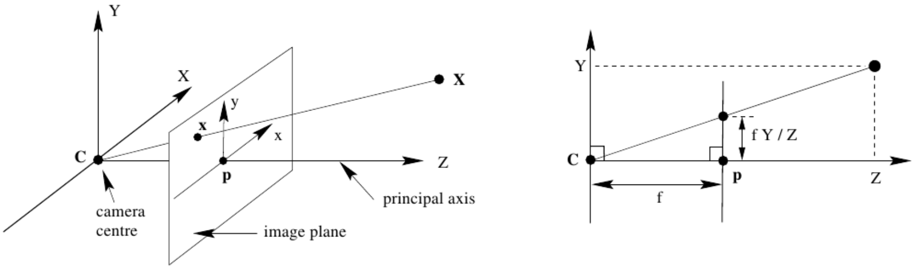


Figure 2.3.3: Pinhole camera geometry. $\boldsymbol{C}$ is the centre of projection, $\boldsymbol{p}$ is the principal point. Two reference system are presented: camera and image reference system.

The camera reference system has the origin at centre of projection $\boldsymbol{C}$ and the Z axis perpendicular at image plane. This axis is called *principal axis.* The point of intersection of principal axis and focal plane is called *principal point* $\boldsymbol{p}$.

With the pinhole camera model a point in space with coordinates $\mathbf{X} = (X, Y, Z)$ is mapped to the point on the image plane $x = (fX/Z, fY/Z, f)$, where a line joining the point $\boldsymbol{X}$ to the centre of projection $\boldsymbol{C}$ meets the image plane.

Using the homogeneous vector representation this mapping can be written in terms of matrix multiplication:

$$\begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{2.3.1}$$

or:

$$x = PX \tag{2.3.2}$$

Where x is the image point, P is called *projection matrix* and X is the point in 3D space.

The Expression 2.3.1 assumes that the origin of coordinates in the image plane is at the principal point and the focal length is considered equal in x an y direction. In practice, this is not true: the pixel may be rectangular and thus the length in x and y direction are different (f_x, f_y). Also the two reference system may be shifted as shown in Figure 2.3.4. .

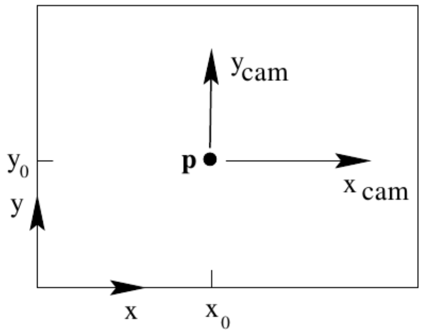


Figure 2.3.4: Image (x,y) and camera (x_{cam}, y_{cam}) reference system

The mapping becomes:

$$\begin{pmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \end{pmatrix} = \begin{bmatrix} f_x & & x_0 & 0 \\ & f_y & y_0 & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{2.3.3}$$

or:

$$x = PX_{cam} = K[\ I \quad |0\ ]X_{cam} \tag{2.3.4}$$

The matrix K is called the *camera calibration matrix.* In 2.3.4 we have written $(X, Y, Z, 1)$ as X_{cam} to emphasize that the camera is assumed to be located at the origin of a Euclidean coordinate system with the principal axis coincident with Z axis direction, and the point X_{cam} is expressed in this coordinate system. Such a coordinate system may be called the *camera coordinate reference system.*

Extrinsic Parameters

In general a point will be defined respect another coordinate system called *world coordinate reference system. Camera* and *world* reference system are related with a rotation R and a translation t.

Figure 2.3.5 shows the relation between a point expressed in a world reference system and its coordinate on the image plane. Dotted arrows represent the two transformation

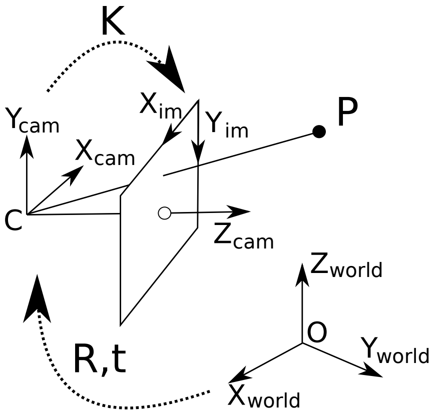


Figure 2.3.5: Camera model used. Relations between world , camera and image reference system.

(and) between the three reference systems (world, camera and image).

This relation can be formalized as follows:

$$X_{cam} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} X_{world} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{2.3.5}$$

Putting together this equation with 2.3.4 we obtain:

$$x = PX_{world} = K \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} X_{world} \tag{2.3.6}$$

This general mapping for a pinhole camera the projection matrix $P = K[R|t]$, has 10 degree of freedom: 4 for K (f_x, f_y, x_0, y_0) and 6 for R and t. The parameters contained in K are called the *internal camera parameters*, or the *internal orientation* of the camera. The parameters of R and t which relate the camera orientation and position to a world coordinate system are called the *external parameters* or the *exterior orientation.*

The calibration procedure aims to identify the parameters that characterize these transformation for each camera.

Lens distortion

The linear model of mapping in Equation 2.3.6 shows deviation in real optical systems. The most important deviation is caused by radial and tangential distortion. The radial effect is predominant.

A point is imaged at a distance from the principal point that is larger (pin-cushion) or smaller (barrel) than the predicted one by the perspective projection equations 2.3.6.The distance is calculated from the central point. The effect of radial distortions are shown in Figure 2.3.6.

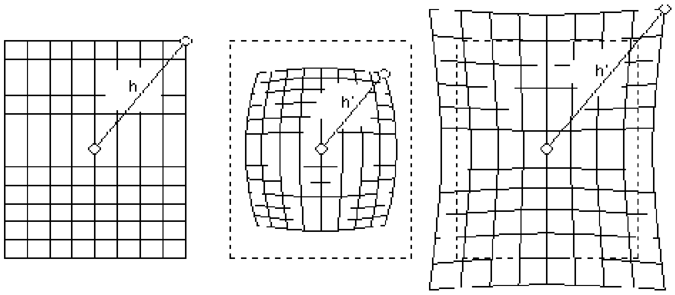


Figure 2.3.6: Image distortion caused by real lens. Left: image not distorted. Centre: pin distortion. Right: barrel distortion. Distortion increases with the distance from the centre of image.

The distortion effects can be compensated using the Brown's distortion model, [13]. In Equation 2.3.7 the distorted coordinate in image plane (x_d, y_d) are undistorted (x_u, y_u) to obtain the point projection in an ideal pin-hole camera.

$$\begin{cases} x_u = (x_d - x_{cam})(1 + \kappa_1 r^2 + \kappa_2 r^4 + ...) + (2\epsilon_1 (x_d - x_{cam})(y_d - y_{cam}) + \epsilon_2 (r^2 + 2(x_d - x_{cam})^2)) \\ y_u = (y_d - y_{cam})(1 + \kappa_1 r^2 + \kappa_2 r^4 + ...) + (\epsilon_1 (r^2 + 2(y_d - y_{cam})^2)) + 2\epsilon_2 (x_d - x_{cam})(y_d - y_{cam})) \end{cases} \tag{2.3.7}$$

3 Algorithms proposed

In this Chapter is described in detail the algorithm proposed to perform shape reconstruction and motion analysis in terms of kinematic and dynamic parameters estimation.

In each Section is described a step of the process of motion capture and motion analysis framework developed in this research work. In Figure 3.0.1 a flow chart shows the relation of various steps.

In Section 3.1 are described the algorithms realized to generate the color coded pattern plotted on the suit as well as the strategies to extract the barycentre, shape and colour of the markers.

Section 3.2 analyses the algorithm proposed to reconstruct the 3D shape starting from the marker extracted from the images captured by cameras.

The multicamera calibration algorithm realized to calibrate the GaMoCap is described in Section 3.3.

Section 3.4 is articulated in three subsections: motion segmentation, kinematic and dynamic parameters estimation.

A detailed analysis of the uncertainty with Monte Carlo Method of the angle at joints estimated with our system is reported in Section 4.2. In this Section is presented also the comparison of the angle at joint estimated with our system and the commercial system Vicon.

3.1 Feature extraction

The feature extraction is the procedure to detect the circular marker plotted on the suit in the image and create a descriptor for each marker. These steps are essential to associate the marker through different cameras and calculate their 3D coordinate. Figure 3.1.1 shows the block on “Feature extraction” with an higher level of detail.

In this Section are described the algorithms proposed to identify the marker position in image, Subsection 3.1.1, and the assignation to each marker of unique descriptor called ID, Subsection 3.1.3.

The feature description is a color based procedure. In Subsection 3.1.2 is described the algorithm implemented to recognize the different colours of the markers.

3.1.1 Feature detection

The feature detection is the procedure of calculation of 2D coordinate on image plane of the coloured markers, identification of their neighbour and the selection of the marker relative to the garment.

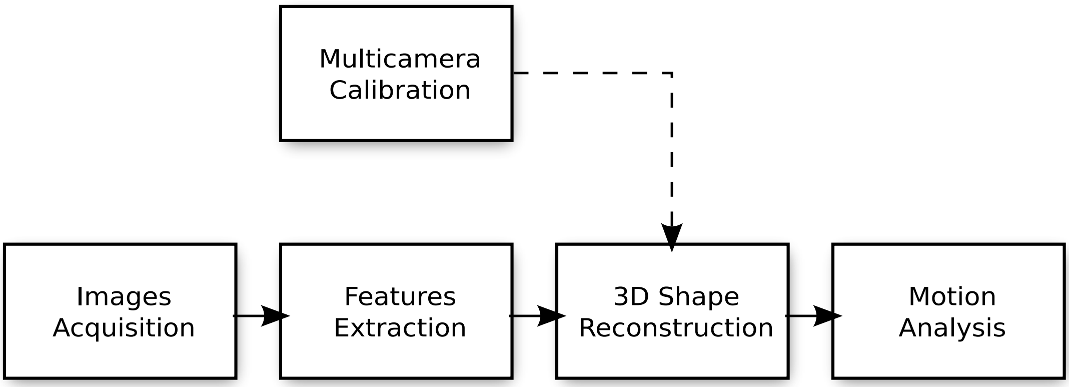


Figure 3.0.1: Flow chart of the motion capture and analysis framework developed.

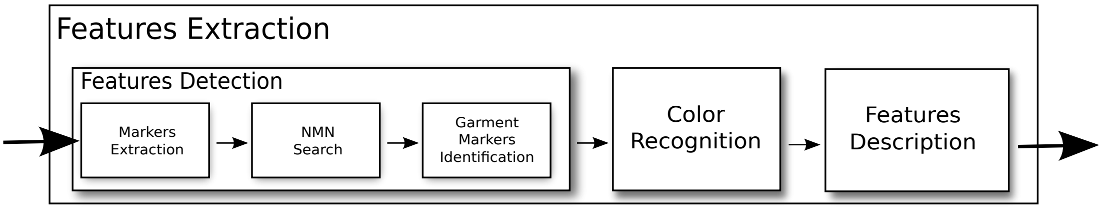


Figure 3.1.1: Flow chart of Features Extraction step. It is divided in Features detection, Color recognition and Features description.

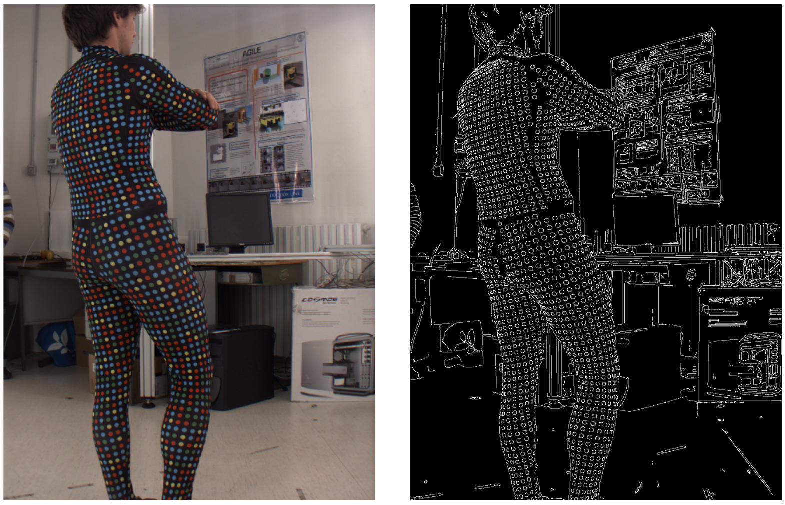


Figure 3.1.2: Canny edge detection result(right) on a image(left) of the GaMoCap suit.

In this Subsection two different algorithms used for marker detection are described: Canny and MSER. Each one has pro and cons depending on the specific application.

Furthermore it is analysed in detail the technique developed to identify the neighbours of the marker even with high distortion.

Once identified the markers and their neighbours ,it is applied a filter operation that keeps only the marker relative to the garment on the basis of their regular disposition.

The result of feature detection algorithm on each image is a matrix that contains only the coordinate of the marker relative to the garment and their color.

Canny

The first algorithm used to detect marker centroid is Canny operator [15]. The RGB image, acquired by the camera, is transformed into a grayscale. Canny edge extraction is performed on the grayscale image. With an opportune tuning of the two thresholds of the Canny algorithm (upper and lower) it is possible to detect most of the circular markers on the image.

The result of a Canny edge extraction on grayscale image is a binary image, in which each pixel is equal to 1 if it is relative to an edge and 0 if it is not, Figure 3.1.2.

From this binary images we can consider each set of connected pixel as a distinct object, called *contour*. These *contours* are identified with an implementation of the algorithm presented by [95].

Figure 3.1.2 shows that among the edges relative to the markers there are many other edges relative to background objects. Since the edges relative to markers are ellipses (projection of circles) it is possible to identify the markers setting a tolerance on the ratio of perimeter and area of each contour, Equation 3.1.1.

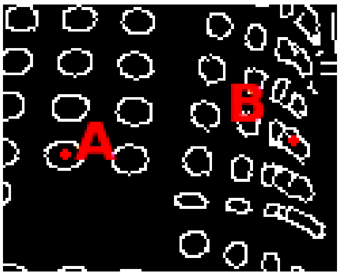


Figure 3.1.3: Edge coalescence effect due to a low resolution

$$\frac{Perimeter}{Area} = \frac{\sqrt{2 \cdot (a^2 + b^2)}}{ab} \pm tolerance \tag{3.1.1}$$

where$a = MajorAxisLength$ and $b = MinorAxisLength$. If this condition is verified, the object is considered as a marker, otherwise it is deleted.

The pro of this feature detection algorithm is the speed of elaboration: 85 ms per image. The main disadvantage is represented by the minimum marker resolution required to have a good detectability.

If the number of pixels is not sufficient (low resolution) the edges coalesce and the markers are wrongly identified, Figure 3.1.3. With edge coalescence fails also color identification of the marker, since color of different markers are mixed together.

This factor limit the subject-camera distance and the detectability of marker with an high inclination respect image plane.

MSER

As an alternative to edge detector to identity markers we used a blob detector algorithm called MSER (Maximally Stable Extremal Regions).

The algorithm was initially proposed by [72] for a robust wide-baseline stereo association.

The algorithm allows to identify regions with stable characteristics. The regions are defined solely by an extremal property of the intensity function in the region and on its outer boundary.

The concept of extraction of this region can be explained as follows: Imagine all the 256 possible threshold images I_t of a 8-bit gray-level image I. Each threshold image I_t is the result of a threshold of the grey image I with the threshold value equal to $t = 1...256$.

If these images I_t are shown sequentially, we would see first a white image ($t = 1$) because all the pixels are above thresholds. Subsequently, with the increase of the threshold value t, black spots corresponding to local intensity minima will appear and grow. The last threshold image I_t, relative to threshold $t = 256$, will be all black. The set of all connected components of all frames of the sequence of images I_t is the set of all maximal regions. In many images, local binarization is stable over a large range of

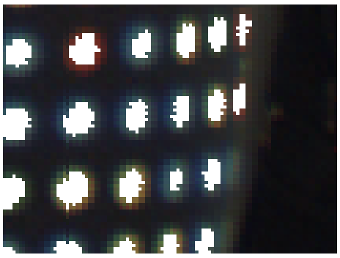


Figure 3.1.4: MSER marker detection. White pixels are relative to marker identified. Markers with high inclination are correctly segmented

thresholds in certain regions. MSER seeks a range of thresholds that leave the connected components effectively unchanged.

The result of MSER applied on a grayscale image of GaMoCap images is shown in Figure 3.1.4.

Once the algorithm has identified the Maximally Stable Extremal Regions in the image, it is possible to calculate the property of area and perimeter of each region. Also in this case it is necessary to distinguish between marker and non-marker regions. The filter operation based on geometry properties is the same used in Canny, Equation 3.1.1.

In Figure 3.1.4 is possible to notice that even marker projection of image plane with few pixels are correctly identified and well separated with MSER.

The main drawback for this algorithm is that is slower than Canny (150 ms per image).

Nearest Neighbour with Mahalanobis

For feature description purposes, Subsection 3.1.3, it is necessary to identify the 8 nearest (euclidean) neighbour markers in image space for each marker on the garment, Figure 3.1.5.

This procedure is a typical classification problem and it was firstly called by [37] as Nearest Neighbour search.

The KNN (k-nearest neighbour) algorithm implemented in [51] is an out of the box solution to detect the 8 (3x3) nearest neighbours of a marker in euclidean space, Figure 3.1.5. While the nearest neighbours of the marker that lay on a surface almost parallel to plane image correspond to the nearest in euclidean space, this is not true for markers with high inclination surface-camera.

When the surface with the marker and its neighbours has an high inclination respect to the image plane, the second crown of neighbours (5x5) gets close to the central marker.

The graph in Figure 3.1.6 shows respectively the case with low (Left) and high (Right) surface-image inclination. In the case of low inclination of surface-image the circular area, defined by the distance between the central marker and the most distant 3x3 neighbours, includes only the 8 neighbours. In the high inclination case the area includes a larger number of neighbours. In this case, in the first 8 nearest neighbours in euclidean space, there are markers which belong to the second crown of neighbours. This leads to a wrong

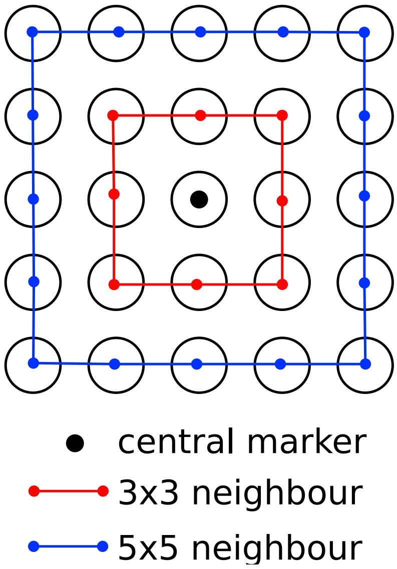


Figure 3.1.5: 3x3 and 5x5 nearest neighbour markers.

identification of 3x3 neighbours.

Figure 3.1.7 shows an example of marker with low and high surface-image inclination.

This problem was solved developing a Nearest Neighbours search based on the use of Mahalanobis distance, called Nearest Mahalanobis Neighbours or NMN.

This technique takes advantage of the fact that the circular markers which lie on a plane with an high inclination respect to the image appear as ellipses with the major axis aligned with the axis of the rotation of the plane, Figure 3.1.6. The major axis of a marker gives thus a hint of the marker surface inclination respect to the plane of the image.

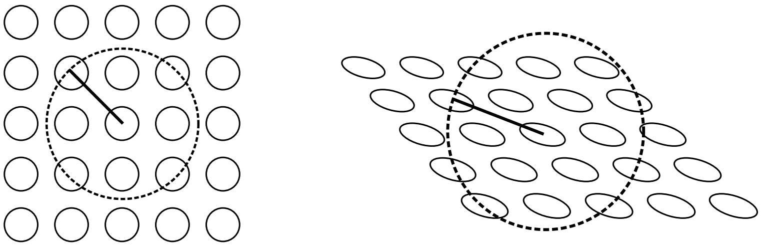


Figure 3.1.6: Left: low inclination surface-image. The circular area encloses only the 8 (3x3) neighbours. Right side: high inclination surface-image, the circular area encloses only the 8 (3x3) neighbours.

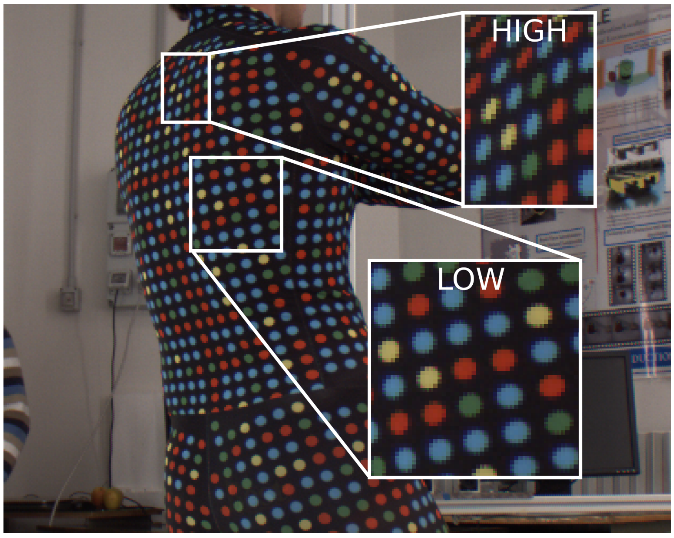


Figure 3.1.7: Example of low and high surface-image inclination.

The NMN algorithm developed is a two step search. The first step is a classical KNN search in euclidean space to find the first 25 nearest neighbours of a central marker. The second step consists in a calculation of the Mahalanobis distance of the central points to the 25 neighbours. The points are then sorted in an ascending order respect to the value of their Mahalanobis distance. The first 8 points are considered to be the 3x3 nearest neighbours.

The number of nearest neighbours was chosen empirically equal to 25. We have indeed observes that even under high inclination surface-image the 8 (3x3) nearest neighbours remain in the 25 euclidean nearest neighbours.

The Mahalanobis distance, [69], is a statistical measure to identify similarity of an unknown sample set to a known one.

By defining two points $P = [p_1, p_2, ..., p_n]$ and $Q = [q_1, q_2, ..., q_n]$, the Mahalanobis distance is defined as:

$$d(P, Q) = \sqrt{(P - Q)^T C^{-1} (P - Q)} \tag{3.1.2}$$

where C is the sum of the covariance matrix C_P and C_Q respectively of the two points P and Q.

In our case the Mahalanobis distance was used not in a statistical sense. In our case the points P and Q are represented by the mean of the pixel coordinates identified as a marker, Equations 3.1.3 and 3.1.4, and their covariance is calculated as the covariance of the coordinate of the pixels, Equations 3.1.5 3.1.6 and 3.1.7.

$$\bar{x_j} = \frac{1}{n} \sum_{i=1}^{n} x_{ij} \tag{3.1.3}$$

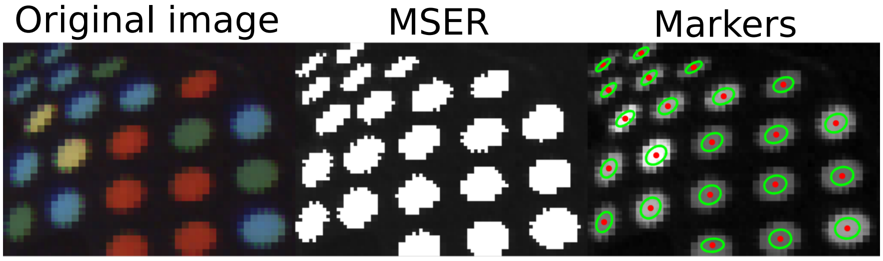


Figure 3.1.8: Markers barycentre and covariance. Left: Original image. Centre: Marker extracted with MSER. The pixel identified as belonging to one marker are white, the others black. Right: Barycentre of the marker (red) and the covariance of each marker (green).

$$\bar{y_j} = \frac{1}{n}\sum_{i=1}^{n} y_{ij} \tag{3.1.4}$$

$$c_{jxx} = \frac{1}{n-1}\sum_{i=1}^{n}(x_{ij} - \bar{x})_j^2 \tag{3.1.5}$$

$$c_{jxy} = \frac{1}{n-1}\sum_{i=1}^{n}(x_{ij} - \bar{x_j})(y_{ij} - \bar{y_j}) \tag{3.1.6}$$

$$c_{jyy} = \frac{1}{n-1}\sum_{i=1}^{n}(y_{ij} - \bar{y_j})^2 \tag{3.1.7}$$

where n is the number of pixels that are identified by MSER and belonging to the marker j, x_i and y_i are respectively the coordinate of the i-th pixel, $\bar{x_j}$and $\bar{y_j}$are the barycentre of the marker and c_{jxx},c_{jxy} and c_{jyy} are respectively the values of the covariance matrix C relative to the marker j.

Figure 3.1.8 shows the result of MSER marker detection with the relative barycentre and the ellipses that represent the covariance. The marker in the top left corner lays on a surface with an high inclination respect to the image and thus has an elliptical shape. The covariance of each markers is represented as an ellipse in Figure 3.1.8-Right.

The NMN approach was tested with both synthetic and real images. The synthetic test consists in the creation of a grid of 25 makers disposed on a 5x5 grid as shown in Figure 3.1.9-Left. To this image it is applied a transformation to simulate a reciprocal rotation between the camera and the marker Figure 3.1.9-Right. It is possible to notice in Figure 3.1.9-Right that the true marker neighbours (red) of the central marker (blue) are correctly identified as nearest neighbours even with an high inclination.

In the real case this can be seen in Figure 3.1.10. The marker 3x3 neighbours are identified in 5 different images captured by different cameras even with high inclination surface-image and low pixel resolution.

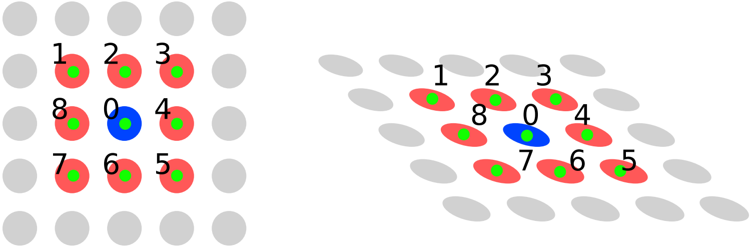


Figure 3.1.9: Nearest Mahalanobis Neighbours search result in synthetic test. Left: central marker (blue) with its 8 neighbours (red) and non neighbours markers (grey). Right: Simulation of a high angle between plane of the marker and the camera one. Neighbours are correctly identified thanks to NMN.

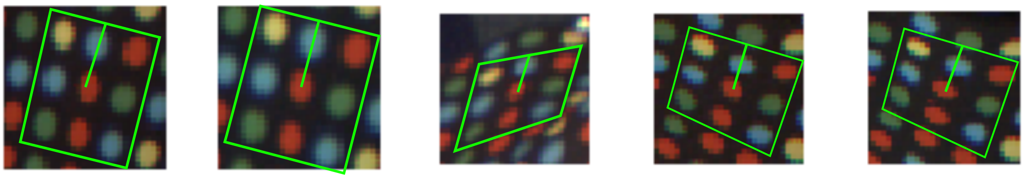


Figure 3.1.10: NMN result on real images. Same marker and its neighbours are identified even if the marker has an high inclination respect image plane.

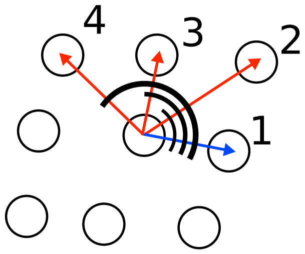


Figure 3.1.11: Neighbours sorting respect to the angle formed with the nearest in euclidean space.

The 8 neighbour, once identify correctly, are sorted with respect to the angle they form with the vector that connects the central marker with the nearest neighbours in euclidean space, Figure 3.1.11. This is a pre-elabration step useful for feature description.

Garment marker identification

After marker extraction it is necessary to identify the markers that belong to the suit and the markers that belong to the background. A Distinctive feature of the markers on the garment it is the disposition on a regular grid. For this reason we realized a filtering operation able to discern the marker of the garment on the base of neighbours disposition. In the non deformed conditions and with no inclination surface-image the neighbours are arranged on the edges and vertices of a square, Figure 3.1.5. In a high inclination surface-image the neighbours are arranged on the edges and vertices of a parallelogram, Figure 3.1.9.

We thus generalized the disposition of the neighbours of a marker with these 3 sentences:

1. vectors that connect two opposite neighbours respect to the central marker have the same length Figure 3.1.12-Left

2. vectors that connect two opposite neighbours respect to the central one must be aligned Figure 3.1.12-Center

3. vectors that connect three neighbour on the same edge are aligned Figure 3.1.12-Right

In the algorithm implemented the markers are filtered by:

1. setting a threshold on the difference of the norms of the vectors v_1 and v_2. The points that respect this inequality, $|\|v_1\| - \|v_2\| < |0.1 \cdot \max(\|v_1\|, \|v_2\|)$, are considered as valid. This condition is verified for all the 4 couples of opposite neighbours.

2. setting a threshold on the difference of the angles of the vectors v_1 and v_2. The inequality to be respected is $|\angle v_1 - \angle v_2 - 180°| < 20°$, where $\angle v_1$ and $\angle v_2$ are

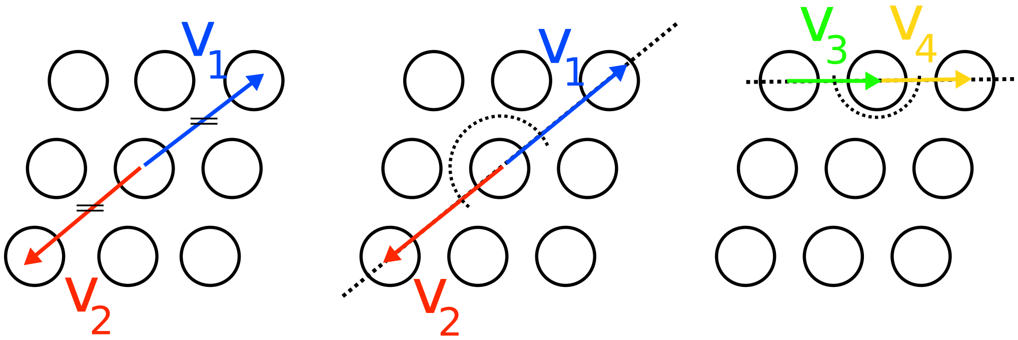


Figure 3.1.12: Distinctive features of markers on the garment. Left: vectors that connect two opposite neighbours respect to the central marker have the same length. Centre: vectors that connect two opposite neighbours respect to the central one must be aligned. Right: vectors that connect three neighbour on the same edge are aligned.

respectively the angles that vectors $v_1$and v_2 form with the horizontal axis. This condition is verified for all the 4 couples of opposite neighbours.

3. setting a threshold on the difference of the angles of vectors $v_3$and v_4. The inequality to be respected is $|\angle v_3 - \angle v_4| < 20°$, where $\angle v_3$ and $\angle v_4$ are respectively the angles that vectors $v_3$and v_4 form with the horizontal axis. This condition is verified for all the 4 edges.

If all these 12 (4+4+4) conditions are satisfied then the marker and its neighbours are considered as belonging to the garment.

Figure 3.1.13 shows the markers identified with MSER and the result of the filter operation described.

3.1.2 Color recognition

The color recognition is the task to recognize the color of a marker and assign one of the 4 possible labels: 0-red, 1-yellow, 2-green, 3-blue. This label is used to encode the identity number of the marker, Subsection 3.1.3. The algorithm that we developed has to be robust and be able to deal with different illumination conditions of the markers (shadows) and with different type of suits. To solve this task and obtain a robust color recognition we proposed an algorithm which rely on the markers disposition on a regular grid to select and clusters colours markers in Hue color space.

The algorithm here proposed to detect the colours of the markers is articulated in two steps:

- marker color conversion to HSV
- color clustering in Hue space

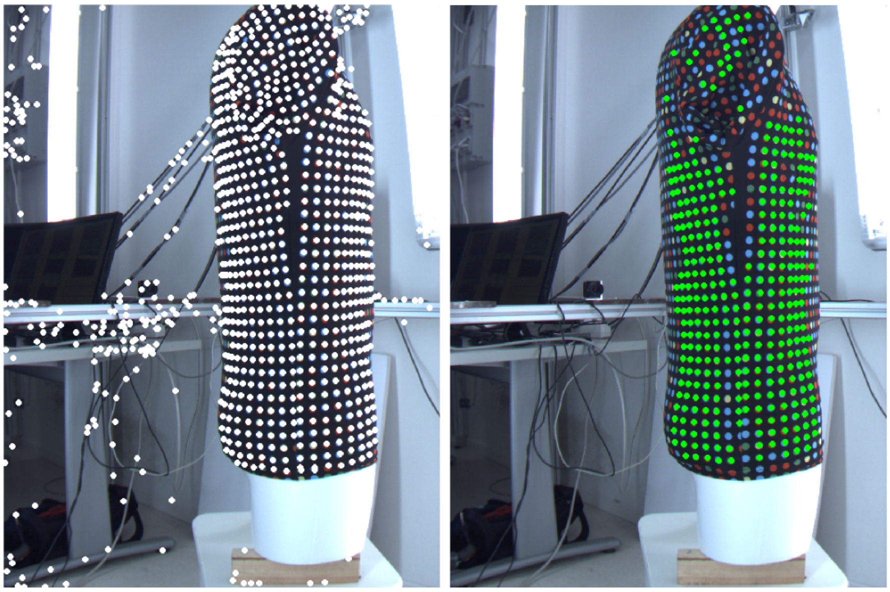


Figure 3.1.13: Garment marker identification based on grid disposition. Left: all markers detected in image. Right: marker which satisfies the three condition of neighbours disposition.

As describe in Subsection 3.1.1 we are able to discern the marker of the suit respect to the markers on the background thanks to their disposition on a regular grid. The RGB mean value of each marker is calculated and stored as a distinctive property. Since we want a robust algorithm which is illumination independent we decided to work in a color space in which we can separate the component of the illumination.

We chose HSV and in particular we use the information of the Hue channel, since *"the hue ... is invariant to surface orientation, illumination direction, and illumination intensity under the assumption of dichromatic reflectance and white illumination"* [39]. Figure 3.1.14 shows the HSV cylindrical-coordinate representations of points in an RGB color model. The Hue starts conventionally from the red (0°) goes to yellow (60°) to green (120°) to blue (180) to return to the red (360°).

Since the numbers of the colours of the markers on the garment are known a priori (4) we cluster the hue value of the markers with a K-means algorithm. Figure 3.1.15 shows on the right a sample image and the marker on which is performed the color clustering analysis. On the left is visible the histogram of the hue of the markers. The 4 groups of colours are easily identifiable.

With this clusterization the color-label assignation is straightforward: the first group represent red markers and these will receive the label equal to 0, the second are yellow markers with a label equal to 1, the third and forth are respectively green and blue markers with labels equal to 2 and 3.

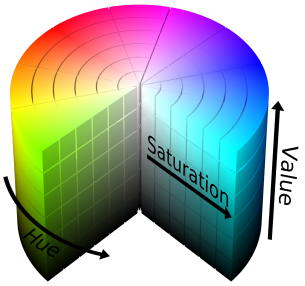


Figure 3.1.14: HSV color coordinate cylindrical representation.

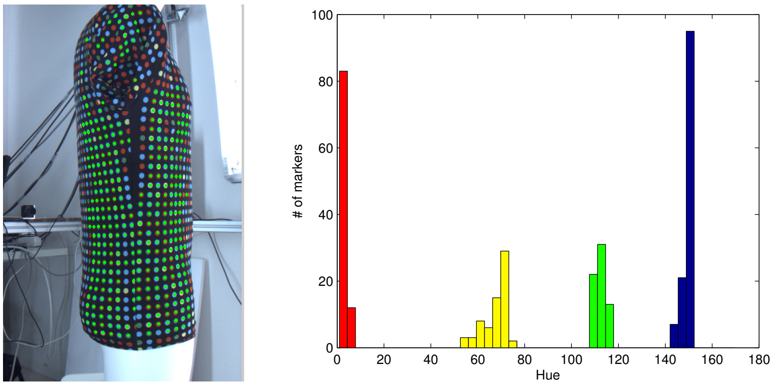


Figure 3.1.15: Color clustering. Left: Marker detected that belong to the suit. Right: Histogram of the colours of the image build on the hue of marker detected.

3.1.3 Feature descriptor

The feature descriptors are widely used in a number of vision related tasks: stereo reconstruction, objection recognition, tracking. Their scope is to give a compact description of an image content. Wanted characteristics of descriptors are:

- Affine invariant. The descriptor must be invariant to change in scale, rotation, viewpoint changes.
- Locality. Features are local, so robust to occlusion, clutter and change in scene illumination
- Distinctiveness. The can differentiate a large database of points.
- Efficiency. Feature extraction must be able to reach real-time performance.

The main purpose of feature descriptor in this work is to solve the correspondence problem, i.e. identify the same marker in different images in order to reconstruct its 3D position.

With a discrete feature definition, as for the markers on the garment, we developed a feature description based on the colours of the $3x3$ neighbour markers. The idea is simple yet powerful: to each marker is assigned a unique identification number, or ID, color coded with its neighbours, Figure 3.1.17.

The ID uniqueness of each marker allows to achieve a robust multicamera matching and tracking even with an high number of markers per cm^2.

This approach was firstly proposed by[77] which developed a pseudo-random colour coded feature descriptor based on 3 colours and the $v \times w$ neighbour.

The code developed consists in a square pseudo-random matrix, M, of size L $(k = l = L)$. To each element m_{ij} within M is assigned one letter from a palette of A possible letters. Since M is a pseudo-random array, the location of each element m_{ij} (with at least 8 neighbours) in M is completely determined by the letters contained in the $v \times w$ neighbourhood of m_{ij}.

[77] used in his system only a projector (to project the grid) and a single camera to reconstruct the shape of a static object.

While in his work there are no limitations (except code repetition) on sequences generation we imposed some others constrains:

1. generates a high number of codes
2. every block of 3×3 points is unique
3. blocks are rotation independent
4. blocks must overlap

The first requirement is due to the fact that if we want to reconstruct the shape of the subject we need a high density of marker per cm^2 that covers all the body. The

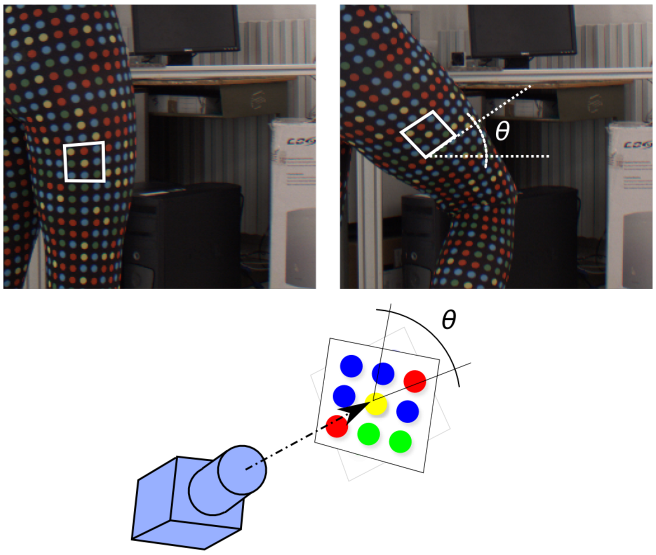


Figure 3.1.16: Camera-marker rotation changes during the motion.

density of markers per cm^2 is a trade-off between relative distance camera-subject and the accuracy with which we want to measure the surface.

Markers size and spacing depend on the application: a full motion capture requires cameras placed a congruent distance (~2-3 meters), bigger markers and higher spacing between them. For a single limb analysis the distance camera-subject can be reduced (~1 meter) as markers diameter and spacing.

We decided to generate unique blocks of 3×3 markers to ensure shape reconstruction of smaller limbs. These blocks are indeed compact and well fits the arm of an adult. The numerical sequence that can be generates is of length equal to 9.

In order to ensure a robust color recognition even in different lighting condition the number of color was chosen equal to 4. The maximum numbers of blocks with these values is $q^n = 4^9 = 262144$. The convention adopted to label the colours is: red=0, yellow=1, green=2, blue=3.

The second requirement ensure no-ambiguity of markers association in different images captured by different cameras.

The third constrains is introduces because when the subject moves the relative rotation between a block and the camera changes, Figure 3.1.16. We have thus no information about where to start to read the sequence of neighbours. This can be resolved in two ways: the first one is to use a cyclic code, the second one is to use special markers to identify the principal directions. In order not to use special marker, which will introduce unnecessary complexity, we choose cyclic code.

A cyclic code means that the sequence can be read starting from any digit. A cyclic code reduces the number of blocks since we have to eliminate the code that can be obtained by a rotation of the same code around the central marker, Figure 3.1.17-Center.

The digit in the sequence which refers to the central marker is always the first and does not change with the rotation. For this reason in next examples the first digit will be omitted.

The rotation of the block around the central marker can be regarded as a shift (forward or backward) of the sequence by d positions.

For example if we consider a block defined by the sequence 02341221, with 1 shift forward we will obtain 10234122. Notice that the last digit is now at the begin of the sequence.

Since we are using a cyclic code the two sequences refers to the same block and are translated to the same ID. For this reason we have to eliminate the blocks that are produces by a rotation (or a shift) of another block.

With length $n = 8$ (because the centre of the block is fixed) we have only four cases of possible shifts, one for each divisor of n:

1. 1 shift, e.g. 00000000
2. 2 shifts, e.g. 10101010
3. 4 shifts, e.g. 01000100
4. 8 shifts, e.g. 10000000

In case 1 we obtain an identical sequence with just one shift (forward or backward). In case 2 we obtain an identical sequence with 2 shift forward of the sequence displayed. In case 3 we obtain an identical sequence of the one displayed with just 4 shifts. In case 8 we obtain a unique sequence that can not be generated by shifting another (8 shifts means that each digit remains at its position).

To compute the number of sequence of $n = 8$ digits that can be generated we use the follow theorem:

Theorem 1 *The strings of length n with an alphabet of q letters have equal shifts for each divisor d of n.*

Denoting by s_d the number of d shifts, we have:

$$s_1 = q$$

$$s_2 = q2 - q$$

provided that $2 \mid n$ and in general:

$$s_d = q^d - \sum_{i<d, i|n} s_i$$

The number of different shifts of all the strings of length n with an alphabet of q letters is:

$$\frac{1}{d}s_d$$

In our case with $n = 8$, $q = 4$ we have $q^n = 4^8 = 65536$ strings (sequences). Their shifts are:

d	s_d	$1/d\, s_d$
1	4	4
2	12	6
4	240	60
8	65280	8160
Σ	q^n	8230

We have only 8160 strings with 8 different shifts. These considerations were made for the 8 points around the centre of a block. Since every centre can assume one of the 4 colours there is a maximum of $q \cdot s8 = 4 \cdot 8160 = 32640$ sequences.

The fourth requirement, Figure 3.1.17-Bottom, allows to generate a compact grid of blocks which can be exploited to filter wrong IDs and add missing IDs. This property is described in details in next subsection.

The sequences are defined on a numerical base of $q = 4$ (colours). In order to manage the sequences with more ease we performed a change of base. We translated the sequence with base 4 to a decimal number with the polynomial form. Each number can indeed expressed as a sum of powers which have for base the base of the numeration system and power the position occupied by the digit:

$$n = \sum_{k=1}^{m} a_k B^k$$

where $B \geq 2$ is the base of numeration, a_k is the k-th digit that can assume values $0 \leq a_k \leq B-1$, $a_k \in \mathbb{N}$, m is the number of the digits by which the number is composed, n is the decimal number.

The numerical sequence of 9 elements can thus expressed as a decimal number with a simple change of base as in this example: $102341221_4 \rightarrow 1 \cdot 4^8 + 0 \cdot 4^7 + 2 \cdot 4^6 + 3 \cdot 4^5 + 4 \cdot 4^4 + 1 \cdot 4^3 + 2 \cdot 4^2 + 2 \cdot 4^1 + 1 \cdot 4^0 \rightarrow 77929_{10}$.

Since we used a cyclic code and there is no a defined starting marker to read the sequence, we have 8 possible sequences to be translated. Each sequences is produced by a shift forward of the precedent. The first digit, correspondent to the central marker, is fixed. For example the sequence 102341221 has 8 possible shifts:

1. 0 shift: 102341221

2. 1 shift: 110234122

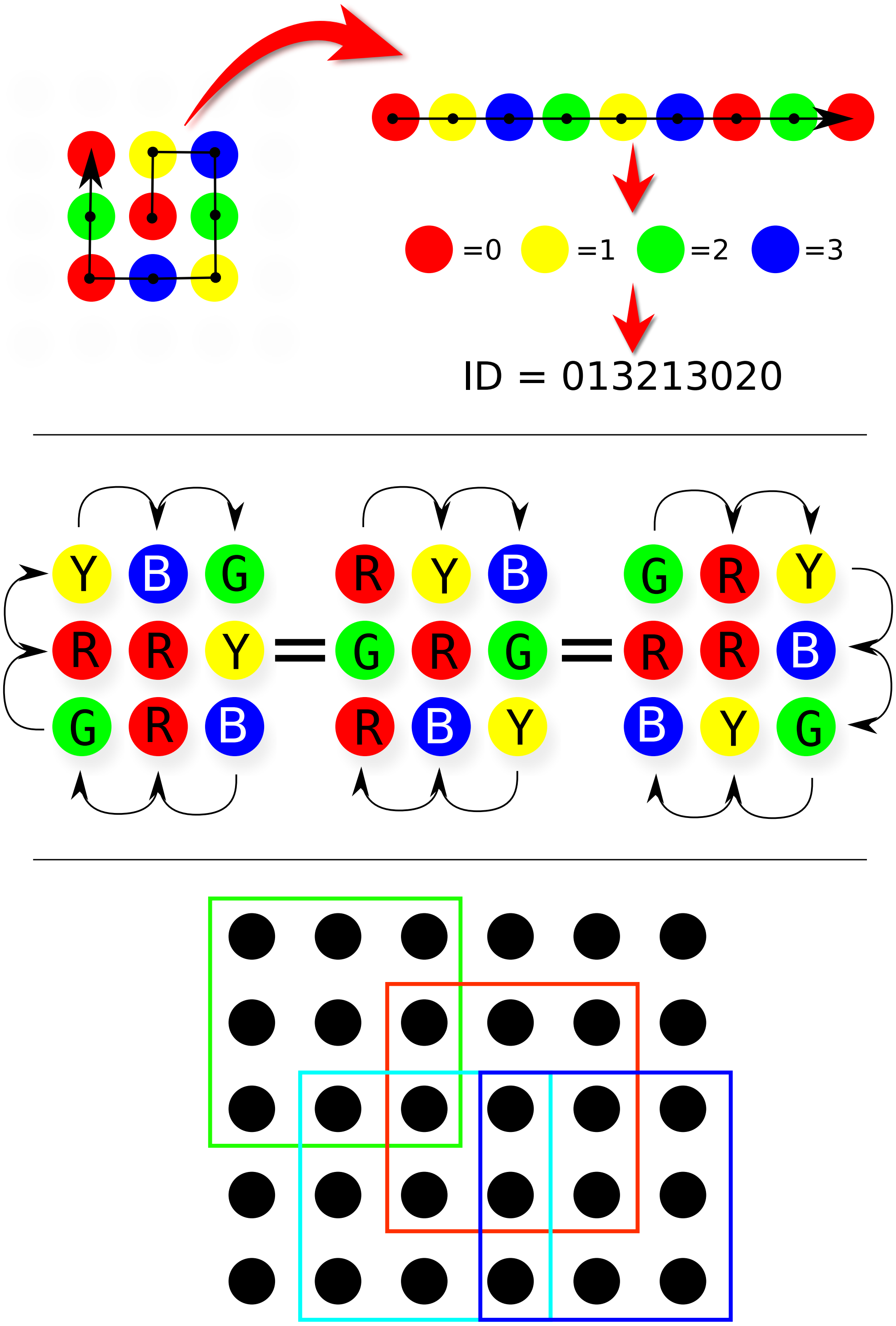


Figure 3.1.17: Marker properties. Top: Color coded ID assignation to blocks. Centre: No rotational property of markers. Bottom: Sequences overlap.

3. 2 shifts:121023412
4. 3 shifts:122102341
5. 4 shifts:112210234
6. 5 shifts:141221023
7. 6 shifts:134122102
8. 7 shifts:123412210

We defined as *main* sequence the one which produce the higher decimal number. This *main* sequence defines also a unique order in which the sequence have to be read. So, after the detection of the markers which are a centre of a sequence, we compute all the 8 possible basis transformation and then we keep the maximum. This is the ID of the marker.

In the example above the sequences, after a change the change of basis, assume respectively the values:

1. $102341221_4 \rightarrow 77929_{10}$
2. $110234122_4 \rightarrow 85018_{10}$
3. $121023412_4 \rightarrow 103174_{10}$
4. $122102341_4 \rightarrow 107713_{10}$
5. $112210234_4 \rightarrow 92464_{10}$
6. $141221023_4 \rightarrow 137803_{10}$
7. $134122102_4 \rightarrow 132754_{10}$
8. $123412210_4 \rightarrow 115108_{10}$

The *main* sequence is the 6-th since it produces the number with maximum value (137803). This is the ID of this marker.

The **Theorem 1** allows only to compute the total number of sequence that can be produced with the desired properties, but not how to arrange them.

In the implementation we thus relayed on a brute force approach: the algorithm firstly generates a list of all possible sequences, then, starting with a seed-block, tries to arrange them. When the algorithm does not succeed to add more sequences undoes the last 3 sequences arranged and start again with a different sequence. The algorithm finishes when no other sequences can be arranged on the grid.

With this algorithm we were able to create a grid of 90 rows and 90 columns for a total of $(rows - 2)(cols - 2) = 7744$ sequences.

This number of sequences is much lower than the theoretical one (32640) because the constrain of overlap. Anyway we were able to produce with this grid a suit with a sufficient density of marker per cm^2 suitable for a shape and motion reconstruction, Figure 3.1.18.

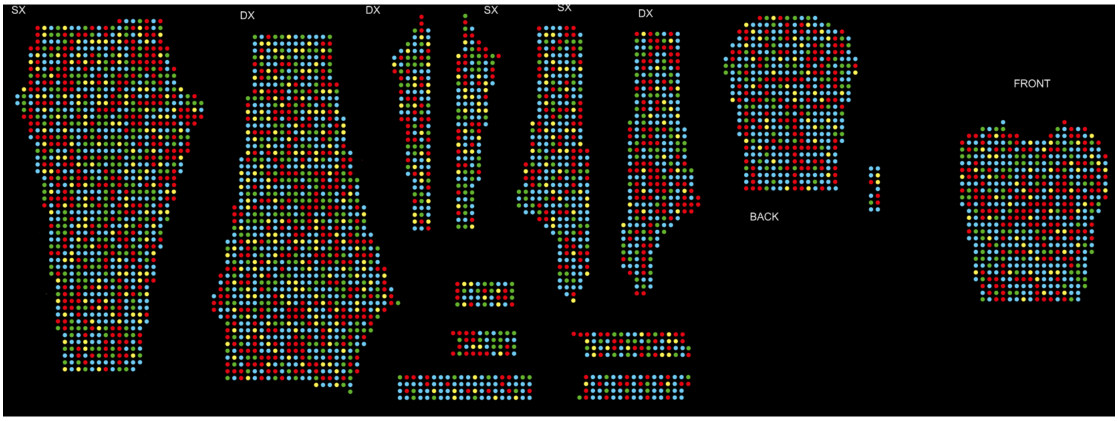


Figure 3.1.18: Pieces of the suit with the coloured coded pattern.

Garment mapping

The realization of the suit to reconstruct the motion was achieved cutting out from the grid of 90 × 90 markers the pieces of the suit, Figure 3.1.18. Since the markers on the cut edge have no 8 neighbours a direct ID assignation for these markers it is not possible, thus the number of the marker with a valid ID with the suit show in the image is 2933.

With the image of Figure 3.1.18 it is possible to build a *map* of the suit. The map will contains information about which IDs are on the suit and also their relative position.

This map of the suit it is useful for two purposes: filter out wrong IDs and add missing IDs.

A wrong ID calculation on the suit be caused by three factors: fold on the garment, ID generation in correspondence of the seams and wrong color association of markers.

The suit has been realized in Spandex, an elastic tissue which assure a good adhesion of the garment to the body of the subject. Yet is possible that, when performing some motions or when the suit it is not stretch enough, the garment fold covers one or more of markers, Figure 3.1.19-Left. Another possible wrong ID assignation is in correspondence of the seams. Different pieces of the suit when attached together can generate new ID getting close two rows of markers, Figure 3.1.19-Right.

A wrong color recognition of a marker occurs with a poor marker resolution. In this case the noise of the image sensor and the color interpolation of the Bayer mask lead to a wrong color association of the marker and thus a wrong ID. Figure 3.1.20 shows an example were a blue marker (label=3) is labelled as a green marker (label=2), because of the poor resolution and the image noise.

A simple solution to these problems is given by the garment mapping. The ID associated of a marker and the ID of the neighbours is checked with the map. If these IDs are congruent with the disposition of the IDs on the map then the ID of the marker is kept, otherwise is eliminated.

Figure 3.1.21 shows and example of wrong ID generation in correspondence of a seam. In Figure it is also visible the marker with the same ID but in the correct position. The

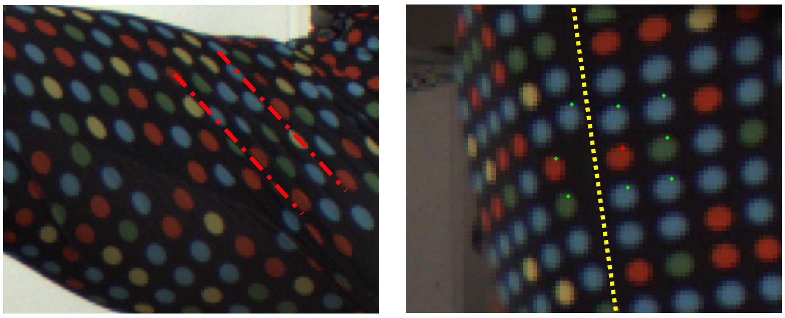


Figure 3.1.19: Possible wrong ID assignation. Left: Two fold of the garment that hide some markers. Right: Seam of two suit pieces that generate a new ID.

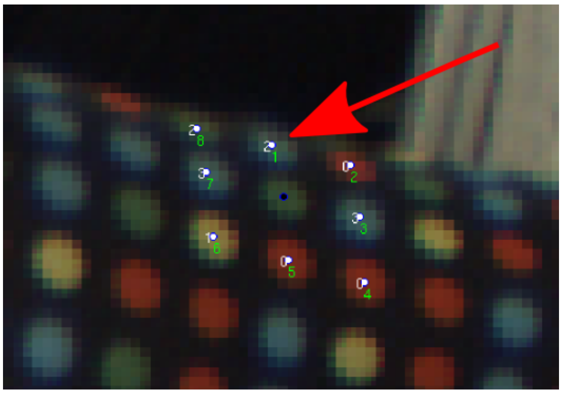


Figure 3.1.20: Wrong marker color association. The arrow indicates the markers that is classified as a green marker (label=2) instead of a blue marker (label=3). The numbers in white are the color-labels of the each marker. The numbers in green indicate the order in which they marker should be read to generate the ID.

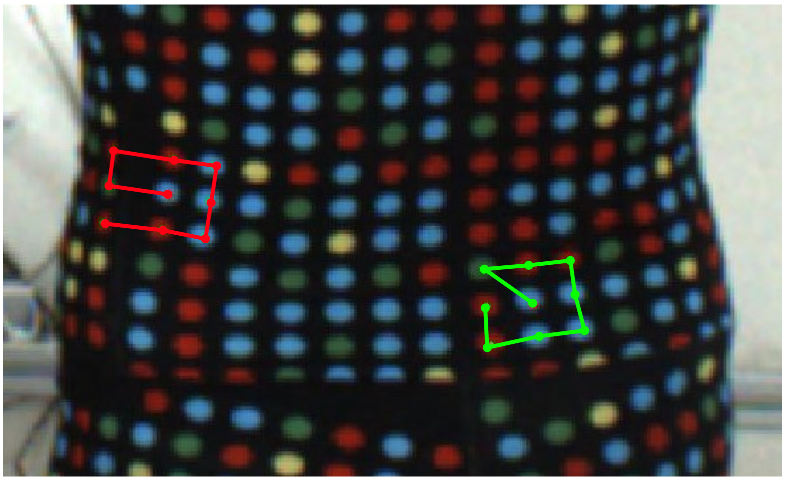


Figure 3.1.21: Wrong ID generation on seam correspondence. In red is represented the ID and the neighbours which generates a ID not compatible with the map. The green one is the correct.

first one will be eliminated.

Garment mapping is also useful to add ID to markers for which it is not possible to calculate directly the ID since one or more of 8 neighbours are occluded. With the knowledge of the map it is sufficient that one or more neighbour have an ID and this information can used to “fill the holes”..

Figure 3.1.22 shows a typical example in which for the marker highlighted with the green circle it is not possible to detect the 8 neighbours because they are occluded by the hand of the user. For this marker and all the markers in the same row it is not possible to calculate directly the ID. The map allows to spread the information of the neighbours to the marker without an ID, Figure 3.1.23, to obtain an higher number of detected markers and thus more complete reconstruction.

Figure 3.1.24 shows the result of ID-adding thanks to the map. Red asterisks indicate markers for which is possible to calculate the ID. Blue asterisks indicate markers that receive an ID thanks to the neighbours ID and the map. It worth noticing that this operation add a lot of markers to 3D shape reconstruction.

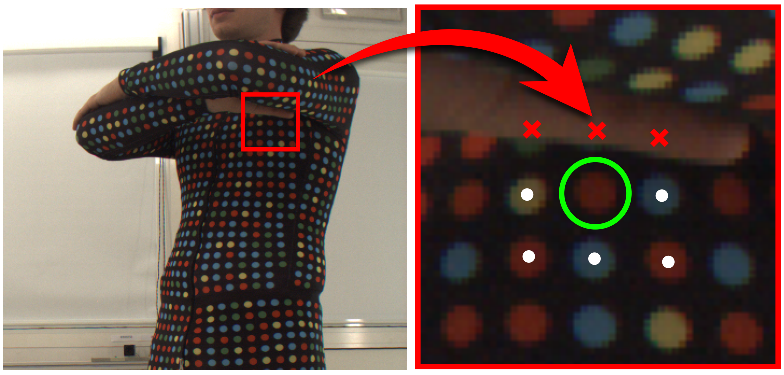


Figure 3.1.22: Occlusions can cause the missed detection of the 8 neighbours and thus the central marker (green circle) will not receive an ID.

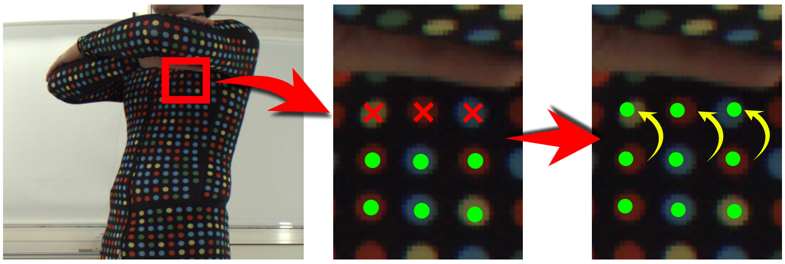


Figure 3.1.23: Marker for which it is not possible calculate and ID will receive one thanks to the ID of the neighbours and the garment map.

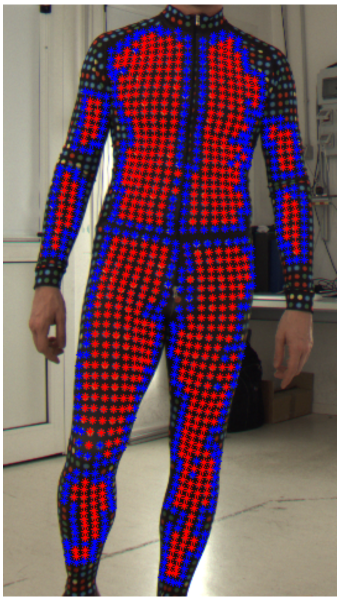


Figure 3.1.24: ID added thanks to the garment map. Red asterisks indicate marker for which the ID is calculated relying on the colours of the neighbours. Blue asterisks indicate markers that receive an ID thanks to the neighbours ID and the map

3.2 Shape reconstruction

In this Section is described the algorithm used to reconstruct the shape of the subject. Respect multi-stereo approach in this work we adopted a pure multicamera strategy.

In previous works, [27, 28], shape reconstruction was performed using a multi-stereo approach. Each couple of camera is considered a system isolated to each other. For each couple of cameras the markers are extracted and associated and a 3D cloud of points is produced.

The main advantage with this approach is that the features have a little distortion since the images captures with the stereo-pair have very similar view point. This speed up the process of intra-image feature association.

One of the most critical point in the multiple stereo approach is originated from the fact that the points cloud generated from each stereo-pairs must be fuse in a singe points cloud. A multi-stereo approach requires also a partial overlapping of the points clouds of different stereo-pairs.

Merging different cloud of points is performed with classic points registration algorithms, like ICP [101], or with more complex approach that take into account uncertainty propagation, compatibility analysis and Bayesian data fusion, [28].

Thanks to the unique feature descriptor developed, 3.1.3, it is possible to directly associate markers among the images captured with all cameras on the base of the ID associated to each marker.

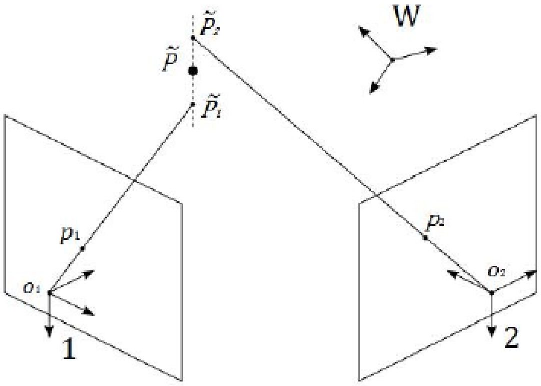


Figure 3.2.1: Mid point triangulation algorithm.

One of the simplest triangulation algorithm is the mid point triangulation, [46]. The mid point triangulation calculate the 3D point coordinates as the point of minimum distance of the two optical rays passing through the optical centre and the projection of the point on the image plane, Figure 3.2.1. This algorithm allows to recover the 3D position of a point taking in account only points projection on two different cameras plane. In order to adapt this method to our multicamera setup, in which are possible more than two marker projection, we developed a two step algorithm. First step consist in the evaluation of a first guess of the 3D position of each marker with an ID associated with a distance weighted mid point algorithm. Second step refine the position all the markers with an optimization phase based on Sparse Bundle Adjustment.

In the first step the triangulation of a marker seen by n cameras is achieved considering all the possible combinations, m, of each couple of cameras. For each combination k of the cameras i and j are stored the 3D coordinates of the mid point, P_k, and the distance of the optical rays, d_k. A low value of the optical ray distance d_k indicates condition near to the ideal case, in which the the two optical ray meet in one point. The optical ray distance is used as a value to filter out possible outlier since an high optical distance to indicates a possible bad ID calculation (fold of the garment, bad color association) or a bad system calibration (camera moved accidentally). The point P_k with a distance between optical ray d_k above 20 mm are considered outliers and thus discarded.

The value of the m points obtained with this method are meaned using as weight the square of the inverse of distance of the optical rays, d_k, as show in Equation 3.2.1.

$$P = \frac{1}{w} \sum_{k=1}^{m} P_k \cdot \frac{1}{d_k^2} \tag{3.2.1}$$

where P is the point in 3D space, w is the sum of the weighting factors $w = \sum_{k=1}^{m} 1/d_k^2$, P_k and d_k are respectively the mid point and the distance of the optical ray obtained by the k mid point calculation.

The second step is a Sparse Bundle Adjustment optimization implemented by [64]

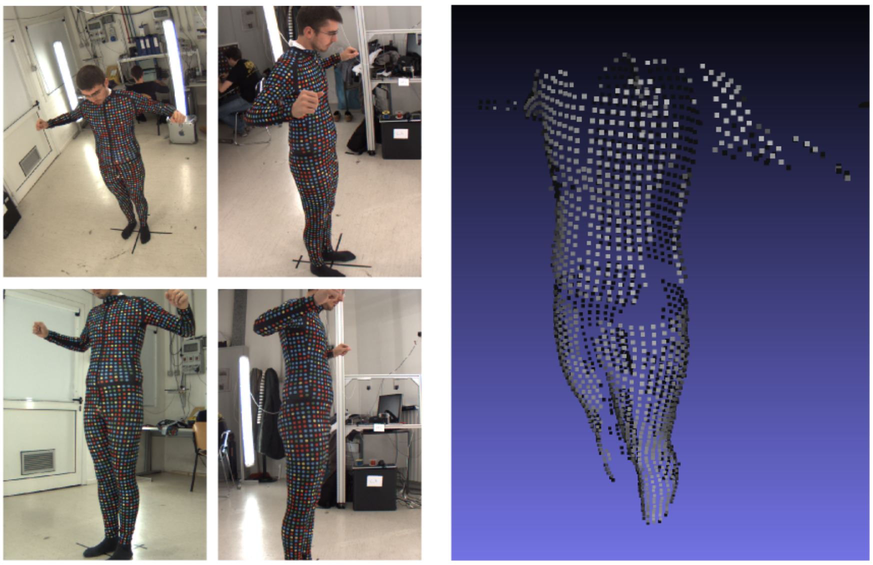


Figure 3.2.2: Images captured by GaMoCap system and the result of shape reconstruction with the algorithm proposed.

which allows to refine the position of the 3D points.

The images captured by the multicamera system and the result of shape reconstruction are shown in Figure 3.2.2.

3.3 Multicamera calibration

In this Section is described the algorithm developed to calibrate the GaMoCap multicamera system.

The algorithm proposed to calibrate the GaMoCap is simple and reliable. The basic idea is to replace the chessboard pattern used by [12] with the same coloured pattern used for the garment, Section 3.1.3, printed on a planar surface. The unique ID of each marker indeed resolves the problem of partial visibility and makes the intra-camera points correspondence straightforward.

Steps of the algorithm proposed are:

1. capture of synchronized images of the calibration pattern
2. marker extraction and ID assignation for each cameras and each frame
3. intrinsic parameters calculation for each camera
4. first guess of extrinsic (camera connection graph and minimum path with Dijkstra)
5. sparse bundle adjustment (SBA) optimization of extrinsic parameters

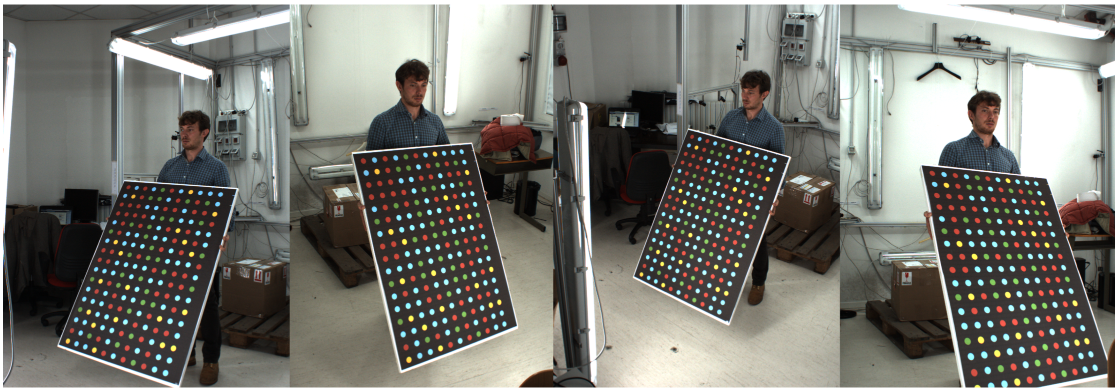


Figure 3.3.1: GaMoCap multicamera calibration: calibration grid captured by 4 different cameras

Intrinsic calibration

The first step in calibration procedure consists in the synchronized capture of images with all cameras of the GaMoCap calibration pattern, Figure 3.3.1. After each shot the pattern orientation and position is changed, taking care to cover all the image area of each camera. Obtain control points at image border is an essential step for good distortion parameters estimation.

Usually for a full-motion capture the camera are placed in a ring configuration (all around the subject) to obtain a full shape reconstruction. While moving the calibration pattern it is important to "close" the ring, i.e. make sure that the first and last cameras of the ring can see the calibration grid. This operation assure a good parameters optimization with SBA.

The second step consists in the extraction of coordinates of the markers from images and the ID assignation. Since coloured pattern of the calibration plane is the same of the garment, the algorithms that perform marker extraction and ID assignation are the same of the one described respectively in Subsection 3.1.1 and Subsection 3.1.3.

The third step is intrinsic parameters calculation for each camera. The algorithm used is the one proposed by [113] and implemented in OpenCV,[51]. For each camera the algorithm takes as input the 2D coordinate of the marker extracted with a valid ID in each image and the correlated 3D coordinate of the calibration plane. The pose (orientation and position) of the calibration plane in each image is initialized with a closed-form solution, followed by a non-linear refinement based on the maximum likelihood criterion of Equation 3.3.1.

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left\|m_{ij} - \check{m}(K,^i R_G^C,^i t_G^C, M_j)\right\| \tag{3.3.1}$$

here $\check{m}(K,^i R_G^C,^i t_G^C, M_j)$ is the projection of point M_j in image i[1] according to the Equa-

[1] In this Subsection the letter i will be used to refer to the i-th image captured by the camera, and also

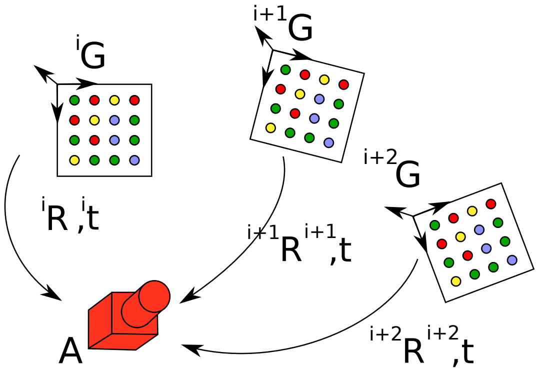


Figure 3.3.2: Grid to camera transformations at time instant i, i+1, i+2

tion 2.3.5 followed by distortion according to 2.3.7.

The algorithm returns the vector of camera distortions, k, the matrix of intrinsic parameters, K, and the n rotation matrices, ${}^{i}R_{G}^{C}$, and translation vectors, ${}^{i}t_{G}^{C}$, that express the transformations of grid (calibration plane) reference system respect camera reference system at i-th time instant, Figure 3.3.2.

Since the calibration grid may not be entirely seen by the camera in this step the number of markers of calibration plane visible in the i-th image are stored. These data will be used to build the camera connection graph described in next section.

Connection graph

For simplicity, instead of using rotation and translation matrices, to express the transformation between references system will be used the homogeneous matrix notation, Equation 3.3.2.

$$ {}^{i}H = \begin{bmatrix} {}^{i}R_{G}^{C} & {}^{i}t_{G}^{C} \\ 0 & 1 \end{bmatrix} \tag{3.3.2} $$

The strategy on which is based the determination of first guess of extrinsic parameters (relative rotation and translation between cameras reference system) is explained through an example.

When the calibration plane is visible by two cameras, called A and B, at time i and its rotation and translation are computed for both camera it is possible to write the roto-translation between the cameras reference systems, Figure 3.3.3.

This is formalized in Equations 3.3.3 and 3.3.4.

$$ \begin{cases} P^{A} =^{i} H_{G}^{A} \quad P^{G} \\ P^{B} =^{i} H_{G}^{B} \quad P^{G} \end{cases} \tag{3.3.3} $$

to the i-th time instant at which the image was capture. Since all the cameras are synchronized, the images i captured by two different cameras refer to the same time instant.

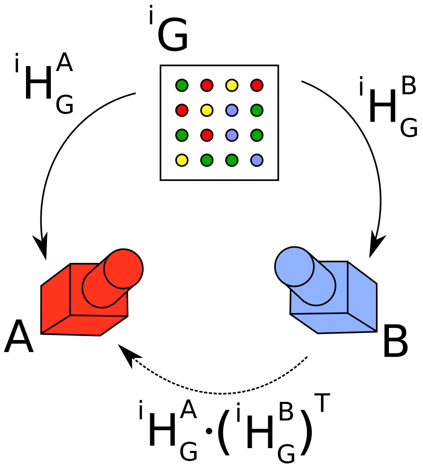


Figure 3.3.3: Homogeneous transformation from Grid to Camera A and B at time i.

where P^A, P^B and P^G are the same point defined respectively in reference system of camera A, B and G.${}^{i}H_G^A$ is the homogeneous transformation from grid (calibration pattern) reference system to system A and ${}^{i}H_G^B$ is the homogeneous transformation from grid (calibration pattern) reference system to system B. Solving the second Equation of the system 3.3.3 respect P^G and substituting into first Equation we obtain Equation 3.3.4, that express the relation between cameras system A and B.

$$P^A = {}^{i}H_G^A \left({}^{i}H_G^B\right)^T P^B \tag{3.3.4}$$

In practice, the calibration object may never be seen by all cameras (taken two by two) simultaneously. It is thus impossible to compute the direct homogeneous transformation between each couple of cameras as in Equation 3.3.4. In order to handle non-overlapping cameras and be able to compute an homogeneous transformation for each couple of cameras we decide to rely on Graph Theory.

Introduced by Euler in 1736, formalized by [26] and [43] a graph can be defined as a pair of sets $G = (N, L)$, where $N = [n_1, ..., n_m]$ is the vector of *nodes* of the graph G and $L = [l_1, ..., l_O]$ is the vector of the *links*. Graph theory can be extended by assigning a weight to each link of the graph. Graphs with weights, are used to represent structures in which connections have some numerical values. Such weights might represent, for example, costs, lengths or capacities.

Another important theoretic concept of the Graph Theory are the *walk*. A *walk* can be defined as a sequence of nodes and edges, with an initial and a final node, and all the intermediate nodes. A *walk* is called a *path* if each link is covered only one time.

In our case the *nodes* are representation of the cameras, the *links* the direct connection of two cameras at a certain time instant as in Equation 3.3.4.

The *weights* of the links express how much a connection between two cameras is good. The *weights* are calculated as the maximum number of points of the grid seen

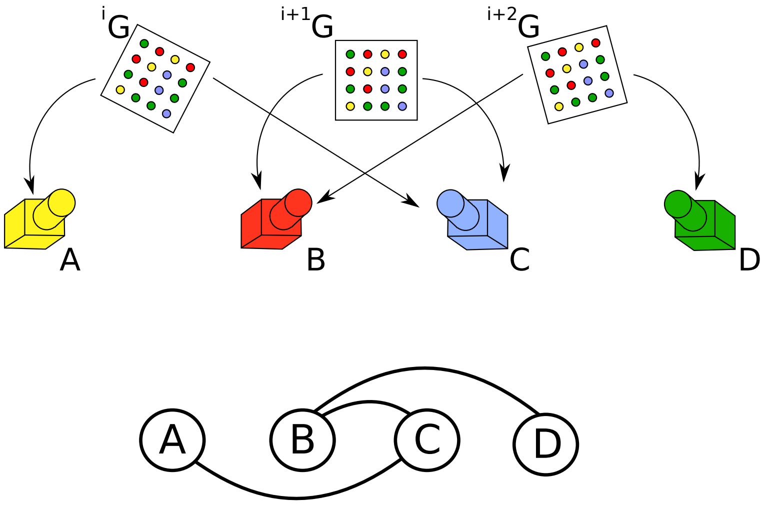


Figure 3.3.4: Graph representation of multicamera connection during the calibration phase.

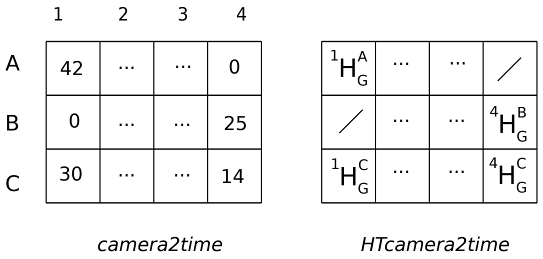
simultaneously by the camera connected for all times instant. A higher *weight* is a stronger bound between cameras.

Figure 3.3.4 show the connection among 4 cameras that see the calibration grid at different time instants. Top: at time i the grid can be see by cameras A and C, at time i+1 by B and C, at time i+2 by cameras B and D. Bottom: graph representation of these connections.

The graph relative to the camera configuration must be built elaborating the data obtained by camera intrinsic calibration. The number of markers in the calibration grid, seen by each camera at each time instant, are stored in a matrix with as many rows as cameras that have to be calibrated and as many columns as image captured by each camera (relative to a well defined time instant). This matrix is called *camera2time*. At the element (c, i) of this matrix is stored the number of points of the grid seen by the camera c at the time instant i.

In a matrix with same dimensions, called *HTcamera2time,* are stored the homogeneous transformation matrices that at a well defined time instant express the grid reference system to a determined camera. At element (d, j) of this matrix is stored the homogeneous transformation matrix, ${}^{j}H_{G}^{d}$, that express the transformation of the calibration grid respect the camera d at the time instant i.

A simplified example involving 3 cameras (A,B,C) and 4 images per camera is shown in Figure 3.3.5. Top-Left: camera2time matrix stores the number of the point of the grid recovered by each camera at each time instant. Top-Right: HTcamera2time matrix, stores the homogeneous transformation matrices of the grid respect each camera at each time instant. Camera B at time t=1 and camera A at time t=4 do not see the calibration grid, thus the homogeneous transformation matrix it is not recovered.



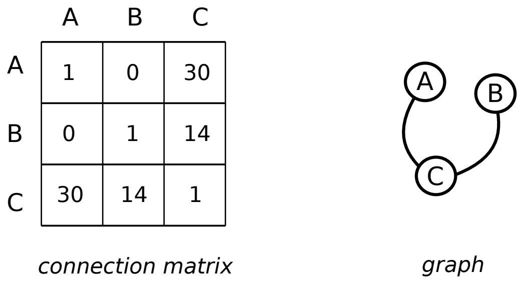
	1	2	3	4
A	42	...	...	0
B	0	...	...	25
C	30	...	...	14

camera2time

${}^{1}H_{G}^{A}$	...	...	/
/	...	...	${}^{4}H_{G}^{B}$
${}^{1}H_{G}^{C}$	...	...	${}^{4}H_{G}^{C}$

HTcamera2time

Figure 3.3.5: Example of calibration of 3 cameras and 4 images taken at 4 different time instant.



	A	B	C
A	1	0	30
B	0	1	14
C	30	14	1

connection matrix

Figure 3.3.6: Matrix representation of the graph connection.

Camera2time and *HTcamera2time* matrices are used to build the graph.

The information of the links between the nodes (cameras) and their weight can be expressed with matrix notation. The matrix, called *connection matrix*, is a symmetric square matrix with has as many rows as cameras that have in each element a numerical value that express the weight of connection between two cameras, Figure 3.3.6. If the element has value 0 means that there is no a direct connection between two cameras. On the diagonal the value is set to 1 because is the connection of the camera with itself.

The *connection matrix* is build evaluating each column (time instant) of *camera2time* matrix. For each column is kept the maximum value that connect two cameras.

Once the camera connection tree is build, we have to determine the best estimation of homogeneous transformation between the camera chosen as reference, called *main* and all the others cameras.

In our case best estimation is the one that exploit the biggest number of points reconstructed of the calibration grid to estimate the transformation between two camera.

This problem is solved with weighted graph theory. The convention of weigh of connection between two cameras is: lower weight → strong connection. We take the reciprocal of each element in the connection matrix. With this convention is possible to find the optimal path with the shortest path theory.

In graph theory, the shortest path problem is the problem of finding a path between two vertices (or nodes) in a graph such that the sum of the weights of its constituent edges is minimized.

The shortest path was found with the implementation of the Dijkstra's algorithm, [30]. For a given source node in the graph, the algorithm finds the path with lowest sum of weight between that node and every other node.

The node taken as the source of the graph correspond to the *main* camera, the camera chose as the *world* reference system. Once identified the shortest path from each node to *main* node it necessary just to compose the homogeneous matrix in the right order to obtain the first guess of extrinsic parameters.

The algorithm implemented can identify if one (or more) node are detached form the source node and thus a path is not computable. This formulation of the estimation of the first guess of the extrinsic parameter allows to deal with non-overlapping cameras and free the camera positioning.

SBA optimization

As last step the extrinsic parameters are optimized with the sparse bundle adjustment implementation of [64]. Provided with initial estimates of extrinsic parameters obtained with graph, BA refines homogeneous transformation matrix among cameras by minimizing the reprojection error between the observed and predicted image points.

The minimization is typically carried out with the aid of the Levenberg-Marquardt (LM) algorithm. The lack of interaction among parameters for different 3D points and cameras results in the underlying normal equations exhibiting a sparse block structure. SBA exploits this sparseness by employing a tailored sparse variant of the LM algorithm that leads to considerable computational gains.

In Figure 3.3.7 are reported the reprojection errors in x and y direction in image plane before and after SBA optimization. After calibration, the mean reprojection error achieved in our setup was lower than 1 pixel.

Figure 3.3.8 shows the camera reference system of a classic setup GaMoCap of 11 cameras. In Figure the reference system chosen as world was computed in order to have the subject moving at origin proximity.

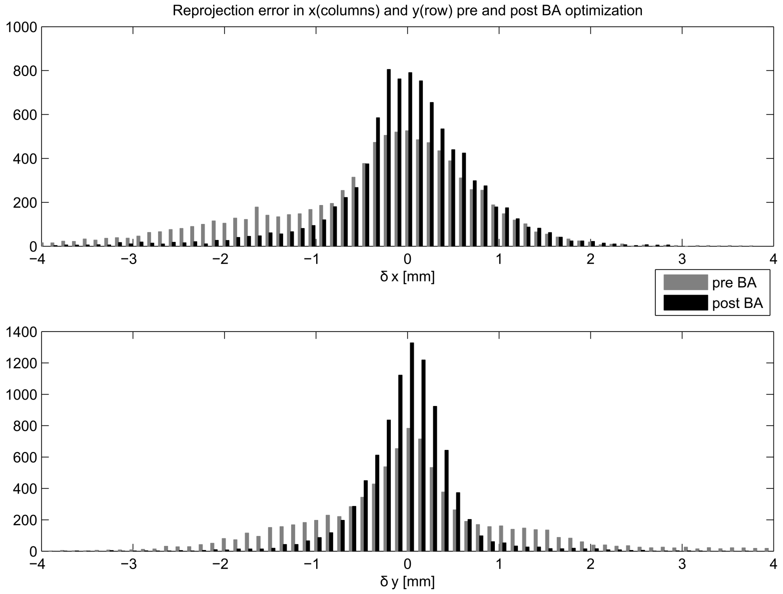


Figure 3.3.7: Reprojection errors before and after SBA optimization in x an y direction.

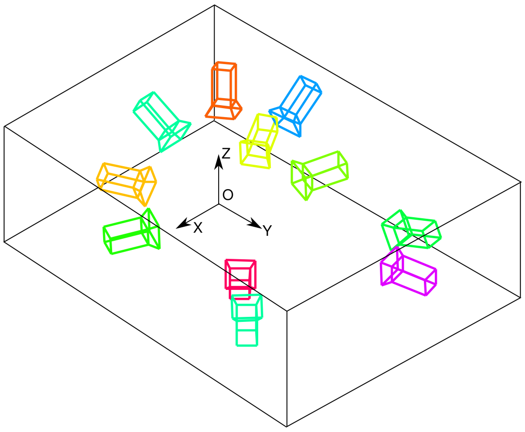


Figure 3.3.8: Reference frames of the 11 cameras respect world reference after calibration procedure.

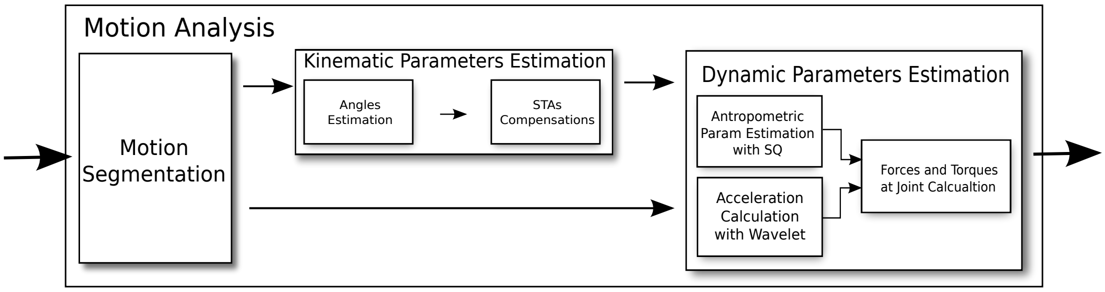

Motion Analysis
Motion Segmentation
Kinematic Parameters Estimation
Angles Estimation
STAs Compensations
Dynamic Parameters Estimation
Antropometric Param Estimation with SQ
Acceleration Calculation with Wavelet
Forces and Torques at Joint Calcualtion

Figure 3.4.1: Flow chart of Motion analysis step.

3.4 Motion analysis

In this Section are described the algorithms proposed to estimate the kinematic and dynamic parameters of motion.

Flow chart of Figure 3.4.1 shows the block relative to Motion analysis with an higher level of detail.

This Section is articulated in 3 Subsections: motion segmentation, kinematic parameters estimation and dynamic parameters estimation.

In Subsection 3.4.1 is addressed the problem to divide the cloud of 3d points reconstructed with GaMoCap in limbs by analysing how they move. Subsection 3.4.2 describes the kinematic parameters estimation and focus the attention on angle estimation at joints and the compensations of Soft Tissue Artefacts (STAs).

Subsection 3.4.3 describes the algorithm to estimate the forces and torques at joint with GaMoCap.

3.4.1 Motion segmentation

The analysis of the kinematics with GaMoCap is possible by elaborating the series of 3D point reconstructed while the subject moves. The first step in this procedure is to determine which point belong to each limb by analysing how it moves, this problem is called motion segmentation.

As pointed out in [100] among the algorithms for motion segmentation, the best performing is the one proposed in [110], called Local Subspace Affinity, or LSA. LSA is a general framework for motion segmentation exploiting the fact that different 3d point trajectories lay in different linear manifolds. The idea is to: estimate the local linear manifold for each trajectory and build an affinity matrix on some measure of the distance between each pair of manifolds. Once the affinity matrix is built a K-means-based clustering algorithm is applied to group the trajectories compatibles with different limbs.

The LSA algorithm can be divided into five main steps:

1. rank estimation of trajectories matrix
2. data transformation

3. subspace estimation

4. build of affinity matrix and clustering

Before starts with LSA it is necessary to reorder the data. The 3D clouds of points reconstructed in each time instant are thus re-organized in a trajectory matrix. This matrix has many columns as many points have been reconstructed and many rows as the time instants for which the motion was recorder. If some points are reconstructed only in few time instants are deleted.

For an acquisition of F time instants with P points reconstructed the resulting trajectory matrix is:

$$W = \begin{bmatrix} w_{11} & \dots & w_{1P} \\ \vdots & \ddots & \vdots \\ w_{F1} & \dots & w_{FP} \end{bmatrix} \tag{3.4.1}$$

Where $w_{ij} = [x_{ij}, y_{ij}, z_{ij}]$ is the 3D position of the j-th point at the i-th time instant.

Building this matrix with GaMoCap clouds is very easy since each point has at each time instant an unique ID, Subsection 3.1.3.

Rank estimation

The first step of LSA consist in the rank estimation of the trajectory matrix W. The rank of the matrix depend on how many different limbs compose the trajectory matrix and the constrains (joints) between them.

The algorithm consists in estimating the rank of W by using Model Selection techniques, [54]. The rank r is estimated as:

$$r = \text{argmin}_r \frac{\lambda_{r+1}^2}{\sum_{k=1}^{r} \lambda_k^2} + kr \tag{3.4.2}$$

Where λ_i is the the $i-th$ singular value of W, and k a parameter that should depend on the noise. The higher the noise level is, the larger k should be used.

Data transformation

As suggested in [110], given the trajectory matrix W and its estimated rank r it is possible to perform a data transformation. This data transformation provides a dimensionality reduction, a normalization of the data and a preparation for next step: the local subspace estimation. Basic idea is to consider each of columns of W as a vector in $\mathbb{R}^3$ and to project them onto the $\mathbb{R}^r$ unit sphere. An example of a 2D point trajectories reprojected in a $\mathbb{R}^3$ sphere is shown in Figure 3.4.2. In the Figure the white dots are trajectories which belong to one object while black dots are trajectories which belong to another object, which undergoes to different motions.

In order to perform this transformation, SVD is applied to the matrix $W_{3F \times P}$. Assuming its rank equal to k the matrix is decomposed as:

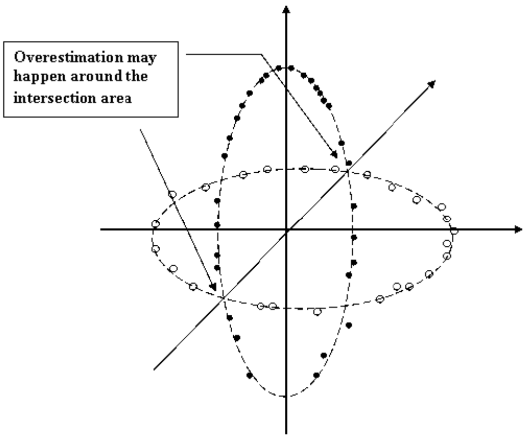


Figure 3.4.2: Example of data transformation and of trajectories that belong to two different subspaces of dimension 2 projected into an $\mathbb{R}^3$ unit sphere. The two circles are the 2 subspaces. The trajectories label are represents colour-coded: black dots are points with trajectories which belong to one motion while black dots are trajectories which belong to another motion.

$$W_{3F\times P} = U_{3F\times k} \cdot D_{k\times k} \cdot V_{k\times P}^T \tag{3.4.3}$$

Each column of $V(:, 1 : k)^T$ is then normalized to project it onto the unit sphere. Then is performed the segmentation on these vectors.

Subspace estimation

As can be noticed in Figure 3.4.2, the points lie on two different subspaces, the two different circles. It worth notice that the closest neighbours of a point lie on the same underlying subspace.

The idea is to estimate the subspaceα of a points by taking local samples form itself and its n neighbours. Tanks to the reprojection on the $\mathbb{R}^k$ sphere, the neighbour of each trajectories can be estimated using the $arccos(\alpha_1^T\alpha_2) \in [0, \pi]$. The value of the n neighbour to take into account must respect $n + 1 \geq d$, where d is the highest dimension of the linear subspace generated by the cluster. If the motion is rigid d is chosen ≤ 4 while in the case of an articulated motion $d \leq 7$. In our case we chose $d = 7$ and thus $n \geq 6$.

For each trajectory from W , the sub-matrix W_β containing only the trajectory β in analysis and its n nearest neighbours are extracted. The rank of W_β is estimated again using the rank selection. Knowing the rank ofW_β and using SVD the bases of the subspace $S(\beta)$ can be computed.

A problem is represented by the point in correspondence to the intersection between two subspaces. Due to noise not all the points that belong to the same body lie exactly

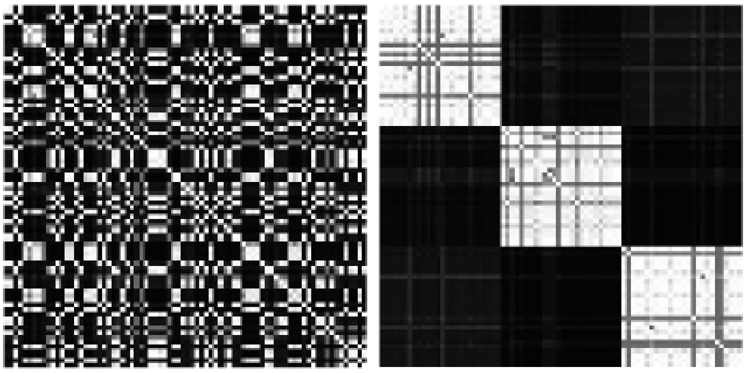


Figure 3.4.3: Left: Affinity matrix relative to trajectories of three clusters. Right: Affinity matrix rearranged after the spectral clustering.

on the same subspace. The sampled nearest neighbours could indeed belong either to "correct" subspace or to the others. This problem will be solved with the Point Cluster Technique, or PCT, Subsection 3.4.2.

Affinity matrix and clustering

Once estimated the subspace for each trajectory it is necessary to build a measure of similarity between trajectories subspaces, these values fill an Affinity matrix $A_{P \times P}$.

The affinity of two trajectories is based on the principal angle, proposed by , and calculated as:

$$A(\alpha_{1,}\alpha_2) = \exp(-\sum_{i=1}^{M} \sin(\theta_i)^2) \tag{3.4.4}$$

Where M is the minimum dimension between the basis $S(\alpha_1)$ and $S(\alpha_2)$.

Figure-Left 3.4.3 shows an Affinity matrix relative to the trajectories of three clusters. Brighter elements corresponds to trajectories with high affinity (the diagonal is equal to 1), darker elements with low affinity.

Once the Affinity matrix is calculated is possible to cluster together trajectories with high subspace affinity with the algorithm described by [90].

Figure-Right 3.4.3 shows the affinity matrix after clusterization.

The trajectories of each point in the same cluster belong to the same limb, with this consideration is possible to estimate the relative rotation between limbs, Subsection 3.4.2, and the torques at joint, Subsection 3.4.3. An example of motion segmentation is shown in Figure 4.2.3. It can be notice that the points at joint proximity are badly segmented.

This problem is due to the noise introduced by Soft Tissue Artefact. The marker located next to the joint have indeed trajectories compatibles with both limbs, i.e. affinity of trajectories to two cluster. The this problem is addressed in next Subsection.

3.4.2 Kinematic parameters estimation

In this Subsection is described the algorithm used to calculate the angle of relative rotation between two limbs and the artefacts caused by soft tissues. We propose also an algorithm to compensate the Soft Tissue Artefacts (STAs) based on Point Cluster technique.

Joint position is computed with the algorithm proposed by [33].

Angle estimation

Once the cloud of points is segmented in different limbs is possible to calculate the relative rotation of two limbs as the composition of rotation of each limb respect the initial time instant.

The rotation of a limb at each time instant is computed with a SVD-based technique, as described in [4].

Suppose that p_i is a $M \times 3$ matrix that contains the coordinates of the M points relative to a body at time instant i. Suppose also p_{i+1} the coordinate of points at time $i+1$. In the case of a rigid motion these two points sets are related by the equation:

$$p_{i+1} =^i R \cdot p_i + t_{i+1} + n_i \tag{3.4.5}$$

where R_i is a 3×3 rotation matrix, t_i the 3×1 translation vector, and n_i the noise vector.

In [4] is described an algorithm to find the least-squares solution of ${}^iR$ and ${}^it$, based on the singular value decomposition (SVD) of the cross correlation matrix C computed as:

$$C = \widetilde{p}_i \cdot \widetilde{p}_{i+1}^T \tag{3.4.6}$$

where $\widetilde{p}_i = p_i - \bar{p}_i$, $\widetilde{p}_{i+1} = p_{i+1} - \bar{p}_{i+1}$ and $\bar{p}_i$ are $\bar{p}_{i+1}$ respectively centroids of the point sets p_i and p_{i+1}. Decomposing the C matrix with SVD, $C = U\Sigma V^T$, rotation matrix ${}^iR$ can be computed as ${}^iR = UV^T$.

With this technique the rotation of the cloud of points in at each time instant is calculated respect the position of the limb at time $i = 0$. For this reason the rotation matrix ${}^iR$ at time $i = 0$ is an 3×3 identity matrix since the rotation is 0.

Given two limbs A and B, relative rotation between them, at time instant i is calculated by composing the direct cosine rotation matrix ${}^iR^A$ of the limb A with the inverse of the other limb B, ${}^iR^B$, Equation 3.4.7.

$${}^iR_A^B =^i R_A \cdot inv({}^iR_B) \tag{3.4.7}$$

Also in this case the rotation matrix between two limbs at time $i = 0$ is an identity matrix.

In order to obtain a numeric value of the angle of rotation between the two limbs the rotation matrix is expressed with the Rodrigues' rotation vector representation.

In this representation the direction of this vector defines the axis of rotation and the magnitude is the value of the angle.

The angular velocity can be estimated from angle trajectories applying a derivative operation. This operation can be approximated by the ratio of the finite difference in joint angles and the video frame rate as in Equation 3.4.8:

$$v = \frac{d\theta}{dt} \approx \frac{\Delta\theta}{T_s} \tag{3.4.8}$$

where v is the angular velocity, $\Delta\theta$ is the difference of the angle measure between two time instants and T_s is the sampling time at which the motion is captured by cameras.

The main limitation with this angle estimation technique is the rigid motion assumption. The presence of points that moves with a non rigid motion introduces an error of the estimation of the angle between joints.

The main cause of non rigid motion of points is imputable to Soft Tissue Artefact and its effect on the angle estimation procedure is analysed in next Section.

Soft tissue artefact

The term Soft Tissue Artefact, according to [17], refers to the experimental artefact associated with the relative movement between markers and underlying bone. In this work the term STAs will be used also to refer to the artefact introduced by the relative movement of the dotted garment and the underlying skin.

The STA affects the LSA motion segmentation, Subsection 3.4.1, and consequently angles estimation procedure.

The segmentation is affected by STAs since it is based on the hypothesis of rigid motion. The points at joint proximity of two different limbs, corrupted by the relative movement of garment respect the underlying bone, can be wrongly segmented. This non rigid motion causes indeed an high affinity of the trajectories of these points with both limbs that results in a bad trajectory segmentation.

Figure 4.2.3 shows an example for which the points at hip proximity at intersection of torso and the leg are mixed each other. A bad segmentation of the cloud of points results in a bad estimation of the relative rotation with the SVD-based technique described in previous section.

We show the effect of a bad segmented points in angle estimation procedure on simulate acquisition. In this acquisition we created two sets of points relative to two limbs, called A and B. Points relative to limb B are rotated with respect to the point O with an angle from 0° to -90° degrees, while points A were kept fixed.

These data were used in two trials. In trial 1 the sets of points were considered as perfectly segmented: points relative to the first limb were labelled as A and points relative to second limb as B, Figure 3.4.4-Top. The angle of relative rotation between B and A was calculated with the SVD technique described. In trial 2 was introduced a segmentation error assigning to two points relative to limb B at joint proximity the label A, 3.4.4-Bottom. After this step the angle of relative rotation between B and A rotation was estimated.

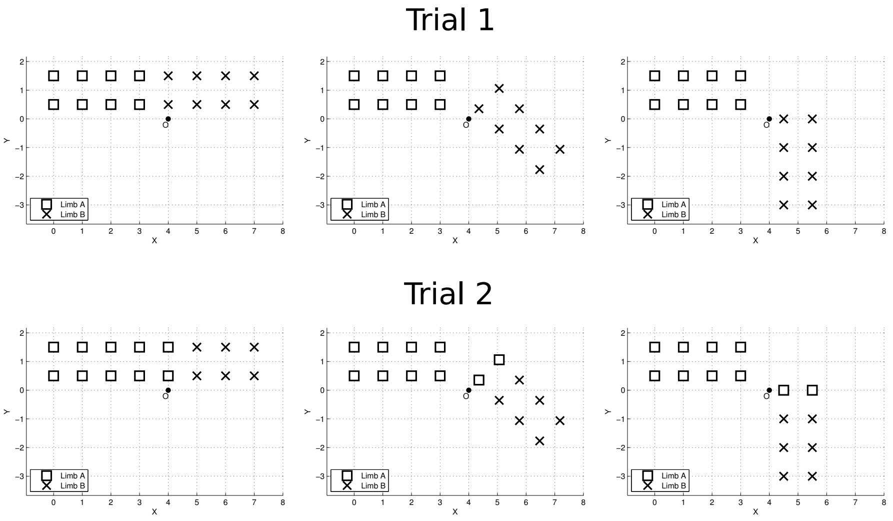


Figure 3.4.4: The bad segmentation effects on angle estimation. Top: Trial 1 perfect points segmentation. Bottom: Trial 2 wrong segmentation of point at joint proximity.

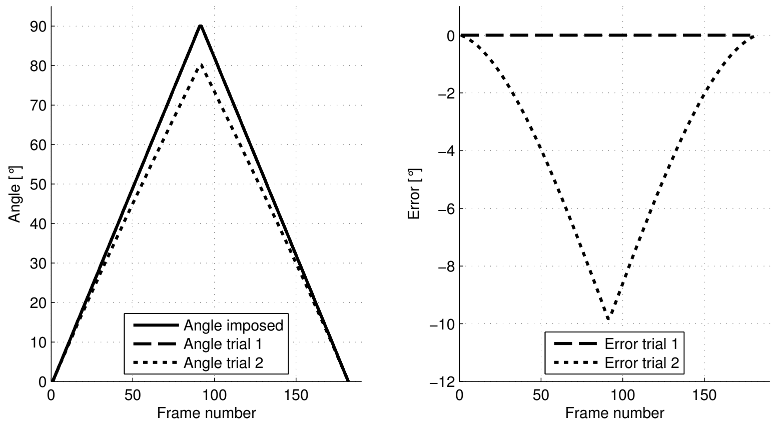


Figure 3.4.5: Bad segmentation effects on angle estimation. Angle estimated in trial 1 and 2 versus the angle imposed. Error is calculated as the difference between the angle of the trial and the angle imposed.

It is noticeable that a perfect segmentation in trial 1 leads to a correct estimation of the angle, instead a wrong segmentation of points in trial 2 reduce the angle estimated of a maximum of about 10%.

Soft Tissue Artefacts compensation

Soft Tissue Artefacts compensation in angles estimation is based on an automatic identification and elimination of points that moves with non rigid motion. The algorithm realized is based on the work of [3]. The technique presented, called Point Cluster Technique or PCT, is particularly suitable to be used with GaMoCap points clouds. One of PCT assumption is to use a cluster of points uniformly distributed on the limb segment. While an high density of marker with traditional mocap is a problem, Section 2.1 the PCT is directly implementable with GaMoCap.

The main idea of PCT is to assign a weight factor to each point and optimize it in order to minimize the inter-frame variation of the inertia tensor of cloud points.

For each cluster of points relative to one limb the centre of mass at time i is calculated as:

$$C^i = \frac{\sum_{j=1}^{M} p_j^i \cdot m_i^j}{\sum_{j=1}^{M} m_i^j} \tag{3.4.9}$$

where M is the total number of points relative to the cluster in analysis, p_j^i is the vector of coordinates of point j at time i and m_j^i is the weight assigned to that point at time i.

The centre of mass is calculated to translate the points at centre of mass of the limb in analysis: $q^i = p^i - C^i$.

The matrix q^i can be seen a the rotation R^i at the time i of the points v^i defined respect the local reference system of the limb: $q^i = R^i \cdot v^i$.

The inertial tensor I^i of the cluster of points q^i at time i is defined as:

$$I^i = \begin{bmatrix} \sum_{j=1}^{M}[(q_{j,y})^2 + (q_{j,z})^2] \cdot m_j^i & \sum q_{j,x} \cdot q_{j,y} \cdot [-m_j^i] & \sum q_{j,x} \cdot q_{j,z} \cdot [-m_j^i] \\ \sum q_{j,x} \cdot q_{j,y} \cdot [-m_j^i] & \sum_{j=1}^{M}[(q_{j,x})^2 + (q_{j,z})^2] \cdot m_j^i & \sum q_{j,z} \cdot q_{j,y} \cdot [-m_j^i] \\ \sum q_{j,x} \cdot q_{j,yz} \cdot [-m_j^i] & \sum q_{j,z} \cdot q_{j,y} \cdot [-m_j^i] & \sum_{j=1}^{M}[(q_{j,x})^2 + (q_{j,y})^2] \cdot m_j^i \end{bmatrix} \tag{3.4.10}$$

The eigenvector of $I^i$are used to estimate the rotation matrix R^i as:

$$R^i = [E_1^i, E_2^i, E_3^i] \tag{3.4.11}$$

where the eigenvector E_g^i have the form:

$$E_g^i = \begin{bmatrix} e_{g,x}^i \\ e_{g,y}^i \\ e_{g,z}^i \end{bmatrix} \quad g = 1, 2, 3 \tag{3.4.12}$$

Associate with each eigenvector there is an eigenvalue λ_g^i, $g = 1, 2, 3$. If non-rigid body movement occurs, the eigenvalues will change their value during movement.

The algorithm developed minimize the eigenvalue changes by redistributing the weight factor m_j^i at each time step. The function to minimize at each time step:

$$F^i = (\Lambda^i - \Lambda^0) \tag{3.4.13}$$

Where $\Lambda^i = \sqrt{(\lambda_1^i)^2 + (\lambda_2^i)^2 + (\lambda_3^i)^2}$ and $\Lambda^0 = \sqrt{(\lambda_1^0)^2 + (\lambda_2^0)^2 + (\lambda_3^0)^2}$. Λ^0 is the sum of the square of the eigenvalues at time instant chosen as reference. Since the function F^i has M unknowns m_j^i (one for each points), the problem was reformulated in order to minimize only one parameter, called ϵ.

This parameter is calculated such that the point with the largest displacement from the rest position, v^0 in the local cluster coordinate system, was assigned the lowest weighting factor:

$$m_j^i = \epsilon^i - \frac{\Delta v_j^i}{\max(\Delta v^i)} \tag{3.4.14}$$

where $\Delta v = v^i - v^0$ represents the displacement of each point respect the time instant chose as reference.

F^i is thus function of only the parameters ϵ and it is minimized with a Levenberg-Marquardt technique. This minimization produce for each time instant a vector of weight m^i.

These vector $m^{i=1\ldots n}$ are used to filter out the points considered affected by STAs by setting a threshold. The threshold is calculated separately for each time instant and is set to 50% of the mean of the weights of the points calculated for that time instant. A point is considered affected by STA and thus disregarded if its weight is under-threshold.

This procedure aims to reduce the effects of STA eliminating the contribution of points behaving weakly like part of rigid bodies. The points that move from the reference frame with a non-rigid transformation will indeed receive a low weight factor.

The results of this filtering operation can be seen in Figure 4.2.3, 4.2.6 and 4.2.9.

3.4.3 Dynamic parameters estimation

In this Subsection are described the algorithms to estimate the forces and torques at joints for a 2D planar motion. The solution of the equations of a free diagram body, 2.2.1, requires the knowledge of anthropometric parameters as length, mass and inertia of the limbs in analysis. We propose here a method to compute these parameters relying on Superquadrics.

Estimation of the forces and torques at joints require also the accelerations (linear and rotational) of the limbs.

Since the GaMoCap measure the displacement of the limbs we calculate accelerations from displacements relying on Wavelets property reported in Equation 2.2.15.

Anthropometric parameters estimation with Superquadrics

Anthropometric parameters estimation with Superquadrics, or briefly SQ, here proposed is an algorithm articulated in three steps:

- calculation of shape parameters $(a_1, a_2, a_3, \varepsilon_1, \varepsilon_2)$ of the SQ

- calculation of the homogeneous transformation of the SQ respect the world reference system
- calculation of volume and moment of inertia of the SQ

Respect [108] we do not have to segment the cloud of 3D point since the segmentation has been already performed on the basis of the motion of the subject in analysis, Subsection 3.4.1.

The three steps of the algorithm are performed on each limb separately. Before the calculation of shape parameters $(a_1, a_2, a_3, \varepsilon_1, \varepsilon_2)$ of the SQ it is necessary a pre-processing operation to centre the 3D points relative to the limb in [0,0,0] and align respect to the principal inertial axis of the cloud. This because the SQ implicit formula, Equation 2.2.4, is defined respect a SQ-reference system centred in the origin.

The cost function used to compute the shape parameters of the SQ is the same proposed by [52]. The cost function is the follow:

$$\min_{\Lambda} \sum_{i=1}^{n} \left(\sqrt{a_1 a_2 a_3}\left(F^{\varepsilon_1}\left(x_i, y_i, z_i, \Lambda\right) - 1\right)\right)^2 \tag{3.4.15}$$

where Λ represents the 5 parameters of shape $(a_1, a_2, a_3, \varepsilon_1, \varepsilon_2)$ and F it the implicit formulation of the SQ, Equation 2.2.4, n are the number of 3D points of the limbs in analysis and x_i, y_i, z_i are the coordinate of the$i$-th point in the local reference system.

In order to help the convergence of the solution the parameters a_1, a_2, a_3 are initialized respectively with the maximum distance of the points in the x, y and z direction. The parameters of ε_1 and ε_2 are initialized to 1.

Once the parameters of the shape of SQ have been calculated it is possible to recover the position and orientation of the SQ respect the world reference system. We decided to use an Homogeneous transformation notation to express the transformation of a 3D point expressed in the SQ reference system P_{SQ} in the world coordinate system P_w:

$$P_w = HP_{SQ}$$

The H matrix is composed as follows:

$$H = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}$$

where R and T are respectively rotation and translation of the SQ reference system respect the world reference system. To express the rotation matrix R we used the Euler angles (ϕ, θ, ψ) which define the orientation in therm of composition of rotation of ϕ around z axis, whit a rotation of θ around the new y axis, with a rotation of ψ about the new z axis.

The cost function used is the same of the shape parameters optimization, Equation 3.4.15, but the minimization is performed acting on the parameters ϕ, θ, ψ and the t_x, t_y, t_z, elements of translation vector$T$.

The cost function is the follow:

$$\min_{\Gamma} \sum_{i=1}^{n} \left(\sqrt{a_1 a_2 a_3}\left(F^{\varepsilon_1}\left(x_i^w, y_i^w, z_i^w, \Gamma\right) - 1\right)\right)^2 \tag{3.4.16}$$

Where: Γ represents the 6 parameters that model the homogeneous transformation $(\phi, \theta, \psi, t_x, t_y, t_z)$, F it the implicit formulation of the SQ, and x_i^w, y_i^w, z_i^w are the coordinate of the$i$-th point in the world reference system.

In order to deal with eventual outliers both minimization function were used in a statistical framework based on RANSAC, [36].

Figure 4.3.3 shows the result of SQ fitting on the points relative to the right leg of the subject.

Once the shape parameters of the SQ and the orientation and position respect the world reference system it is possible to calculate the length of the limb and the distance of the barycentre of the SQ from the rotational axis. We suppose that the barycentre of the limb and of the SQ are coincident. The length of the limbs is calculated as $2 \cdot a_3$, where a_3 is the parameter relative to the superquadric in the direction with maximum length.

With the Equations 2.2.8 and 2.2.12 it is possible to calculate the volume and the inertia of the limb. With the assumption of a uniform limb density it is also possible to have an estimate of the mass and inertia of the limb. In Section 4.3 are reported the results of a real test.

3.5 Uncertainty estimation and Vicon comparison

Uncertainty estimation of the angle measured and comparison with state-of-the-art system are two essential steps to check the goodness of the results obtained.

In this Chapter is described in detail the procedure that was used to numerically quantify the uncertainty of the angle estimated by means of the Monte Carlo Method (MCM) and the procedure of comparison of results of GaMoCap and Vicon.

3.5.1 Uncertainty propagation with Monte Carlo

The evaluation of the uncertainty of the angle estimated is achievable through an uncertainty propagation stage, according to the GUM standard [9].

The main stages for the procedure for uncertainty propagation are:

1. *Formulation*:
   - define the output quantity *Y,* the quantity intended to be measured (the measurand);
   - determine the N input quantities $\boldsymbol{X} = (X_1, ..., X_N)^T$ upon which Y depends;
   - develop a model relating Y *and* $\boldsymbol{X}$;
   - on the basis of available knowledge assignment of PDFs—Gaussian (normal), rectangular (uniform), etc. to the X_i. Assign a joint PDF to those input variables X_i that are not independent.
2. *Propagation*: propagate the PDFs for theX_i through the model to obtain the PDF for *Y.*
3. *Summarizing*: use the propagated PDF for Y to obtain:
   - the expectation of *Y;*
   - the standard deviation of *Y;*
   - (optional) the coverage interval containing Y with a specified probability;

Crucial element is the *Propagation* stage, that can be implemented in several ways. The two mainly used are:

- uncertainty propagation based on replacing an explicit model by its first-order Taylor series approximation — the law of propagation of uncertainty [30 5.1.2];
- numerical methods [30 G.1.5] that implement the propagation of distributions, specifically using Monte Carlo Method (MCM).

In practice, only for simple cases, where an explicit and close to linearity model is available, the law of propagation of uncertainty can be applied. The problem of uncertainty estimation of 3D point reconstruction and of motion estimation was analysed in [35].

In our case due to the non-linearities introduced by the triangulation algorithm, Subsection 3.2, by the motion segmentation algorithm, Subsection3.4.1, and the procedural complexity of the model MCM was chosen.

Formulation phase is the step in which is computed the model that relates the inputs and the outputs. This step, for angle at joint calculation, has been already described in Subsections 3.4.1 and 3.4.2

Monte Carlo Method

According to the GUM the MCM can be used as an implementation of the propagation of distributions. The basic idea of the MCM is very simple, yet powerful: repeated sampling from the PDFs for the input quantities X_i and the evaluation of the model in each case. Expectations and variance of the output quantity, Y, can be determined directly from the set of model values obtained.

As shown in the graphical representation of the MCM, Figure 3.5.1, the method can be stated in a step-by-step procedure:

1. selection of number M of trial Monte Carlo to be made;
2. generate M vectors, by sampling from the assigned PDFs, as realizations of the (set of N) input quantities X_i;
3. for each such vector, form the corresponding model value of Y , yielding M model values;
4. sort these M model values into strictly increasing order, using the sorted model values to provide $\boldsymbol{G}$, a discrete representation of the distribution for the output quantity Y;
5. use $\boldsymbol{G}$ to form the expectation of Y and the standard uncertainty *u(y)*;

MCM implementation

Figure 3.5.2 shows the block diagram of the procedure for angle estimation of GaMoCap and represent the relation model between the inputs and the output of the system.

The two input variables, represented in Figure 3.5.2 as triangles, are coordinates of the centroid of the marker extracted from the image and the intrinsic and extrinsic calibration parameters.

Coordinates of markers extracted from images influence the triangulation process, 3D point cloud reconstruction and consequently the evaluation of the angle between the limbs.

The assignment of PDF to marker extracted is a non-trivial task. It requires a complex study on the variables that influence the image formation and marker extraction (image digitalization, illumination, image focus, marker inclination respect plane image, colour of the marker, ...). Instead of doing this we decided to use the centroid of the markers

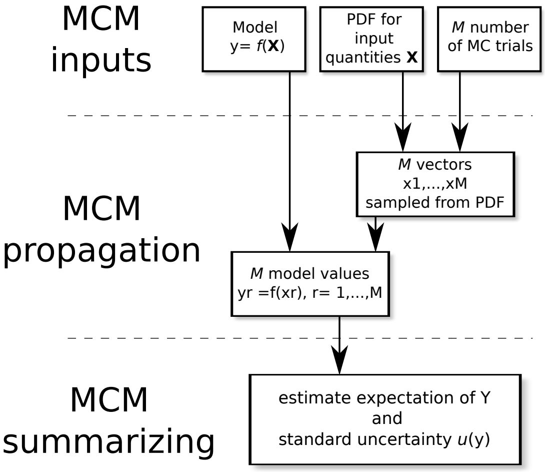

MCM inputs
Model y= f(X)
PDF for input quantities X
M number of MC trials
MCM propagation
M vectors x1,...,xM sampled from PDF
M model values yr =f(xr), r= 1,...,M
MCM summarizing
estimate expectation of Y and standard uncertainty u(y)

Figure 3.5.1: The propagation and summarizing stages of uncertainty evaluation using MCM to implement the propagation of distributions.

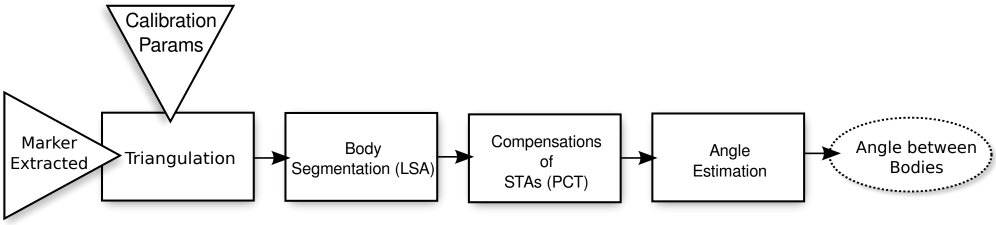


Figure 3.5.2: Block diagram of the procedure for angle estimation. The input variables are represented in triangles , the output variable in ellipse and the functional blocks in rectangles.

extracted from the images directly as a sample from their PDF. The number of markers extracted from each image (~1000), multiplied by the number of cameras (8) and by the number of frames for each motion analysed (100), generates a number of samples ($8 \cdot 10^5$) that provides a good statistics of this influencing factor.

Second input variables are the intrinsic and extrinsic calibration parameters obtained by the multicamera calibration, as described in Section 3.3. Intrinsic and extrinsic camera influence the position of the markers extracted (through distortion coefficients) and the triangulation results (projection matrix).

The marker ID assignation is not represented in Figure 3.5.2 and it is not considered as an input variable of the process because the color recognition process is considered not to be affected by error.

The output variable corresponds to the measured quantities, i.e. the angle between the limbs.

The selection of the trials number M is chosen equal to 100 after a convergence study.

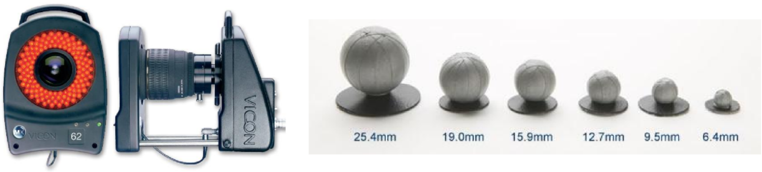


Figure 3.5.3: Left: Vicon camera with active IR illumination. Right: Spherical retro-reflective markers with different sized.

The first step of MCM propagation stage consists in the generation of M vectors, $(x_1, ..., x_m)$, that contains the sample from the PDF of the input variables.

In this step a hybrid procedure is adopted, in which synthetic, obtained by PDF sampling, and real data, obtained directly by image elaboration, are used. The use of real data is necessary to overcome the PDF generation process for marker extraction. The calibration parameters sampling, due to variables correlation, is performed from the covariance matrix obtained by calibration algorithm [12].

The second step of propagation consists in the model evaluation for 100 times with the input data previously generated. The model that we employed for angle calculation is the one proposed in Subsections3.4.1 and 3.4.2.

As last step, the 100 output values (angle between limbs) obtained numerically through the model are used to estimate the mean and uncertainty of the angles, with a level of confidence set at 65%. The uncertainty range corresponding to this confidence level is obtained according to GUM as the minimum.

Results of this analysis are reported in Section 4.2.

3.5.2 Comparison with Vicon

In order to evaluate the results obtained with the GaMoCap system, estimate angles were compared to the commercial MoCap system Vicon (http://www.vicon.com/).

Vicon is an optoelectronic MoCap based on triangulation of passive retro-reflective markers and active IR cameras, Figure 3.5.3.

Performances of accuracy reconstruction of the Vicon system were extensively studied during the years [31, 106] confirming an high accuracy and repeatability of the results. Therefore the Vicon system is widely used to analyse and measure human motion of upper [1, 19] and lower limbs [81, 11].

The procedure to acquire the motion with Vicon can be divided in four steps: system setup, subject preparation, motion acquisition, post-processing.

The system setup consists in the placing of the cameras to cover the capturing space and their calibration. The Vicon calibration is the procedure to estimate the intrinsic parameters of the Vicon cameras and their relative positions. This operation is accomplished moving a calibration wand with three markers a well defined distances, Figure 3.5.4, in front of the cameras. Once the calibration software has computed the intrinsic and extrinsic parameters it is possible to set a common reference frame for all the

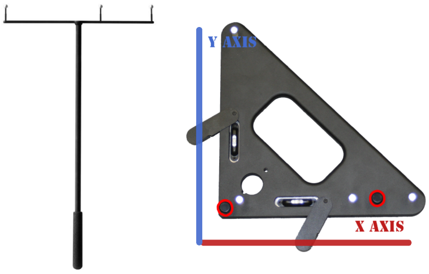


Figure 3.5.4: Calibration tools of Vicon system. Left: calibration wand with 3 retro-reflective markers. Right: “L-frame” used to set the *world* reference system.

cameras, called *world*. This is achievable capturing a frame in which is visible the “L frame”, Figure 3.5.4.

Since Vicon and Gamocap are used to acquire the same movements in order to compare the results a common reference frame was defined. This was achieved attaching 6 Vicon markers to the Gamocap calibration plane and capturing a shot of the scene, Figure 3.5.5, with both systems. With this simultaneous capture is possible to define in the two system the same reference plane.

The subject preparation is a delicate phase because the marker positions determine the accuracy of the results, [42], in particular, the larger part of the error is associated with the relative movement between externally located markers and the underlying bone, due to the interposition of both passive and active soft tissues, [16].

The markers positions on the subject were chosen in order to reduce STAs and maximize visibility to the cameras. The maximum number of markers for the Vicon system is limited by the problem of markers swapping. This problem occurs when performing marker tracking. If two markers are placed very near one to each other it is possible that their label are interchanged leading to a trajectories swapping. This problem can be solved ans require the user intervention that manually reassign the correct label to each marker. This procedure can be time expensive for long acquisitions.

Since the comparison of GaMoCap and Vicon requires the simultaneous use of the Vicon marker and the GaMoCap suit, the bony landmarks, [7], are not directly accessible. These are indeed covered by the suit of GaMoCap. This problem was overcome placing the markers in clusters of 3 for each limb in analysis. Their position on the limb is central in order to avoid the effect of the stretches of the garment, which is more strong at joint proximity.

A cluster disposition of markers, as pointed out in [16] shows several advantages: easier marker placing on patients, optimal selection of cluster location so as to minimise STAs, reduction number of cameras needed for data collection. Figures that show marker positions for the various movements analysed are reported in Section 4.2.

After subject preparation is possible to start the acquisition of the motion. The capturing frequency of Vicon cameras was set to 100Hz. In post-processing takes place

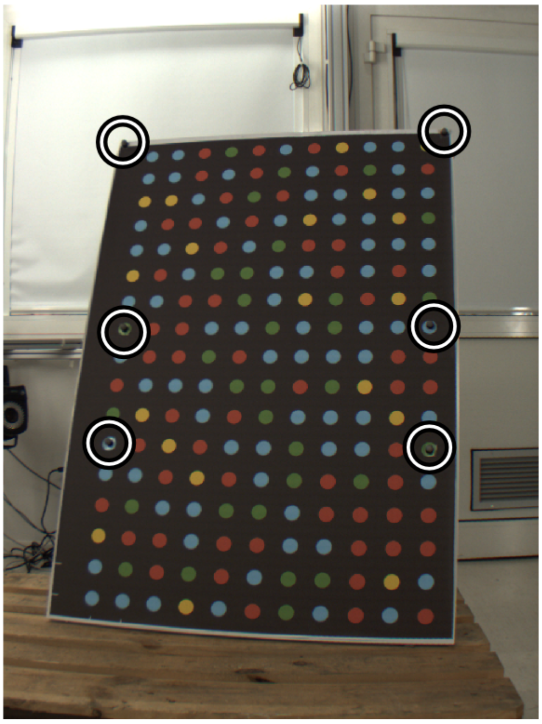


Figure 3.5.5: Definition of a common reference system for GaMoCap and Vicon. In circles are marked the 6 Vicon markers.

the marker labelling process. The operator must label the marker and check that no swapping error have occurred.

The angle at joints is computed as reciprocal rotation of the cluster of markers relative to two limbs in analysis, Subsection 3.4.2.

The comparison of the angle estimated with GaMoCap and Vicon are reported in detail in Section 4.2 for each motion analysed.

4 Experimental results

In order to test the proposed algorithms we developed a prototype of the multicamera system GaMoCap. In this Chapter are reported the hardware setup of the GaMoCap and the results of uncertainty estimation with Monte Carlo Method and the Vicon comparison. In the last section the algorithms proposed for kinematic and dynamic analysis are tested in a real experiment.

4.1 Hardware setup

This Section describes the hardware chosen and the layout of the GaMoCap. Two words describe the design of GaMoCap prototype and the hardware chosen: *flexibility* and *scalability*. Flexibility since the hardware chosen must not constrain the elaboration process. For this reason in the realization of the prototype no special hardware was chosen. The cameras are indeed standard industrial cameras and the computers are personal computers.

The scalability was chosen in order to assure the functionality of the system with 2 up as many cameras available and to allow either a full or a single limb motion analysis with the same system.

Due to the high amount of data to manage, in order to maintain the principle of *scalability* of the GaMoCap, instead of use a single "big elaboration", rely on many small PC, called *secondary*, controlled by one central unit, called *main*.

Each couple of camera is connected to one *secondary* PC. Role of the secondary PCs is to acquire the images directly on RAM, transfer over LAN to *main* PC maintaining a copy of the images on the internal hard disk for redundancy, Figure 4.1.1.

The *main* PC controls also the triggering device that synchronizes cameras.

Figure 4.1.1 shows the configuration of the hardware setup employed in GaMoCap.

Figure 4.1.2 shows two photos of the system realized at University of Trento.

Cameras

Multicamera setup of the GaMoCap is composed by two different type of cameras:

- 8 triggerable USB PointGrey Chameleon with a resolution of 1280x960 pixels
- 4 cameras Matrix Vision (MV) mvBlueCOUGAR-X 124C (GigE , 1600x1200, C-mount, 1/1.8')

PointGrey Chameleon mount standard C-mount lenses with focal length 6mm and f-number 1.4. Matrix Vision Cougar mount CS-mount lenses with focal length 8mm.

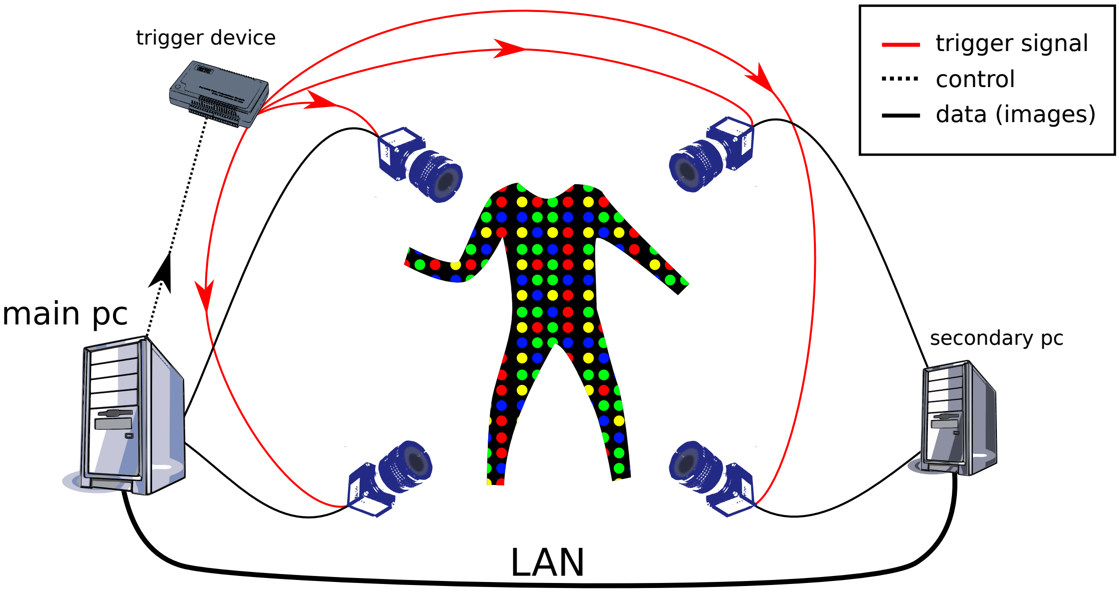


Figure 4.1.1: Hardware setup scheme of the GaMoCap system

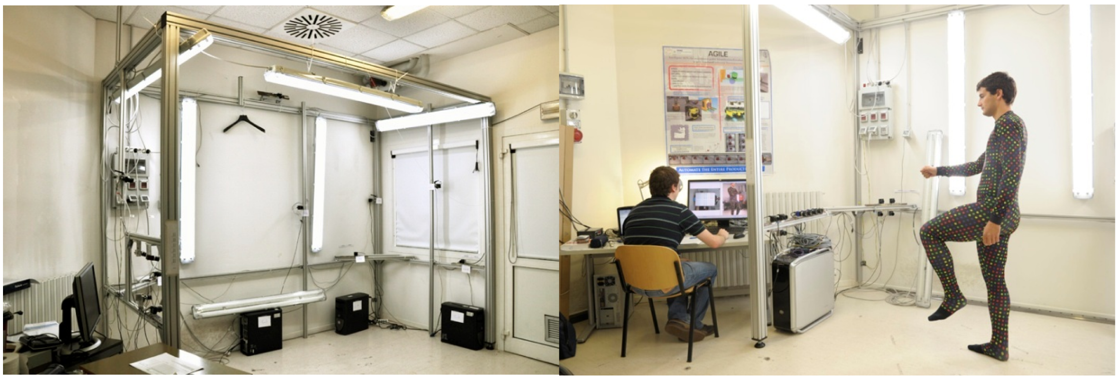


Figure 4.1.2: Photos of the GaMoCap system realized at University of Trento.

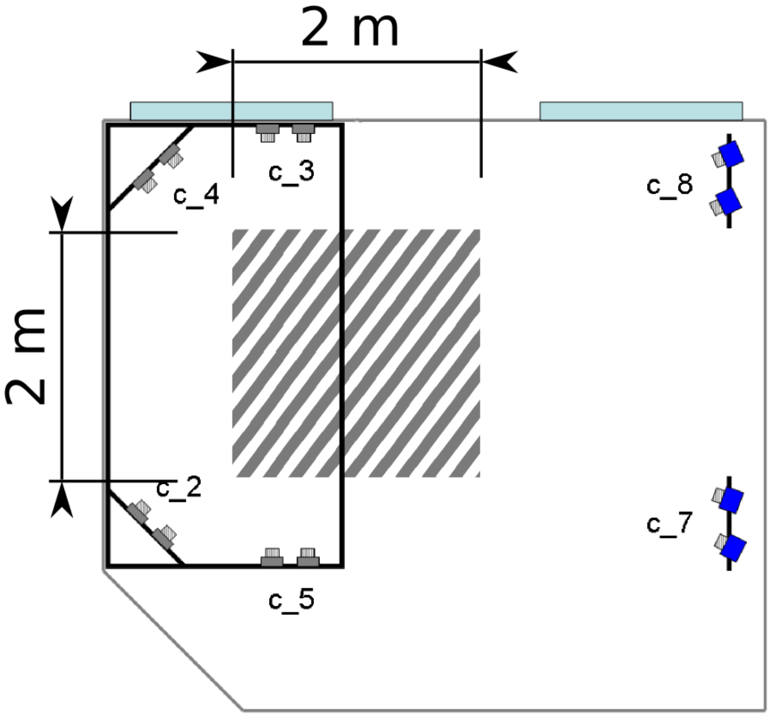


Figure 4.1.3: Camera disposition of GaMoCap to define a working area of $4m^2$

Their disposition was chosen in order to have a working area of $4m^2$, Figure 4.1.3. The MV cameras were chosen because can be placed an higher distance respect the other thanks to the higher resolution.

Cameras synchronization is assured with the device USB-4751L Advantech which generates a TTL square signal with the desired frequency. The duty cycle of the pulse given by the trigger device is used to set the exposure time of the cameras.

Due to the acquisition frequency (10Hz) to avoid loss of data, images captured are written directly into the RAM of computer and then transferred to hard disk. This element limits the acquisition time to the maximum quantity of RAM available on the PC. In this setup the *secondary* PCs, with an amount of 2 GB of RAM, allow to acquire motions of about 10 seconds. While this acquisition time seems quite short was adequate for the purposes of this work.

Illumination

Due to the number of colours to recognize (4) and the high frequency of acquisition (10 Hz) the illumination was chosen carefully.

The fact that the feature description of GaMoCap is based on color, Subsection 3.1.3, impose a constrain on the light condition. In order to have a good color recognition is necessary to have stable lighting conditions.

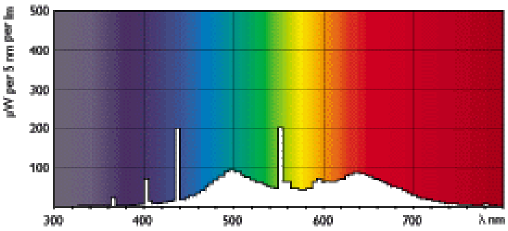


Figure 4.1.4: Spectrum of emission of Philips MASTER TL-D 90 Graphica 58W/965

The use of fluorescent lamps with common ballast is not suitable for the application, since they produce a flicker in illumination at same frequency of power supply at which they operates, 50/60 Hz. This flicker introduces a sensible change in the illumination that affect the color recognition process.

Instead of common ballast where chosen electronic ballast which supply power to the lamp at a frequency of 10,000 Hz, thus removing the flickering effect.

To assure a spectrum of emission uniform in all the field of visible were chosen the fluoresces lamps of Philips MASTER TL-D 90 Graphica 58W/965, Figure 4.1.4. These lamps allow a high colour rendering and are compliant with European Standard EN 12464-1.

Suits

A full body suit was realized to test the reconstruction and motion analysis capabilities of the GaMoCap. The suit has a total of 2933 markers uniformly distributed on the body. The garment chose is Spandex for its elastic property. An essential requirement is in fact that the suit is thigh, in order to follow the human body shape. We chosen a black garment to assure good contrast and detectability of the markers.

Figure 4.1.5 shows the suit realized.

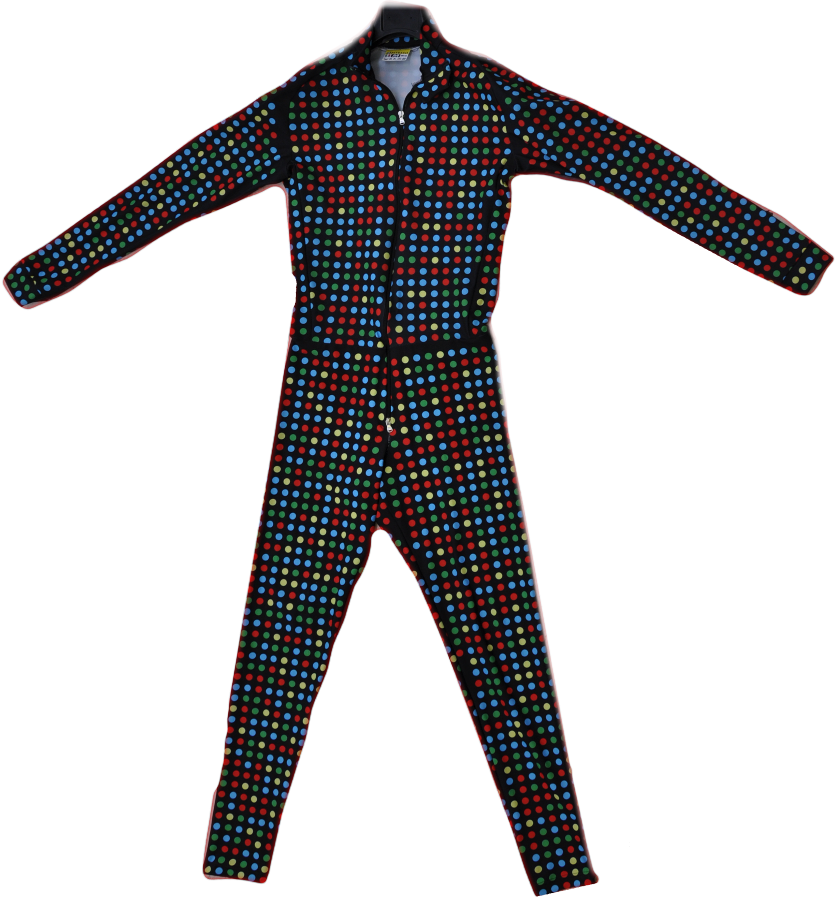


Figure 4.1.5: Suit employed in GaMoCap acquisitions.

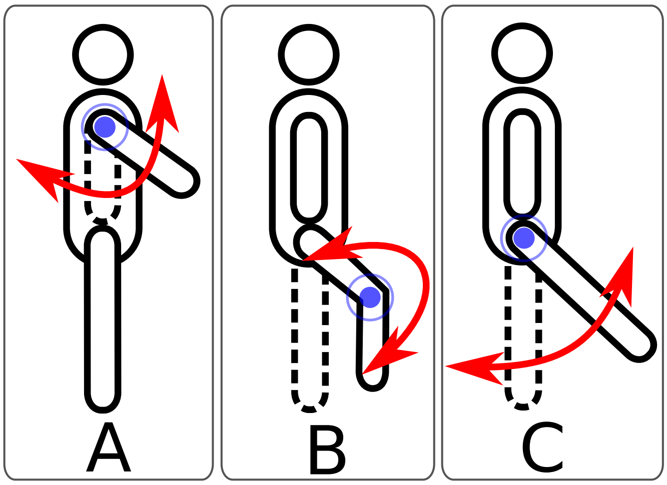


Figure 4.2.1: Motions analysed for uncertainty analysis angle estimation comparison. (A) shoulder flexion/extension, (B) knee flexion/extension, (C) hip flexion/extension

4.2 Uncertainty analysis and Vicon Comparison

In this Section are reported the results of: uncertainty analysis with MCM, Subsection 3.5.1, angle estimation with and without STAs compensation, Subsection 3.4.2, and comparison with Vicon, Subsection 3.5.2.

The uncertainty analysis, angle estimation and the comparison are preformed on 3 different motions. Next Subsection describes the motions analysed.

In this Section is also reported the markers disposition chosen for Vicon setup.

Motions analysed

The comparison of results are performed on three different motions: one relative to the shoulder, one to the knee, one to the hip. We will refer to these motions respectively as A, B and C.

Motion A consists in the shoulder flexion/extension. The subject, starting with arm relaxed along the torso, raises the extended arm in sagittal path until the maximal extension and then returns to starting position. Motion B consists in a knee flexion/extension: starting from a standing position the subject lifts the leg, bending the knee. Motion C is an hip flexion/extension: starting from a standing position the subject raises the leg in a sagittal path and then pushes his/her leg behind as much as possible and returns to the starting position. Figure 4.2.1 show the three motions analysed in this Section. The circled dot identify the axis of rotation of the limb and the arrows indicate the direction of the motion of the limb. All the motions are performed in about 10 seconds.

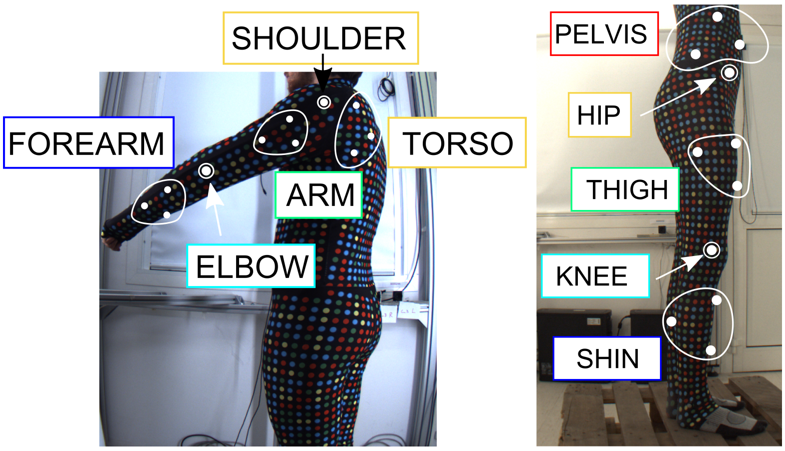


Figure 4.2.2: Vicon marker disposition on the GaMoCap suit. Left: Vicon markers clusters used to evaluate the motion A. Right: Vicon markers clusters used to evaluate the motion B and C.

Vicon marker

In Figure 4.2.2are shown the marker disposition relative to Motion A (image on the left) and Motion B and C (image on the right).

The angle at joint estimated with vicon are computed as relative rotation between two cluster of markers. The angle of rotation is computed with the algorithm described in Subsection 3.4.2.

For each motion, different couple of clusters were used to compute the angle at joints.

- Motion A: relative rotation of cluster labelled “Torso” and “Arm”, Figure 4.2.2-Left

- Motion B: relative rotation of cluster labelled “Thigh” and “Shin”, Figure 4.2.2-Right

- Motion C: relative rotation of cluster labelled “Pelvis” and “Thigh”, Figure 4.2.2-Right

Results Motion A

Figure 4.2.3 shows segmentation results for motion A with LSA (left) and LSA+PCT (right) methods.

Figure 4.2.4 shows the comparison of the angle estimated with algorithm described in Subsection 3.4.2, and Vicon. The points used for the algorithm proposed are the one obtained with LSA and LSA+PCT. The uncertainty bands of angles estimated with MCM for LSA and LSA+PCT are shown on the plot. Error bands, relative to standard uncertainty, show that the three methods lead to compatible results.

Figure 4.2.5 shows the standard uncertainty of the angle estimated with LSA and LSA+PCT respect to Vicon. LSA+PCT have a standard uncertainty of 3.8°. LSA instead show a standard uncertainty which increase linearly with the angle estimated. The standard uncertainty can be expressed as linear expression as follow: $3.3° + 0.08 \cdot AV$. Where AV is the value of the angle estimated with Vicon expressed in degree. This effect is more evident for Motion B and C.

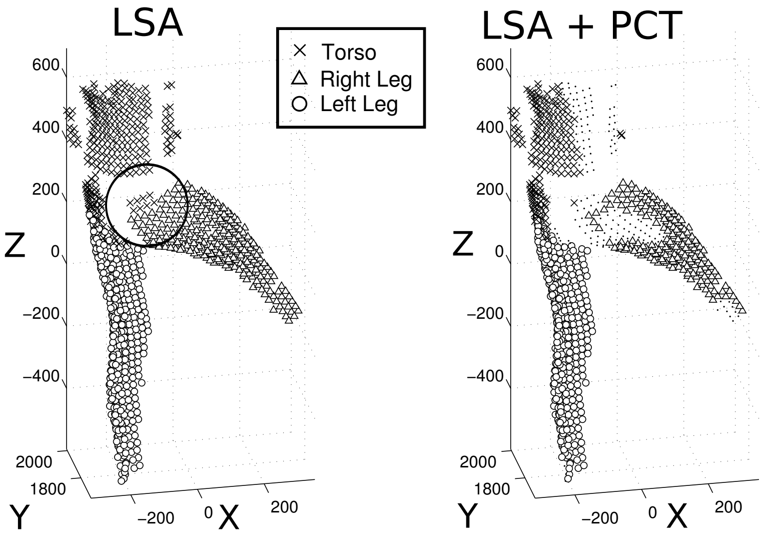


Figure 4.2.3: Motion A. Comparison of results of segmentation operation with LSA and LSA+PCT.

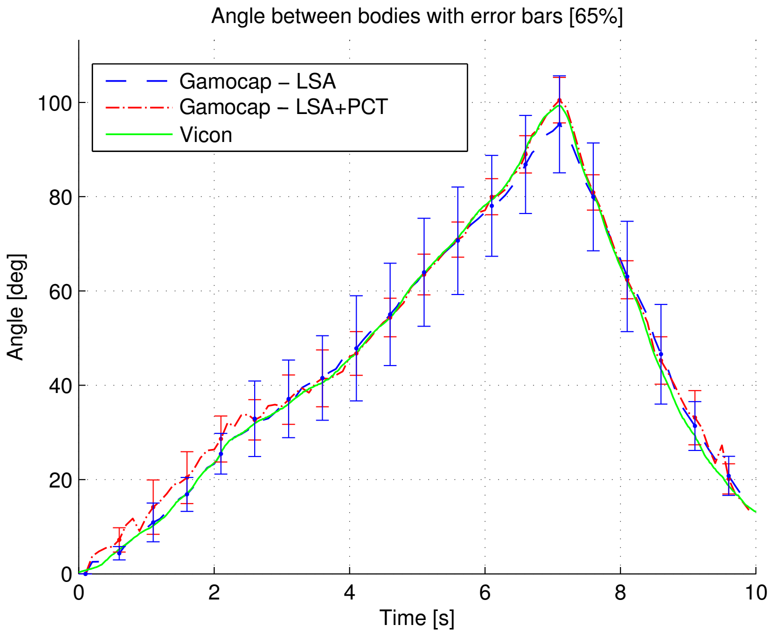


Figure 4.2.4: Comparison of angle estimation results obtained for motion A with GaMoCap (LSA and LSA+PCT) and Vicon.

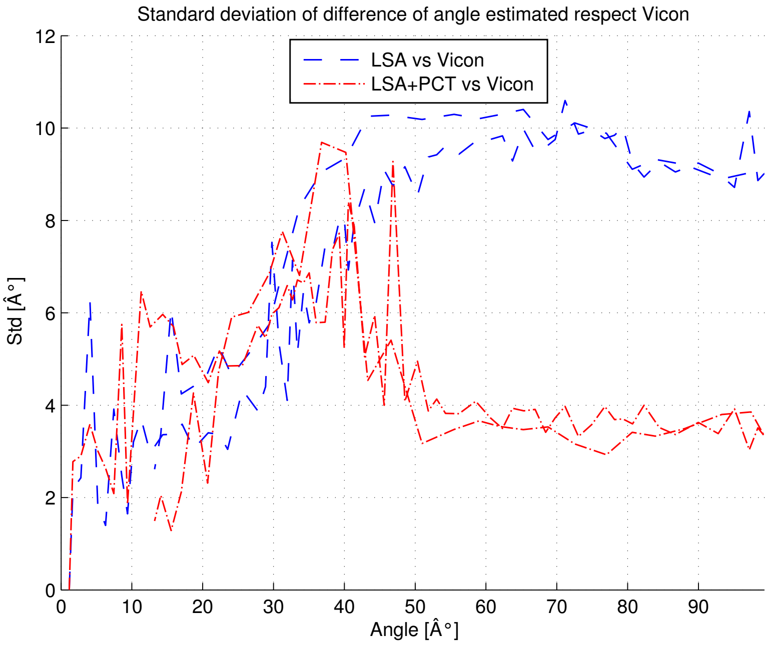


Figure 4.2.5: Standard uncertainty of the angle estimated with LSA and LSA+PCT respect to Vicon for Motion A.

Results Motion B

Figure 4.2.6 shows segmentation results for motion B with LSA (left) and LSA+PCT (right) methods. It is noticeable that many points in correspondence to the joints (within circles) are wrongly segmented with LSA.

This is due to the fact that the points at joints do not move with a rigid motion. LSA+PCT it is not affected by this effect since points near to the joints have been pruned.

In Figure 4.2.6-Right in LSA+PCT points highlighted are the one used to calculate the angle.

Figure 4.2.7 shows angles calculated with the three methods. It is noticeable that LSA+PCT is close to Vicon angle estimation while LSA deviates strongly in terms of lower angle estimation (-20° respect Vicon) and wider bands of error. These two effects are justified by the wrong segmentation of LSA of points close to the joints and the non-determinism of trajectory clustering operation in LSA.

Non rigid motions of points in proximity of joints introduce an error of angle underestimation as described in Subsection 3.4.2.

Wider bands of error are justified by the fact that clustering step of trajectories in LSA is performed with a K-means-based algorithm. In this algorithm points, chosen as seed for segmentation, are selected randomly at each trial of MCM. This generates a segmentation slightly different in each trial, especially for the points at joint proximity. The variation of segmentation of these points in each MCM trial causes a wider error bands in angle estimation.

Figure 4.2.8 shows the standard uncertainty of the angle estimated with LSA and LSA+PCT respect to Vicon. LSA+PCT have a standard uncertainty of 1.6°. This value remains constant with the angle estimated. LSA instead shows a standard uncertainty which increase linearly with the angle estimated. The standard uncertainty can be expressed as linear expression as follow: $1.1° + 0.13 \cdot AV$. Where AV is the value of the angle estimated with Vicon, expressed in degree.

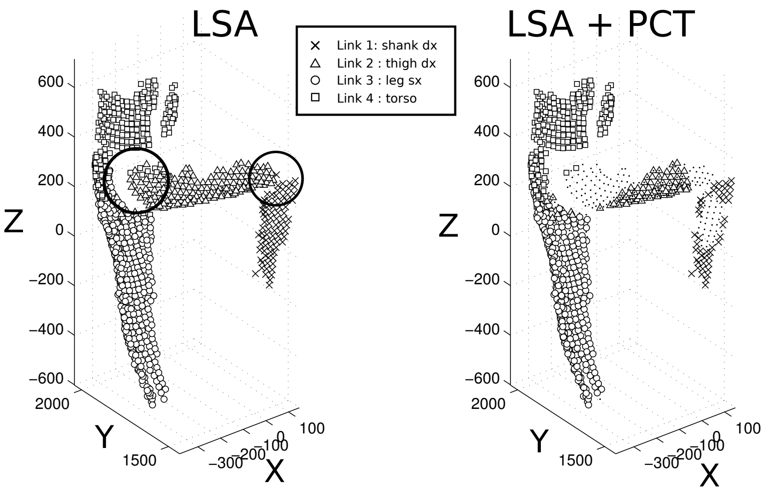


Figure 4.2.6: Motion B. Comparison of results of segmentation operation with LSA and LSA+PCT.

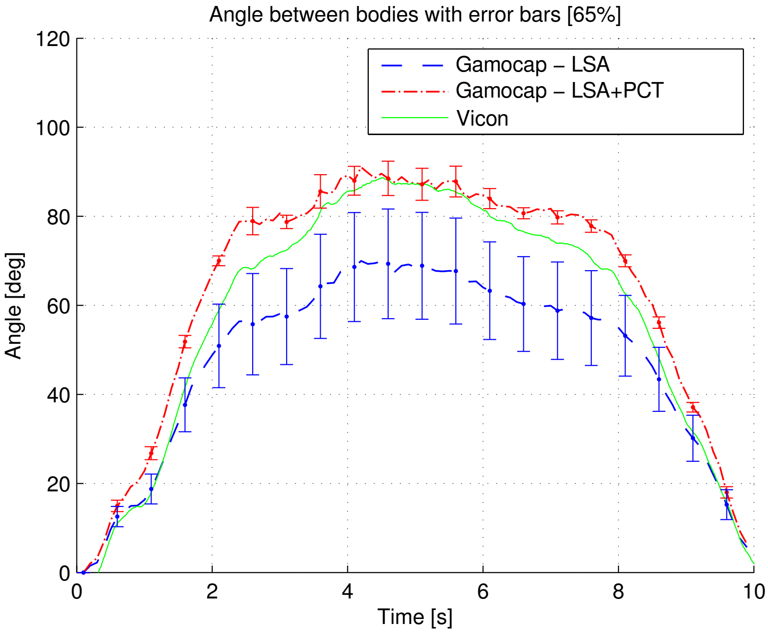


Figure 4.2.7: Comparison of angle estimation results obtained for motion B with GaMoCap (LSA and LSA+PCT) and Vicon.

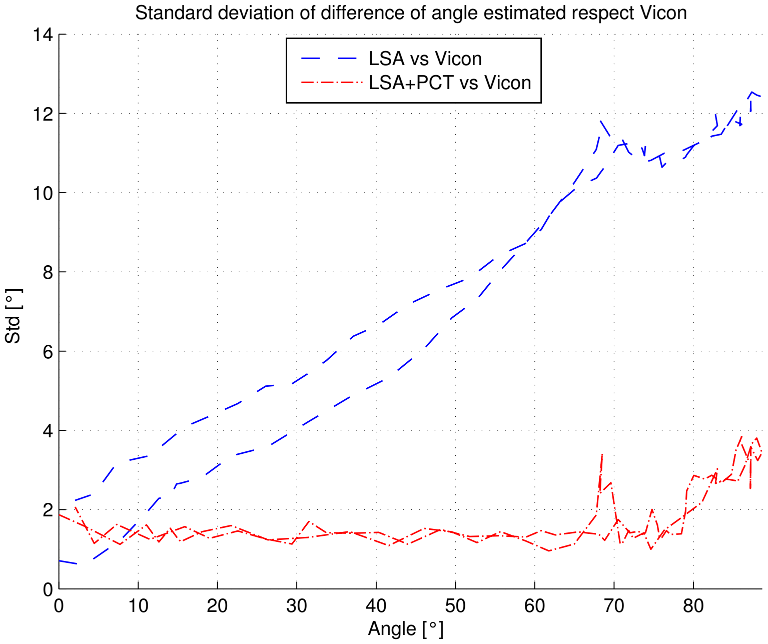


Figure 4.2.8: Standard uncertainty of the angle estimated with LSA and LSA+PCT respect to Vicon for Motion B.

Results Motion C

Figure 4.2.9 shows segmentation results for motion C with LSA (left) and LSA+PCT (right) methods. It is noticeable that many points in correspondence to the hip (within circles) are wrongly segmented with LSA.

Figure 4.2.10 shows that LSA underestimate the maximum angle at hip of about 14° respect Vicon.

Figure 4.2.11 shows the standard uncertainty of the angle estimated with LSA and LSA+PCT respect to Vicon. LSA+PCT have a standard uncertainty of 0.8°. This value remains constant with the angle estimated. LSA instead shows a standard uncertainty which increase linearly with the angle estimated. The standard uncertainty can be expressed as linear expression as follow: $0.2° + 0.05 \cdot AV$. Where AV is the value of the angle estimated with Vicon, expressed in degree.

The difference of repeatability for motion A and B compared with C is explained by a greater number of points reconstructed on which the angle is calculated.

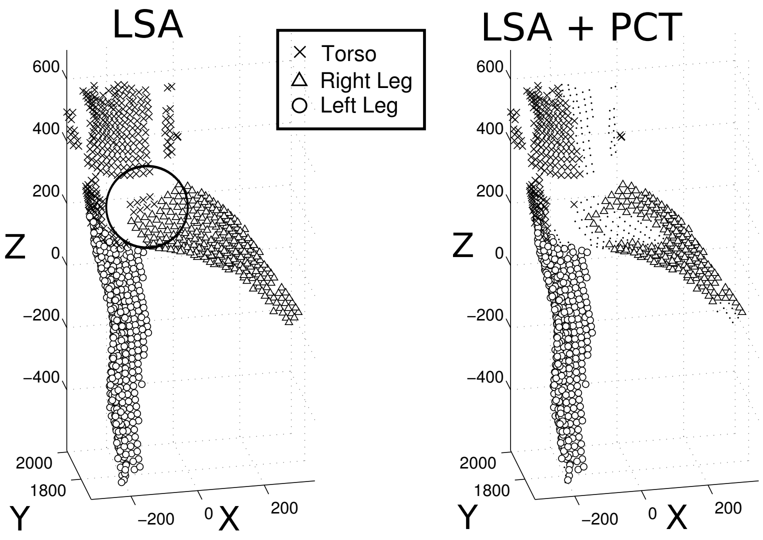


Figure 4.2.9: Motion C. Comparison of results of segmentation operation with LSA and LSA+PCT.

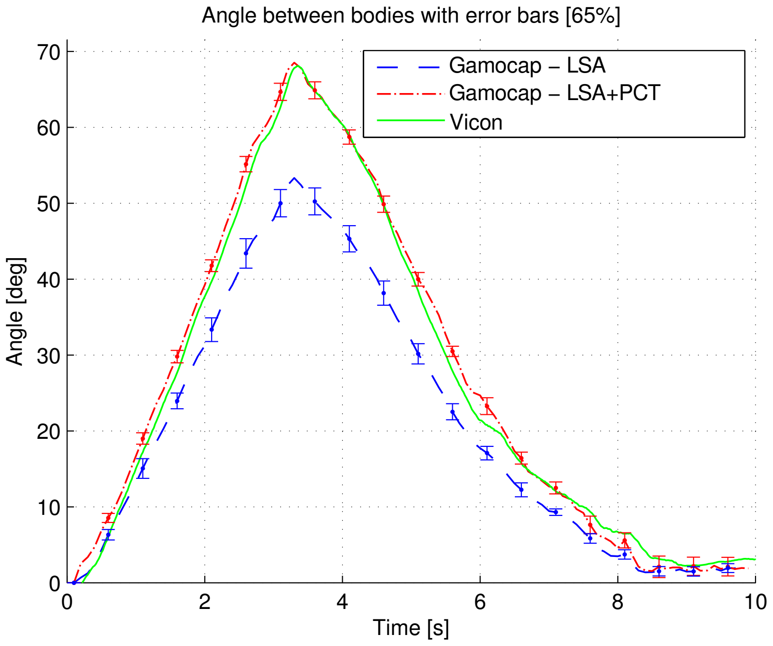


Figure 4.2.10: Comparison of angle estimation results obtained for motion C with GaMoCap (LSA and LSA+PCT) and Vicon.

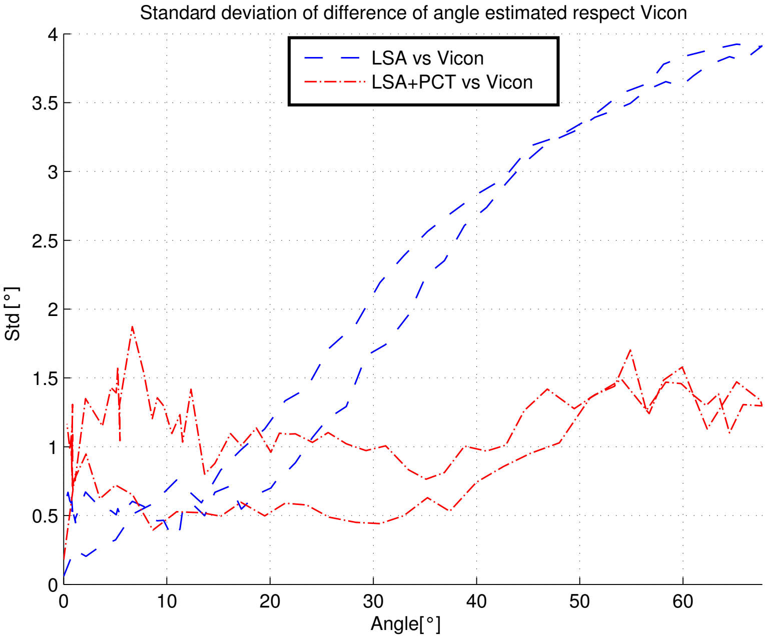


Figure 4.2.11: Standard uncertainty of the angle estimated with LSA and LSA+PCT respect to Vicon for Motion C.

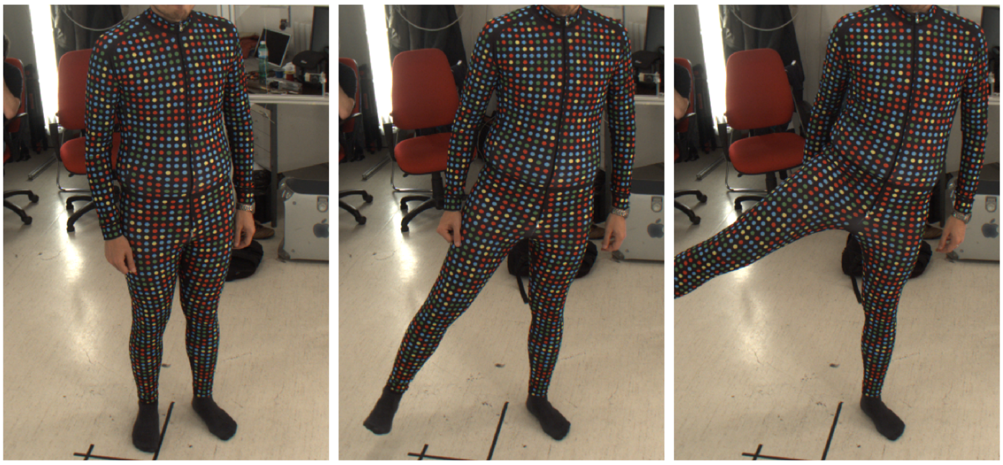


Figure 4.3.1: Key frames of the motion analysed

4.3 Kinematic and Dynamic parameters estimation of a motion

In this Section we show the results of the algorithms proposed in this work to estimate the kinematic and dynamic parameters. Key frames of the motion analysed captured by one camera are shown in Figure 4.3.1.

The motion consists in a hip abduction/adduction. The subject, starting in a standing position with the right foot lifted, raises the leg in a frontal plane and then pushes his/her leg laterally as much as possible and then returns to the starting position. The joint in analysis is the hip.

The requirement of the lifted foot is imposed since we do not have any sensors to measure the force exchanged by the subject with the ground.

While for the kinematic analysis (motion segmentation, angle estimation, joint position) the problem is considered 3D in the case of dynamic analysis (forces and torques at joint) the movement is considered to be performed only on a frontal plane. This motion is considered to be thus 2D.

Figure 4.3.2 shows the results of the shape reconstruction of the subject and the motion segmentation results. The algorithm proposed identifies correctly 3 bodies: torso, left leg, right leg.

Since the knee of the subject does not bend during the motion the lifted leg is considered as one segment and thus fitted with only one superquadrics.

Figure 4.3.3 shows the results of joint position estimation and the result of superquadrics fitting.

Dynamic estimation of the parameters of motion consists in the solution of the Equation 2.2.1 of the *free-body diagram*, Figure 4.3.4.

With the hypothesis of a 2D motion and that the only limbs that is moving is the right leg, the equations to be solve to calculate the forces and torques at joints are:

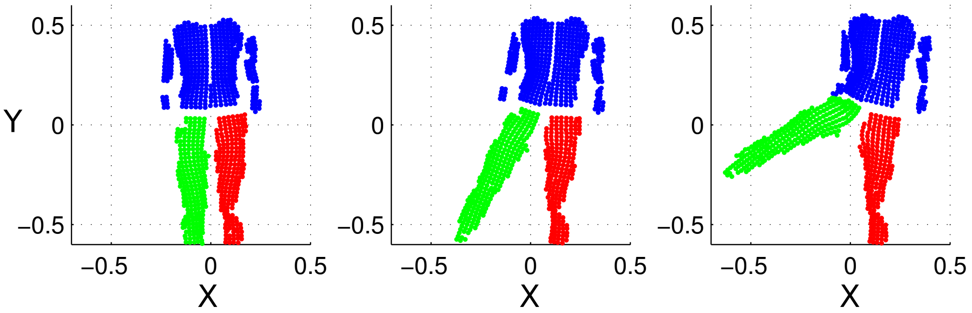


Figure 4.3.2: Key frames of 3D point reconstructed and segmented.

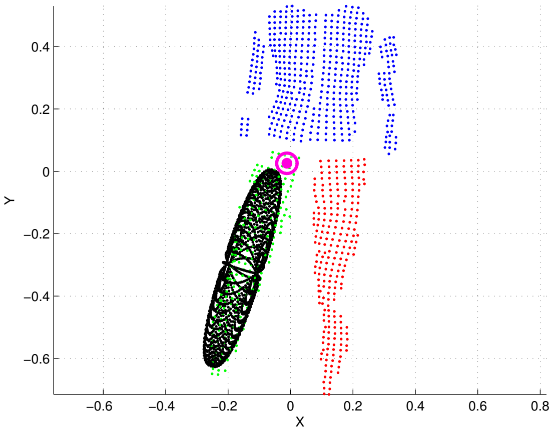


Figure 4.3.3: Result of Superquadric fitting on the leg of the subject. The pink circle indicates the axis of rotation of the leg identified with the algorithm proposed.

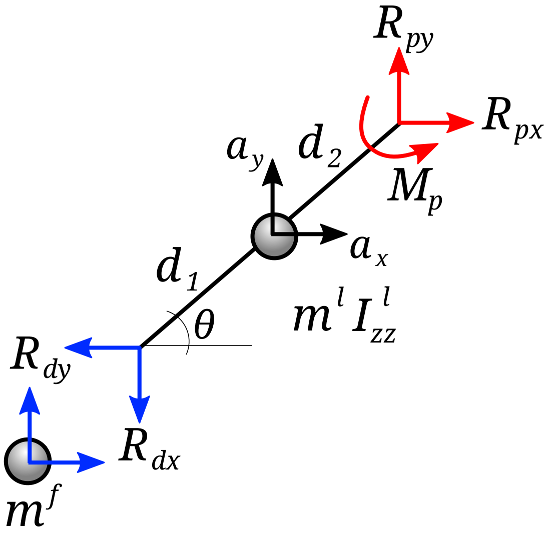


Figure 4.3.4: 2D free-body diagram used.

$$\begin{cases} R_{xp} - R_{xd} = m^l a_x^l \\ R_{yp} - R_{yd} - m^l g = m^l a_y \\ M_p - M_d - (R_{xd}d_1 + R_{xp}d_2)\sin(\theta) - (R_{yd}d_1 + R_{yp}d_2)\cos(\theta) = I_{zz}^l\ddot{\theta} \end{cases}$$

where $R_{xp}$and $R_{yp}$are the resulting reaction forces at hip joint. $a_x^l$and $a_y^l$are the acceleration of the COM of the cloud of points (coincident with the SQ centre). m^l and I_{zz}^l are respectively mass and inertia of the leg calculated with SQ. $d_1$and $d_2$are respectively the distance of the COM from the joint of the foot and the hip. Both are set equal to half of the leg length calculated with SQ parameters.

R_{xd} and R_{yd} are the reaction forces acting at distal end of segment, in our case the foot joint. As a simplification we have considered the mass of the foot equal to 1.09 kg concentrated at distal joint position. The mass of the foot is calculated as a fraction of the body weight $(0.0145 \cdot M_{tot})$ as described by [107]. The moment of inertia is neglected. Since the subject is not loading the foot there are no reaction forces with the ground. The reaction forces at the joint are calculated as follows:

$$\begin{cases} R_{xd} = m^f a_x^f \\ R_{yd} = m^f a_y^f + m^f g \end{cases}$$

where m^f is the mass of the foot, a_x^f and a_y^f the acceleration of the foot.

Figure 4.3.5 shows a comparison of 1° and 2° derivative calculation leg rotation with finite differences differentiation and wavelet property, Equation 2.2.15. The Figure shows the fast degradation of the signal with the finite differences derivative calculation.

In Figure 4.3.6 are shown the displacements of COM of the leg in x and y direction and the rotation of the leg. Figure also shows the velocity and the acceleration obtained

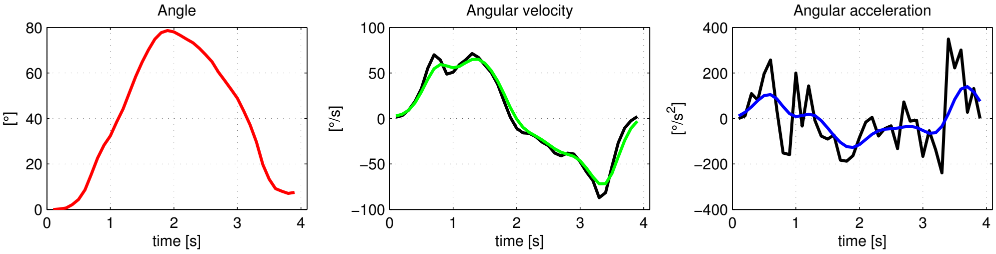


Figure 4.3.5: Comparison of 1° and 2° derivative calculation of leg rotation with finite differences (black) and wavelet (color).

with wavelet differentiation.

The results of the torque M_p and reaction forces R_{px} and R_{dy} estimation at joints are shown in Figure 4.3.7.

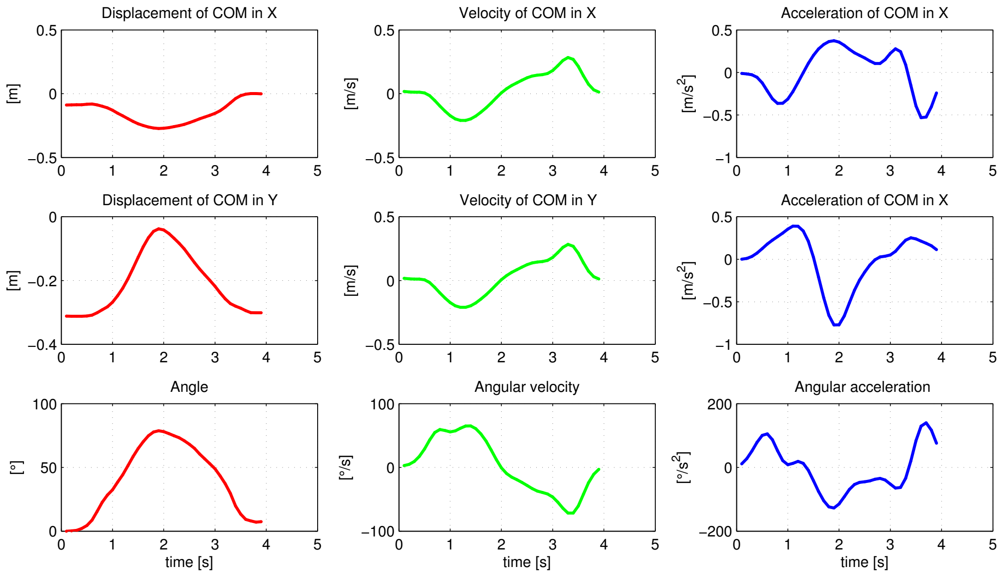


Figure 4.3.6: Displacements (linear and rotational), velocity and acceleration of COM of leg.

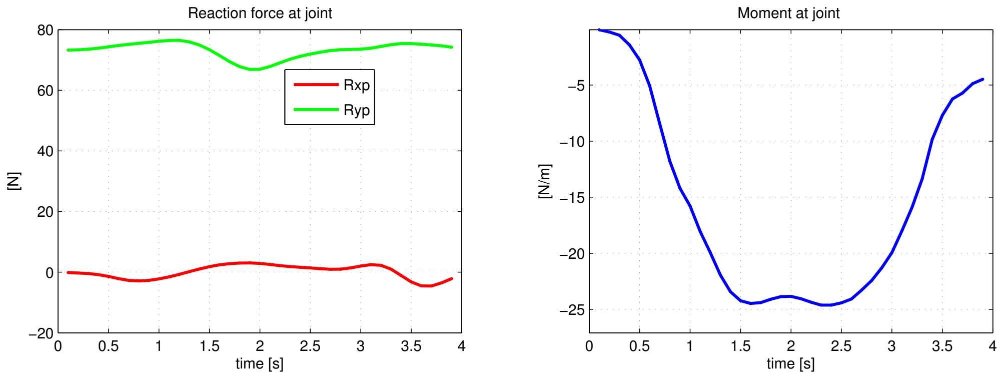


Figure 4.3.7: Left: reaction forces at hip joint. Right: torque at hip

5 GaMoCap applications

In this Chapter are described two main application in which GaMoCap system was used as measurement tool: Veritas and Memscon. In Veritas project GaMoCap was used to measure the kinematic parameters of elderly and impaired people. In Memscon project GaMoCap was used to measure displacements of a concrete structure undergoing a shake sequence that simulate a shock-wave of an earthquake.

5.1 Veritas Project

VERITAS is an acronym that stands for: **V**irtual and augmented **E**nvironment and **R**ealistic user **I**nteractions **T**o achieve embedded **A**ccessibility design**S**. The core concept of VERITAS is to conduct research and development of an open framework for providing built-in accessibility support at all the stages of realization of mainstream ICT and non-ICT technologies. The project aims at delivering to product/software developers 'generic' instructions - embedded in an empowering virtual reality platform, for exploring new concepts, designing new interfaces and testing interactive prototypes that will inherit universal accessibility features, including compatibility with established assistive technologies. As a consequence VERITAS project aims to analyse and to create detailed virtual user models for some selected disability.

In VERITAS project GaMoCap was used to estimate the parameters of mobility of the virtual user with different types and degree of disabilities. The analysis was performed on 6 different types of motions: 3 for upper body and 3 for lower body kinematic parameters. Each movement is here described and the results obtained are reported and discussed.

For this purpose special suits were realized to face the restricted articular mobility of the patients. The key concept which guided the realization of these garment were: fast and easy suit setup, unaffected subject mobility and maximum contact between suit and the limb of the patient. Were realized suit in 3 different sized (S,M,L).

The result of this design phase are shown in Figure 5.1.1. Figure 5.1.1-Right shows the trousers used to measure the lower body parameters. Figure5.1.1-Left show the 3 pieces suit used in upper body parameter estimation. The two pieces are a corset and a 2 sleeves (arm and forearm). The corset is closed with a velcro strip on the back to be easily worn.

The patients

The patients in analysis are affected by two articular diseases: gonarthrosis and coxarthrosis.

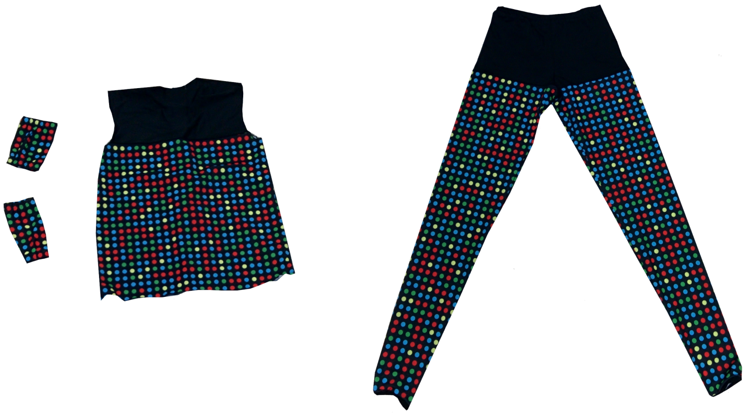


Figure 5.1.1: Veritas 4 pieces suit. Left: 3 pieces suit used for upper limb analysis. Right: trousers for lower limb analysis.

Gonarthrosis and coxarthrosis are degenerative diseases which lead to a deterioration respectively of the cartilage of the hip and of the knee. The bones, without the protection of the cartilage, start rubbing directly against each other causing pain and limited joint mobility, Figure 5.1.2.

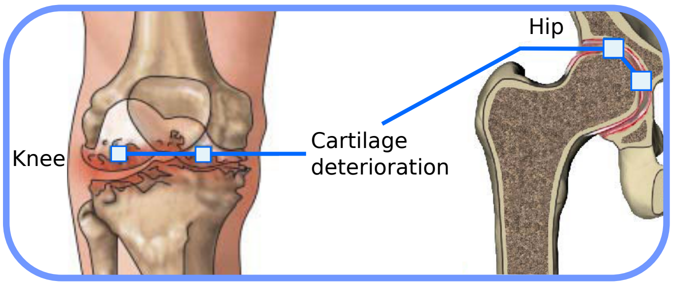


Figure 5.1.2: Deterioration of the cartilage at joint. Left: deterioration of the cartilage at the knee (gonarthrosis). Right: deterioration of the cartilage of the hip (coxarthrosis).

The degree of the motion disability is quantified with the Barthel scale, [70]. This is an ordinal scale used to measure performance in activities of daily living. The values assigned to each item are based on time and amount of actual physical assistance required if a patient is unable to perform the activity. The values range from 0 to 15. In this case Barthel index is obtained as a sum of the indexes relative to the ability to ambulate and the ability to ascend/descend the stairs.

For example the Barthel index of ambulation is equal to 15 if patient can walk at least 50 meters without help or supervision, 10 if patients needs help or supervision but can walk at least 50 meters.

The patients analysed have different Barthel indexes which go from 15 to 25.

Lower body motions

Lower body motions analysed consists in 3 different articular motions: knee flexion, hip flexion/extension and hip adduction. Motions need to test the maximum angle at joint achievable by the patients.

1. Knee flexion. This motion tests the maximum knee angle. Starting from a standing position (with the necessary support if required) the subject bends his knee as much as possible, Figure 5.1.3.

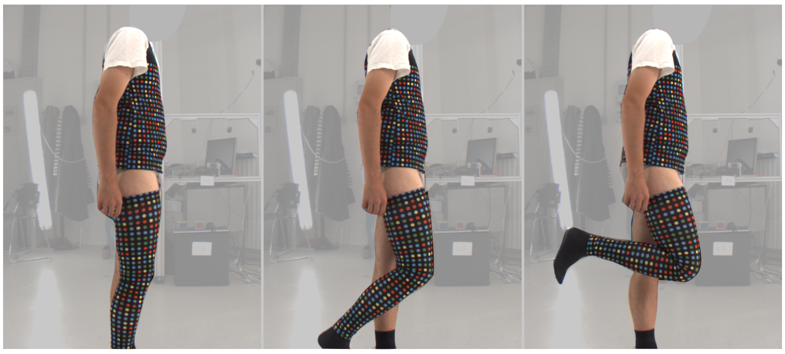


Figure 5.1.3: Veritas motion analysed: Knee flexion

2. Hip flexion and extension. This motion tests the maximum hip flexion angle. Starting from a standing position (with the necessary support if required) the subject raises the leg in a sagittal path as much as possible and then push his/her leg behind as much as possible. Bending knee is allowed. Figure 5.1.4.

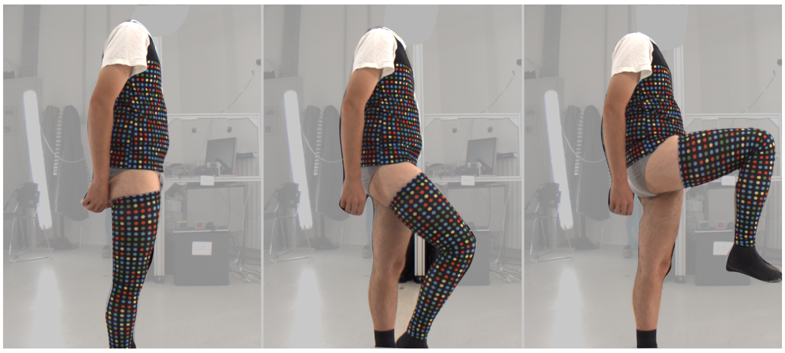


Figure 5.1.4: Veritas motion analysed: Hip flexion/extension

3. Hip adduction. With this motion is tested the maximum hip adduction angle. Starting from a standing position (with the necessary support if required) the subject raises the extended leg in a lateral path as much as possible. Figure 5.1.5.

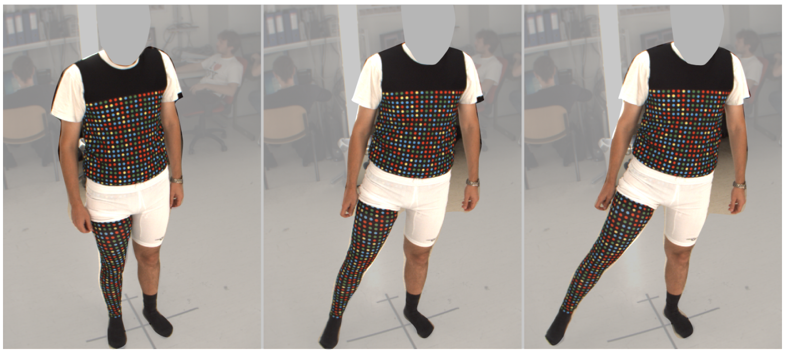


Figure 5.1.5: Veritas motion analysed: Hip adduction

Upper body motions

Upper body motions analysed consists in 3 different articular motions: shoulder sagittal, lateral and horizontal rotations.

1. Shoulder sagittal rotation. Starting with an arm relaxed along the torso, the subject raise the extended arm in sagittal path until the maximal extension, then come back and go down/behind as much as possible. Figure 5.1.6.

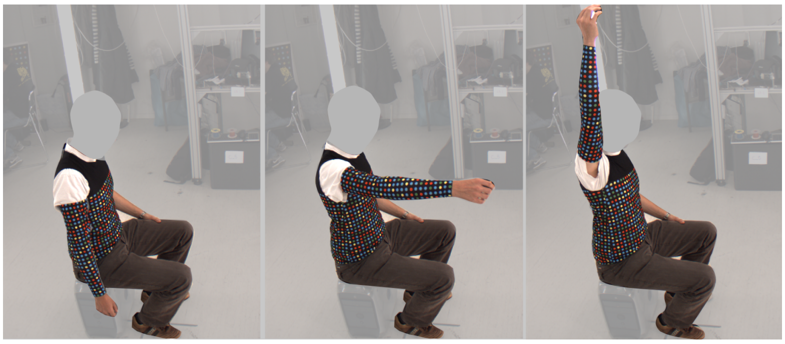


Figure 5.1.6: Veritas motion analysed: Shoulder sagittal rotation

2. Shoulder lateral rotation. This motion evaluates the maximum shoulder lateral rotation angle. Starting with an arm relaxed along the torso, the subject raises the extended arm in lateral path until reaches the maximal extension. Figure 5.1.7.

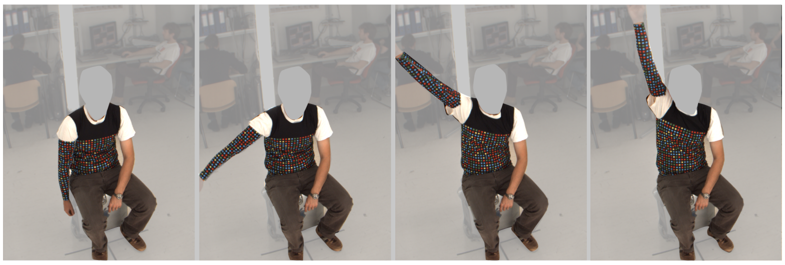


Figure 5.1.7: Veritas motion analysed: Shoulder lateral rotation

3. Shoulder horizontal rotation. Starting with arm horizontally extended in front of himself, the subject perform an horizontal rotation of the arm in right and left direction as much as possible. Figure 5.1.8.

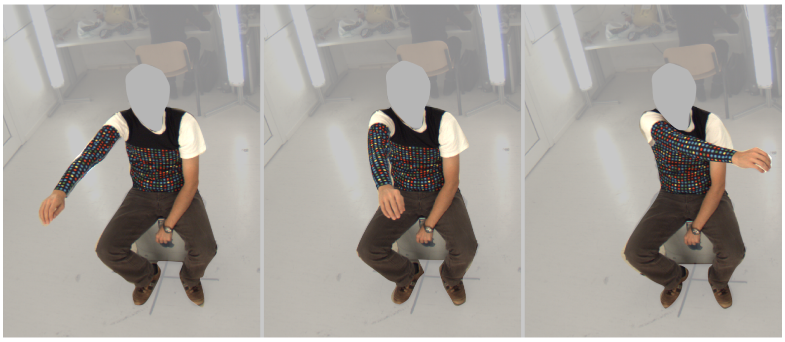


Figure 5.1.8: Veritas motion analysed: Shoulder horizontal rotation

Results

Figure 5.1.9 shows the result relative to the maximum angle spanned by patients with gonarthrosis for three different articular motions (knee flexion, hip flexion/extension, hip adduction) as a function of their Barthel index.

Figure 5.1.10 shows the result relative to the maximum angle spanned by patients with coxarthrosis for three different articular motions: knee flexion, hip flexion/extension, hip adduction.

These results will be considered when generating a virtual model of a subject with gonarthrosis and coxarthrosis to take into consideration the maximum angles achievable.

The GaMoCap system proved to be a valid tool for this type of analysis especially for its fast setup. This feature has shown to be essential when working with patient with reduces articular mobility.

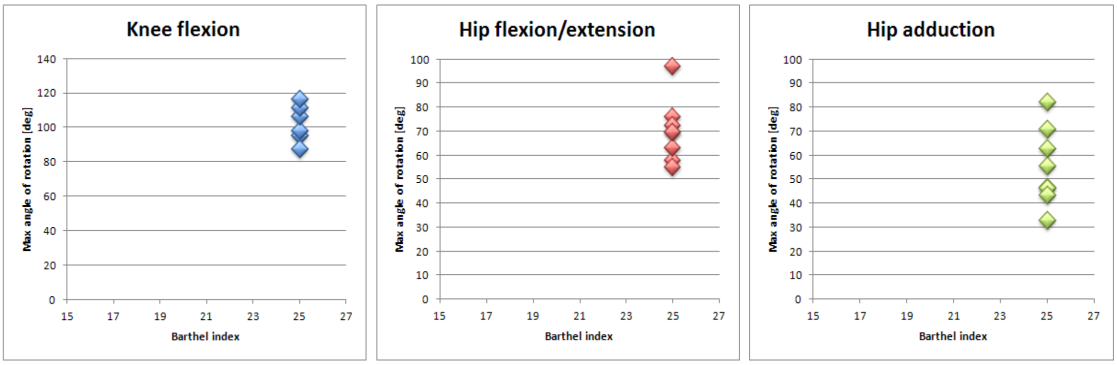


Figure 5.1.9: Maximum angle spanned by patients with gonarthrosis for different lower limbs motions.

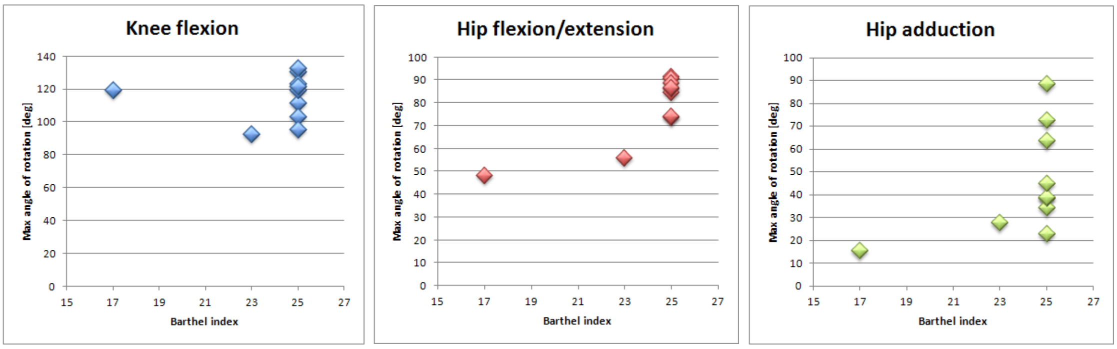


Figure 5.1.10: Maximum angle spanned by patients with coxarthrosis for different lower limbs motions.

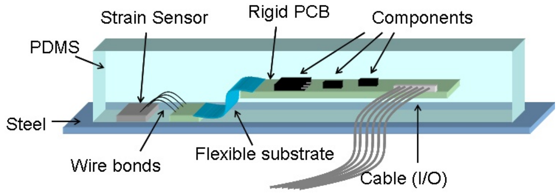


Figure 5.2.1: Memscon wireless sensor

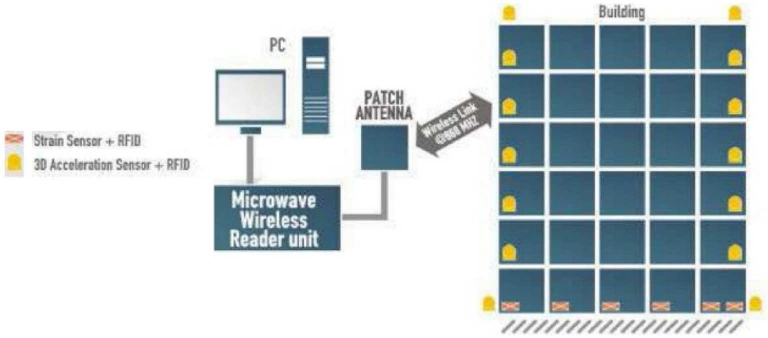


Figure 5.2.2: Memscon system layout

5.2 Memscon

Memscon[1] is and European Funded research project that aims to develop a structural monitoring system to be installed in reinforced concrete buildings, for damage assessment just after a seismic event. The system consists of a set of accelerometers and strain sensors, Figure 5.2.1, drowned in the concrete structure of the building with wireless transmission of the data to an acquisition centre outside the building, where the data is interpreted and damage analysed, Figure 5.2.2.

The reliability of the displacement time history estimated by double integration of the accelerations are compared with the displacement measurements recorded with an opportune setup of GaMoCap multicamera system.

The whole Memscon system was validated under laboratory conditions, with the implementation of the system in a three-dimensional reinforced concrete frame, Figure 5.2.3, which underwent a shake sequence of increasing amplitude.

In Figure 5.2.3 it is possible to see the two actuators used to: simulate the vertical dead load and the live lateral loads. The actuator in vertical position have the role to simulate the vertical load of the concrete structure (about 150 kN per column). The actuator in horizontal position is used to induce an horizontal displacement that simulates the earthquake. The dimensions of the frame are of $3.20 \times 6.35\, m^2$ in plan and $3.90m$ height above ground.

[1] http://www.memscon.com

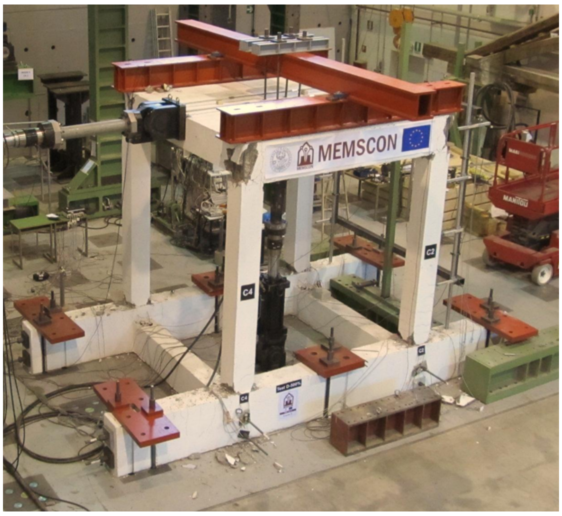


Figure 5.2.3: Memscon concrete structure laboratory setup.

Setup

The GaMoCap setup used is visible in Figure 5.2.4. A set of 4 Matrix BlueCougar arranged stereo-couple is used. Two cameras point at the base of a column to record the deformation, while two at the top of the structure measure the structure displacement.

In each area of analysis is reproduced the color coded pattern with adequate geometric characteristics. At the base of the column, since the cameras are very close to the structure it is used a grid with step of $40mm$ and diameter of markers of $8mm$, Figure 5.2.5-Left. A the top of the structure the marker have a diameter of $25mm$ and are arranged on a square grid with step of $150mm$, Figure 5.2.5-Right.

Results

The sequence of synchronized images captured at frequency of 10 Hz is elaborated, as described in Section 3.2, to obtain the 3D coordinates of each marker at each time instant. The 3D point reconstructed, relative to time instant $i = 0$, are shown in Figure 5.2.6-Left. In Figure 5.2.6-Right are shown the displacement of the points highlighted on the left during the earthquake simulation.

The GaMoCap frequency of acquisition is a trade-off between speed and length of acquisition. The maximum length of the acquisition is constrained by the amount of RAM installed on the PC used for this elaboration.

The results of displacements estimated with GaMoCap are compared to the displacement sensor attached to the horizontal actuator and displayed in Figure 5.2.7.

It is noticeable that displacement estimation is accurate except for the discrepancy at time instant $i = 34.4s$. The GaMoCap does not accurately detect the maximum displacement of the first cycle, due to an insufficient acquisition frequency (10 Hz), suitable to record only ~7 samples per structure oscillation (~1.5 Hz).

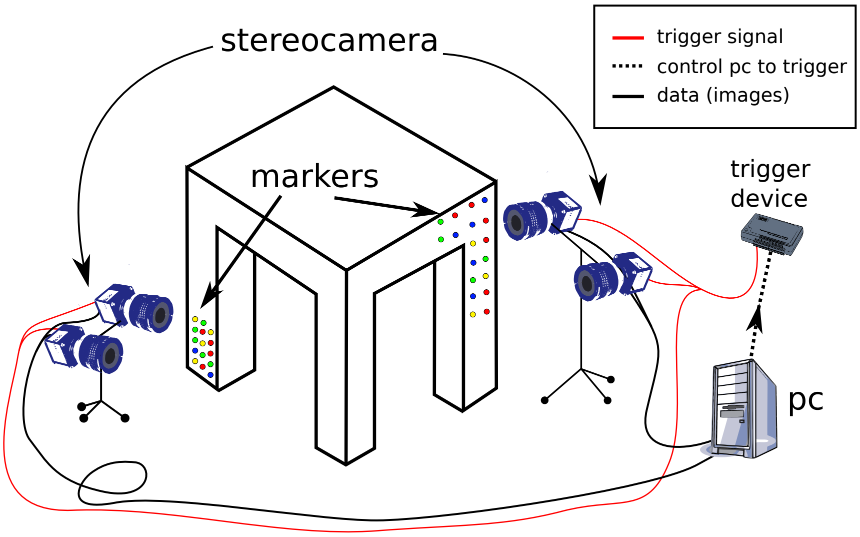


Figure 5.2.4: GaMoCap setup used in Memscon laboratory test.

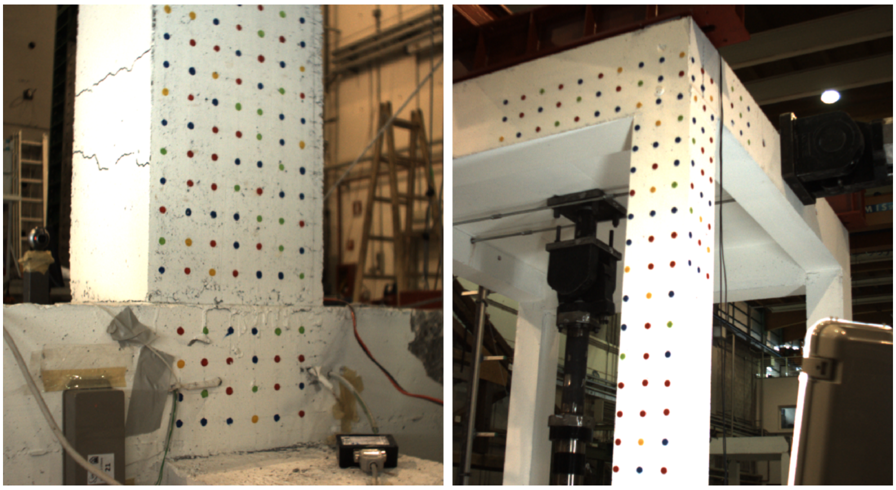


Figure 5.2.5: GaMoCap marker position at the base(left) and at the top(right) of the concrete structure.

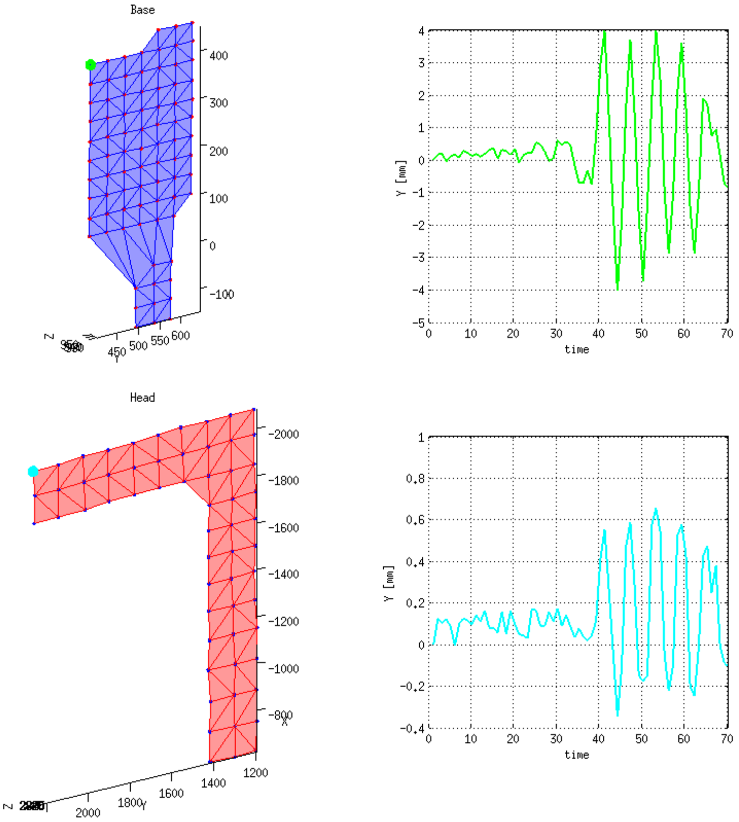


Figure 5.2.6: Left: 3D point reconstruction of markers placed at the base and at the top of the concrete structure. Right: Trajectory over time of the point highlighted during the earthquake simulation.

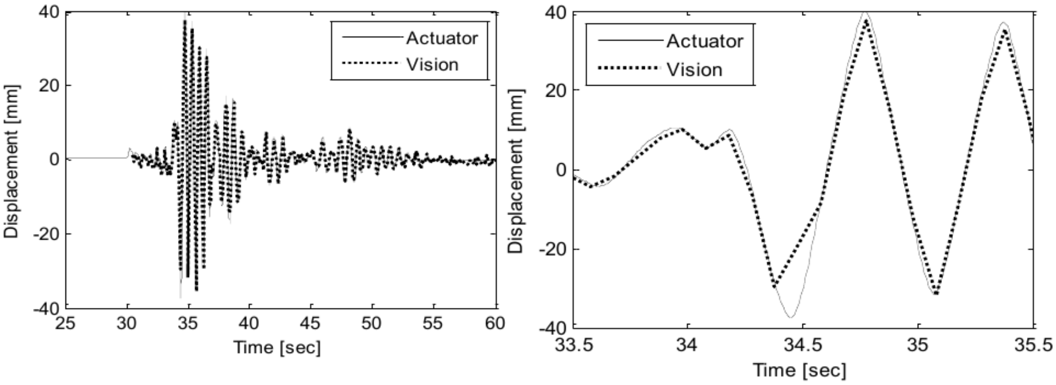


Figure 5.2.7: Comparison between the displacement induced by the actuator and displacement recorded by the GaMoCap (vision) system (left) and zoom of the first cycles (right).

This system however proved to be a valid tool to measure displacement in laboratory setup. Despite the low resolution in time the GaMoCap showed desirable features for a experimental setup: measurement of many points simultaneously, easy and fast setup time, possibility to measure the 3D motion of the structure.

6 Conclusions

This work presented a new concept garment-based mocap system, GaMoCap, capable of a simultaneous reconstruction of motion and shape. This is a important characteristic of this system since combines the features of 3D scanner with classic mocap systems. In particular the aim of this work was to realize a framework of kinematic and dynamic analysis of human motion without a-priori model.

The main contributes of our work are summarized in the next paragraphs.

Garment-based Mocap Multicamera system realized was designed and developed for a generic reconstruction of shape and motion. The algorithm of image elaboration implemented were realized with the principles of flexibility, robustness and scalability. The color recognition was resolved with a simple and fast solution. The color coded pattern realized is an elegant solution to solve some intrinsic problem of computer vision application like feature description and association. The pattern allows also to break the limits of number of makers of classical motion capture systems. The high density pattern allows a reconstruction of the shape with good accuracy for most of the applications.

With the conjunct features of dense reconstruction of the shape and the motion, GaMoCap is effectively a 2in1 measurement tool.

Multicamera calibration The multicamera calibration toolbox implemented overcome some of the classic problem of multicamera calibration toolbox like partial grid visibility. We generated a multicamera calibration toolbox flexible and reliable also for partial overlapping fields of view. The multicamera calibration allows to recalibrate the system with no difficulty and with a reduced time, assuring good quality reconstruction and motion results.

Kinematic analysis The kinematic algorithms were designed to work without a model of the user known a-prior. Our aim was to develop a fully-automated framework which requires limited user interaction to produce the results. Thanks to the colour coded pattern was possible to apply a data driven technique to kinematic analysis, LSA. STAs compensation technique paired with GaMoCap data allows to produce results comparable with high level commercial system (Vicon).

Dynamic analysis The dynamic analysis was performed on a simplified 2D problem. However the algorithm proposed can be generalized to a 3D case with a minimum effort. We proposed a mass and inertia estimation based on a compact parametric model. The solution proposed is fast and easy to implement.

We exploited a property of Wavelet transformations to calculate the limbs acceleration from GaMoCap displacement data. This method is particular useful since in able to work even in presence of noisy data.

The quality of the kinetic parameters estimated was analysed in detail with a numerical uncertainty propagation method. The procedure adopted was hybrid, mixing synthetic and real data. Results obtained are compatible with the requirements of clinical applications.

Algorithms All the algorithm proposed in this work were implemented in C/C++ language to assure a fast speed of execution. The process of feature extraction was parallelized for each camera to further reduce the elaboration time. In order to assure the usability of these algorithms was also developed a multi-platform graphical interface with which is possible to control the steps of shape reconstruction and motion analysis.

Applications The framework developed was employed in an experimental environment to test its kinematic and dynamic analysis capabilities.

The GaMoCap was also used in one European project in which it proved to be a valid framework for human motion analysis. Its easy and quick setup proved to be a essential feature especially for the motion analysis of impaired and elder people.

The system proved to have desirable features also outside the human motion analysis. A special GaMoCap setup was prepared for the analysis of a civil engineering concrete structure. Whit a precision comparable to classic displacement sensor (LVDT), a higher number of points measured and a lower cost of setup is a valid and cost-effective alternative as a laboratory measurement tool.

Finally, some limitations need to be considered.

Despite the fact that algorithms of shape reconstruction and motion analysis are quite fast was not possible to build a real-time mocap system. This was caused mainly by the limited time and budget to develop a real-time solution.

The color coded pattern realized is a powerful but not error-free solution. Garment fold, overlap and bad illumination can generate wrong or duplicate IDs that cannot be corrected with the garment mapping. A solution to this problem would be to introduce code correction capabilities in the coloured sequence via standard techniques of communication theory.

STAs compensation based on the PCT proposed tends to eliminate a lot of 3D points. While the number of points that remains is sufficient to perform the motion analysis, some better STAs compensation solutions can be studied.

Even with the objective to realize a full-automatized motion analysis framework, some steps of kinematic and dynamic motion still require a user intervention. A further effort in this direction will produce an automatic, easy-to use framework of motion analysis.

In order to have a more complete and robust framework of motion analysis in a future work possible improvements are:

- to implement code correction techniques in the color coded pattern to improve IDs assignation stability and reconstruction
- to study better STAs compensation technique in order to maximize the number of points after motion segmentation
- to develop an algorithm for 3D dynamic analysis

Bibliography

[1] Amin Ahmadi, D Rowlands, and D James. Deriving upper arm rotation from vicon to enhance the first serve in tennis. *Journal of Science and Medicine in Sport*, 12:S29–S30, 2009.

[2] Eugene J Alexander and Thomas P Andriacchi. Correcting for deformation in skin-based marker systems. *Journal of Biomechanics*, 34(3):355–361, 2001.

[3] TP Andriacchi, EJ Alexander, MK Toney, C Dyrby, and J Sum. A point cluster method for in vivo motion analysis: applied to a study of knee kinematics. *Journal of biomechanical engineering*, 120(6):743–749, 1998.

[4] K Somani Arun, Thomas S Huang, and Steven D Blostein. Least-squares fitting of two 3-d point sets. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 5(5):698–700, 1987.

[5] Luca Ballan and Guido Maria Cortelazzo. Multimodal 3d shape recovery from texture, silhouette and shadow information. In *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pages 924–930. IEEE, 2006.

[6] Alan H Barr. Superquadrics and angle-preserving transformations. *IEEE Computer graphics and Applications*, 1(1):11–23, 1981.

[7] MG Benedetti, A Cappozzo, F Catani, and A Leardini. Anatomical landmark definition and identification. *EU project CAMARC II deliverable, March*, 15:1994, 1994.

[8] Georg Biegelbauer and Markus Vincze. Efficient 3d object detection by fitting superquadrics to range image data for robot's object manipulation. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 1086–1091. IEEE, 2007.

[9] BIPM, IEC, IFCC, ISO, IUPAP, and OIML. *Guide to the Expression of Uncertainty in Measurement.* ISO - International Organization for Standardization, 1993.

[10] Andrea Bottino and Aldo Laurentini. A silhouette based technique for the reconstruction of human movement. *Computer Vision and Image Understanding*, 83(1):79–95, 2001.

[11] Vicky Bouffard, Mickael Begon, Annick Champagne, Payam Farhadnia, Pascal-Andr&eacute; Vendittoli, Martin Lavigne, and Fran&ccedil Prince. Hip joint center localisation: A biomechanical application to hip arthroplasty population. *World J Orthop*, 3(8):131–6, 2012.

[12] Jean-Yves Bouguet. Complete camera calibration toolbox for matlab, 2004.

[13] Duane C Brown. Decentering distortion of lenses. *Photometric Engineering*, 32(3):444–462, 1966.

[14] Thomas D Brown, Leon Sigal, Glen O Njus, Nina M Njus, Robert J Singerman, and Richard A Brand. Dynamic performance characteristics of the liquid metal strain gage. *Journal of biomechanics*, 19(2):165–173, 1986.

[15] John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 6(6):679–698, 1986.

[16] Angelo Cappello, Aurelio Cappozzo, Pier Francesco La Palombara, Luigi Lucchetti, and Alberto Leardini. Multiple anatomical landmark calibration for optimal bone pose estimation. *Human movement science*, 16(2):259–274, 1997.

[17] Aurelio Cappozzo, Fabio Catani, Alberto Leardini, MG Benedetti, and Ugo Della Croce. Position and orientation in space of bones during movement: experimental artefacts. *Clinical biomechanics*, 11(2):90–100, 1996.

[18] Laurent Chevalier, Fabrice Jaillet, and Atilla Baskurt. Segmentation and superquadric modeling of 3d objects, 2003.

[19] WM Chung, S Yeung, WW Chan, and R Lee. Validity of vicon motion analysis system for upper limb kinematic measurement–a comparison study with inertial tracking xsens system. *Hong Kong Physiotherapy Journal*, 29(2):97, 2011.

[20] Sean Clarkson, Simon Choppin, John Hart, Ben Heller, and Jon Wheat. Calculating body segment inertia parameters from a single rapid scan using the microsoft kinect, 2012.

[21] Charles E Clauser, John T McConville, and John W Young. Weight, volume, and center of mass of segments of the human body. Technical report, DTIC Document, 1969.

[22] Renato Contini. Body segment parameters, part ii. *Artificial limbs*, 1:1, 1972.

[23] Stefano Corazza, Emiliano Gambaretto, Lars Mundermann, and Thomas P Andriacchi. Automatic generation of a subject-specific model for accurate markerless motion capture and biomechanical applications. *Biomedical Engineering, IEEE Transactions on*, 57(4):806–812, 2010.

[24] Stefano Corazza, Lars Mündermann, and Tom Andriacchi. A framework for the functional identification of joint centers using markerless motion capture, validation for the hip joint. *Journal of biomechanics*, 40(15):3510–3515, 2007.

[25] Stefano Corazza, Gianluigi Pillonetto, Ruggero Frezza, and Claudio Cobelli. Numerical approach to skin artifacts correction in stereophotogrammetry. In *Proceedings of the 27th Annual Meeting of the American Society of Biomechanics*, pages 25–27, 2003.

[26] H.S.M. Coxeter. *Regular Polytopes*. Courier Dover Publications, June 1973.

[27] M De Cecco, L Baglivo, G Parzianello, M Lunardelli, F Setti, and M Pertile. Uncertainty analysis for multi-stereo 3d shape estimation. In *Advanced Methods for Uncertainty Estimation in Measurement, 2009. AMUEM 2009. IEEE International Workshop on*, pages 22–27. IEEE, 2009.

[28] Mariolino De Cecco, Marco Pertile, Luca Baglivo, Massimo Lunardelli, Francesco Setti, and Mattia Tavernini. A unified framework for uncertainty, compatibility analysis, and data fusion for multi-stereo 3-d shape estimation. *Instrumentation and Measurement, IEEE Transactions on*, 59(11):2834–2842, 2010.

[29] Wilfrid Taylor Dempster. Space requirements of the seated operator: geometrical, kinematic, and mechanical aspects of the body, with special reference to the limbs. Technical report, DTIC Document, 1955.

[30] Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.

[31] RD Dorociak and TJ Cuddeford. Determining 3-d system accuracy for the vicon 370 system. *Gait & Posture*, 3(2):88, 1995.

[32] Rudolfs Drillis, Renato Contini, and Maurice Bluestein. Body segment parameters. *Artificial limbs*, 8(1):44–66, 1964.

[33] J Fayad, Alessio Del Bue, Lourdes Agapito, and Pedro MQ Aguiar. Human body modelling using quadratic deformations. In *7th EUROMECH Solid Mechanics Conference, Lisbon, Portugal*. Citeseer, 2009.

[34] Mark Fiala. Artag, a fiducial marker system using digital techniques. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 590–596. IEEE, 2005.

[35] Pezhman Firoozfam and Shahriar Negahdaripour. Theoretical accuracy analysis of n-ocular vision systems for scene reconstruction, motion estimation, and positioning. In *Proceedings of the 3D Data Processing, Visualization, and Transmission, 2nd International Symposium*, 3DPVT '04, pages 888–895, Washington, DC, USA, 2004. IEEE Computer Society.

[36] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[37] Evelyn Fix and Joseph L Hodges Jr. Discriminatory analysis-nonparametric discrimination: Small sample performance. Technical report, DTIC Document, 1952.

[38] S Fukashiro, PV Komi, M Järvinen, and M Miyashita. Comparison between the directly measured achilles tendon force and the tendon force calculated from the ankle joint moment during vertical jumps. *Clinical Biomechanics*, 8(1):25–30, 1993.

[39] Theo Gevers, Arjan Gijsenij, Joost Van de Weijer, and Jan-Mark Geusebroek. *Color in computer vision: Fundamentals and applications*, volume 24. Wiley. com, 2012.

[40] MJ Gittoes and David G Kerwin. Component inertia modeling of segmental wobbling and rigid masses. *Journal of applied biomechanics*, 22(2):148, 2006.

[41] DL Glos, DL Butler, ES Grood, and MS Levy. In vitro evaluation of an implantable force transducer (ift) in a patellar tendon model. *Journal of biomechanical engineering*, 115(4A):335–343, 1993.

[42] George E Gorton III, David A Hebert, and Mary E Gannotti. Assessment of the kinematic variability among 12 motion analysis laboratories. *Gait & posture*, 29(3):398–402, 2009.

[43] T. Harju. Graph theory. Technical report, University of Turku, Department of Mathematics, 01 2011.

[44] E. Harless. The static moments of human limbs (in german). *Treatises of the Math.-Phys. Class of the Royal Acad. of Sc. of Bavaria*, 8:69–96 and 257–294, 1860.

[45] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[46] Richard I Hartley and Peter Sturm. Triangulation. *Computer vision and image understanding*, 68(2):146–157, 1997.

[47] Hans Theodore Edward Hertzberg, Clarence Wesley Dupertuis, and Irvin Emanuel. *Stereophotogrammetry as an anthropometric tool*, volume 58. Wright Air Development Center, 1957.

[48] Stanley Hoppenfeld, Richard Hutton, and Hugh Thomas. Physical examination of the spine and extremities, 1976.

[49] James G Howe, Clay Wertheimer, Robert J Johnson, Claude E Nichols, Malcolm H Pope, and Bruce Beynnon. Arthroscopic strain gauge measurement of the normal anterior cruciate ligament. *Arthroscopy: The Journal of Arthroscopic & Related Surgery*, 6(3):198–204, 1990.

[50] Joseph P Hunter, Robert N Marshall, and Peter J McNair. Relationships between ground reaction force impulse and kinematics of sprint-running acceleration. *J Appl Biomech*, 21(1):31–43, 2005.

[51] Intel. *Open Source Computer Vision Library, Version 2.4*, 2013.

[52] Ales Jaklic, Ales Leonardis, and Franc Solinas. *Segmentation and recovery of superquadrics*, volume 20. Springer, 2000.

[53] Takaaki Kaiga, Takashi Yukawa, Kazutaka Mitobe, Takeshi Miura, Hideo Tamamoto, and Noboru Yoshimura. Magnetic motion capture system measuring movements of hands and a body simultaneously. In *ACM SIGGRAPH 2007*, page 79. ACM, 2007.

[54] Ken-ichi Kanatani. Motion segmentation by subspace separation and model selection. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 586–591. IEEE, 2001.

[55] Donghoon Kang, Yujin Jung, Anjin Park, and Jinwook Kim. Human body motion capture system using magnetic and inertial sensor modules, 2011.

[56] Dimitrios Katsoulas, Christian Cea Bastidas, and Dimitrios Kosmopoulos. Superquadric segmentation in range images via fusion of region and boundary information. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(5):781–795, 2008.

[57] Takeshi Kawano, Yoshihiro Ban, and Kuniaki Uehara. A coded visual marker for video tracking system based on structured image analysis. In *Mixed and Augmented Reality, 2003. Proceedings. The Second IEEE and ACM International Symposium on*, pages 262–263. IEEE, 2003.

[58] Kevin G Keegan, GJ Baker, MJ Boero, GJ Pijanowski, and JW Phillips. Evaluation of support bandaging during measurement of proximal sesamoidean ligament strain in horses by use of a mercury strain gauge. *American journal of veterinary research*, 53(7):1203, 1992.

[59] Neelesh Kumar, Nissan Kunju, Amod Kumar, and BS Sohi. Active marker based kinematic and spatio-temporal gait measurement system using labview vision. *Journal of scientific & industrial research*, 69(8):600–605, 2010.

[60] MA Lafortune, PR Cavanagh, HJd Sommer, and A Kalenak. Three-dimensional kinematics of the human knee during walking. *Journal of Biomechanics*, 25(4):347–357, 1992.

[61] Ales Leonardis, Ales Jaklic, and Franc Solina. Superquadrics for segmenting and modeling range data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(11):1289–1295, 1997.

[62] JL Lewis, WD Lew, JA Hill, P Hanley, K Ohland, S Kirstukas, and RE Hunter. Knee joint motion and ligament forces before and after acl reconstruction. *Journal of biomechanical engineering*, 111(2):97, 1989.

[63] Frank Lindner, Karen Roemer, and Thomas L Milani. Analysis of skeletal motion kinematics for a knee movement cycle. In *ISBS symposium, Ouro Preto, Brazil*, 2007.

[64] M.I. A. Lourakis and A.A. Argyros. Sba: A software package for generic sparse bundle adjustment. *ACM Trans. Math. Software*, 36(1):1–30, 2009.

[65] Luca Lucchese. Geometric calibration of digital cameras through multi-view rectification. *Image and Vision Computing*, 23(5):517–539, 2005.

[66] Arne Lundberg, Ola K Svensson, Carin Bylund, and Goran Selvik. Kinematics of the ankle-foot complex - part 3: influence of leg rotation. *Foot & Ankle International*, 9(6):304–309, 1989.

[67] Q-T Luong and Olivier D Faugeras. Self-calibration of a moving camera from point correspondences and fundamental matrices. *International Journal of computer vision*, 22(3):261–289, 1997.

[68] GE Lutz, RA Palmitier, KN An, and EY Chao. Comparison of tibiofemoral joint forces during open-kinetic-chain and closed-kinetic-chain exercises. *The Journal of bone and joint surgery. American volume*, 75(5):732–739, 1993.

[69] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2:49–55, 1936.

[70] Florence I Mahoney. Functional evaluation: the barthel index. *Maryland state medical journal*, 14:61–65, 1965.

[71] Stephane Mallat. *A wavelet tour of signal processing: the sparse way.* Access Online via Elsevier, 2008.

[72] Jiri Matas, Ondrej Chum, Martin Urban, and Tomás Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and vision computing*, 22(10):761–767, 2004.

[73] Stephen J Maybank and Olivier D Faugeras. A theory of self-calibration of a moving camera. *International Journal of Computer Vision*, 8(2):123–151, 1992.

[74] John T McConville, Charles E Clauser, Thomas D Churchill, Jaime Cuzzi, and Ints Kaleps. Anthropometric relationships of body and body segment moments of inertia. Technical report, DTIC Document, 1980.

[75] Thomas B Moeslund and Erik Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3):231–268, 2001.

[76] Thomas B Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, 104(2):90–126, 2006.

[77] Raymond A Morano, Cengizhan Ozturk, Robert Conn, Stephen Dubin, Stanley Zietz, and J Nissano. Structured light using pseudorandom codes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(3):322–327, 1998.

[78] L Mündermann, S Corazza, and Thomas P Andriacchi. The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications. *Journal of NeuroEngineering and Rehabilitation March 2006*, 3:6, 2006.

[79] Mitsuru Nakazawa, Ikuhisa Mitsugami, Yasushi Makihara, Hozuma Nakajima, Hitoshi Habe, Hirotake Yamazoe, and Yasushi Yagi. Dynamic scene reconstruction using asynchronous multiple kinects. In *21st International Conference on Pattern Recognition*, 2012.

[80] James F. O'Brien, Robert E. Bodenheimer, Gabriel J. Brostow, and Jessica K. Hodgins. Automatic joint parameter estimation from magnetic motion capture data. In *Proceedings of Graphics Interface 2000*, pages 53–60, May 2000.

[81] Ö Fuad Öken, Öznur Öken, A Özgür Yıldırım, Güneş Yavuzer, Murat Gülçek, Vuslat Sema Ünal, and Ahmet Uçaner. Gait symmetry in patients with unilateral partial hip arthroplasty. *Eklem hastalıkları ve cerrahisi= Joint diseases & related surgery*, 21(2):86, 2010.

[82] Yuji Oyamada, Pascal Fallavollita, and Nassir Navab. Single camera calibration using partially visible calibration objects based on random dots marker tracking algorithm. *ISMAR, Workshop on Tracking Methods and Applications, 2012*, 1:–, 2012.

[83] Julien Pilet, Andreas Geiger, Pascal Lagger, Vincent Lepetit, and Pascal Fua. An all-in-one solution to geometric and photometric calibration. In *Mixed and Augmented Reality, 2006. ISMAR 2006. IEEE/ACM International Symposium on*, pages 69–78. IEEE, 2006.

[84] Katrin Pirker, Matthias Rüther, Horst Bischof, Falko Skrabal, and Georg Pichler. Human body volume estimation in a clinical environment. *AAPR/OAGM: Challenges in the Biosciences: Image Analysis and Pattern Recognition Aspects, Stainz Austria*, 1:–, 2009.

[85] Bérangère Ravary, Philippe Pourcelot, Claude Bortolussi, Stanislas Konieczka, and Nathalie Crevier-Denoix. Strain and force transducers used in human and veterinary tendon and ligament biomechanical studies. *Clinical biomechanics*, 19(5):433–447, 2004.

[86] C Reinschmidt, AJ Van Den Bogert, BM Nigg, A Lundberg, and N Murphy. Effect of skin movement on the analysis of skeletal knee joint motion during running. *Journal of biomechanics*, 30(7):729–732, 1997.

[87] Daniel Roetenberg, Henk Luinge, and Per Slycke. Xsens mvn: full 6dof human motion tracking using miniature inertial sensors, 2009.

[88] Peter Sandilands, Myung Geol Choi, and Taku Komura. Capturing close interactions with objects using a magnetic motion capture system and a rgbd sensor. In Marcelo Kallmann and Kostas E. Bekris, editors, *MIG*, volume 7660 of *Lecture Notes in Computer Science*, pages 220–231. Springer, 2012.

[89] Xueguang Shao and Chaoxiong Ma. A general approach to derivative calculation using wavelet transform. *Chemometrics and intelligent laboratory systems*, 69(1):157–165, 2003.

[90] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, 2000.

[91] Takaaki Shiratori, Hyun Soo Park, Leonid Sigal, Yaser Sheikh, and Jessica K Hodgins. Motion capture from body-mounted cameras. In *ACM Transactions on Graphics (TOG)*, volume 30, page 31. ACM, 2011.

[92] Jamie Shotton, Toby Sharp, Alex Kipman, Andrew Fitzgibbon, Mark Finocchio, Andrew Blake, Mat Cook, and Richard Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013.

[93] Franc Solina and Ruzena Bajcsy. Recovery of parametric models from range images: The case for superquadrics with global deformations. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(2):131–147, 1990.

[94] Joshua A Staples and Timothy A Davis. Motion capture for the masses, 2006.

[95] Satoshi Suzuki et al. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1):32–46, 1985.

[96] Tomáš Svoboda, Daniel Martinec, and Tomáš Pajdla. A convenient multicamera self-calibration for virtual environments. *PRESENCE: teleoperators and virtual environments*, 14(4):407–422, 2005.

[97] William R Taylor, Rainald M Ehrig, Georg N Duda, Hanna Schell, Petra Seebeck, and Markus O Heller. On the influence of soft tissue coverage in the determination of bone kinematics using skin markers. *Journal of orthopaedic research*, 23(4):726–734, 2005.

[98] Jing Tong, Jin Zhou, Ligang Liu, Zhigeng Pan, and Hao Yan. Scanning 3d full human bodies using kinects. *Visualization and Computer Graphics, IEEE Transactions on*, 18(4):643–650, 2012.

[99] Bill Triggs. Autocalibration from planar scenes. In *Computer Vision-ECCV'98*, pages 89–105. Springer, 1998.

[100] Roberto Tron and René Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[101] Emanuele Trucco, Andrea Fusiello, and Vito Roberto. Robust motion and correspondence of noisy 3-d point sets with missing data. *Pattern recognition letters*, 20(9):889–898, 1999.

[102] Michael Warren, David McKinnon, and Ben Upcroft. Online calibration of stereo rigs for long-term autonomy. In *International Conference on Robotics and Automation (ICRA)*, Karlsruhe, 2013.

[103] Alexander Weiss, David Hirshberg, and Michael J Black. Home 3d body scans from noisy image and range data. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1951–1958. IEEE, 2011.

[104] Greg Welch and Eric Foxlin. Motion tracking: No silver bullet, but a respectable arsenal. *Computer Graphics and Applications, IEEE*, 22(6):24–38, 2002.

[105] Theodore Wild. Simplified volume measurement with the polar planimeter. *Surveying and Mapping*, 14(2):218–220, 1954.

[106] Markus Windolf, Nils Götzen, and Michael Morlock. Systematic accuracy and precision analysis of video motion capturing systems exemplified on the vicon-460 system. *Journal of biomechanics*, 41(12):2776–2780, 2008.

[107] David A Winter. *Biomechanics and motor control of human movement.* Wiley. com, 2009.

[108] Yijun Xiao and J Siebert. Building superquadric men from 3d whole-body scan data. In *4th IEEE Chapter Conference on Applied Cybernetics*, pages 82–88, 2005.

[109] S. Yabukami, H. Kikuchi, M. Yamaguchi, K. I. Arai, K. Takahashi, A. Itagaki, and N. Wako. Motion capture system of magnetic markers using three-axial magnetic field sensor. *Magnetics, IEEE Transactions on*, 36(5):3646–3648, 2000.

[110] Jingyu Yan and Marc Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *Computer Vision–ECCV 2006*, pages 94–106. Springer, 2006.

[111] Maurice R Yeadon. The simulation of aerial movement. a mathematical inertia model of the human body. *Journal of Biomechanics*, 23(1):67–74, 1990.

[112] V Zatsiorsky and V Seluyanov. The mass and inertia characteristics of the main segments of the human body. *Biomechanics VIII-B*, 56(2):1152–1159, 1983.

[113] Zhengyou Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 666–673. Ieee, 1999.

[114] Wenping Zhao, Jinxiang Chai, and Ying-Qing Xu. Combining marker-based mocap and rgb-d camera for acquiring high-fidelity hand motion data. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '12, pages 33–42, Aire-la-Ville, Switzerland, Switzerland, 2012. Eurographics Association.

[115] Ke Zheng, Qiang Zhu, Yueting Zhuang, and Yuehe Pan. Motion processing in tight-clothing based motion capture. *Robot Vision. Auckland*, 1:1–5, 2001.