Surface Measurement and Tracking of Human Body Parts from Multi Station Video Sequences

Nicola D'Apuzzo

Zurich, October 2003

This publication is an edited version of:

Diss. ETH No. 15271

Surface Measurement and Tracking of Human Body Parts from Multi Station Video Sequences

A dissertation submitted to the
SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of
Doctor of Technical Sciences

presented by
NICOLA MICHELE D'APUZZO
Dipl. Masch. Ing. ETH

born July 11, 1970
citizen of Bellinzona, TI

accepted on the recommendation of

Prof. Dr. Armin Grün, examiner
ETH Zurich, Switzerland

Prof. Dr. Henrik Haggrén, co-examiner
Helsinki University of Technology, Espoo, Finland

Prof. Dr. Hans-Peter Meinzer, co-examiner
German Cancer Research Center, Heidelberg, Germany

Zurich 2003

IGP Mitteilung Nr. 81

Surface Measurement and Tracking of Human Body Parts
from Multi Station Video Sequences
Nicola D'Apuzzo

Institut für Geodäsie und Photogrammetrie
Eidgenössische Technische Hochschule Zürich
ETH Hönggerberg
CH-8093 Zürich

ISSN 0252-9335
ISBN 3-906467-44-9

.

Ce que peut la vertu d'un homme
ne se doit pas mesurer par ses efforts,
mais par son ordinaire.

The strength of a man's virtue
must not be measured by his efforts,
but by his ordinary doings.

Blaise Pascal (1623-1662)

Abstract

This work pertains to surface measurement and surface tracking of human body parts using video sequences acquired by multiple cameras. Traditionally, and even today, research and commercial applications were concentrated either on the measurement and modeling of the human face and body or on the capture of movement of the whole body and facial expressions. In this work, a method is presented to treat both the surface measurement aspects and the surface tracking aspects in a unique process. The proposed method can be applied to measure either the surface of a static human body part or the moving surface of a dynamic event. In the latter case, the 3-D data gained can be of two different types: surface measurement of the part of interest in the form of a 3-D point cloud for each recorded time step or surface tracking in form of a vector field of 3-D trajectories.

The process is composed of seven steps: (1) calibration of the system, i.e., establishing the internal and external orientation of the cameras and the parameters modeling the lens distortions; (2) acquisition of multi-image sets and/or multi-image sequences; (3) matching process, i.e., establishing correspondences in the multi-images; (4) computation of the 3-D point cloud for each matched multi-image set; (5) surface tracking in the multi-image sequences; (6) establishing a 3-D vector field of trajectories (position, velocity, acceleration); and (7) tracking key-points in the vector field of trajectories. In the case of surface measurement, only the first four steps are required.

The accurate measurement of a human body part starts with an adequate acquisition of the required data. In the case of static surface measurement are used multi-images (multiple images acquired from different positions in the space describing the same scene), while in the case of dynamic surface measurement and surface tracking are used multi-image sequences (multi-images acquired during a time interval). For the acquisition of multi-images different systems can be applied with differing levels of quality depending on the cameras used.

The orientation and calibration processes establish the position and orientation of the camera sensors in 3-D space, the parameters describing the internal geometry of the imaging device, and the parameters modeling the distortions caused by the optical system. A thorough determination of all the parameters is required for accurate measurement using photogrammetric techniques.

The goal of the automatic matching process is the determination of a dense set of corresponding points in the multi-images on the part of surface of interest. The process uses a stereo matcher based on least squares matching techniques. The automatic matching process begins by defining several seed points. Starting from them, a dense and robust set of corresponding points covering the entire interested region is generated. The 3-D coordinates of the matched points are then computed by forward ray intersection using the results of the calibration process. The strategy is customized to the characteristics of the surface of the human body. Moreover, it is designed to reduce the required processing time to a minimum.

The basic idea of the multi-image tracking process is tracking corresponding points in the multi-images through the sequence and computing their 3-D trajectories. Velocities and accelerations are also computed at each time step. The process is based on least squares matching techniques. These are applied to determine the spatial correspondences between the images acquired simultaneously from different views, as well as to determine the temporal correspondences between subsequent frames.

The proposed process can be used to track well defined points on the human body surface. Trajectories of single points, however, are not sufficient to understand and record the motion and movement of a human or the changes of the surface of human body parts. Accordingly, the tracking process is extended to simultaneously track a dense set of points belonging to a common surface. In this case, the result of the tracking process can be considered to be a 3-D vector field of trajectories.

To solve additional problems caused by occlusions, lack of texture, loss of tracked points and the appearance of new points, the concept of key-points is introduced. The key-points are 3-D regions, defined in the vector field of trajectories, whose size can vary and whose position is defined by their center of gravity. The key-points are interactively defined in a graphical user interface and tracked simply; the position in the next time step is computed as the mean value of the displacements of all the trajectories contained inside the 3-D region.

Graphical user interfaces were developed and implemented for all of the processes employed, including the multi-image acquisition, the calibration and orientation procedures, the automatic matching process and the visualization of the results as 3-D point clouds and 3-D vector field of trajectories.

To demonstrate the multiple functionality of the method, three different applications are presented: high accuracy measurement of human faces using five CCD cameras, measurement of a blood vessel branching casting fixed in a rotating frame using three CCD cameras and full body motion capture without markers using video sequences acquired by two or three synchronized CCD cameras.

Riassunto

Questo lavoro tratta la misurazione e il tracking di superfici di parti del corpo umano utilizzando videosequenze registrate con più telecamere. Finora, la ricerca e le applicazioni commerciali in questo settore si sono concentrate unicamente sulla misurazione e modellizzazione del viso e del corpo umano e sulla registrazione del movimento del corpo umano o di espressioni del viso. In questo lavoro, invece, viene presentato un metodo che integra in un unico processo le procedure sia di misurazione di superfici sia di registrazione del movimento del corpo umano. Infatti, il metodo proposto può venire utilizzato per la misurazione di superfici statiche di parti del corpo umano, come pure per la misurazione di superfici in movimento. In quest'ultimo caso, i dati tridimensionali ottenuti possono produrre una misurazione della superficie della parte interessata sotto forma di nuvola di punti tridimensionali per ogni passo temporale oppure un tracking della superficie sotto forma di campo vettoriale di traiettorie tridimensionali.

L'intero processo è composto dai seguenti passaggi: (1) calibrazione del sistema, determinazione dell'orientamento interno ed esterno delle telecamere, inclusi i parametri di modellizzazione delle distorsioni ottiche; (2) acquisizione di multi-immagini e/o sequenze di multi-immagini; (3) processo di matching, determinazione di corrispondenze nelle multi-immagini; (4) calcolo delle nuvole di punti tridimensionali per ogni gruppo di multi-immagini sottoposte al processo di matching; (5) tracking di superfici nelle sequenze di multi-immagini; (6) determinazione del campo vettoriale tridimensionale di traiettorie (posizione, velocità, accelerazione); (7) tracking di punti chiave nel campo vettoriale di traiettorie. Nel caso della misurazione di superfici, sono richiesti solamente i primi quattro passaggi.

La misurazione accurata di parti del corpo umano inizia con un'adeguata acquisizione di dati. Nel caso di misurazioni statiche di superfici vengono utilizzate delle multi-immagini (immagini multiple che descrivono la stessa scena da posizioni differenti), mentre nel caso di misurazioni dinamiche o di tracking di superfici vengono utilizzate delle sequenze di multi-immagini (multi-immagini acquisite durante un intervallo di tempo). Per l'acquisizione di multi-immagini vari sistemi possono essere utilizzati con differenti livelli di qualità a dipendenza delle telecamere impiegate.

I processi di orientamento e di calibrazione hanno lo scopo di determinare la posizione e l'orientamento dei sensori, stabilire i parametri che descrivono la geometria interna delle telecamere e ottenere i parametri necessari per la modellizzazione delle distorsioni

generate dai sistemi ottici. La determinazione precisa di tutti i parametri è necessaria per ottenere un'accurata misurazione.
L'obiettivo del processo automatico di matching consiste nella determinazione di punti corrispondenti nelle multi-immagini nella parte di superficie interessata. Il processo automatico di matching utilizza uno stereo-matcher costruito con tecniche ai minimi quadrati. La strategia sviluppata per il processo automatico di matching consiste nella definizione di alcuni punti base, dai quali viene determinato automaticamente un denso gruppo di punti corrispondenti coprendo l'intera parte interessata. Le coordinate tridimensionali dei punti corrispondenti sono successivamente calcolate tramite intersezione dei raggi utilizzando i risultati del processo di calibrazione. La strategia è stata sviluppata interamente considerando le caratteristiche della superficie del corpo umano ed è stata implementata in modo da ridurre al minimo il tempo di elaborazione.
L'idea base del processo di tracking consiste nel seguire punti corrispondenti nelle multi-immagini durante la sequenza completa e nel calcolare poi le relative traiettorie tridimensionali. Velocità e accelerazione possono pure venire calcolate per ogni passo temporale. Il processo è costruito su tecniche di matching ai minimi quadrati. Queste vengono utilizzate per la determinazione delle corrispondenze spaziali fra immagini acquisite contemporaneamente da diversi punti di vista come pure per la determinazione di corrispondenze temporali fra immagini consecutive.
Il processo proposto può essere utilizzato per il tracking di punti ben definiti sul corpo umano. Le traiettorie tridimensionali di singoli punti non sono sufficienti per comprendere e descrivere il movimento di una persona o i cambiamenti della superficie di una parte del corpo umano. Per questo motivo, il processo di tracking viene esteso per seguire simultaneamente un denso gruppo di punti appartenenti a una superficie comune e il risultato può venire considerato come un campo vettoriale di traiettorie tridimensionali.
Per risolvere problemi causati da occlusioni, mancanza di texture, perdita di punti e apparizioni di nuovi punti, vengono introdotti i punti chiave. I punti chiave sono regioni tridimensionali di dimensioni variabili definiti nel campo vettoriale di traiettorie e la loro posizione è definita dal centro di gravità. Essi vengono scelti e posizionati interattivamente in una interfaccia grafica ed il processo di tracking avviene in mariera semplice: la posizione nel prossimo passo temporale è calcolata come il valore medio dello spostamento di tutte le traiettorie contenute nella regione tridimensionale.
Interfacce grafiche sono state sviluppate per tutti i processi, compresi l'acquisizione delle multi-immagini, la calibrazione e l'orientamento, il processo automatico di matching e la visualizzazione dei risultati come nuvola di punti tridimensionali e/o campo vettoriale di traiettorie tridimensionali.
Per dimostrare le funzionalità multiple del metodo proposto, sono presentate tre differenti applicazioni: misurazione ad alta precisione del viso umano tramite cinque telecamere CCD, misurazione di un calco arteriale tramite tre telecamere CCD e registrazione del movimento del corpo umano senza l'utilizzo di punti marcanti, tramite due o tre telecamere CCD sincronizzate.

Contents

Chapter 1

Introduction

This work pertains to the surface measurement and surface tracking of human body parts.
In the past and even presently, the research and commercial applications were concentrated either on the measurement and modeling of the human face and body or on the capture of the movement of the whole body and facial expressions.
In this chapter, the existing techniques for the surface measurement of human body parts, for human face modeling, for full body modeling and for motion capture are described. At the end, the methods proposed in this work are described shortly.

1.1 SURFACE MEASUREMENT OF HUMAN BODY PARTS

In recent years, the measurement of the surface of the human body has gained importance in medical applications. The relevant disciplines are orthopedics (Commean et al., 1994; Ono, 1995; Hackenberg et al., 2000), orthodontics (Motegi et al., 1996; Höflinger, 1996), physiology (Youmei, 1994; Tukuisis et al., 2001), plastic surgery (Mao et al., 2000; Okada, 2001), dermatology (Nebel, 2000; Frankowski et al., 2001) and biomechanics (Ronsky et al., 1999; Savatier et al., 2001). The common characteristic of the existing techniques applied in medical applications is the required high accuracy.
To measure the surface of human body parts, four different optical methods can be used: laser scanning, silhouette extraction, structured light and photogrammetric approaches. Laser scanning (Brunsman et al., 1997; Okada, 2001; Tukuisis et al., 2001) and structured light methods (Maas, 1992; Youmei, 1994; Wolf, 1996; Frankowski et al., 2000; Hackenberg et al., 2000; Sitnik and Kujawinska, 2000) are widely used to measure objects in a macroscopic scale and are also applied for the measurement of the human body surface in a microscopic scale (Frankowski et al., 2001). They have common acquisition and processing characteristics. The precision of the measurement and the simplicity of use and the wide range of software packages available for the processing, editing and modeling, have made them the most widely used systems for surface measurement. However, depending on the size of the surface to measure, the acquisition time can range from seconds to half minute, during which the subject must remain stationary. This fact makes it difficult to obtain stable measurement results, because a human always moves slightly during the data acquisition (Marshall and Gilby, 2001). Moreover, these techniques cannot be used to measure dynamically moving surfaces.

In the case of silhouette extraction based methods (Zheng, 1994; Matsumoto et al., 1999; Savatier et al., 2001), multiple cameras, a single moving camera or a single camera combined with a rotating platform are used to acquire a set of images around the object. The images are processed to extract silhouettes. Intersecting the surfaces generated by silhouettes and projection centers of each image, a coarse 3–D model of the imaged body part can be determined. The detail of the model increases with the number of acquired images. However, the accuracy of this method is limited. Systems using multiple cameras can acquire images simultaneously, allowing the recording of dynamic events (e.g., containing movements).
Photogrammetric techniques (Frobin and Hierholzer, 1983; Banda et al., 1992; Gaebel et al., 1992; Mitchell, 1992; Minakuchi et al., 1995; Boersma, 2000; Nebel et al., 2001; D'Apuzzo, 2002c) offer instead accurate surface measurement, even of dynamic processes (Dorffner, 1996; Maas, 1997), by the simultaneous acquisition of images from different directions. A thorough calibration of the camera system and an accurate establishment of image correspondences are required to achieve high accuracy.

1.2 HUMAN FACE MODELING

Modeling and measurements of the human face have wide applications including medical purposes (Banda et al., 1992; Gaebel and Kakoschke, 1996; Koch et al., 1996; Motegi et al., 1996; Thomas and Newton, 1996; D'Apuzzo, 1998; Okada, 2001), computer animation (Pighin et al., 1998; Blanz and Vetter, 1999; Lee and Magnenat-Thalmann, 2000; Liu et al., 2000; Marschner et al., 2000; Sitnik and Kujawinska, 2000; Ju and Siebert, 2001b), video surveillance (C.N.N., 2001), lip reading systems (Minakuchi et al., 1995), video teleconferencing and virtual reality (De Carlo et al., 1998; Borghese and Ferrari, 2000; Fua, 2000; Shan et al., 2001). The issues that must be considered to model the face of a real person are: how realistic and accurate the obtained shape is, how long it takes to get a result, how simple the equipment is and how much the equipment costs.
The different approaches to enable the reconstruction of a human face can be classified depending on the requirements. For animation, virtual reality and teleconferencing, the photorealistic aspect is essential. In contrast, high accuracy is required for medical applications. Two different groups of methods can also be distinguished based on their data source: the first using range digitizers and the second using only images.
To date, the most popular measurement technique is laser scanning (Motegi et al., 1996; Hasegawa et al., 1999; Marschner et al., 2000; Okada, 2001); for example, the head scanner of Cyberware^{TM}(Cyberware, 2003). However, these scanners are expensive and the data are usually noisy, requiring manual editing. Another solution is offered by the structured light range digitizers (Proesmans and Van Gool, 1996; Wolf, 1996; Sitnik and Kujawinska, 2000) which are usually composed of a stripe projector and one or more CCD cameras. These can be used for face reconstruction with relatively inexpensive equipment as compared to laser scanners. The accuracy of both systems is satisfactory for static objects; however, the acquisition time required for a face scan can range from seconds to half a minute depending on the used system. Thus, a person must remain stationary during the measurement. Not only does this place a burden on the subject, but it is also difficult to obtain stable measurement results. In fact, even when the acquisition time is short, the person moves slightly

involuntarily.
A different approach to face modeling uses images as source data. Various image-based techniques have been developed. They can be distinguished by the type of source data used, e.g., a single image, two orthogonal images, a set of images, video sequences, multiple images acquired simultaneously.
Parametric face modeling techniques (Blanz and Vetter, 1999) start from a single image to generate a complete 3–D model of the face. Exploiting the statistics of a large data set of 3–D face scans, the face model is built by applying pattern classification methods. The results are impressively realistic, however the reconstructed shape is an approximation of the real face and can not be used for accurate measurements.
Computer face models can also be created from two orthogonal views (Ip and Yin, 1996). Manual intervention is required for the modeling process by selecting feature points in the images. It is a simplified method to produce realistic models of human faces. However, also in this case, the shape obtained represent an approximation of the real face. To solve this problem, some solutions (Lee and Magnenat-Thalmann, 2000) work in combination with range data acquired by laser scanners.
Another image-based method consists of automatically extracting the contour of the head from a set of images acquired around the person (Zheng, 1994; Matsumoto et al., 1999). The data are then combined to form a volumetric model of the head. The set of images can be acquired by moving a single camera around the head or having the camera fixed and the face turning. The systems are fast and completely automatic, however the accuracy of the methods is low.
Video sequences based methods (Fua and Miccio, 1997; Pighin et al., 1998; Fua, 1999, 2000; Liu et al., 2000; Shan et al., 2001) use photogrammetric techniques to recover 3–D data from the images acquired by video cameras. A generic 3-D face model is then deformed to fit the recovered data, which is usually noisy. These techniques are fully automatic but may perform poorly on faces with unusual features or other significant deviations from the normal.
High accuracy measurement of real human faces can be achieved by photogrammetric solutions which combine a thorough calibration process with the use of synchronized CCD cameras to acquire simultaneously multi-images (Gruen and Baltsavias, 1989; Banda et al., 1992; Minakuchi et al., 1995; D'Apuzzo, 1998; Borghese and Ferrari, 2000; D'Apuzzo, 2001b). To increase the reliability and robustness of the results some techniques use the projection of an artificial texture on the face (Banda et al., 1992; D'Apuzzo, 2002b). The high accuracy potential of this approach can however result in a time expensive processing.

1.3 FULL BODY MODELING

In the last few years, the demand for 3-D models of human bodies has dramatically increased. The applications include medicine, biometry and manufacturing of objects to be fitted to a specific person or group of persons. However, currently the fields where 3-D models of humans are mostly used are virtual reality, the movie industry, pure animation and computer games (Anil and Gabor, 2001). This can be explained by the high costs of the required hardware and software available to date.
Two issues have also to be considered: the portability of the equipment, its simplicity of use and the achieved accuracy and resolution of the obtained 3-D models. For

virtual reality, the movie industry, pure animation and computer game applications (Balder et al., 1999; Badler, 2000; Siebert and Ju, 2000; Ju and Siebert, 2001a), where the shape of the human body is first defined and then animated, only an approximative measurement is required. On the other hand, a precise measurement of the body surface is required in medical applications (Bhatia et al., 1994) and manufacturing applications, for example, in the space and aircraft industry for the design of seats and suits (McKenna, 1999) or more generally in the clothes or car industries (Jones et al., 1993; Certain and Stuetzle, 1999; Bradtmiller and Gross, 1999). For these purposes, methods have been developed to extract biometric information from range data (Dekker et al., 1999). Recently, anthropometric databases have been defined (Paquet and Rioux, 1999; Robinette and Daanen, 1999); besides the shape information, they contain also other records of the person, which can be used for commercial or research purposes.

The currently used approaches for building such models are laser scanning, structured light methods, silhouette extraction based methods and photogrammetry. Laser scanners are quite standard in human body modeling, because of their simplicity of use, the acquired expertise (Brunsman et al., 1997) and the related market of modeling software (Burnsides, 1997). Structured light methods are well known and used for industrial measurement to capture the shape of parts of objects with high accuracy (Wolf, 1996) and have already been applied to build full body scanners (Bhatia et al., 1994; Horiguchi, 1998; Siebert and Marshall, 2000). The acquisition time of both laser scanning and structured light systems ranges from a couple of seconds to half minute. In the case of human body modeling, this can pose problems with accuracy caused by the need for a person to remain immobile for several seconds (Daanen et al., 1997; Marshall et al., 2001).

The silhouette extraction based methods can also be employed for full body measurements (Narayanan et al., 1998; Delamarre and Faugeras, 1999; Ertaud et al., 1999; Hilton and Gentils, 1999). Multiple cameras or a single moving camera are used to acquire a set of images around a person. A coarse 3-D model is determined by intersecting the surfaces generated by silhouette of the person and projection center of each acquired image. The system based on multiple cameras can also measure dynamic events (e.g., a person performing some movement); however, high accuracy cannot be achieved. By combining the extracted silhouettes with stereo data (Vedula et al., 1998), more robust results can be obtained.

Photogrammetric solutions have already been used successfully for the measurement of human body parts (D'Apuzzo, 2002c) and can be extended to the measurement of the entire body surface. Multiple synchronized cameras acquire images of a person simultaneously from different directions. The photogrammetric method can measure the surface of the human body very precisely if the quality of the acquired image is sufficient. In this case, the unconscious movements of the subject do not affect the measurement because of the simultaneous acquisition of all the images; moreover, it is not requested to stay immobile and (voluntary) movements can also be performed. The main advantages of this method over the others are the high accuracy potential, the possibility to measure moving persons and the lower cost of the required equipment. The disadvantage is the thorough processing. Combined solutions fit complex human body models to data extracted photogrammetrically from video sequences (Fua et al., 1998; Plänkers, 2001a).

1.4 MOTION CAPTURE

Motion capture systems are mainly used for two applications: in computer animation to increase the level of realism digitizing the desired movements performed by an actor and in biomechanics to precisely measure the movement of joints. The motion capture systems can be divided into three major groups: magnetic, optical and mechanic systems. Different characteristics can be taken into account to classify them, e.g., accuracy, processing time, method used, costs and portability of the system.
The magnetic systems (e.g., Ascension^{TM}, Polhemus^{TM}) use electromagnetic sensors connected to a computer unit which can process the data and produce 3–D data in real time. The major advantage of these systems is the direct access to the 3–D data without complex processing. For this reason they are very popular in the animation community. Wireless systems have also been developed to solve the disadvantage of restricted freedom of movement caused by the cabling.
Optical systems (e.g., Motion Analysis^{TM}, Vicon^{TM}, Qualisys^{TM}) are mostly based on photogrammetric methods where the trajectories of signalized target points on the body are measured very accurately (Tsuruoka et al., 1995; Boulic et al., 1997, 1998). They offer complete freedom of movement and interaction of different actors is also possible. In the last years, several improvements have been introduced, such as the use of smart cameras and CMOS sensors to achieve real-time 3–D data acquisition (Richards, 1999; Kadaba and Stine, 2000).
Electro-Mechanical systems (e.g., Analogus^{TM}) have recently appeared in the market: in this systems the person moving has to wear a special suit with integrated electro-mechanical sensors that register the motion of the different articulations. This method also has the advantage of real-time data transfer from the sensors to the computer with minimal processing. Moreover, it is less expensive.
Motorized video theodolites in combination with a digital video camera have also been used for human motion analysis (Anai and Chikatsu, 1999, 2000).
Motion capture can also be achieved by image-based methods. They can be divided into single camera and multiple cameras systems. Single camera systems use sequences of images acquired by a single camera. To gain three-dimensional information from single video clips, knowledge of human motion must be used. Some systems learn from provided sample training data and apply statistical methods to get the performed 3–D motion (Mahoney, 2000; Song et al., 2000; Rosales and Sclaroff, 2000). Other systems perform the tracking of defined human body models with constraints by sophisticated filtering processes (Cham and Rehg, 1999; Segawa and Totsuka, 1999; Deutscher et al., 2000).
Multiple camera systems use sequences of images acquired simultaneously by two or more cameras. Some systems assume very simple 3–D human models (e.g., articulated objects made of cylinders) whose characteristical sizes and joint angles are determined by comparing the projections of the model into the different images with the extracted silhouettes of the moving person (Delamarre and Faugeras, 1999; Cheung et al., 2000) or the extracted edges (Kinzel and Behring, 1995; Gravila and Davis, 1996). Other systems use image based tracking algorithms to track three-dimensionally the surface of the human body (D'Apuzzo, 2000) or different body parts (Ohno and Yamamoto, 1999). Mathematical models of human motion can also be used to track directly in 3–D data, which can be trajectories of known key points (Iwai et al., 1999) or dense disparity maps (Jojic et al., 1999). More sophisticated methods fit generic human

body models to extracted 3–D data from video sequences (Fua et al., 2000; Herda et al., 2000; Plänkers, 2001b; Fua et al., 2002).

1.5 CONSIDERATIONS

The four fields described above correspond with the recent research and commercial applications for surface and motion measurement regarding the human body. In this work, a method is presented to treat both the surface measurement and the surface tracking aspects in a unique process. The proposed method can be applied to measure either the surface of a static human body part or the moving surface of a dynamic event. The entire process is composed of seven steps (four, in case of surface measurement):

1. calibration of the system: establishment of the internal and external orientation of the cameras and the parameters modeling the lens distortion;
2. acquisition of multiple images and/or multiple image sequences;
3. matching process: establishment of correspondences in the multiple images (at least for the first set);
4. surface measurement: computation of the 3–D point cloud for each matched multiple image set;
5. surface tracking in the multiple image sequences;
6. establishment of a 3–D vectorfield of trajectories (position, velocity, acceleration);
7. tracking key-points in the vectorfield of trajectories.

To compare the proposed method for surface measurement (first four steps) with other systems, five aspects can be considered: the accuracy of the measurement, the time required to acquire the data, the ease of use, the total processing time and the hardware costs. Figure 1.1 shows the five aspects for three surface measurement methods: multi image photogrammetry (the proposed method), structured light digitizer and laser scanning.
The accuracy of the surface measurement system is one of the most important and significant aspect. The greatest accuracy potential is held by structured light digitizers, followed by multi-image photogrammetry and laser scanner. However, when measuring human body parts, the effective accuracy of the measurement depends strongly on the required acquisition time. In fact, even if stabilized and immobilized, humans move unconsciously (e.g., breathing, muscle contractions). Therefore, only a short acquisition time can assure the accuracy potential of the used system. Increasing the time required for data acquisition will affect negatively the accuracy of the measurement. About this aspect, the proposed method is certainly more adequate than structured light digitizers or laser scanners, since multi-images can be acquired simultaneously by using synchronized cameras. Moreover, the simultaneous acquisition of the images allows the measurement of moving surfaces and even the recording of dynamic events (surface tracking).
Two negative aspects of the proposed method compared to the others are the total processing time and the ease of use. In fact, multi image photogrammetry implies

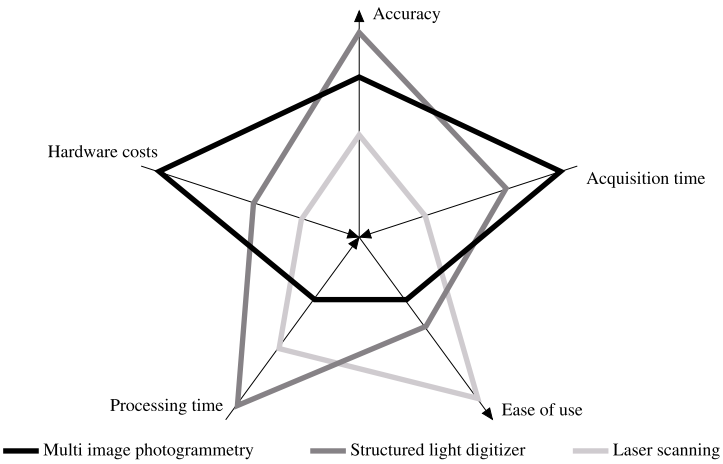

Fig. 1.1 Benchmarking of the three surface measurement techniques: multi image photogrammetry, structured light digitizer, laser scanning. Accuracy of the measurement, acquisition time, ease of use, processing time and hardware costs are considered. The farther from the center of the graph, the better it is (note the different directions of the axes).

different processing steps (system calibration, matching, computation of 3–D coordinates) whereas the other systems deliver the results quite fast. Similar is the aspect about the easiness of use: multi image photogrammetry requires the setting up of the camera system, its calibration and the image acquisition whereas for, e.g., laser scanners, the scanning device has simply to be placed and the measurement process can begin. These aspects are however less relevant.
The last considered aspect is the cost of the hardware which is in case of multi image photogrammetry limited to the cameras, lenses and frame grabber. A large variety of hardware exists with large costs differences depending of the quality, performance and characteristics. However, structured light digitizers and laser scanners always require more expensive hardware.
Considering simultaneously all aspects, the method proposed in this work result more adequate for the measurement of human body parts.

1.6 CONTENTS

The system setup and the procedure of acquiring the data are described in Chapter 2. The calibration process is presented in Chapter 3. Chapter 4 explains the matching process. Chapter 5 describes how the surface of the human body is computed. Two applications are presented: human face modeling and the measurement of a blood vessel branching. Chapter 6 presents the validation of the proposed method for surface measurement. The process of tracking (single points, surface and key-points) is explained in Chapter 7. At the end of the chapter a method where full body motion capture can be achieved using the process is presented. A short application of tracking facial expressions using image sequences acquired by a single video camera is also presented to show the flexibility of the proposed method. Chapter 8 gives the conclusions of this work. The Appendix A describes the implemented graphical user interface for the developed software and Appendix B a newly implemented acquisition system. Appendix C gives some examples of parameter sets for the surface matching and tracking processes.

Chapter 2

Data Acquisition

This chapter presents the background of CCD sensors, video signal and frame grabbers, followed by the description of the different camera systems that can be used for the acquisition of multi-images and/or multi-image sequences.

2.1 CCD CAMERAS, VIDEO SIGNAL AND FRAME GRABBERS

2.1.1 CCD Cameras and Video Signal

Cameras used in the field of digital image processing have two major components: the image acquisition unit and the image output unit. Currently, the acquisition unit is based dominantly on *CCD* (*Charge Coupled Device*) chips, which consist of separated light sensitive elements, each of which represents one pixel. A CCD pixel transforms light into a charge, which during readout is transformed into a voltage. CCD chips are small in size and weight, have high dynamic and high linearity. Moreover, due to mass production for the consumer market, they are relatively inexpensive. These reasons made them the most popular acquisition units. For machine vision applications, the sensor chips require additional features such as sensor flatness and regularity of the pixel position.

One of the basic parameters of CCD chip is the number of effective pixels or pixel elements. A typical CCD chip used in video cameras, has 768 pixels in the horizontal direction (columns) and 576 pixels in the vertical direction (rows). If larger format of the images is required, more expensive CCD arrays having more pixels may be used.

The output unit of a camera generates a video signal which is suitable for succeeding image processing devices. In standard monochrome cameras, the acquired image is transformed into a video signal conforming to one of the internationally accepted standards. For Europe this is *CCIR* (*Comité Consultatif International des Radiocommunications*) and for the United States it is RS-170 which was defined by the *EIA* (*Electronics Industries Association*). For color cameras, the additional standards (*PAL* for Europe and *NTSC* for the United states) are modified version of the original monochrome standards.

These standards are based on the constraints of traditional tube cameras and screens and seem antiquated in view of modern CCD chips and flat screen displays. Nevertheless, they are still used in most current camera devices. Therefore, it is interesting to describe how they were originally generated in traditional devices. In the tube cameras, the beam of electrons has to be controlled to scan the light sensitive area of

the pick-up tube; the same applies for the beam which generates the image on tube screens. The scan starts in the top left corner (see figure 2.1 top); encountering the end of the line (after 52 ms for CCIR) the deactivated ray runs back (*retrace*, 12 ms for CCIR) to the beginning of the third (not the second) line. In this manner, the first scan covers the field of odd lines while the second scan works on the field of even lines. This method (called *interlacing*) diminishes the flicker of monitors but leads to problems in image processing. A complete scan (called *frame*) consists of 625 lines and lasts 1/25th of a second for CCIR. During the retrace time, the horizontal sync pulse (*hsync*) is added to the video signal to indicate the beginning of the next line. Figure 2.1 shows how a standard video signal is built.

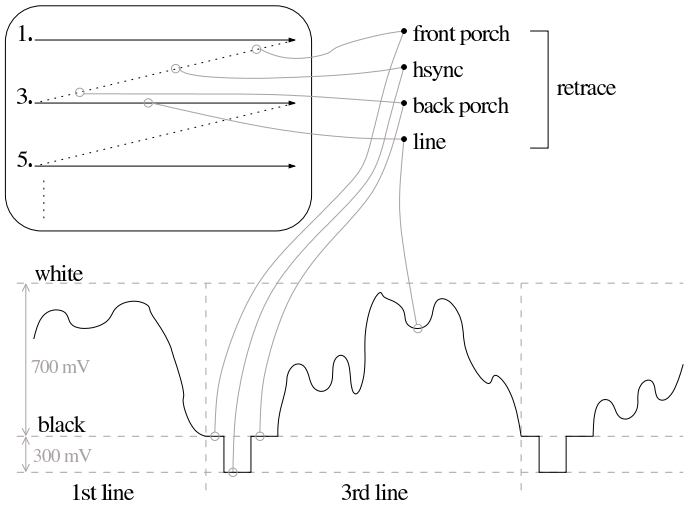

Fig. 2.1 Analog video signal according to CCIR/EIA standards (The Imaging Source, 2001a).

The vertical sync pulse (*vsync*) triggers the beginning of a new field. The vsync is a sequence of several pulses spread over 50 lines for CCIR. Therefore these lines are not usable for the standard video signal, which leads to 575 visible lines for CCIR. The synchronization of standard video signal transmission is exclusively based on the horizontal and vertical sync pulses (hsync, vsync).

Most machine vision video cameras can be synchronized externally so that the hsyncs and vsyncs of all of the cameras can be put in sync with each other. The technical term for this process is *external synchronization* or *genlock*. The simplest method is to build a synchronization chain in which the video output from one camera is used as the synchronization input of the next one. However, when the synchronization must be very accurate, an extra sync generator should be used; this option is usually offered by most machine vision frame grabbers.

Another remand from the video standards is the interlace technique which is a curse for metrology. The camera first scans the odd lines of the image and secondly the even lines. If a moving object is recorded, it would be in different positions between the first and second scans, resulting in the even and the odd lines being horizontally disarranged, causing a saw pattern effect. The simplest solution to this problem is to use only the odd lines, however the resolution in vertical direction would be halved. If the full resolution is required, special non-interlaced cameras have to be used, called *progressive scan* cameras, which are able to acquire an image in one pass. Obviously they are more expensive and require special frame grabbers.

2.1.2 Frame Grabbers

Frame grabbers read the analog video signal of a camera and transform it into a digital image sequence. Figure 2.2 shows the basic structure of a frame grabber. First, the horizontal and vertical synchronization pulses are separated from the incoming video signal (*sync separation*), indicating the beginning of new lines and the beginning of new fields. This procedure can take some time (usually a couple of frames), it can however be avoided by synchronizing externally the video sources so that the hsyncs and vsyncs of all of the cameras are in synchronization with the frame grabber (this operation is called *genlock*). Once stable synchronization of the lines and frames has been achieved, the next part of the process of acquiring an image concerns the generation of the pixels themselves. In accordance to the video standards, the *sample and hold* unit digitizes 767 pixels per line (for CCIR) collecting them in an *image buffer* which stores at least one complete frame and is used if the bandwidth of the connection to the PC (*bus*) is too small to transport without loss the digitized video data stream in the main memory of the computer.

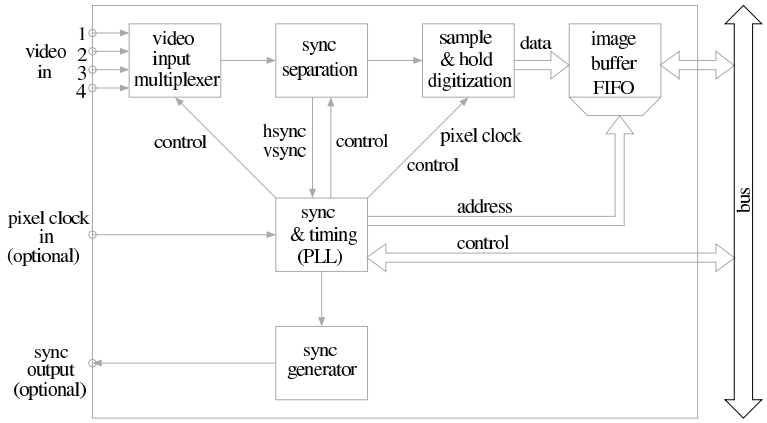

Fig. 2.2 Basic structure of a frame grabber (The Imaging Source, 2001b).

Different types of frame grabbers exist (see figure 2.3). For mobile applications, frame grabbers are available on PCMCIA cards (e.g., DFG/VPP from ImagingSource) or Video-to-FireWire converters (e.g., DFG/1394-1 from ImagingSource). The quality of the digitized images is not as high as the machine vision frame grabbers but they can be used for demonstration purposes. Low-cost machine vision frame grabbers (e.g., Matrox Meteor-II) with on board between-storage are able to acquire image sequences with full frame rate (25 Hz for CCIR) and have usually multiple composite video inputs. Frame grabbers with three independent RGB inputs can be generally used to acquire up to three monochrome video signals at the full frame rate.
Frame grabbers with multiple (usually two or three) RGB inputs (e.g., Matrox Meteor-II Multi Channel) can be used to acquire images synchronously from multiple color cameras or multiple monochrome standard CCD cameras (up to six in case of two multiple RGB inputs and up to nine for three multiple RGB inputs). These frame grabbers are however more expensive (mid-cost class).
Special frame grabbers (e.g., Matrox Meteor-II/Camera Link/1394/Digital) are required for higher resolution, for higher frame rate, for use with progressive scan cameras or for use with digital cameras. The hardware costs are substantially higher.

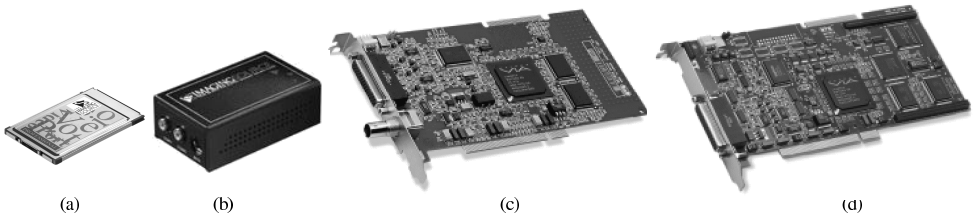

Fig. 2.3 Different types of frame grabbers: (a) PCMCIA frame grabber (DFG/VPP) with two composite video inputs; (b) Video-to-IEEE-1394 converter (DFG/1394-1) with two composite video inputs; (c) Low-cost machine vision frame grabber (Matrox Meteor-II) with 4 composite video inputs: (d) Multi-channel frame grabber (Matrox Meteor-II Multi Channel with two RGB inputs.

2.2 MULTI IMAGE ACQUISITION SYSTEMS

In this work, the term *multi-image* refers to multiple images acquired from different positions in the space describing the same scene and *multi-image sequence* refers to multi-images acquired during a time interval.
In this work are treated only cameras with standard video format (24 bit color or 8 bit black and white, 756x578 or 640x480 pixels) and frame rate (25 or 30 Hz). Other cameras with special characteristics (e.g., high speed, high resolution, high sensitivity) are not considered here because of the elevated costs.
To record a scene with movement, the multiple images have to be acquired simultaneously. The precision of the synchronization of the multiple imaging devices plays an essential role for the accuracy potential of the measurement achieved using the images.
On the other hand, to record static scenes, the multi-images can be acquired at different times without a loss of accuracy. However, for applications involving recording people, the human body cannot be considered as a static object, because a person always moves slightly unconsciously due to, for example, breathing or muscle contraction. Therefore, for surface measurement of human body parts, it is always recommended to precisely synchronize the multiple cameras.
Various methods can be used to acquire multi-image sequences. Although different camera systems have similar resolution and quantization, different levels of quality can be achieved depending on the system. They are listed below in order of decreasing accuracy potential:

- synchronized machine vision progressive scan CCD cameras,
- synchronized machine vision interlaced CCD cameras,
- digital video camcorders, synchronized using a clapper.

For surface measurement applications without tracking, only multi-images are required. In this case, also the following systems can be used for the acquisition (listed with decreasing accuracy potential):

- multiple digital still cameras,
- a single digital still camera or video camcorder, acquiring in different position,
- multiple digital web cameras.

In the following sections, the characteristics of the listed systems are described in more detail.

2.2.1 Synchronized Machine Vision Progressive Scan CCD Cameras

In term of quality and performance, the best imaging tool for the acquisition of multi-image video sequences are synchronized progressive scan CCD cameras. They differ from the more common and less expensive interlaced CCD cameras (described in section 2.2.2) in the way that a full frame is acquired simultaneously offering full resolution of the sensor also with moving objects. The cameras are more expensive and special frame grabbers are required.

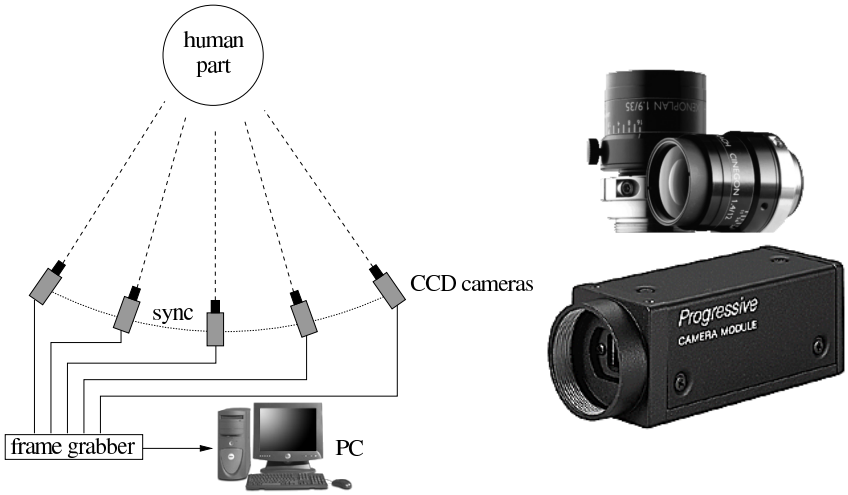

Fig. 2.4 Progressive scan CCD camera multi-image acquisition system. Left: setup of the system, the cameras are synchronized together and connected to a frame grabber which digitizes the images and transfers them to a PC. Right: progressive scan CCD camera (Sony XC-55)(bottom) equipped with machine vision c-mount lens (top).

Figure 2.4 shows the setup of an acquisition system built up with progressive scan CCD cameras. The precise electronic synchronization guarantees the simultaneous acquisition of multi-images. Depending on the number of cameras used, one or more frame grabbers are required to digitize the multi-image sequences. They are stored in the memory of a PC first and then moved to the disks.
Recently developed progressive scan cameras (e.g., Dragonfly of Point Grey Research) use the IEEE-1394 port. No frame grabber is in this case required. See the Appendix B for the description of a multi-image acquisition system based on such cameras.

2.2.2 Synchronized Machine Vision Interlaced CCD Cameras

Machine vision interlaced CCD cameras offer a less expensive but still high quality solution. The main advantage of this type of camera over the progressive scan camera is the standard video output (CCIR/EIA). In this case, low-cost frame grabbers can be used. They are usually provided with separate RGB inputs that allow the connection of three synchronized CCD cameras. This results in the least expensive high quality multi-camera acquisition system. If more than three cameras are required, multi-channel frame grabbers or multiple frame grabbers have to be used.
The setup of a multi-image acquisition system based on machine vision CCD cameras (figure 2.5) is the same as for progressive scan CCD cameras (section 2.2.1). The cameras are synchronized together and connected to a frame grabber. The image sequences are first stored in the memory of a PC and then moved to the disks.

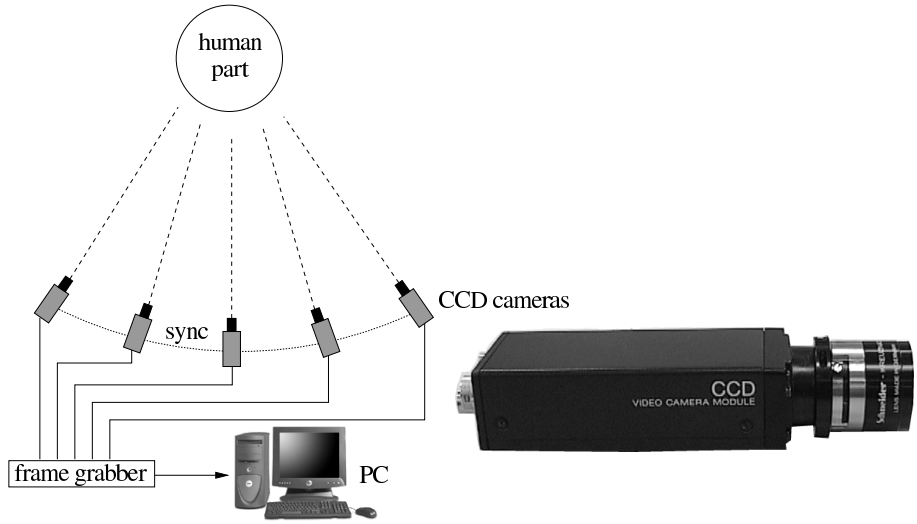

Fig. 2.5 Multi-image machine vision CCD camera acquisition system. Left: setup of the system, the cameras are synchronized together and connected to a frame grabber which digitizes the images and transfers them to a PC. Right: interlaced machine vision CCD camera (Sony XC-77 with 35mm lens).

A disadvantage of using this type of CCD cameras is the interlace effect which is caused by the fact that odd and even lines are acquired in different times. This results in saw pattern effects when movement is present, as shown in figure 2.6.
A solution for solving this problem is to use only the odd lines, duplicating them in the images and deleting the even lines. The result is shown on the right of figure 2.6, the saw pattern disappears at the cost of the resolution which is halved in the vertical direction.

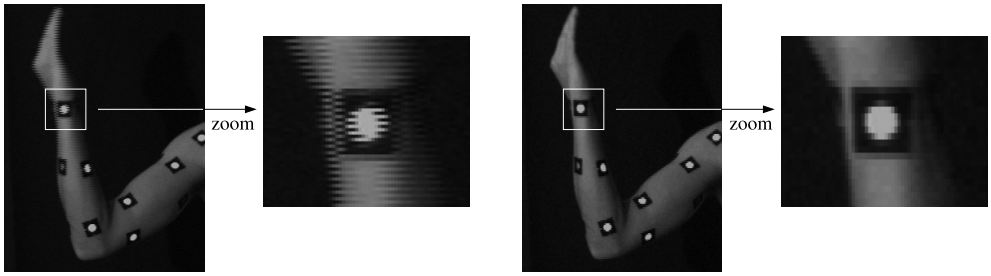

Fig. 2.6 Interlace effect. Left: the hand is moving to the right, the interlace effect can be seen as a saw pattern. Right: result after resolution reduction: the even lines are removed and the odd lines are duplicated.

2.2.3 Digital Video Camcorders

Multiple digital video camcorders can also be used for the acquisition of multi-image video sequences. This is a less expensive solution because no frame grabber is required. The digital video camcorders (for example the Sony digital HandyCam DCR-VX700E, figure 2.7 right) store the image sequence digitally on mini DV tapes, in DV format with a size of 720x576 pixels and 24 bit color resolution. The DV format is a Sony proprietary digital video and audio recording standard. A single image in this

compressed format has a size of about 140kB. The image sequences stored on the mini DV tape can be transferred without loss of quality to a PC by FireWire connection and converted into common image formats.
Figure 2.7 shows the setup of a multi-image acquisition system based on multiple digital video camcorders. No frame grabber is required. The multiple cameras can be synchronized using a *clapper* (figure 2.7 left) or something generating a "clap" sound. A "clap" sound has a clear defined start peak which can be recognized in the audio signal of the different video sequences (see figure 2.8). The sequences are then manually synchronized by aligning the audio signals. A maximum error of half frame (corresponding to 1/50 of a second, for CCIR) can be expected. Depending on the applications (e.g., slow movement), this may be acceptable or not.

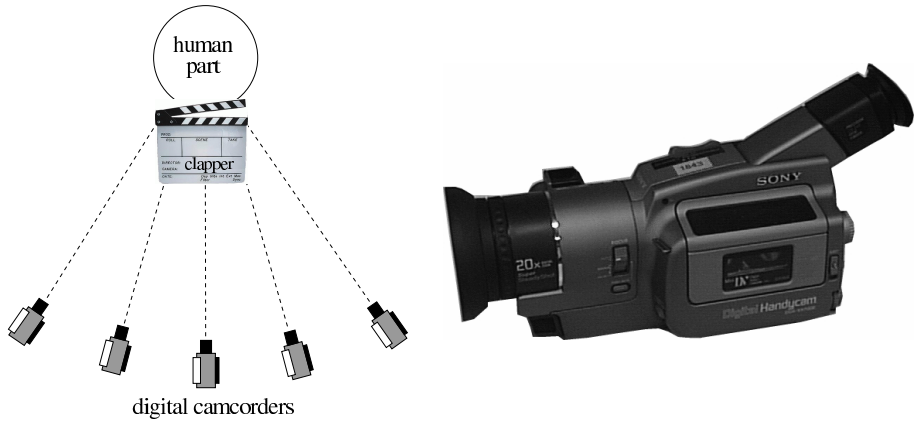

Fig. 2.7 Multi-image acquisition system based on multiple digital video camcorders (Sony DCR-VX700E). Left: setup of the system, a *clapper* is used for synchronization purposes.

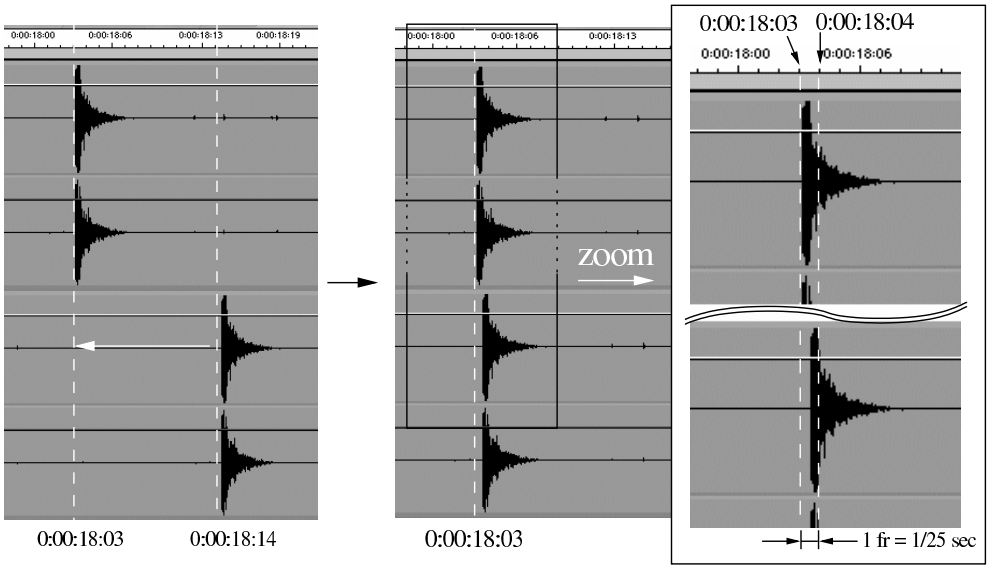

Fig. 2.8 Synchronization with the audio signal. Right: audio signal of two acquired sequences, the beginning of the "clap" sound is clearly defined. Left: the two sequences are synchronized aligning the audio signals, the maximum error is half a frame (0.04 seconds for CCIR).

2.2.4 Multiple Digital Still Cameras

Digital still cameras decrease in price from year to year. An interesting feature of some of them is the possibility to control the camera from a PC, allowing the acquisition of images from multiple cameras in short times. The acquired images are stored on memory cards and can be transferred to a PC afterward. For example, figure 2.9 (right) shows a Kodak DC215 which can acquire images with 1152x864 or 640x480 pixels. The use of larger image formats is not treated in this work.

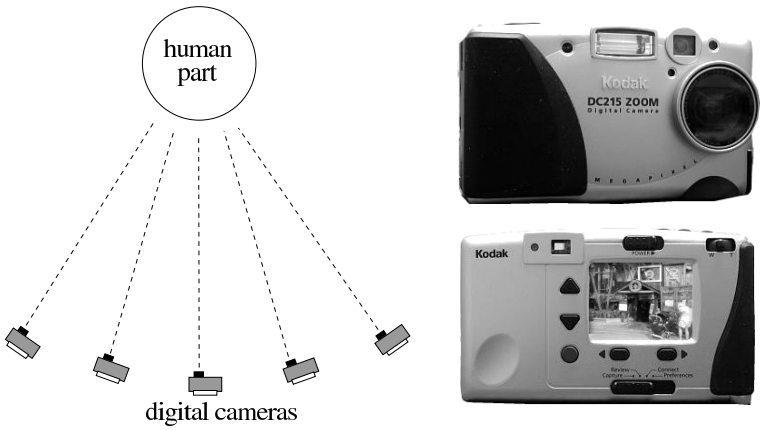

Fig. 2.9 Multiple digital still camera acquisition system. Left: setup of the system. Right: a digital still camera (Kodak DC215), bottom: back view, LCD monitor.

With this acquisition system, it is obviously not possible to acquire multi-image sequences but only multi-images. Moreover, there is a short delay (less than 1 second) between the acquisition of the different images. Therefore, the recording of movement is not possible and only static surface measurement processes can be achieved using the images acquired with this system. Since humans move slightly unconsciously during the acquisition, the accuracy potential of this system decreases with the increment of the time delay between the acquisition from the different cameras.

2.2.5 Single Moving Digital Still Camera or Digital Video Camcorder

A less expensive solution is offered by using a single digital still camera or a single video camcorder and acquiring the images by moving the camera at different positions (see figure 2.10). Although the images can be acquired relatively fast (e.g., less than ten seconds for five multi-images of a face), this method can be used only for the measurement of immobile or fixed human body parts.
A second disadvantage of this method is the requirement of more complex orientation and calibration procedures. To utilize the method implemented for this work and described in section 3.3, reference points with known 3–D coordinates have to be imaged together with the human body part (reference points on figure 2.10).

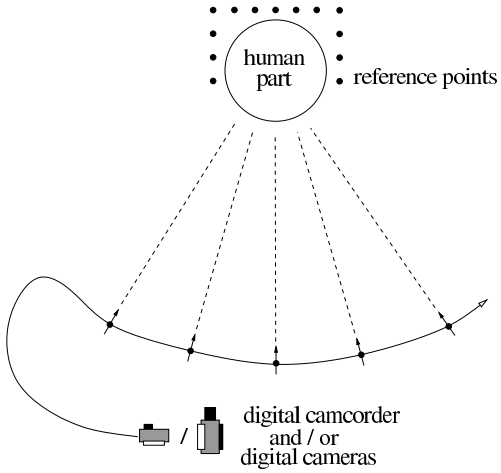

Fig. 2.10 A digital camcorder or a digital still camera is moved around the human body part to acquire multi-images. Note the reference points required by the implemented method for the establishment of the external orientation of the camera at different positions.

2.2.6 Digital Web Cameras

The least expensive solution for the acquisition of multi-images is offered by FireWire web cameras. Their cost is in fact today under 100 US$ per camera and no frame grabber is required. The digital web cameras are in fact connected to the PC by the IEEE-1394 port (also called *FireWire* or *i-link*). The advantage of this type of connection is that only a single input port on the PC is required; the multiple cameras can be connected serially to each other (see figure 2.11).

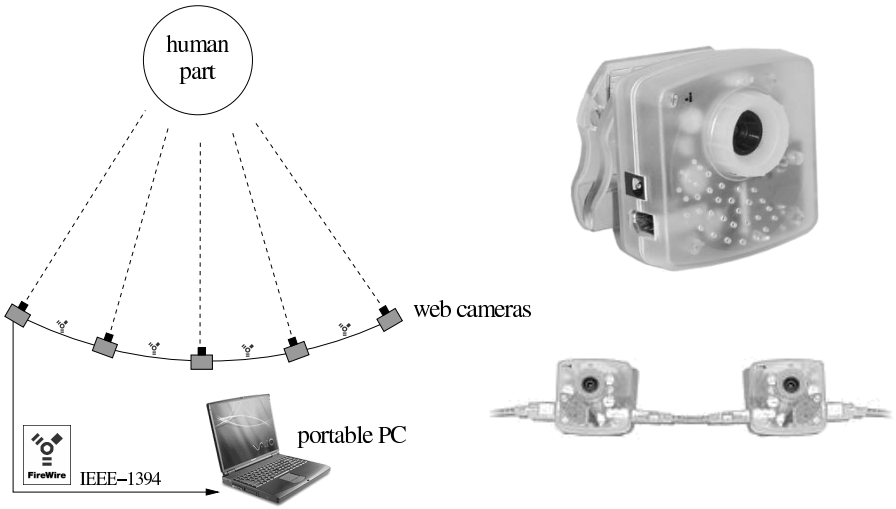

Fig. 2.11 Multi-image acquisition system using web cameras. Setup of the system: the FireWire digital camera (Unibrain Fire-i)(right) are connected together serially through the IEEE-1394 cable (right bottom), only one camera has to be connected to a (portable) PC.

For example, the Unibrain Fire-i camera (see figure 2.11, right) has digital output and several formats (color uncompressed, color compressed and grayscale uncompressed), several resolutions (640x480, 320x240 and 160x120 pixels) and several frame rates (7.5, 16 and 25 Hz) can be chosen. The serial connection allows the acquisition of

multi-images even with a portable computer (see figure 2.12); PCMCIA cards are readily available, moreover some portable computers (e.g., Sony VAIO and Apple i-Book) are provided with an integrated FireWire input.

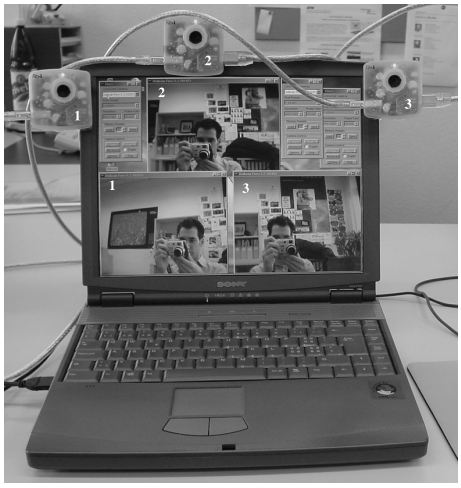

Fig. 2.12 Three FireWire digital cameras (Unibrain Fire-i) connected simultaneously to a portable computer.

Although the image quality is low and the noise is high, digital web cameras are a convenient and inexpensive solution for demonstration purposes (figure 2.12 shows three web cameras connected to a laptop).

2.2.7 Considerations

The simultaneity of the acquisition of the multi-images is essential for an accurate measurement of moving human body parts. A precise synchronization of the multiple acquisition units is available with only the two described machine vision acquisition systems (sections 2.2.1, 2.2.2). Systems based on multiple digital still cameras (section 2.2.4) and systems using digital web cameras (section 2.2.6) can be set up to acquire multi-images simultaneously. However, the cameras cannot be electronically synchronized with each other; the maximum time delay which can occur between the acquisition of the different images is therefore half a frame (1/50 of a second for CCIR). The same synchronization accuracy can be achieved with systems using multiple video camcorders (section 2.2.3), aligning the audio signals of the different sequences. Systems based on a single camera (section 2.2.5) can evidently not simultaneously acquire multi-images. In some cases, e.g., when high accuracy of the measurements is not required or if the human body part can be immobilized, the longer time delay can however be acceptable.

Chapter 3

System Calibration

As *system calibration* is defined the simultaneous calibration and orientation of all the components involved in the acquisition system. Camera *calibration* refers to the determination of the parameters describing the internal geometry of the individual imaging devices and other parameters modeling the systematic errors caused by the optical system and other sources. Camera *orientation* includes the determination of the parameters of exterior orientation to define the camera station and camera axis in the 3–D space. A thorough determination of all the parameters is required for an accurate measurement.
In the next sections, the mathematical model for the projection of the object space onto the digital image coordinate system will be described first. The different methods used in this work to calibrate the imaging systems are then presented and finally the implemented single camera bundle calibration model is described.

3.1 OBJECT SPACE-TO-IMAGE SPACE MATHEMATICAL MODEL

In this section the mathematical model for the projection of a point in the object space onto the focal plane of an imaging device, as shown in figure 3.1, is described along with the transformation into image coordinates (image space).
The object space-to-focal plane projection can be expressed by the *collinearity condition*, i.e., a point in the object space, its projection on a plane and the projection center have to lie on a straight line (dotted line in the figure 3.1). Mathematically it can be described by the following equation:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \lambda \mathbf{R} \begin{bmatrix} x' - x_p \\ y' - y_p \\ -c \end{bmatrix} + \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} \tag{3.1}$$

where

X, Y, Z	object space coordinates of point P,
x', y'	coordinates of point P' on the focal plane,
x_p, y_p	coordinates of the principal point H' on the focal plane,
c	camera constant,
$\mathbf{R}$	rotation matrix between sensor and object space coordinate systems,
X_0, Y_0, Z_0	object space coordinates of projection center O',
r_{xx}	coefficients of rotation matrix $\mathbf{R}$,
λ	scale factor for each imaging ray.

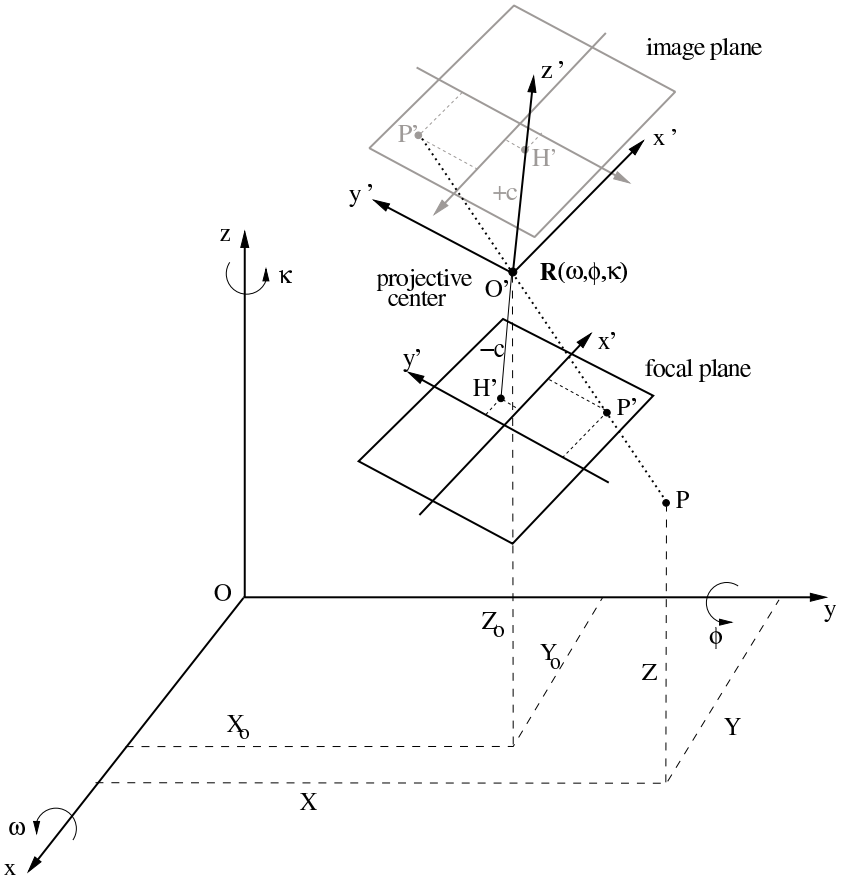

Fig. 3.1 Projection of a point P in the object space $Oxyz$ onto the focal plane.

In equation 3.1 the interior orientation is defined by the principal point x_p, y_p and the camera constant c while the exterior orientation is defined by the object coordinate of the projection center O' X_0, Y_0, Z_0 and the rotation matrix $\mathbf{R}\,(\omega, \varphi, \kappa)$ describing the rotation from the object coordinate system $Oxyz$ to the sensor coordinate system $O'x'y'z'$ where ω, φ, κ are the Euler angles around the x, y, z axes:

$$\begin{aligned}\mathbf{R} &= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \\ &= \begin{bmatrix} \cos\omega\cdot\cos\kappa & -\cos\varphi\cdot\sin\kappa & \sin\varphi \\ \cos\omega\cdot\sin\kappa+\sin\omega\cdot\sin\varphi\cdot\cos\kappa & \cos\omega\cdot\cos\kappa-\sin\omega\cdot\sin\varphi\cdot\sin\kappa & -\sin\omega\cdot\cos\varphi \\ \sin\omega\cdot\sin\kappa-\cos\omega\cdot\sin\varphi\cdot\cos\kappa & \sin\omega\cdot\cos\kappa+\cos\omega\cdot\sin\varphi\cdot\sin\kappa & \cos\omega\cdot\cos\varphi \end{bmatrix}. \end{aligned} \tag{3.2}$$

The three components in equation 3.1 are reduced to two by canceling out the scale factor λ, resulting in the *collinearity equations*:

$$\begin{aligned} x' &= x_p - c\cdot\frac{r_{11}\cdot(X-X_0)+r_{21}\cdot(Y-Y_0)+r_{31}\cdot(Z-Z_0)}{r_{13}\cdot(X-X_0)+r_{23}\cdot(Y-Y_0)+r_{33}\cdot(Z-Z_0)} \\ y' &= y_p - c\cdot\frac{r_{12}\cdot(X-X_0)+r_{22}\cdot(Y-Y_0)+r_{32}\cdot(Z-Z_0)}{r_{13}\cdot(X-X_0)+r_{23}\cdot(Y-Y_0)+r_{33}\cdot(Z-Z_0)} \end{aligned} \tag{3.3}$$

To model the distortions occurring in the focal plane, a simplified version of the parameter set introduced by Brown (1971) is used. Two additional parameters modeling the shearing and the differential scaling in x are introduced in the model as described by Beyer (1992). Thus the coordinates on the focal plane are:

$$\begin{aligned} x' &= \bar{x}' + d\bar{x}' = \bar{x}' - sc \cdot \bar{x}' + sh \cdot \bar{y}' + dx' \\ y' &= \bar{y}' + d\bar{y}' = \bar{y}' + sh \cdot \bar{x}' + dy' \end{aligned} \tag{3.4}$$

where

x', y'	coordinates in focal plane [mm], with distortion,
$\bar{x}', \bar{y}'$	coordinates in focal plane [mm], without distortion, according to eq. 3.3,
$d\bar{x}', d\bar{y}'$	distortion terms,
dx', dy'	symmetric radial and decentering lens distortion terms,
sc	scale factor in x,
sh	shear factor;

where

$$\begin{aligned} dx' &= \bar{x}' \cdot \left(k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6\right) + p_1 \cdot \left(r^2 + 2 \cdot \bar{x}'^2\right) + 2 \cdot p_2 \cdot \bar{x}' \cdot \bar{y}' \\ dy' &= \bar{y}' \cdot \left(k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6\right) + 2 \cdot p_1 \cdot \bar{x}' \cdot \bar{y}' + p_2 \cdot \left(r^2 + 2 \cdot \bar{y}'^2\right) \\ r &= \sqrt{\bar{x}'^2 + \bar{y}'^2} \end{aligned} \tag{3.5}$$

where

k_1, k_2, k_3	symmetric radial lens distortion coefficients,
p_1, p_2	decentering lens distortion coefficients.

The last equation for the description of the mathematical model is the transformation between metric coordinate system on the focal plane and pixel image coordinate system (see figure 3.2). This can be described as an affine transformation (equation 3.6).

$$\begin{aligned} x' &= \left(u - \frac{n_x}{2}\right) \cdot \mu_x \\ y' &= -\left(v - \frac{n_y}{2}\right) \cdot \mu_y \end{aligned} \tag{3.6}$$

where

u, v	coordinate in the image [pixel],
n_x, n_y	image size [pixel],
μ_x, μ_y	pixel size [mm].

The equation 3.6 can be inverted to obtain the transformation metric to pixel coordinate system:

$$\begin{aligned} u &= \frac{x'}{\mu_x} + \frac{n_x}{2} \\ v &= -\frac{y'}{\mu_y} + \frac{n_y}{2} \end{aligned} \tag{3.7}$$

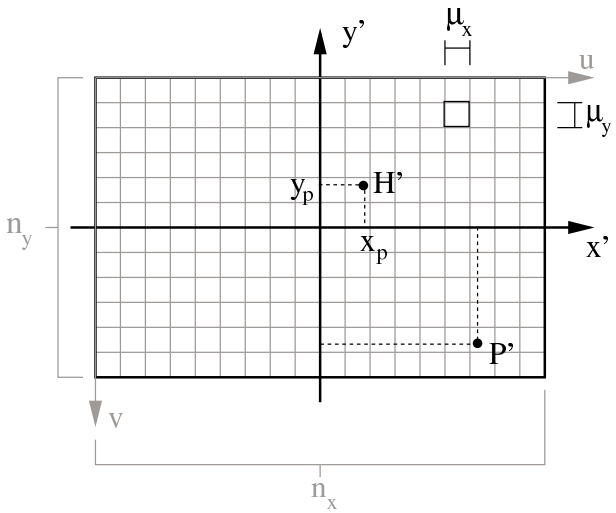

Fig. 3.2 Metric coordinate system on the focal plane (x',y') and pixel image coordinate system (u,v).

3.2 CALIBRATION METHODS

To orient and calibrate camera systems, various methods can be used. However, two characteristics of multiple camera systems have relevant importance for choosing adequate and appropriate calibration and orientation procedures: (a) the multiple cameras have usually either a fix position or they are displaced all together without changing their relative positions; (b) the multiple cameras have to be calibrated and oriented very often (e.g., at every acquisition sessions) because of the need of small adjustments (e.g., focus and iris, and ev. position and direction). For these reasons, a simultaneous calibration and orientation (*system calibration*) of the multiple cameras is more appropriate. The two problems can be solved jointly with a spatial point array of unknown coordinates plus additional scales or with a suitable array of control points. Both methods were applied in this work.

In the first case, the spatial point array was achieved by moving through the object space a reference bar with two retroreflective target points during the acquisition of the multi-image sequence (*reference bar method*). The required additional scales are provided by the known distance between the two points on the bar. The image coordinates of the two target points are automatically measured and tracked along the sequence with a least squares matching based process (see figure 3.3). As described by Maas (1998), the multiple camera system can then be calibrated by self calibrating bundle adjustment (Gruen and Beyer, 2001) with the additional information of the known distance between the two points on the bar at every recorded locations. The main advantage of the method is the only requirement of the reference bar, the disadvantage of the method is a thorough processing of the data to calibrate the system.

The second case can be solved using a 3–D reference field with signalized points whose coordinates in space are known. The calibration procedure is in this case simpler and full automation can be achieved using coded target points (see figure 3.4 and 3.5 left), that can fully automatically be recognized and measured in the images (Niederoest, 1996).

Fig. 3.3 Automatically tracked and measured image coordinates of the two points on the reference bar.

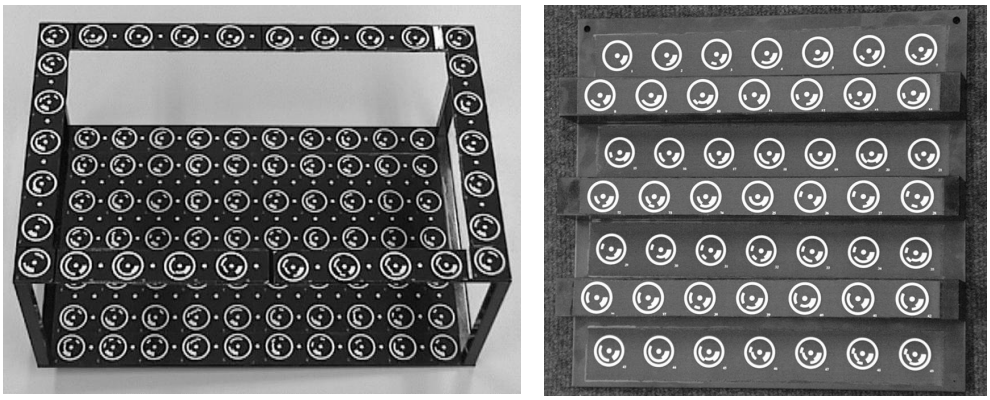

Fig. 3.4 3–D calibration frames with coded targets.

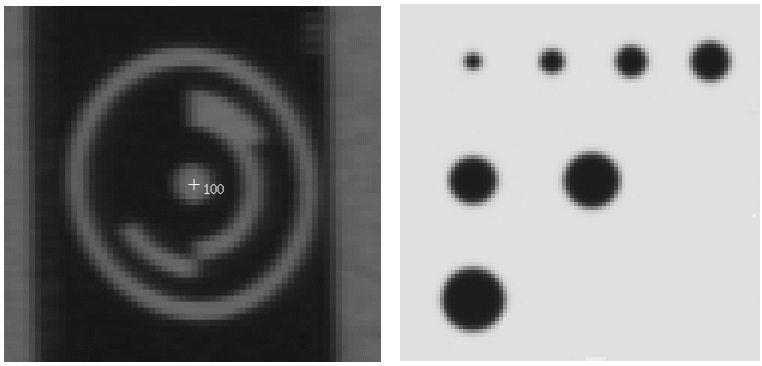

Fig. 3.5 Left: coded target #100. Right: template images used for the measurement of the targets with LSM.

The result of the calibration process are the exterior orientation parameters of the cameras (position X_0, Y_0, Z_0 and rotations ω, φ, κ), the interior orientation parameters of the cameras (camera constant c and principle point x_p, y_p), parameters for the radial and decentering distortion of the lenses and optic systems (k_1, k_2, k_3, p_1, p_2) and two additional parameters modeling effects as differential scaling and shearing (sc, sh). A thorough determination of all these parameters is required to achieve high accuracy in the measurement. Four parameters defining the sensor characteristics (sensor size n_x, n_y and pixel size μ_x, μ_y) have to be known in advance.
The implemented software for the calibration process (see section A.2.2) is based on

the bundle calibration method using a single image of a reference field. The next section will briefly describe the mathematical procedure for calibrating the cameras using this method. For the details regarding the self calibrating bundle adjustment method used for the calibration with the reference bar method, the reader is referred to the literature (Maas, 1998).

3.3 BUNDLE CALIBRATION

3.3.1 Spatial Resection

Assumed is a set of known control points object space coordinates X_i, Y_i, Z_i, their measured image coordinates x'_i, y'_i and approximations for the interior and exterior orientation. The resulting set of collinearity equations (equations 3.3 and 3.4) can be considered as observation equations relating the observation x'_i, y'_i to the parameters $x_0, y_0, z_0, \omega, \varphi, \kappa, c, x_p, y_p, sc, sh, k_1, k_2, k_3, p_1, p_2$, leading to the set of observation equations:

$$\begin{aligned} x'_i &= f_x\left(x_0, y_0, z_0, \omega, \varphi, \kappa, c, x_p, y_p, sc, sh, k_1, k_2, k_3, p_1, p_2, X_i, Y_i, Z_i\right) \\ y'_i &= f_y\left(x_0, y_0, z_0, \omega, \varphi, \kappa, c, x_p, y_p, sc, sh, k_1, k_2, k_3, p_1, p_2, X_i, Y_i, Z_i\right) \end{aligned} \tag{3.8}$$

After linearization of the equations 3.8 and the introduction of a true error vector e, the system of equations becomes:

$$l - e = \mathbf{A}x \tag{3.9}$$

with the following unknown vector x, design matrix $\mathbf{A}$ and observation vector l:

$$\begin{aligned} x &= \left(dx_0, dy_0, dz_0, d\omega, d\varphi, d\kappa, dc, dx_p, dy_p, dsc, dsh, dk_1, dk_2, dk_3, dp_1, dp_2\right)^T \\ \mathbf{A} &= \begin{bmatrix} \left(\frac{\partial f_x}{\partial x_0}\right)_1 & \left(\frac{\partial f_x}{\partial y_0}\right)_1 & \left(\frac{\partial f_x}{\partial z_0}\right)_1 & \left(\frac{\partial f_x}{\partial \omega}\right)_1 & \cdots & \left(\frac{\partial f_x}{\partial p_2}\right)_1 \\ \left(\frac{\partial f_y}{\partial x_0}\right)_1 & \left(\frac{\partial f_y}{\partial y_0}\right)_1 & \left(\frac{\partial f_y}{\partial z_0}\right)_1 & \left(\frac{\partial f_y}{\partial \omega}\right)_1 & \cdots & \left(\frac{\partial f_y}{\partial p_2}\right)_1 \\ \left(\frac{\partial f_x}{\partial x_0}\right)_2 & \cdots & & & \cdots & \left(\frac{\partial f_x}{\partial p_2}\right)_2 \\ \left(\frac{\partial f_y}{\partial x_0}\right)_2 & \cdots & & & \cdots & \left(\frac{\partial f_y}{\partial p_2}\right)_2 \\ \vdots & & & & & \vdots \\ \left(\frac{\partial f_x}{\partial x_0}\right)_n & \cdots & & & \cdots & \left(\frac{\partial f_x}{\partial p_2}\right)_n \\ \left(\frac{\partial f_y}{\partial x_0}\right)_n & \cdots & & & \cdots & \left(\frac{\partial f_y}{\partial p_2}\right)_n \end{bmatrix} \\ l &= \left(x'_1 - \hat{x}'_1, y'_1 - \hat{y}'_1, x'_2 - \hat{x}'_2, y'_2 - \hat{y}'_2, \cdots, x'_n - \hat{x}'_n, y'_n - \hat{y}'_n\right)^T \end{aligned} \tag{3.10}$$

where

$dx_0, dy_0, ..$	changes of the unknown parameters from the given approximations,
n	number of control points,
x'_i, y'_i	observed (i.e., measured) image coordinates of the control point *i*,
$\hat{x}'_i, \hat{y}'_i$	the estimated image coordinates of the control point *i*, i.e., the control point, is backprojected onto the image according to equations 3.3 and 3.4, using the estimated parameters.

For the estimation of x the Gauss-Markov model of least squares is used and leads to:

$$\begin{array}{lcll} \hat{x} & = & \left(\mathbf{A}^T\mathbf{P}\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{P}l & \text{solution vector,} \\ v & = & \mathbf{A}\hat{x} - l & \text{residual vector,} \\ \hat{\sigma}_0^2 & = & \dfrac{v^T\mathbf{P}v}{2n-u} & \text{variance factor,} \\ \mathbf{Q} & = & \hat{\sigma}_0^2 \cdot \left(\mathbf{A}^T\mathbf{P}\mathbf{A}\right)^{-1} & \text{covariance matrix,} \\ \hat{\sigma}_i^2 & = & \mathbf{Q}_{ii} & \text{variance of the single unknowns;} \end{array} \tag{3.11}$$

where

$u = 16$	the number of unknowns,
$\mathbf{P}$	the weight coefficient matrix.

The weight matrix $\mathbf{P}$ has usually diagonal form. Adding an identity part of the unknowns $dc, dx_p, dy_p, dsc, dsh, dk_1, dk_2, dk_3, dp_1, dp_2$ to the equation system 3.10, single unknown parameters can be excluded from the adjustment process by assigning a very large value of the respective weight. The new design matrix $\mathbf{A}$ and observation vector l become:

$$\begin{array}{lcl} \mathbf{A} & = & \begin{bmatrix} \widetilde{\mathbf{A}} & & 0 \\ & 1 & \\ & \ddots & \\ 0 & & 1 \end{bmatrix} \\ l & = & \left(\widetilde{l}, dc', dx'_p, dy'_p, dsc', dsh', dk'_1, dk'_2, dk'_3, dp'_1, dp'_2\right)^T \end{array} \tag{3.12}$$

where

$\widetilde{\mathbf{A}}$	design matrix according to equation 3.10,
$\widetilde{l}$	observation vector according to equation 3.10,
$dc', dx'_p, ..$	observed changes of the unknown parameters.

The partial derivatives of $x_0, y_0, z_0, \omega, \varphi, \kappa, c$ are computed numerically in the iteration process according to equation 3.13 (the derivatives for the other parameters are obtained in an analogue form):

$$\left(\frac{\partial f_x}{\partial x_0}\right)_i = \frac{xd'_i - x'_i}{dm} \tag{3.13}$$

where

x'_i	observed (i.e., measured) image coordinates of the control point *i*,
xd'_i	backprojected control point *i* by $x_0 = x_0 + dm$.
dm	increment.

The other derivatives are determined algebraically as follows:

$$\begin{array}{ll}
\left(\frac{\partial f_x}{\partial x_p}\right)_i = 1 - sc & \left(\frac{\partial f_y}{\partial x_p}\right)_i = sh \\
\left(\frac{\partial f_x}{\partial y_p}\right)_i = sh & \left(\frac{\partial f_y}{\partial y_p}\right)_i = 1 \\
\left(\frac{\partial f_x}{\partial sc}\right)_i = -x'_i & \left(\frac{\partial f_y}{\partial sc}\right)_i = 0 \\
\left(\frac{\partial f_x}{\partial sh}\right)_i = y'_i & \left(\frac{\partial f_y}{\partial sh}\right)_i = x'_i \\
\left(\frac{\partial f_x}{\partial k_1}\right)_i = x'_i \cdot r_i^2 & \left(\frac{\partial f_y}{\partial k_1}\right)_i = y'_i \cdot r_i^2 \\
\left(\frac{\partial f_x}{\partial k_2}\right)_i = x'_i \cdot r_i^4 & \left(\frac{\partial f_y}{\partial k_2}\right)_i = y'_i \cdot r_i^4 \\
\left(\frac{\partial f_x}{\partial k_3}\right)_i = x'_i \cdot r_i^6 & \left(\frac{\partial f_y}{\partial k_3}\right)_i = y'_i \cdot r_i^6 \\
\left(\frac{\partial f_x}{\partial p_1}\right)_i = 2x'^2_i + r_i^2 & \left(\frac{\partial f_y}{\partial p_1}\right)_i = 2x'_i \cdot y'_i \\
\left(\frac{\partial f_x}{\partial p_2}\right)_i = 2x'_i \cdot y'_i & \left(\frac{\partial f_y}{\partial p_2}\right)_i = 2y'^2_i + r_i^2
\end{array} \tag{3.14}$$

where

$$r_i = \sqrt{x'^2_i + y'^2_i} \ .$$

The system is solved iteratively computing the estimated unknown vector $\hat{x}$ according to equation 3.11, actualizing the design matrix **A** and the observation vector *l* and computing again the estimated unknown vector, until the changes of all the unknown parameters are smaller than a threshold.

Chapter 4

Matching Process

This chapter describes the matching process in all its components. First, in section 4.1 is described the stereo matcher which determines the position of corresponding points in an image pair. The automatic multi-image matching process based on the stereo matcher is presented and described in detail in section 4.2; the result is a dense set of corresponding points in the multi-images covering the entire interested surface parts. Finally, section 4.3 describes the optional filtering processes that can be applied to the matching results.

4.1 STEREO MATCHER

The stereo matcher is based on the adaptive least squares method (Gruen, 1985) with the additional geometrical constraint of forcing the matched point to lie on the epipolar line. The following sections describe the mathematical models for the least squares matching process and the geometrical constraint, software implementation and the evaluation of the quality of the results.

4.1.1 Least Squares Matching

The least squares estimation model is described briefly here; for a complete detailed description the reader is referred to the reference (Gruen, 1985).
Assumed are two image windows (called *image patches*) given as discrete functions $f(u,v)$, $g(u,v)$, wherein f is the *template*, g is the *search* image window and u, v are the image coordinates. Since the two image windows are not identical, the comparison of them results in:

$$f(u,v) - e(u,v) = g(u,v) \tag{4.1}$$

wherein $e(u,v)$ is the true error vector. The idea of *least squares matching* (*LSM*) is as follows: equation 4.1 is treated as a nonlinear observation equation which models the vector of observation $f(u,v)$ with a function $g(u,v)$, whose location in the search image needs to be estimated. In order to account for the different viewing angles, image shaping parameters in form of an affine transformation (equation 4.2) must also be estimated. Additionally, a radiometric correction factor is introduced to correct different lighting conditions.

The affine transformation is applied with respect to an initial position u_0, v_0 :

$$\begin{aligned} u &= a_0 + a_1 \cdot u_0 + a_2 \cdot v_0 \\ v &= b_0 + b_1 \cdot u_0 + b_2 \cdot v_0. \end{aligned} \tag{4.2}$$

After linearization of the function $g(u, v)$, equation 4.1 becomes:

$$f(u,v) - e(u,v) = g(u_0, v_0) + \frac{\partial g(u_0, v_0)}{\partial u} \cdot du + \frac{\partial g(u_0, v_0)}{\partial v} \cdot dv + rs. \tag{4.3}$$

With the simplified notation:

$$g_u = \frac{\partial g(u_0, v_0)}{\partial u}, \; g_v = \frac{\partial g(u_0, v_0)}{\partial v}$$

and adding the differentiation of equation 4.2, equation 4.3 results in:

$$\begin{aligned} f(u,v) - e(u,v) = \; & g(u_0, v_0) + g_u da_0 + g_u u_0 da_1 + g_u v_0 da_2 + \\ & + g_v db_0 + g_v u_0 db_1 + g_v v_0 db_2 + rs. \end{aligned} \tag{4.4}$$

Combining the parameters of equation 4.4 in the parameter vector x

$$x^T = (da_0, da_1, da_2, db_0, db_1, db_2, rs) \tag{4.5}$$

and discretizing the equation 4.3, i.e., having a number of n pixels in the image windows, the coefficients in the design matrix $\mathbf{A}$ and the vector difference l become:

$$\mathbf{A} = \begin{bmatrix} {}^1g_u & {}^1g_u \cdot {}^1u & {}^1g_u \cdot {}^1v & {}^1g_v & {}^1g_v \cdot {}^1u & {}^1g_v \cdot {}^1v & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ {}^ng_u & {}^ng_u \cdot {}^nu & {}^ng_u \cdot {}^nv & {}^ng_v & {}^ng_v \cdot {}^nu & {}^ng_v \cdot {}^nv & 1 \end{bmatrix} \tag{4.6}$$

$$l^T = \left({}^1f - {}^1g, \ldots, {}^nf - {}^ng \right)$$

where

${}^iu, {}^iv$	local coordinates of the pixel *i* in the (not transformed) image window,
${}^ig_u, {}^ig_v$	discretized horizontal and vertical gradients at the pixel *i*,
${}^if, {}^ig$	grey values of the pixel *i* in the template and in the search image window.

For the search image, the value of ${}^ig$ is bilinearly resampled at the affinely transformed position (equation 4.2) according to:

$$g(x,y) = \left(sgn \cdot g(x, y_1) + \frac{y - y_1}{y_2 - y_1} \right) \cdot (g(x, y_2) - g(x, y_1)) \tag{4.7}$$

where

x, y	position to be resampled,
x_1, y_1, x_2, y_2	the support points for resampling,
$g(x, y_1), g(x, y_2)$	the linear (to x) resampled grey values, from equation 4.8,
sgn	$sign\left(g(x, y_2) - g(x, y_1)\right)$.

Linear resampling in x is computed as:

$$g(x) = \left(sgn \cdot g(x_1) + \frac{x - x_1}{x_2 - x_1} \right) \cdot (g(x_2) - g(x_1)) \tag{4.8}$$

where

x	position to be resampled,
x_1, x_2	the support points for resampling,
sgn	$sign\,(g(x_2) - g(x_1))$.

Introducing the weight coefficient matrix $\mathbf{P}$ as:

$$\mathbf{P} = \begin{bmatrix} p_{da0} & & & & & & & 0 \\ & p_{da1} & & & & & & \\ & & p_{da2} & & & & & \\ & & & p_{db0} & & & & \\ & & & & p_{db1} & & & \\ & & & & & p_{db2} & & \\ 0 & & & & & & & p_{rs} \end{bmatrix}, \tag{4.9}$$

the Gauss-Markov least squares estimation model leads to:

$$\begin{array}{lcll} \hat{x} & = & \left(\mathbf{A}^T\mathbf{P}\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{P}l & \text{solution vector,} \\ v & = & \mathbf{A}\hat{x} - l & \text{residual vector,} \\ \hat{\sigma}_0^2 & = & \frac{1}{n-7}v^T\mathbf{P}v & \text{variance factor,} \\ \mathbf{Q} & = & \hat{\sigma}_0^2 \cdot \left(\mathbf{A}^T\mathbf{P}\mathbf{A}\right)^{-1} & \text{covariance matrix,} \\ \hat{\sigma}_i^2 & = & \mathbf{Q}_{ii} & \text{variance factor of the single parameters.} \end{array} \tag{4.10}$$

For the usual cases, the weight coefficient matrix $\mathbf{P}$ is set as identity.

4.1.2 Geometrical Constraints

To increase the robustness and accuracy of the matching process, geometrical constraints can be introduced. When multi-images are acquired simultaneously, they display the same object space viewed from different directions. Figure 4.1 shows graphically the case of an image pair. Given a point $P = (x_t, y_t)$ on the image plane 1, the corresponding point P' on the image plane 2 must lie on the *epipolar line* . This can be determined as the intersection of the image plane 2 with the plane formed by $O_1, \overrightarrow{O_1P}, \overrightarrow{B}$, called *epipolar plane*, where O_1, O_2 are the projection centers of the two imaging devices and $\overrightarrow{B}$ is the *basis* vector between them.
Therefore, the three vectors $\overrightarrow{O_1P}, \overrightarrow{B}, \overrightarrow{O_2P'}$ have to lie on the same plane; their coplanarity condition can be mathematically expressed as:

$$\left(\overrightarrow{O_1P} \times \overrightarrow{O_2P'}\right) \cdot \overrightarrow{B} = 0\,. \tag{4.11}$$

After the required transformations, equation 4.11 results in the following equation expressed in the coordinate system of image plane 2 :

$$y_2 = k \cdot x_2 + h \tag{4.12}$$

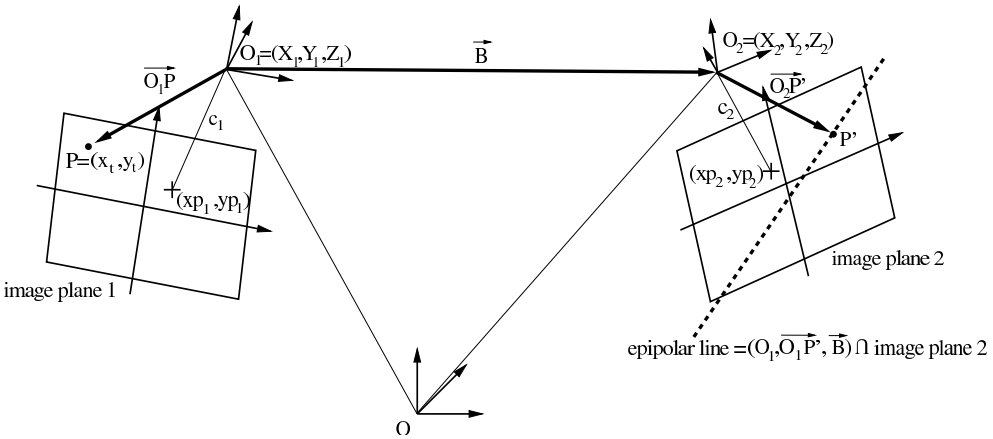

Fig. 4.1 Construction of the epipolar line from an image pair.

with:

$$
\begin{aligned}
k &= \frac{zm_2 \cdot by_2 - ym_2 \cdot bz_2}{zm_2 \cdot bx_2 - xm_2 \cdot bz_2} \\
h &= \frac{bx_2 \cdot (zm_2 \cdot yp_2 - ym_2 \cdot c_2)}{zm_2 \cdot bx_2 - xm_2 \cdot bz_2} + \frac{by_2 \cdot (xm_2 \cdot c_2 - zm_2 \cdot xp_2)}{zm_2 \cdot bx_2 - xm_2 \cdot bz_2} + \\
&\quad + \frac{bz_2 \cdot (ym_2 \cdot xp_2 - xm_2 \cdot yp_2)}{zm_2 \cdot bx_2 - xm_2 \cdot bz_2}
\end{aligned}
\tag{4.13}
$$

where:

$$
\begin{aligned}
\overrightarrow{B_2} &= \begin{bmatrix} bx_2 \\ by_2 \\ bz_2 \end{bmatrix} = \mathbf{R}_2 \cdot \begin{bmatrix} X_2 - X_1 \\ Y_2 - Y_1 \\ Z_2 - Z_1 \end{bmatrix} \\
\overrightarrow{O_1P_2} &= \begin{bmatrix} xm_2 \\ ym_2 \\ zm_2 \end{bmatrix} = \mathbf{R}_2 \cdot \overrightarrow{O_1P}_0 = \mathbf{R}_2 \cdot {\mathbf{R}_1}^T \cdot \overrightarrow{O_1P}_1 = \\
&= \mathbf{R}_2 \cdot {\mathbf{R}_1}^T \cdot \begin{bmatrix} x_t - xp_1 \\ y_t - yp_1 \\ -c_1 \end{bmatrix}
\end{aligned}
\tag{4.14}
$$

and:

c_2	camera constant for image 2,
xp_2, yp_2	principal point for image 2,
$\overrightarrow{B_2}$	the basis vector expressed in the coordinates system O_2,
$\overrightarrow{O_1P}_i$	the vector $\overrightarrow{O_1P}$ expressed in the three coordinate systems O, O_1, O_2,
$\mathbf{R}_1$	rotation matrix of coordinate system O_1 from O,
$\mathbf{R}_2$	rotation matrix of coordinate system O_2 from O.

Different methods can be used to integrate the epipolar geometrical constraint in the least squares matching process. Since for this work, the processing time has to be limited, the geometrical constraint is chosen to be expressed as a single additional observation:

$$da_0 - c_e \cdot db_0 = 0, \tag{4.15}$$

describing the fact that the position in the search image must move along the epipolar line, i.e., with a defined slope c_e, called here *epipolar constant*. The next section describes how the value of c_e is determined from the epipolar line of equation 4.12. The new design matrix $\mathbf{A}$ and the vector difference l become:

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} {}^1g_u & {}^1g_u \cdot {}^1u & {}^1g_u \cdot {}^1v & {}^1g_v & {}^1g_v \cdot {}^1u & {}^1g_v \cdot {}^1v & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ {}^ng_u & {}^ng_u \cdot {}^nu & {}^ng_u \cdot {}^nv & {}^ng_v & {}^ng_v \cdot {}^nu & {}^ng_v \cdot {}^nv & 1 \\ 1 & 0 & 0 & -c_e & 0 & 0 & 0 \end{bmatrix} \\ l^T &= \left({}^1f - {}^1g, \ldots, {}^nf - {}^ng, 0\right). \end{aligned} \tag{4.16}$$

A line in weight coefficient matrix $\mathbf{P}$ must also be introduced

$$\mathbf{P} = \begin{bmatrix} p_{da0} & & & & \\ & \ddots & & & \\ & & p_{rs} & & \\ & & & p_e & \end{bmatrix}. \tag{4.17}$$

The value of p_e can be chosen to increase or decrease the effect of the epipolar constraint to the solution.

4.1.3 Software Implementation

The stereo matcher has to be flexible, since the matching process has to be used for different purposes, e.g., matching corresponding points in multi-images, signalized point measurements, tracking processes. For this reason, it is implemented with and without epipolar constraint.

The following inputs are required to begin the process: the image patch size, the position in the template image and an approximative position in the search image (u_0, v_0 of equation 4.2). Then, the process continues iteratively computing the estimation of the changes of the affine transformation parameters $da_0, da_1, da_2, db_0, db_1, db_2$ and the radiometric correction factor rs according to equation 4.10, updating the design matrix $\mathbf{A}$ and the observation vector l according to equations 4.6 and 4.16 and estimating again the solution vector $\hat{x}$, until the change of the variance factor ($\hat{\sigma}_0$ of equation 4.10) is smaller than a preset threshold (usually 2%).

The given approximative position in the search image must be sufficiently close to the correct position to allow a successful convergence of the process. Figure 4.2 shows an example: if the starting location is too distant from the correct point, the process cannot converge correctly.

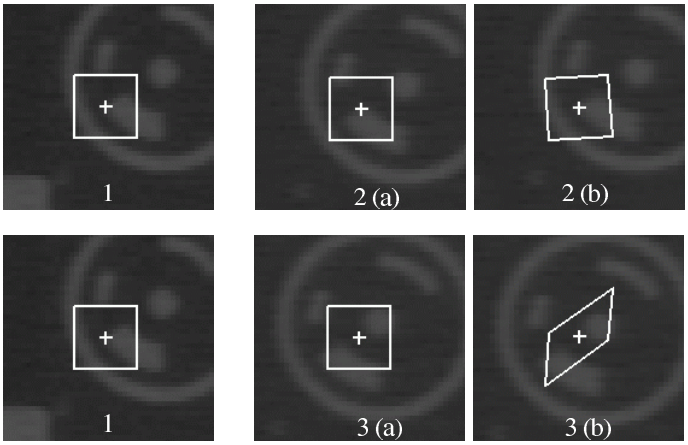

Fig. 4.2 Successfully and unsuccessful LSM. 1: template image, 2: (a) starting location in the search image and (b) successful convergence of the matching algorithm to the correct position, 3: (a) starting location in the search image and (b) unsuccessful matching result, the distance of the starting position to the correct solution was too large.

It is possible that the matching process may converge to false results producing mismatches. Therefore, it is recommended to define indicators for the quality of the matching result for evaluation purposes. This is discussed more in detail in section 4.1.4.

In case of given orientation and calibration information, the geometrical constraint can be used to increase the robustness of the matching process. In this case, some operations have to be performed before starting the least squares process. The epipolar constrained LSM is implemented in this work such that the location of the point in the search image moves with a defined slope (epipolar constant of equation 4.15), assuming the point is lying on the epipolar line. Therefore, it is required that the starting position is on the epipolar line before executing the LSM process. Moreover, the epipolar line has to be corrected for the distortion caused by the lens. In fact, the epipolar line is curved in the image instead of completely straight. This effect has to be corrected before applying the implemented LSM. To accelerate the matching process it is suitable to determine the curvature of the epipolar line only around the interested point and not in the entire image. For this reason, only a short segment of the epipolar line will be computed in the image coordinate system; this will be called the *epipolar segment*. A detailed description of the process to determine it will follow. Practically, the complete matching procedure is composed of four steps: (1) compute the epipolar line in the metric coordinate system in the search image, (2) compute the epipolar segment in the image coordinate system around the selection, (3) project the selected point on the epipolar segment and (4) apply the geometrically constrained LSM at that position.

The flowchart of figure 4.3 gives the overview of the four steps necessary to start the epipolar constrained LSM.

Step 1. Computation of the epipolar line. First, the point selected in the template image (u_t, v_t) has to be transformed from the image coordinate system to the metric coordinate system according to equation 3.6. Then, the obtained metric template image coordinates $x_t{}', y_t{}'$ have to be corrected from the lens distortion. A direct inversion of the equation 3.4 is not possible; the problem has to be solved iteratively subtracting from the image coordinates $x_t{}', y_t{}'$ (rather than adding) the distortion $d\bar{x}', d\bar{y}'$

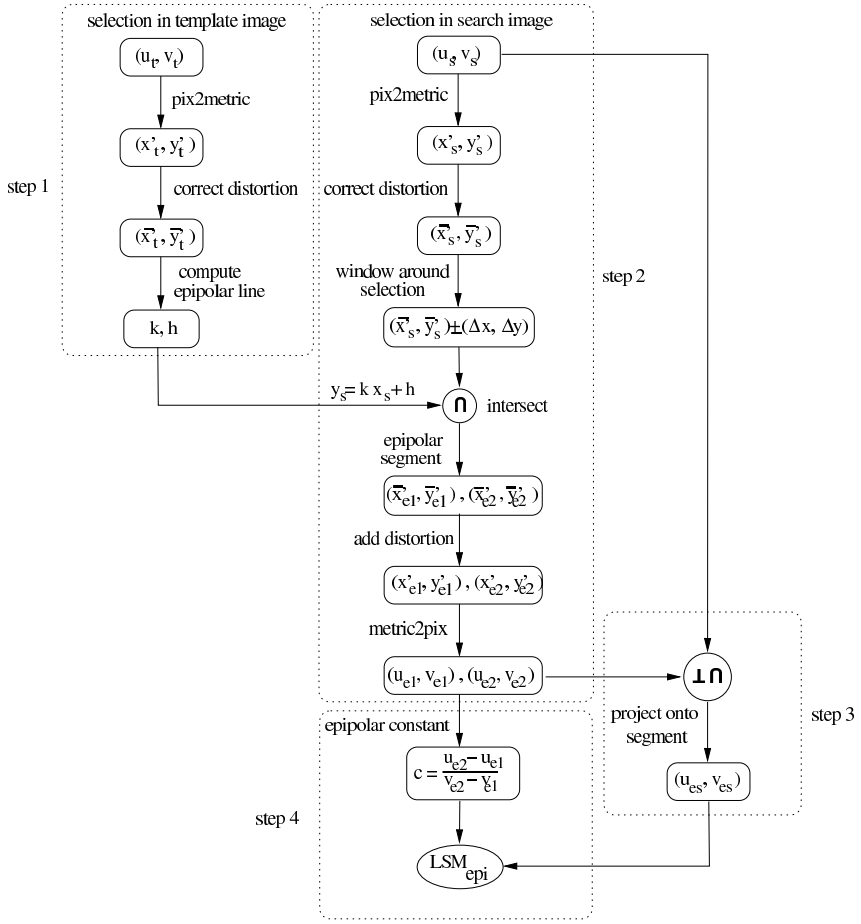

Fig. 4.3 Flowchart: start the LSM process with epipolar constraint.

computed at the corrected position; algorithmically:

$$
\begin{aligned}
&x_{correct}{}' := x_t{}' \\
&y_{correct}{}' := y_t{}' \\
&\textit{repeat} \qquad\qquad\qquad\qquad\qquad\qquad //usually\ only\ 1\ iteration\ required \\
&\quad d\bar{x}', d\bar{y}' := f\,(x_{correct}{}', y_{correct}{}') \;\; //according\ to\ eqs.\ 3.4\ and\ 3.5 \\
&\quad x_{correct}{}' := x_t{}' - d\bar{x}' \\
&\quad y_{correct}{}' := y_t{}' - d\bar{y}' \\
&\textit{end.}
\end{aligned}
\tag{4.18}
$$

The epipolar line, i.e., the k and h values, in the search image is then determined according to equations 4.13 and 4.14.

Step 2. Computation of the segment of the epipolar line around the selection in the search image. The selected point in the search image (u_s, v_s) is first transformed in the metric coordinate system $(x_s{}', y_s{}')$ according to equation 3.6, the lens distortion is then corrected according to the algorithm 4.18. The epipolar line is then intersected with a window around the selected point $(\bar{x_s}{}', \bar{y_s}{}')$, resulting in two points defining the epipolar segment $((x_{e1}{}', y_{e1}{}'), (x_{e2}{}', y_{e2}{}'))$. The distortion caused by the lens according to equations 3.4 and 3.5 has to be added and the two points are then transformed into the image coordinate system $((u_{e1}, v_{e1}), (u_{e2}, v_{e2}))$ according to equation 3.7.

Step 3. The selected point is projected onto the epipolar segment. It is required since the implemented geometrically constrained least squares matching algorithm assumes the selected point to be on the epipolar line. The selected point (u_s, v_s) is orthogonally projected on the epipolar segment as:

$$\begin{aligned} u_{es} &= \frac{b-d}{c-a} \\ v_{es} &= a \cdot u_{es} + b \end{aligned}$$

$$\begin{aligned} where \quad a &= \frac{v_{e2} - v_{e1}}{u_{e2} - u_{e1}} \\ b &= v_{e1} - a \cdot u_{e1} \\ c &= \frac{u_{e1} - u_{e2}}{v_{e2} - v_{e1}} \\ d &= v_{e2} - a \cdot u_{e2}. \end{aligned} \tag{4.19}$$

Step 4. The geometrically constrained LSM is applied at the position u_{es}, v_{es} using as value for the epipolar constant the slope of the epipolar segment as:

$$c_e = \frac{u_{e2} - u_{e1}}{v_{e2} - v_{e1}}. \tag{4.20}$$

Figure 4.4 shows an example of the geometrical constrained LSM process; the template image with the selected template point (black cross) and the image patch (black box) is displayed on the left; in the center, is shown the search image with the epipolar line drawn in white and the given approximative selection (black cross) and starting image patch (black box); on the right, is shown the result of the geometrically constrained LSM, i.e., the affinely transformed image patch drawn as black box and the epipolar segment drawn in white. For convenience, the size of the window for the intersection with the epipolar line is equal to the image patch size.

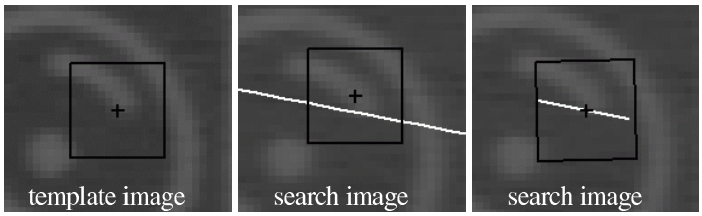

Fig. 4.4 LSM process with epipolar constraint. Left and center: in black the selection in the template and search images, in white the epipolar line. Right: result in the search image after LSM, in black the affinely transformed image patch, in white the epipolar segment.

The stereo matching process is implemented only for an image pair, i.e., an image is used as template and the other as search image. In the case of multi-images, one image is used as template and the others as search images. The stereo matcher is then applied to each search image independently using the same template image patch (see figure 4.5). A complete multi–image matching process taking in account all the images in a single step (Gruen and Baltsavias, 1988; Baltsavias, 1991) could be implemented. This would produce more robust and accurate results (e.g., mismatches as presented in figure 4.6 would not be possible). In the section regarding future works (see section 8.2.2) it is indeed suggested to define and implement a fast geometrical constrained multi–image matching process. Anyhow, the results achieved by the simpler implemented multi–image matching process were enough robust and accurate. Moreover,

the processing time required by the proposed method to determine a dense set of corresponding points was within the desired limits. Therefore, no further investigation was performed to define and implement more complex and accurate matching algorithms.

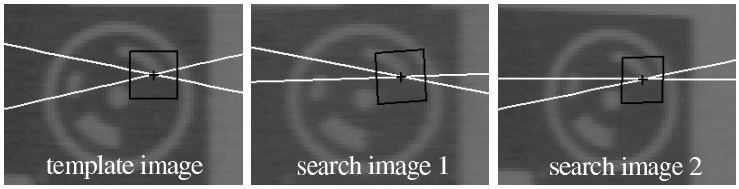

Fig. 4.5 Multi-image geometrical constrained LSM is achieved using an image as template and applying the stereo matching process for each search image independently using the same template image patch. The black boxes represent the patches selected in the template image (left) and the affine transformed in the search images, the epipolar lines are drawn in white.

4.1.4 Quality Evaluation

To evaluate the quality of the results of the matching, different indicators are used: the resulting a posteriori standard deviation of unit weight of the least squares adjustment (σ_0 from equation 4.10), the resulted standard deviation of shifts in x and y directions ($\sigma_x = \sigma_{a_0}, \sigma_y = \sigma_{b_0}$ from equation 4.10), the displacement from the start position in x and y direction (a_0, b_0) and in the case of geometrically constrained LSM, the distance to the epipolar line. Thresholds for these values can be defined for different cases, according to the texture and the type of the images. Quality tests are necessary to evaluate the results of the matching process.
The advantage of using multi-images is shown in figure 4.6. Shown on the left is the template image with the template point, in the center and right, the matching process resulting in two different search images. Additionally, in each image are drawn also the two epipolar lines. The matching process in the two images converges with acceptable value for the single control checks ($\sigma_0, \sigma_x, \sigma_y, a_0, b_0$, distance to epipolar line); however, it can be observed that the matched points do not lie in the intersection of both epipolar lines. Checking the distance to all the epipolar lines is indeed a straightforward and fast way to find mismatches.

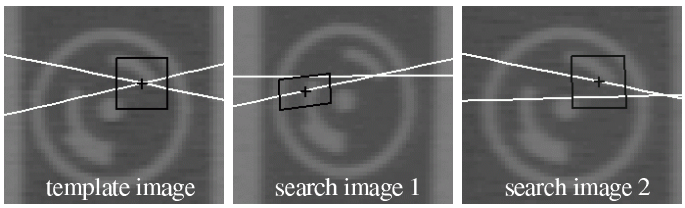

Fig. 4.6 Mismatch of the geometrical constrained LSM. Left: template image, center and right: two search images. In the search image 1 the LSM converged to a false position. The mismatch can be recognized because the results do not lie on the intersection of the two epipolar lines.

A complete multi–image matching process taking in account all the images in a single step (Gruen and Baltsavias, 1988; Baltsavias, 1991) and producing a simultaneous solution in the multi–images would not allow the kind of mismatches shown in figure 4.6. Indeed, the implementation of this feature is proposed for the further development of the process (see section 8.2.2).

4.1.5 Options

The options that have to be chosen for the stereo matching process are the image patch size in x and y direction, which of the affine transformation parameters are used, the use of the radiometric correction factor, the use of the epipolar constraint and the weight of the epipolar constraint in the weight matrix.

4.2 AUTOMATIC MATCHING PROCESS

The role of the automatic matching process is the establishment of a dense and robust set of corresponding points in entire regions of the images. The method described here is entirely developed regarding the mainly smooth characteristic of the surface of human body parts. Moreover, the process strategy is especially developed to minimize the required processing time. The following sections describe in detail the methodology and the functionality of the process.

4.2.1 Seed Point Definition

The automatic matching process that produces a dense and robust set of corresponding points, starts from few seed points. The location, the number and the distribution of the seed points play an essential role for the time required by the automatic matching process to cover entire regions of the images. In the ideal case of a perfectly smooth surface without discontinuities and with a good level of texture, the number of defined seed points does not affect the results neither the required processing time. A single or more seed points would not produce substantial differences in the results. However, the imaged human body part can present some discontinuities (e.g., nose, overlapping parts) and some difficult regions with low texture (e.g., eyebrows). In this case, the seed points serve to define regions that are locally smooth and continuous. These regions are treated separately, allowing a fast coverage of the entire image. The process will be clearly explained in section 4.2.2. Depending on the case, the ideal distribution of seed points has to be chosen. Figure 4.7 shows three examples of well distributed seed points.
The cues for the definition of the seed points in the region to be matched are:

- usually, the seed points should be distributed regularly throughout the region of interest;
- in case of elongated shapes (e.g., legs, arms) seed points are preferable along its long axis (e.g., points 3-6, 12-15 and 16-20 on the central image or points 10-15 and 16-20 on the right image of figure 4.7);
- seed points should be placed symmetrically on both sides of problematic areas such as very dark or very shiny zones (e.g., teeth, eyebrows) (e.g., points 4 and 5 on the left image of figure 4.7), steep surfaces (e.g., nose, under the chin) (e.g., points 13, 14 or 15, 12 on the left image of figure 4.7) or discontinuities in the surface (e.g., overlapping body parts) (e.g., points 4, 8 of the central image of figure 4.7. The problematic area will, in that way, lie at the border of the local regions defined by the seed points (see figure 4.18).

Seed points can be generated in three different modes: fully automatically, semi-automatically (*complete mode*: defining them only in one image or *partial mode*

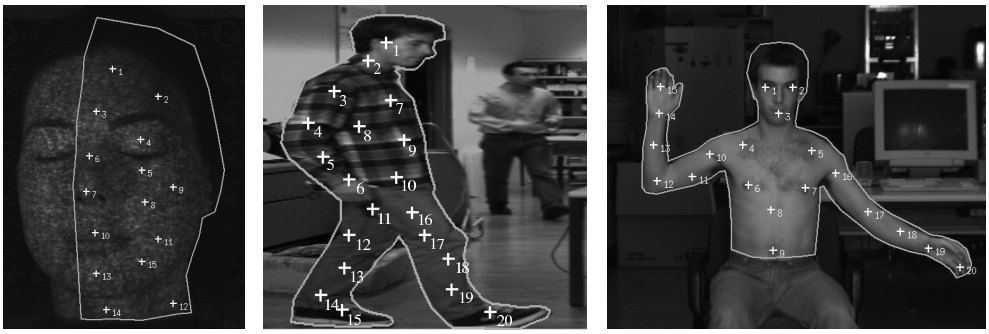

Fig. 4.7 Examples of ideal seed points distribution.

selecting approximatively the location also in the search images) or defined manually in each image.

4.2.1.1 Manual Mode. This is used for special cases where the automated modes could fail. In this mode, the seed points have to be selected manually (with an approximation of few pixels) in each image, LSM is then applied to find the exact positions. Figure 4.8 shows an example: on the top without using epipolar constraint, on the bottom using epipolar constraint. In the latter case, the drawn epipolar line facilitates the user. Additionally, the user has also to decide if the quality of the matching result is sufficient. The manual mode is a time consuming procedure, but in some special cases (e.g., a fine repeated pattern in the image) it is required for the convergence of LSM to the exact position.

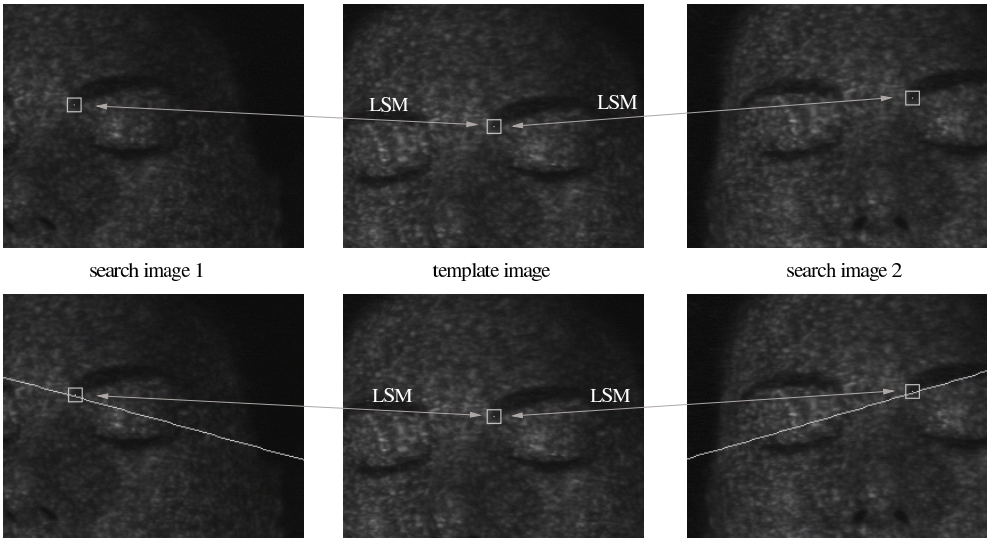

Fig. 4.8 Manual seed point definition. The user has to select with few pixels accuracy the positions in the template and in the two search images. Top: without epipolar constraint, bottom: with epipolar constraint.

4.2.1.2 Semi-Automated Mode. This is a faster way to define seed points. The principle is to define the position of the seed point only in the template image and to automatically establish the corresponding points in the search images. Both *partial* and *complete* semi-automated modes are possible. In the *partial* version, the user has

to select also a rough approximative location in the two search images, whereas in the *complete* version this is not required. Figures 4.9 and 4.10 illustrate the procedure, with and without epipolar constraint.

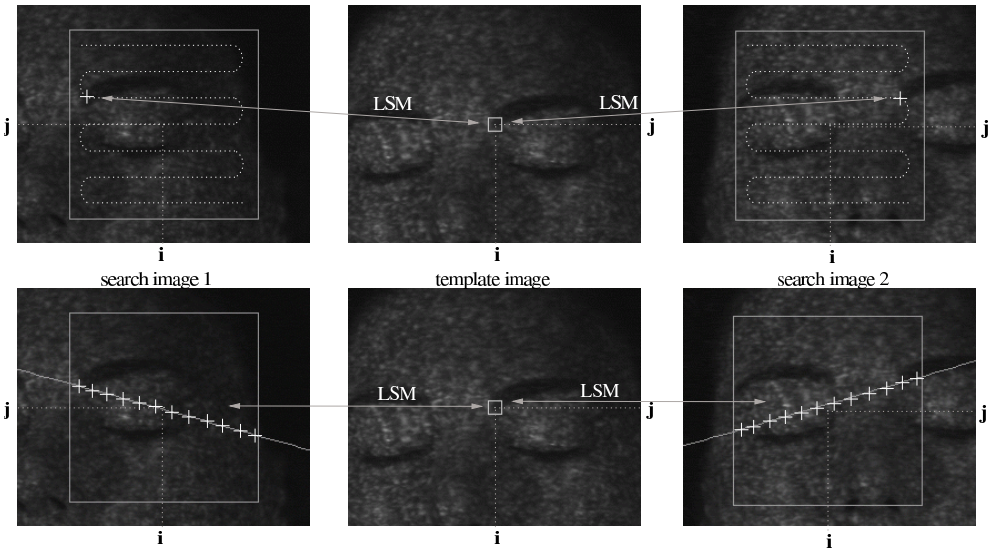

Fig. 4.9 Semi automated seed point definition. *Complete* version: the only selection made by the user is in the template image. Top: without epipolar constraint, bottom: with epipolar constraint.

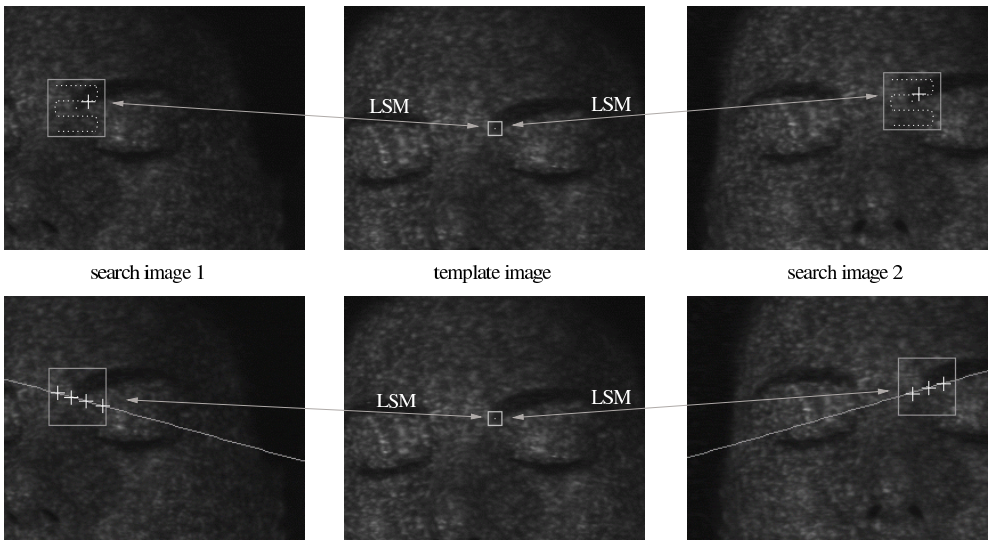

Fig. 4.10 Semi automated seed point definition. *Partial* version: the user select also an approximative location in the two search images, the searching region is smaller. Top: without epipolar constraint, bottom: with epipolar constraint.

When calibration and orientation information is available, the corresponding point in the search images is automatically determined by searching the best matching results along the epipolar line in a restricted region (the white boxes in figures 4.9 and 4.10 bottom). For the *partial* version (figure 4.10 bottom) the search region is smaller, reducing the number of matches that have to be performed and therefore reducing

consistently the required computation time. For the *complete* version of the semi-automatic mode (figure 4.9 bottom), the search region has to be considerably larger; for convenience it is centered around the same position as for the template image (image coordinates i, j in figure 4.9).
When the calibration and orientation information cannot be used or is unavailable, a more complex strategy is used: the search area is in this case scanned along a path (dotted white curve in the top of figures 4.9 and 4.10) to find the position with the best cross correlation value; once this position is found (white cross in the top of figures 4.9 and 4.10), LSM is applied at that location to find the corresponding points.
The results are evaluated in both cases (*partial* and *complete* versions) automatically, checking the thresholds described in section 4.1.4. If the epipolar constraint is used, the distances to both epipolar lines in the search images are also checked. The correctness of the seed points is essential for the automatic matching process, as no mismatches are allowed. The quality thresholds are therefore set high.
The semi-automated mode is the most convenient one for normal cases of static surface measurement: it is in fact fast but leaves the choice of the seed points to the user.

4.2.1.3 Fully Automatic Mode. This applies the Foerstner interest operator (Foerstner and Guelch, 1987) on the template image to automatically determine locations where the matching process may perform robustly. The corresponding points in the other images are then established with the same procedure as in the complete version of the semi-automated mode.
The template image is first divided into regular regions whose size can be chosen. For each region, interest points are established with the Foerstner operator; the points with the best values are chosen as candidates. The complete version of the semi-automated seed points definition procedure is then applied to all the candidates and the matching results are accepted as seed points if the quality is sufficient.
The full automatic procedure works well in case of good texture of the human body part surface. However, some regions without seed points may remain. Figure 4.11 shows a comparison between seed points defined semi-automatically (left) and the result of the full automatic mode (right).

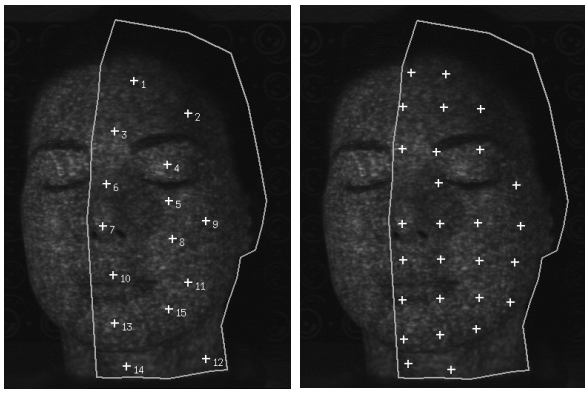

Fig. 4.11 Example of semi-automatically selected seed points (left) and fully automatically generated seed points (right).

It is evident that the fully automatic mode generates seed points that are regularly distributed in the image without any knowledge of the human body part to be measured. On the contrary, in the semi-automated mode, the user can choose to place the seed

points in key locations decreasing the time required by the automatic matching process. The full automatic mode is therefore useful in cases where the number of multi-image sets to be processed may be very large, e.g., for dynamic surface measurements or for tracking from multi-image video sequences. In these cases, the required manual intervention has to be minimal.

4.2.2 Matching Strategy

The matching strategy defines how the entire region of interest in the image is covered starting from the seed points. A dense set of corresponding points is the result expected at the end of the process. An additional goal of the implemented matching strategy is to minimize the entire processing time. The time required to cover the entire image can vary greatly depending on the distribution and the number of the defined seed points. A very simple method to dramatically reduce the processing time is the definition of a region of interest (*ROI*) in the template image. It can be given as a sequence of points or manually defined in the implemented graphical user interface (see section A.2.3). The matching process will then be limited to the ROI.
After the definition of the seed points, the template image is divided into polygonal regions by *Voronoi tessellation* (see figure 4.13 left). The Voronoi tessellation divides the space between individual seed points into polygonal regions such that the boundaries surrounding each seed point enclose an area that is closer to that seed point than to any other neighboring points. For the software implementation, discretization is used to increase the speed of the tessellation process. The image is divided into a regular grid and the closest seed point is found for each element. Then all of the pixels included in the element are set to belong to that seed point region. The flowchart in figure 4.12 describes the process.

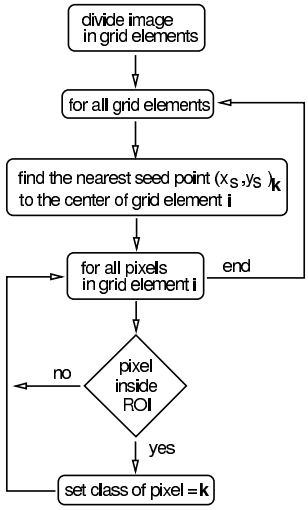

Fig. 4.12 Flowchart of Voronoi tessellation.

Starting from the seed points, the automatic matching process produces a dense set of corresponding points in each polygonal region by sequential horizontal and vertical shifts. The figure 4.13 shows graphically the process and the flowcharts in figures 4.14 and 4.15 describe it in detail.

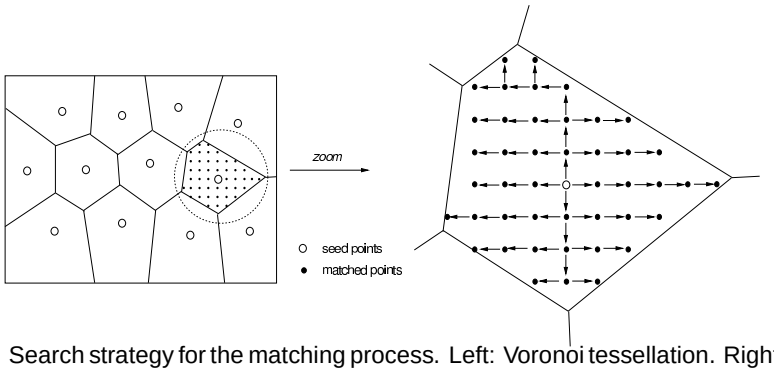

Fig. 4.13 Search strategy for the matching process. Left: Voronoi tessellation. Right: starting from the seed points, each region is covered by sequential horizontal and vertical shifts.

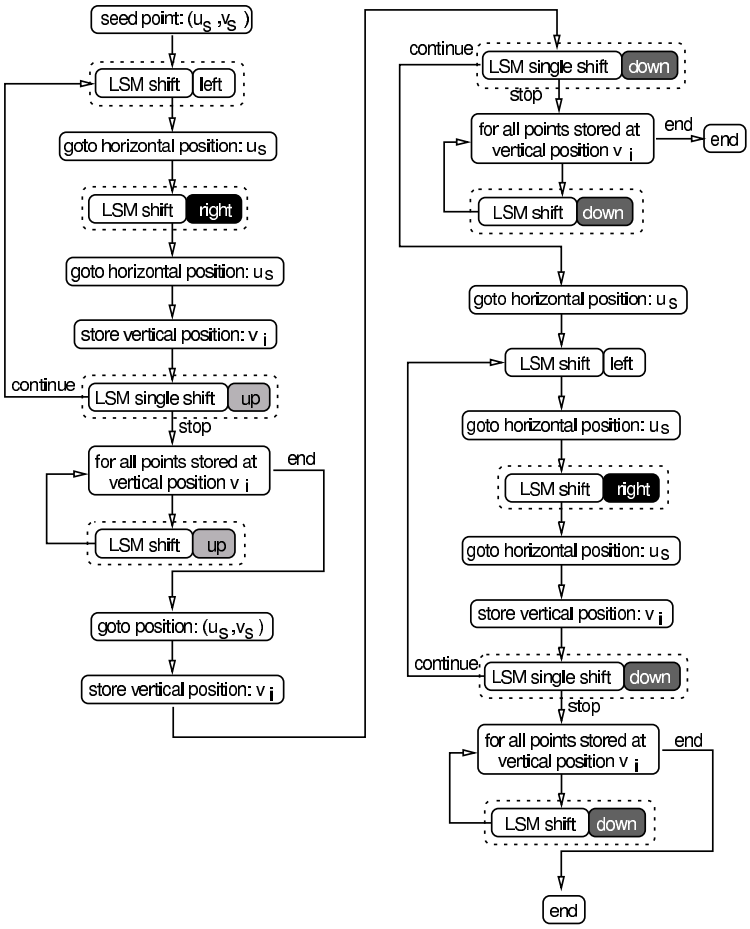

Fig. 4.14 Flowchart of the search strategy for the matching process, for the dotted bounding boxes see figure 4.15.

The process works adaptively at each shift, changing some parameters (e.g., smaller shift, bigger size of the patch) if the quality of the matching result is not satisfactory. Several indicators are used to define the quality (see section 4.1.4): a posteriori standard deviation of the least squares adjustment, standard deviation in x and y directions, displacement from the start position in x and y direction and distance to the epipolar lines. Thresholds for these values are defined at the begin of the process according to the texture and the type of the images.

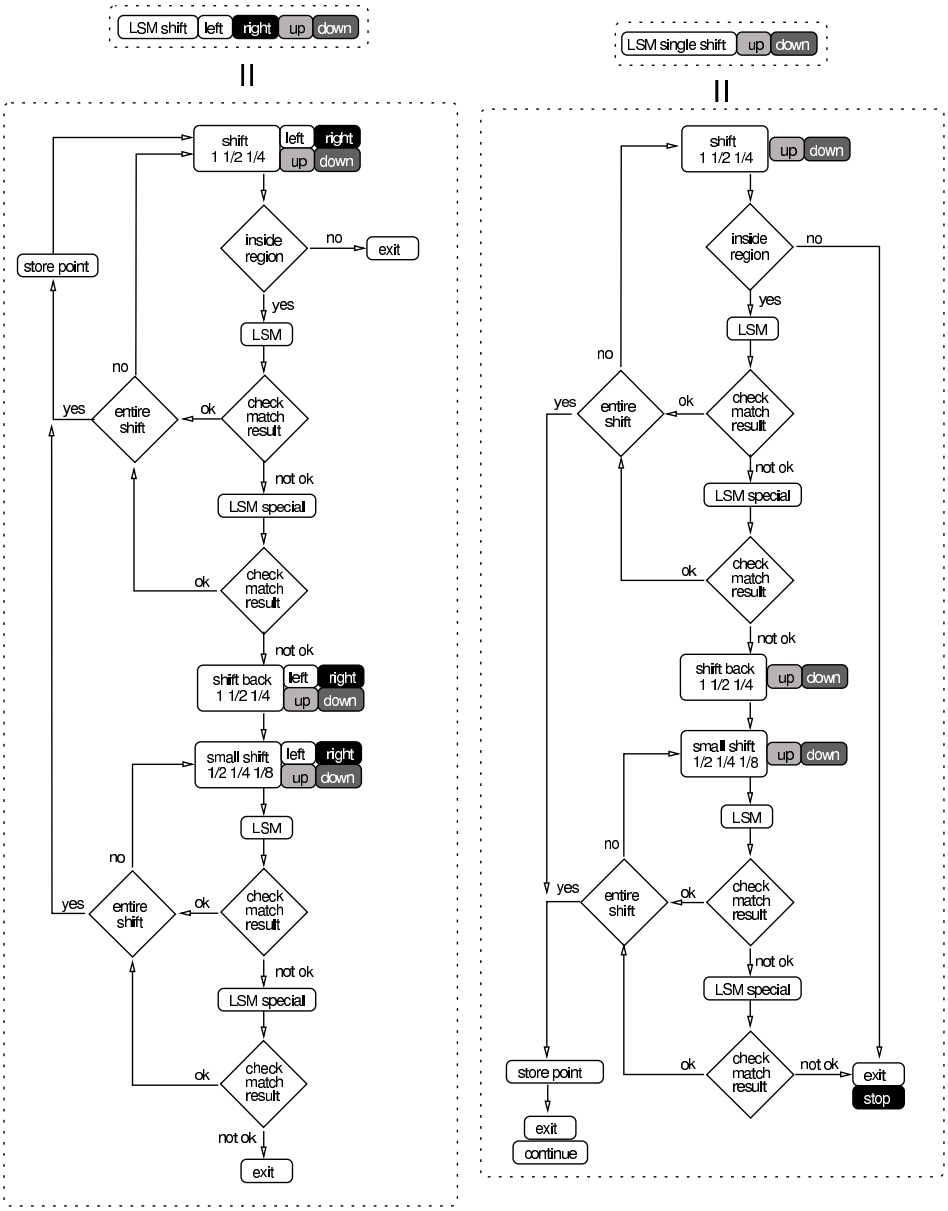

Fig. 4.15 Flowcharts of the search strategy for the matching process, from figure 4.14

The key parameter for the matching strategy is the *pixel shift*. It may have different values in the x and y directions; but they are usually equal. It defines the amount of pixels to shift, in the template and search images, from the seed point and from the previously matched points. Optionally, it is possible to perform the matching process by shifting half or quarter of given pixel shift value (option *shift 1 1/2 1/4* in figure 4.15). In this case, only the matched points at the full shift locations would be stored. This option is useful if the amount of stored data has to be limited without compromising the performance and accuracy of the matching process.
If quality of the results of the matching is not sufficient, the matching may be repeated using a bigger patch (option *LSM special* in figure 4.15) and/or a smaller shift value (option *small shift 1/2 1/4 1/8* in figure 4.15).

When either the boundaries of the polygonal region are reached or when the matching fails, the process continues from the central position to the next upper or lower shift level (see figure 4.18, left). The coverage of the entire image is achieved by repeating the process for all the polygonal seed points regions. However, it is possible that at the end of the process, holes of unanalyzed areas occur in the set of matched points (see figure 4.16).

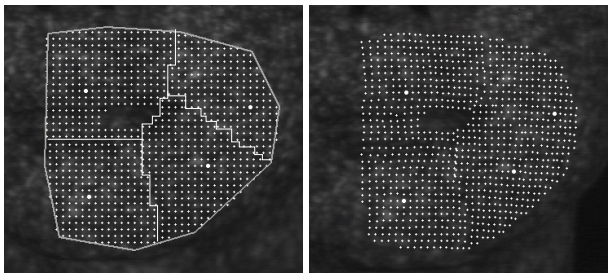

Fig. 4.16 Result of the automatic matching process. An area of unmatched points remains. Left: template image, right: search image. The bigger points are the seed points, the white stepped lines in the template image are the boundaries of the polygonal region determined by the Voronoi tessellation.

The causes of the presence of holes can have different origins, such as the encountering of discontinuities in the surface or lack of texture (see figure 4.18). Therefore, it is highly probable that the holes can be reduced by searching from other directions. Indeed, the process *close the gaps* attempts to match the missing points by searching from all directions around the holes. It is composed of three steps: (1) finding the areas where matched points are missing, (2) attempting to match missing points, (3) continuing to match until no new points are matched.
To find the area where matched points are missing (1) the image is divided into a regular grid whose size can be chosen (see figure 4.17, center).

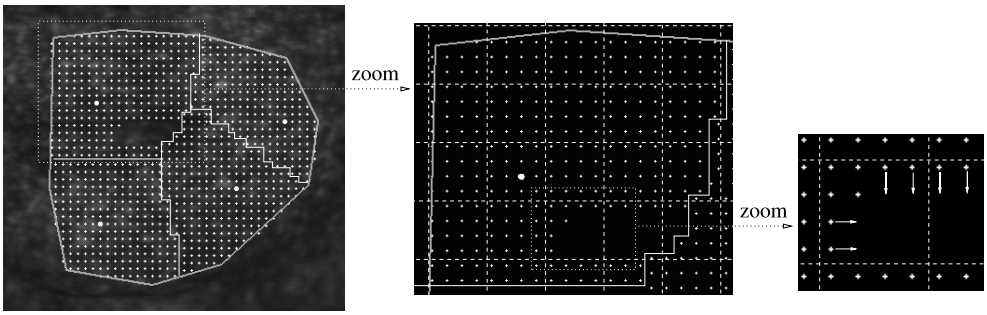

Fig. 4.17 Close the gaps. Left: a gap in the set of matched points. Center: the area is divided into a regular grid (dashed lines) to find the regions where points are missing. Right: shifting from matched points to match the missing points.

The number of matched points is counted for each grid element and if it is less than a set percentage the total number of matched points that can fit in the grid element (e.g., 80%), then the grid element is classified as a region with missing points. Grid elements containing no points are not considered. The size of the grid elements and the threshold percentage are parameters that may be selected by the user.
The grid elements with missing points are then processed in step (2). The matching process is applied starting from all of the matched points contained in the grid element and shifting in each direction (left, right, up, down) (see figure 4.17 right).

In the third phase of the process, step (3), a matching loop is performed starting from all the new matched points and shifting in each direction (left, right, up, down), until no new points are matched. This last phase of the process is intended to close definitively all of the gaps.
The flowchart in figure 4.19 describes the process *close the gaps* in detail. The shifting procedures are the same as described above and the same options can be selected (1 1/2 1/4 1/8 shifts, bigger patch, smaller shift).
The *close the gaps* process is a time consuming process that can strongly affect the speed of the entire matching process. The number and size of the gaps in the data has therefore to be limited. The location and number of the seed points play an essential role. In case of discontinuities (see figure 4.18), two seeds points have to be placed symmetrically to them. The boundary of the polygonal regions would then coincide with the difficult area. In this way, the size of the hole would be smaller and ,therefore, the time required by the entire matching process shorter.

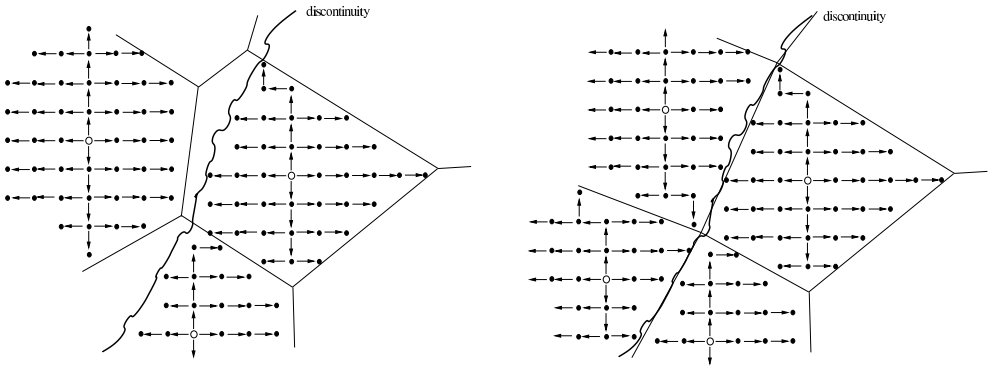

Fig. 4.18 Matching process. The presence of a discontinuity in the surface produces a hole in the set of the matched points (left). The boundary of the polygonal regions will coincide with the discontinuity when the location of the seed points is symmetrical to the discontinuity (right).

4.2.3 Options

Different options and parameters can be chosen for the automatic matching process. As explained in section 4.2.1 different methods can be used to define the seed points. At least a single seed point is required to start the process.
A relevant option that can be chosen for the matching process is whether or not to use the epipolar constraint. The complete matching process (definition of seed points and automatic matching) can in fact be performed without orientation and calibration information if desired. This functionality is useful, for example, when the orientation is not accurate enough or unknown. In these special cases, the least squares matching algorithm is not geometrically constrained. Obviously, the robustness of the results of the process decreases. However, the quality of the set of matched points may be satisfactory.
The complete set of parameters to be set for the automatic matching process is shown in figures 4.20 and 4.21 which shows three option windows of the graphical user interface of the developed software (see section A.2.3).

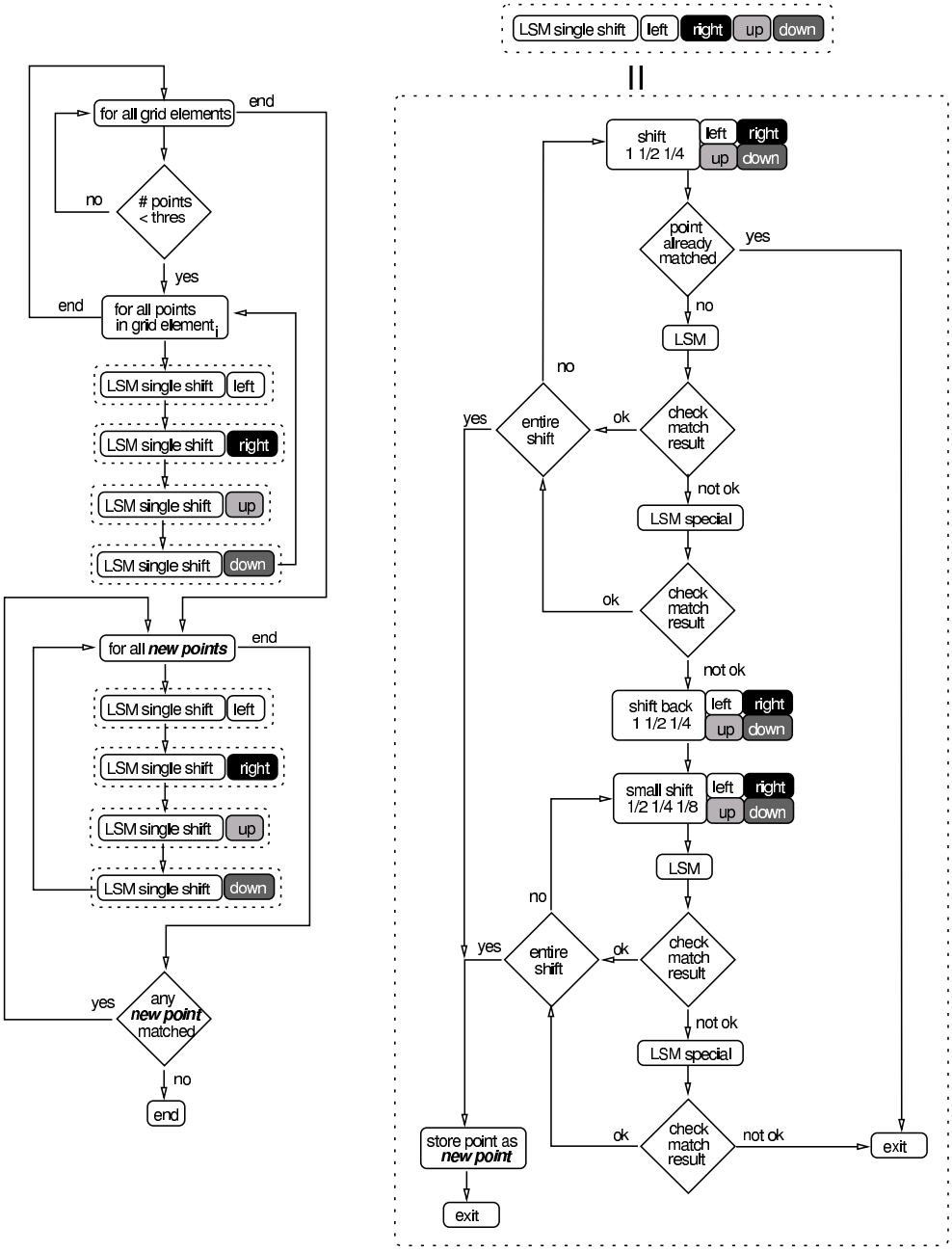

Fig. 4.19 Flowchart of the process *close the gaps*.

4.2.3.1 Preprocessing. Figure 4.20 shows the parameters required for preprocessing. On the left, for the optional Wallis filtering process (top) and for the full automatic generation of seed points (bottom); on the right for the semi-automated and manual definition of seed points. Detailed description about the parameters regarding Wallis filter can be found in (Wallis, 1976).

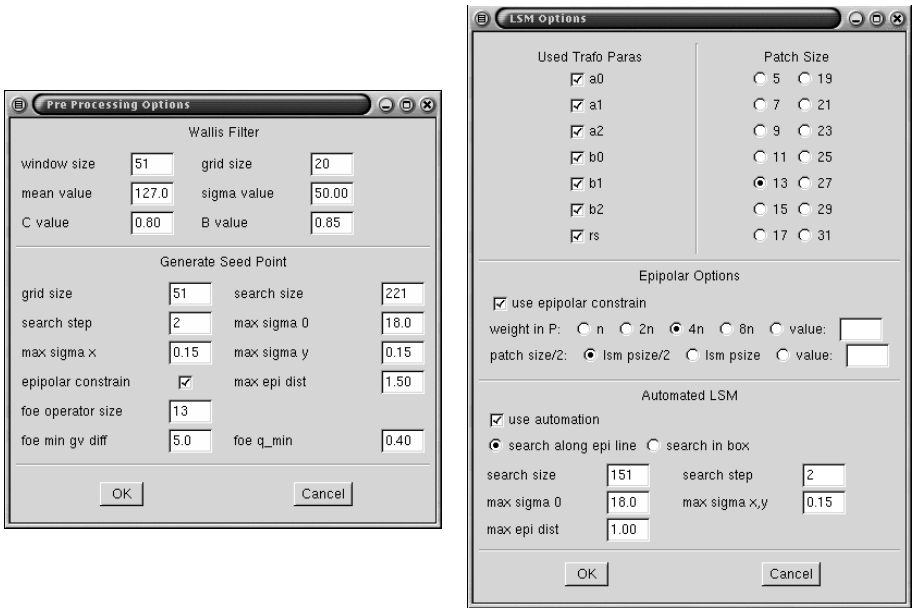

Fig. 4.20 Left: preprocessing options, Wallis filtering and parameters for the full automatic definition of seed points. Right: parameters for the semi-automated and manual definition of seed points.

The required parameters for the **full automatic definition of the seed points** are the following (their names refers to figure 4.20):

- *grid size*: size of the regular grid elements for dividing the template image [pixel]. This parameter control the number of the generated seed points.
- *search size*: size of the search region in the search images [pixel], i.e., the size of the white box in figure 4.9.
- *search step*: pixel step along the epipolar line, respectively along the search path in the search region.
- *max sigma 0*: threshold of $\hat{\sigma}_0$ (see equation 4.10) to accept the matching result.
- *max sigma x,y*: thresholds of $\hat{\sigma}_x = \hat{\sigma}_{a_0}$ and $\hat{\sigma}_y = \hat{\sigma}_{b_0}$ (see equation 4.10) to accept the matching result.
- *epipolar constraint*: on/off.
- *max epi dist*: threshold of distance [pixel] of the matched point to the two epipolar lines to accept the matching result.

The last three parameters are required by the Foerstner interest operator. For detailed information, the reader is referred to the reference (Foerstner and Guelch, 1987). The missing LSM parameters are defined in the next parameter set.
Figure 4.20, right, shows the parameters for the **manual** and **semi-automated seed point definition** divided into three groups: LSM parameters, epipolar parameters and automation.

For the **LSM** process the following parameters have to be chosen:

- *used trafo paras*: used affine transformation parameters ($a_0, a_1, a_2, b_0, b_1, b_2$) and radiometric correction factor (*rs*). Usually, all the parameters are used.
- *patch size*: image patch size in pixels. The ideal value for this parameter has to be found by testing LSM manually on the images. It depends strongly on the texture in the images and on the image format.

The parameters for the **epipolar constraint** are:

- *use epipolar constraint*: use or not use the epipolar constraint.
- *weight in P*: weight (p_e) for the epipolar constraint in the weight coefficient matrix $\mathbf{P}$ (equation 4.17), defined as a multiple of the number *n* of pixels inside the image patch (*n, 2n, 4n, 8n*) or as *value*.
- *patch size/2*: half window size to intersect with the epipolar line to find the epipolar segment, i.e., $\Delta x, \Delta y$ in flowchart of figure 4.3. This parameter is usually set as *lsm psize/2*, i.e., half of the LSM patch size (e.g., as in figure 4.4); *lsm psize* is adequate for small LSM patch sizes (7, 9 pixels); an arbitrary *value* can also be given.

The last set of parameters are for the **semi-automated mode**:

- *use automation*: use or not use the semi-automated mode.
- *search*: searching mode, along the epipolar line (*along epi line*) or inside a square region along a path (*in box*).
- *search size*: size of the search region in the search images, i.e., the size of the white box in figure 4.9.
- *search step*: pixel step along the epipolar line, resp. along the search path in the search region.
- *max sigma 0*: threshold of $\hat{\sigma}_0$ (see eq. 4.10) to accept the matching result.
- *max sigma x,y*: thresholds of $\hat{\sigma}_x = \hat{\sigma}_{a_0}$ and $\hat{\sigma}_y = \hat{\sigma}_{b_0}$ (see eq. 4.10) to accept the matching result.
- *max epi dist*: threshold of distance (in pixels) of the matched point to the two epipolar lines to accept the matching result.

The values for the thresholds of $\hat{\sigma}_0$, $\hat{\sigma}_x$ and $\hat{\sigma}_y$ have to be found by testing manually LSM on the images. The same applies for the two parameters regarding the automated LSM (*search size, search step*), they strongly depend on the images. No rules or advises can be given regarding these parameters. Some examples of parameter sets can be found in appendix C.

4.2.3.2 Automatic Matching Process.

The parameters required for the automatic matching process are displayed in figure 4.21. They can be divided into four groups: search strategy, LSM parameters, epipolar parameters and parameters for the automatic matching process.

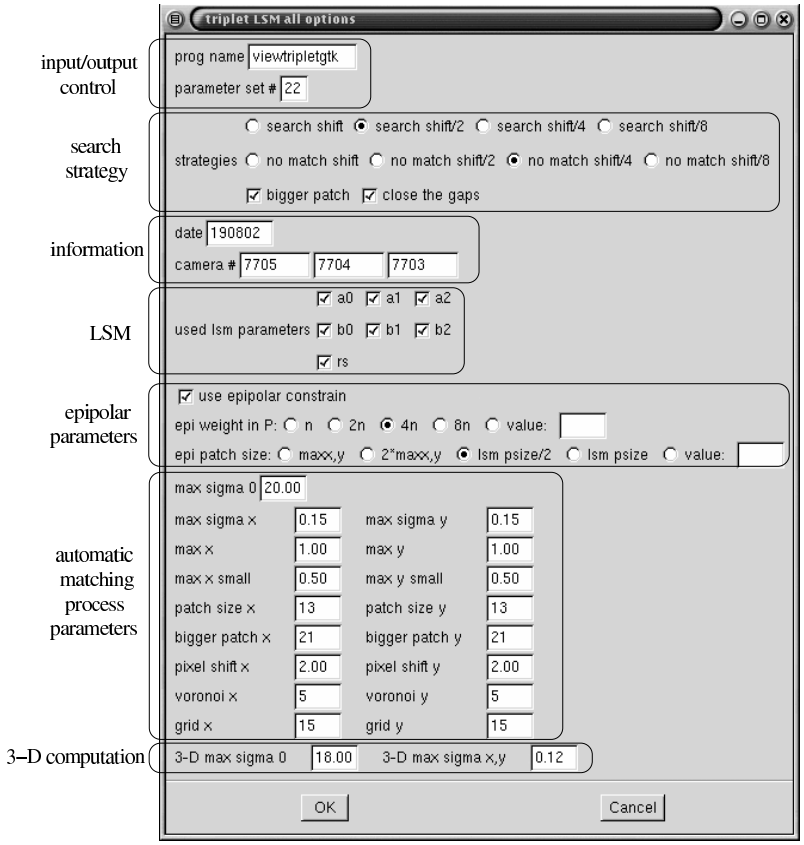

Fig. 4.21 Parameters of the automatic matching process.

The parameters defining the **search strategy** are:

- *search shift*: entire, half, one quarter or one eighth of the pixel shift value which is going to be used for the matching and close-the-gaps processes (*shift* in flowcharts of figures 4.15 and 4.19). This strategy options can be used for reducing the amount of stored data without affecting the matching results. With, e.g., the option *search shift/2* and *pixel shift* as 2 pixels, the matching process will match each pixel in the template image but it will store only every two pixels.
- *no search shift*: entire, half, one quarter or one eighth of the pixel shift value in the case of an unsuccessful match (*small shift* in flowcharts of figures 4.15 and 4.19). It must be smaller than the *search shift*; if it is equal to the *search shift* then the *small shift* strategy is not used.
- *bigger patch*: use or not use this strategy: a bigger patch for LSM is used in case of an unsuccessful match (see *LSM special* in flowcharts of figures 4.15 and 4.19).
- *close the gaps*: use or not use this strategy (flowchart of figure 4.19).

The parameters used for the **LSM process** ($a_0, a_1, a_2, b_0, b_1, b_2, rs$) can be individually switched on or off; however, they are usually all used.

For the **epipolar constraint**, the following options can be selected:

- *use epipolar constraint*: use or not use epipolar constraint.
- *weight in P*: weight (p_e) for the epipolar constraint in the weight coefficient matrix $\mathbf{P}$ (equation 4.17), defined as a multiple of the number *n* of pixels inside the image patch (*n, 2n, 4n, 8n*) or as *value*.
- *epi patch size*: window size to intersect with the epipolar line to find the epipolar segment, i.e., $\Delta x, \Delta y$ in flowchart of figure 4.3. This parameter is usually set as *lsm psize/2*, i.e., half of the LSM patch size (e.g., as in figure 4.4); *lsm psize* is adequate for small LSM patch sizes (7, 9 pixels); it can also be defined as equal to the parameter *maxx,y*, or double of it or as an arbitrary *value*.

The rest of the parameters for the **automatic matching process** are the following:

- *max sigma 0*: threshold of $\hat{\sigma}_0$ (see eq. 4.10) to accept the matching result.
- *max sigma x,y*: thresholds of $\hat{\sigma}_x = \hat{\sigma}_{a_0}$ and $\hat{\sigma}_y = \hat{\sigma}_{b_0}$ (see eq. 4.10) to accept the matching result.
- *max x,y*: thresholds of the displacement of the result of the matching process from the starting position (the two resulting parameters a_0 and b_0). It is usually set as the effective pixel shift performed by the matching process (e.g., half of the value of the parameter *pixel shift* if the strategy *search shift/2* is used).
- *max x,y small*: same as above for the cases of *small shift* (see flowcharts of figures 4.15 and 4.19)(e.g., one quarter of the value of the parameter *pixel shift* if the strategy *no search shift/4* is used).
- *patch size x,y*: LSM patch size.
- *bigger patch size x,y*: LSM patch size in case of *LSM special* (see flowcharts of figures 4.15 and 4.19), i.e., if the strategy *bigger patch* is used.
- *pixel shift*: value of pixel shift.
- *voronoi x,y*: sizes of grid elements for the computation of the Voronoi tessellation. This parameter is not relevant, it depends on the image format (examples can be found in the appendix C).
- *grid x,y*: size of the grid elements for the *close the gaps* process (see flowchart of figure 4.19). This parameter depends on the image format (examples can be found in the appendix C).

As explained previously, some parameters have to be determined by manually testing the LSM process on the images. These are the thresholds for the quality control (*max sigma 0, max sigma x,y*) and the image patch sizes for the matching process (*patch size x,y* and *bigger patch size x,y*). No rules or advises can be given regarding these parameters. Some examples of parameter sets can be found in appendix C.
The most important parameter for the automatic matching process is the effective pixel shift. It affects the amount of matched data as well as the time required by the matching process. Anyhow, different values of this parameter do not influence relevantly the accuracy of the performed measurement (see chapter 6).

4.3 FILTERING

The quality controls of the matching process described in section 4.1.4 serve to minimize the number of mismatches. Anyway, errors may be expected in the set of corresponding points produced. Filters can be applied to the set of matched points before computing their 3–D coordinates, reducing therefore the number of possible blunders.

4.3.1 Regularization of the Grid

Since the seed points do not have to be snapped onto a regular grid, the matching process has the undesired effect of producing an irregular grid of points in the template image (see figure 4.23, left). Before the application of any filter, the set of matched points has to be uniformed to a regular grid (center of figure 4.23). This is easily achieved by matching all the points shifted to the regular grid. The flowchart of figure 4.22 describes the process.

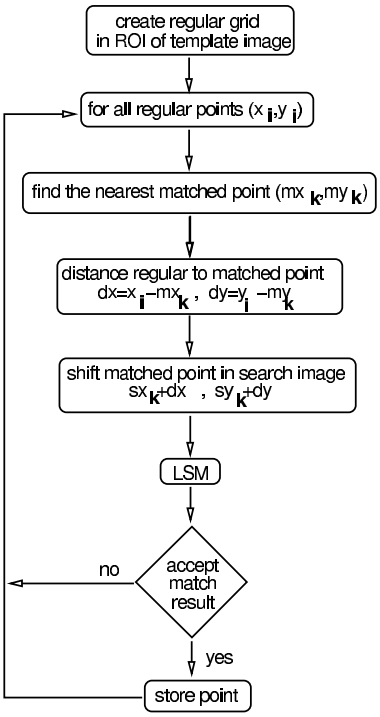

Fig. 4.22 Flowchart of the process regularization of grid.

First, a regular grid is generated in the region of interest (selected manually) of the template image. The nearest matched point (mx_k, my_k) is searched for each element (i) of the regular grid. In the search image, the matched point (sx_k, sy_k) is shifted by the distance of the template point to the regular element $(dx = x_i - mx_k, dy = y_i - my_k)$. LSM is then computed again and the point is stored only if the result pass the quality check, as described in section 4.1.4.

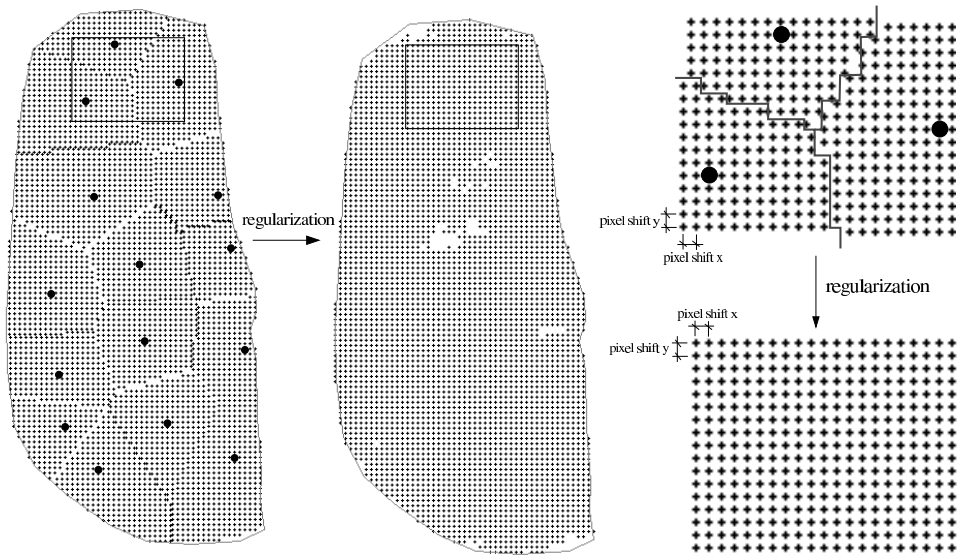

Fig. 4.23 Regularization of grid. Left: seed points and matched points in the template image after the automatic matching process (note, the Voronoi tessellation can be clearly seen). Center: after regularization of the grid. Right: zoom, before and after regularization.

4.3.2 Neighborhood Filtering

For the definition of the filter, the smooth characteristic of the surface of the human body is taken into account. As can be seen in figure 4.24, the transformed image patches of neighbor points belonging to a common smooth surface have similar shapes.

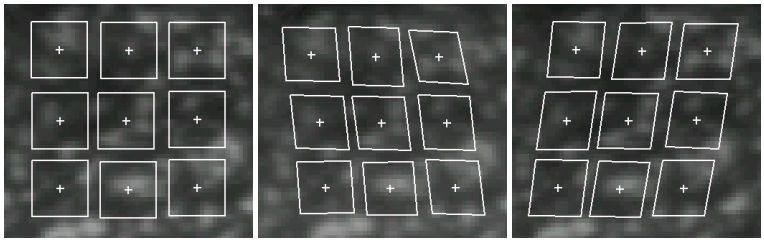

Fig. 4.24 Points matched in the neighborhood have similar characteristics. Left: template image, center and right: search images. The white crosses are the matched points and the white boxes represent the affinely transformed patches.

A neighborhood filter can indeed be defined to check the local uniformity of the shape of the transformed image patches in the set of matched points. To characterize the shape of the transformed patches, the angles of the two major axes to the horizontal and vertical axes and the size of a (1,1) transformed patch are used (see figure 4.25). As illustrated in figure 4.25, left, the two angles can be computed as:

$$\left.\begin{aligned} \alpha_1 &= \arctan\left(\frac{b_1}{a_1}\right) \\ \alpha_2 &= \arctan\left(\frac{b_2}{a_2}\right) \end{aligned}\right\} \tag{4.21}$$

where a_1, a_2, b_1, b_2 are the affine parameters of the transformed patch.

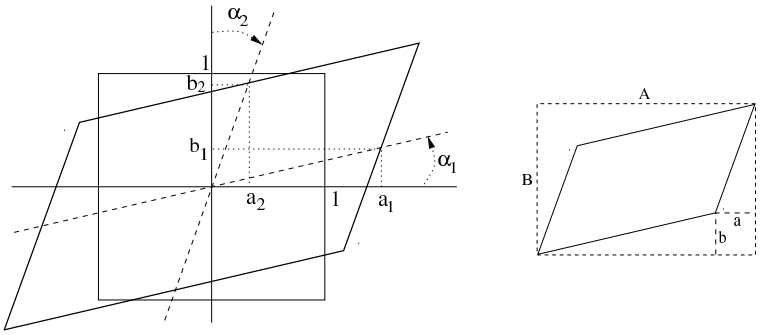

Fig. 4.25 Neighborhood filter: defining the three values which characterize the shape of the transformed patch. Left: angles α_1, α_2; right: help values (*A*,*B*,*a*,*b*) for the definition of the size.

As shown in figure 4.25, right, the size of the (1,1) transformed patch can be computed as:

$$size = A \cdot B - a \cdot B - A \cdot b \tag{4.22}$$

with:

$$\begin{array}{rcl} A & = & 2 \cdot |a_1 + a_2| \\ B & = & 2 \cdot |b_1 + b_2| \\ a & = & 2 \cdot |a_2| \\ b & = & 2 \cdot |b_1| \end{array} \tag{4.23}$$

where a_1, a_2, b_1, b_2 are the affine parameters of the transformed patch.
The three shape characteristic values of the transformed patch (α_1, α_2 and the size of the (1,1) transformed patch) are computed for each matched point. These values are compared to the mean values of the points matched in the neighborhood (defined with a mask, e.g., 3x3 points). If the differences are larger than the thresholds, then the matched point is rejected. The flowchart of figure 4.26 shows the process.

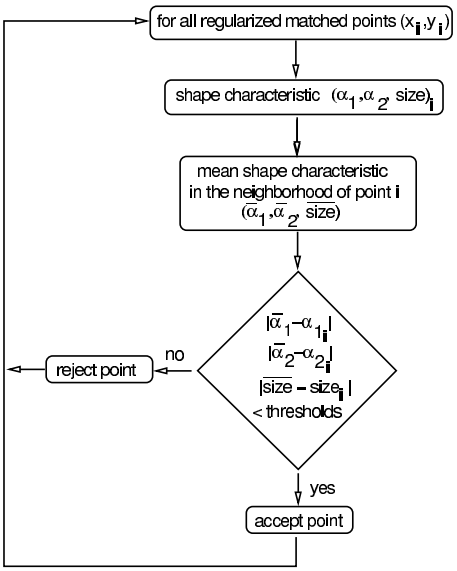

Fig. 4.26 Flowchart of the neighborhood filter.

The options to be chosen for the filtering process, shown in figure 4.27, are the mask size defining the neighborhood, the thresholds for the difference of angles (in degrees) and the threshold for the difference of the sizes (in percentages).

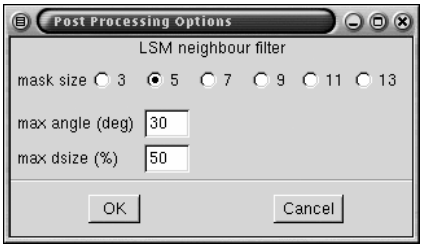

Fig. 4.27 Options for the neighborhood filter.

Chapter 5

Surface Measurement

This chapter describes the process for the measurement of the surface of human body parts. In section 5.1.1 it will be first described how the 3–D point cloud is computed from the matched points. Section 5.2 will then treat the visualization and modeling aspects. Finally two different applications are presented (human face modeling and measurement of blood vessel branching casting) to prove the functionality of the proposed method.

5.1 3–D POINT CLOUD

This section describes how the 3–D coordinates of the matched points are computed and presents a simple 3–D filter.

5.1.1 Forward Ray Intersection

Figure 5.1 shows a typical situation: three images with known interior and exterior orientation are displayed with the image coordinate of the same object point (black circle) in each of them; the rays *projection center–image coordinate* are also displayed.

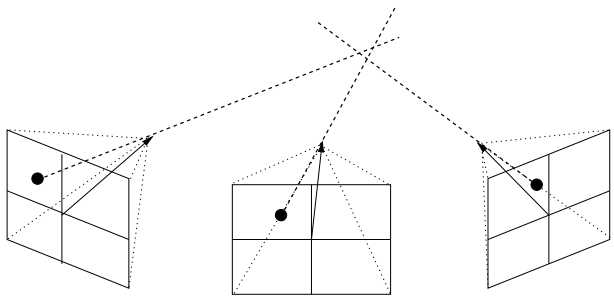

Fig. 5.1 Forward ray intersection.

In case of a perfect calibration of the cameras and a perfect matching of the corresponding points in the images, the three rays would intersect in a common point, i.e., the object point. This is usually not the case and the three rays do not intersect precisely. However, a fictitious intersection point can be found by minimizing the distances to all the rays, this process is called *forward ray intersection*. The same mathematical model of section 3.1 describing the projection of the object space onto the sensor plane of an imaging device is used here. It is modeled by the collinearity

equation (equation 3.3) and the distortion terms ($d\bar{x}'$, $d\bar{y}'$ from equations 3.4 and 3.5):

$$\begin{aligned} x' &= x_p - c \cdot \frac{r_{11} \cdot (X - x_0) + r_{21} \cdot (Y - y_0) + r_{31} \cdot (Z - z_0)}{r_{13} \cdot (X - x_0) + r_{23} \cdot (Y - y_0) + r_{33} \cdot (Z - z_0)} + d\bar{x}' \\ &= \bar{x}' + d\bar{x}' \\ y' &= y_p - c \cdot \frac{r_{12} \cdot (X - x_0) + r_{22} \cdot (Y - y_0) + r_{32} \cdot (Z - z_0)}{r_{13} \cdot (X - x_0) + r_{23} \cdot (Y - y_0) + r_{33} \cdot (Z - z_0)} + d\bar{y}' \\ &= \bar{y}' + d\bar{y}'. \end{aligned} \tag{5.1}$$

The forward intersection problem can be solved by least squares estimation. In this case the observations are the image coordinates of corresponding points (x'_i, y'_i) in the n images of different views and the unknowns are the object space coordinates (X, Y, Z) of the observed points.
The derivatives of the uncorrected terms $(\bar{x}', \bar{y}')$ of the collinearity equation with respect to X, Y, Z are (Luhmann, 2000):

$$\begin{aligned} \left(\frac{\partial \bar{x}'}{\partial X}\right)_i &= -\frac{c_i}{N_i^2} \cdot (N_i r_{11i} - Z x_i r_{13i}) & \left(\frac{\partial \bar{y}'}{\partial X}\right)_i &= -\frac{c_i}{N_i^2} \cdot (N_i r_{12i} - Z y_i r_{13i}) \\ \left(\frac{\partial \bar{x}'}{\partial Y}\right)_i &= -\frac{c_i}{N_i^2} \cdot (N_i r_{21i} - Z x_i r_{23i}) & \left(\frac{\partial \bar{y}'}{\partial Y}\right)_i &= -\frac{c_i}{N_i^2} \cdot (N_i r_{22i} - Z y_i r_{23i}) \\ \left(\frac{\partial \bar{x}'}{\partial Z}\right)_i &= -\frac{c_i}{N_i^2} \cdot (N_i r_{31i} - Z x_i r_{33i}) & \left(\frac{\partial \bar{y}'}{\partial Z}\right)_i &= -\frac{c_i}{N_i^2} \cdot (N_i r_{32i} - Z y_i r_{33i}) \end{aligned} \tag{5.2}$$

where:

$$\begin{aligned} Zx_i &= r_{11i} \cdot (X - x_{0i}) + r_{21i} \cdot (Y - y_{0i}) + r_{31i} \cdot (Z - z_{0i}) \\ Zy_i &= r_{12i} \cdot (X - x_{0i}) + r_{22i} \cdot (Y - y_{0i}) + r_{32i} \cdot (Z - z_{0i}) \\ N_i &= r_{13i} \cdot (X - x_{0i}) + r_{23i} \cdot (Y - y_{0i}) + r_{33i} \cdot (Z - z_{0i}). \end{aligned} \tag{5.3}$$

The derivatives of the correcting term $d\bar{x}', d\bar{y}'$ are very small and can be therefore omitted, thus:

$$\begin{aligned} \left(\frac{\partial x'}{\partial X}\right)_i &\cong \left(\frac{\partial \bar{x}'}{\partial X}\right)_i & \left(\frac{\partial y'}{\partial X}\right)_i &\cong \left(\frac{\partial \bar{y}'}{\partial X}\right)_i \\ \left(\frac{\partial x'}{\partial Y}\right)_i &\cong \left(\frac{\partial \bar{x}'}{\partial Y}\right)_i & \left(\frac{\partial y'}{\partial Y}\right)_i &\cong \left(\frac{\partial \bar{y}'}{\partial Y}\right)_i \\ \left(\frac{\partial y'}{\partial Z}\right)_i &\cong \left(\frac{\partial \bar{x}'}{\partial Z}\right)_i & \left(\frac{\partial y'}{\partial Z}\right)_i &\cong \left(\frac{\partial \bar{y}'}{\partial Z}\right)_i. \end{aligned} \tag{5.4}$$

The unknown vector x, the design matrix $\mathbf{A}$ and the observation vector l result therefore in:

$$x = (dX, dY, dZ)^T$$

$$\mathbf{A} = \begin{bmatrix} \left(\frac{\partial \bar{x}'}{\partial X}\right)_1 & \left(\frac{\partial \bar{x}'}{\partial Y}\right)_1 & \left(\frac{\partial \bar{x}'}{\partial Z}\right)_1 \\ \left(\frac{\partial \bar{y}'}{\partial X}\right)_1 & \left(\frac{\partial \bar{y}'}{\partial Y}\right)_1 & \left(\frac{\partial \bar{y}'}{\partial Z}\right)_1 \\ \vdots & \vdots & \vdots \\ \left(\frac{\partial \bar{x}'}{\partial X}\right)_n & \left(\frac{\partial \bar{x}'}{\partial Y}\right)_n & \left(\frac{\partial \bar{x}'}{\partial Z}\right)_n \\ \left(\frac{\partial \bar{y}'}{\partial X}\right)_n & \left(\frac{\partial \bar{y}'}{\partial Y}\right)_n & \left(\frac{\partial \bar{y}'}{\partial Z}\right)_n \end{bmatrix} \tag{5.5}$$

$$l = (x'_1 - \hat{x}'_1, y'_1 - \hat{y}'_1, \cdots, x'_n - \hat{x}'_n, y'_n - \hat{y}'_n)^T$$

where

dX, dY, dZ	changes of the unknown object coordinates from the estimations,
x'_i, y'_i	observed (i.e., measured) image coordinates of the point in image *i*,
$\hat{x}'_i, \hat{y}'_i$	the estimated image coordinates, computed by backprojecting the estimated 3–D point onto the image *i* according to equation 5.1,
n	number of images.

For the estimation of x, the Gauss-Markov model of least squares is used:

$$\begin{array}{lcll} \hat{x} & = & \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T l & \text{solution vector,} \\ v & = & \mathbf{A}\hat{x} - l & \text{residual vector,} \\ \hat{\sigma}_0^2 & = & \frac{v^T v}{2n-3} & \text{variance factor,} \\ \mathbf{Q} & = & \hat{\sigma}_0^2 \cdot \left(\mathbf{A}^T\mathbf{A}\right)^{-1} & \text{covariance matrix,} \\ \hat{\sigma}_i^2 & = & \mathbf{Q}_{ii} & \text{variance factor of the single unknowns.} \end{array} \tag{5.6}$$

The system is solved iteratively, i.e., the estimated unknown vector $\hat{x}$ is computed according to equation 5.6, the design matrix **A** and the observation vector *l* are updated and the estimated unknown vector $\hat{x}$ is computed again, this until the changes of the unknowns are smaller than a threshold (usually 0.5%).
The precision of the estimated parameters $\hat{\sigma}_i$ resulting from equation 5.6 expresses the theoretical precision of the computed 3–D coordinates in the three axis directions.

5.1.2 Filtering

To reduce the amount of data, a simple 3–D filter may be applied. The object space is divided into voxels of variable dimensions and the points contained in each voxel are reduced to its center of gravity (figure 5.2).

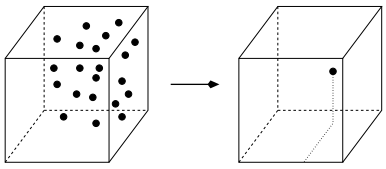

Fig. 5.2 Simple 3–D filter: points in each voxel are reduced to its center of gravity.

The data resulting after this filtering process have a density more uniform and the amount of data is reduced. However, the accuracy of the measurement will also decrease. Figure 5.3 shows an example of the result after applying the filtering process (with a voxel size of 2x2x2 mm) to a selection of points measured on a human face. This simple filter does not take into account the fact that the points can lie on a curved surface. It can therefore work properly only if the 3–D point cloud is sufficient dense and the voxel size sufficient small, so that each voxel would contain points that could be assumed to lie on a plane; in this case, the center of gravity would lie on the same plane. If strong noise is present in the data or if the point cloud is not dense enough, more complex Gaussian filters (Borghese and Ferrari, 2000) have to be applied to the data (see, e.g., section 5.4.2).

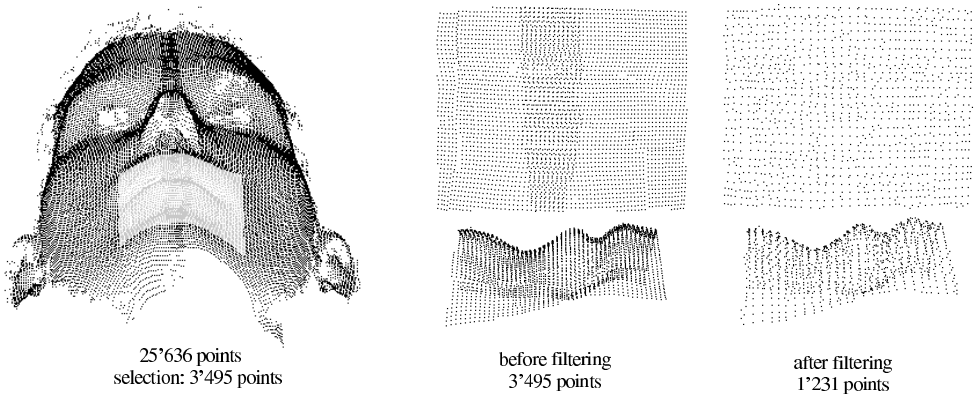

Fig. 5.3 Simple 3–D filter: left: original 3–D data (25'636 points), in light grey the selected points; center: selected points before filtering (3'495 points), right: selected points after filtering with a voxel size of 2x2x2 mm (1'231 points).

5.2 VISUALISATION AND MODELING

This section describes the methods implemented in this work to visualize efficiently 3–D point clouds and some basic modeling procedures used for presentation purposes.

5.2.1 3–D Point Cloud Visualisation

The 3–D information in form of a point cloud alone is not sufficient for visualisation purposes. Some radiometric information is required in this case. For this reason, the 3–D point could is backprojected onto a calibrated and oriented image according to the collinearity equation (equations 3.3, 3.4 and 3.5). The radiometric information is then read directly from the image, as greyscale value or RGB color information depending on the type of the images. Eventually, if required, bilinear resampling can be applied to the image (according to equations 4.7 and 4.8).
A 3–D viewer with some basic editing functions was developed for an efficient visualisation of the point cloud. The detailed description of the viewer will follow in section A.2.5. Figure 5.4 shows two examples, on the left the result of a full body laser scan without radiometric information and on the right the result of the surface measurement process described in this chapter with radiometric information.
Some basic editing functions were added to the viewer: single points can be selected getting information about the point number, its 3–D coordinates and the color information (RGB values) if available; multiple points can be selected as rectangular box or as a (clockwise) hand drawn region; the selected points can then be deleted from the data set or stored as separate data set. This function allows a simple and easy removal of outliers or unuseful points from the data (e.g., points from the background or points not useful for modeling purposes). Of basic importance is also the functionality that allows to get the 3–D coordinates of selected points. These can indeed be used as control points to determine the orientation of uncalibrated texture images (see next section).

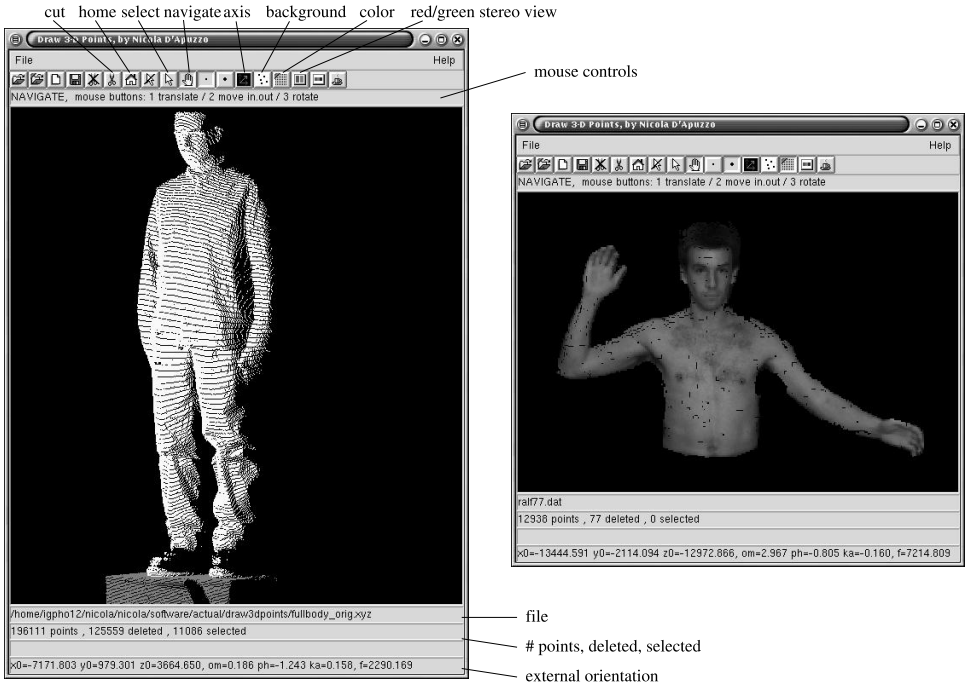

Fig. 5.4 3–D point viewer. Left: a full body 3–D point cloud (determined with laser scanning), right: 3–D point cloud with radiometric information; each point has a color value.

5.2.2 Modeling

Basic surface modeling procedures, such as 2 1/2–D triangulation, Gaussian smoothing and texture mapping with a single image, were applied to achieve a nice visualisation of the surface measurement results. In house existing software and free software was used for the triangulation and smoothing tasks. A a simple software was developed to solve basic texture mapping problems. To achieve better qualitative modeling results, commercial software can be used.

To generate a triangulated surface from the 3–D point cloud, 2 1/2–D Delauney triangulation is used (IGP software *DTMZ*). Unfortunately, this method can process only 3-D data with 2 1/2–D characteristic, i.e., the surface defined by the 3–D point cloud can be projected onto a plane without having overlapping areas. For this reason only landscape-like surface models can be generated (e.g., face masks). In case of required 3–D surface triangulation, commercial software (e.g., Geomagic Wrap^{TM}) can be used.

After the computation of a triangulated surface, the next step for the modeling process is the texture mapping. The standard format used nowadays for 3–D models is the *Virtual Reality Modeling Language* (*VRML*). Free software can be found for the visualization of VRML files. This format allows two different modes for texture mapping definition: texture mapping draping the entire texture image over the 3–D model or defining the texture for each triangle of the model.

In the first case, five parameters have to be defined: two translations and two scaling factors both in horizontal and vertical directions and a rotation. The entire image is then draped over the 3–D model as shown in figure 5.5. The disadvantages of this mode are the manual determination of the five parameters and the approximative result.

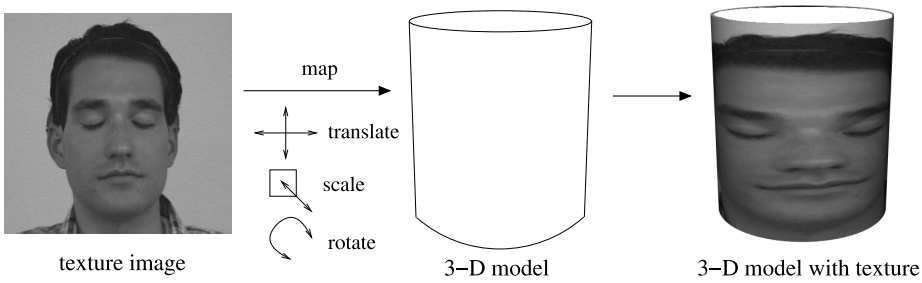

Fig. 5.5 Texture mapping with entire image: the texture image is mapped over the 3–D model; translations, scaling factors and rotations have to be given.

In the second texture mapping mode allowed by the VRML format, each triangle of the 3–D model has a texture information. This is achieved by assigning to each vertex a *texture coordinate*. Texture coordinates differ from the normal image coordinates in the way that the origin is in the bottom left corner of the image and their value ranges from 0 to 1. The advantage of this definition is the possibility to change the format of the texture image without modifying the 3–D model file. Figure 5.6 shows an example.

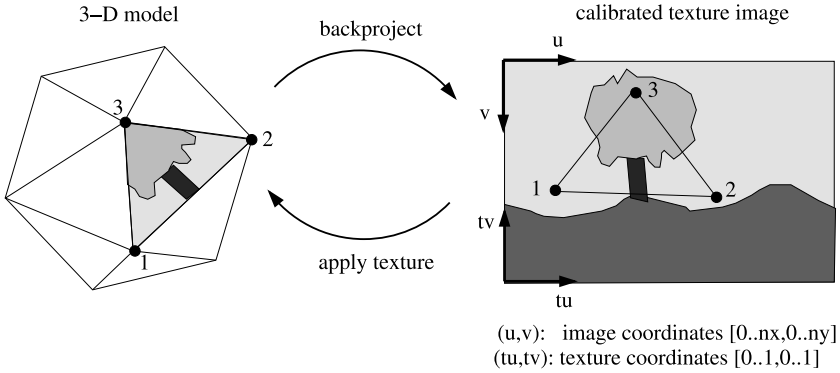

Fig. 5.6 Single triangle texture mapping: each triangle has a texture information stored as texture coordinates of a given image for each vertex of the model.

The figure shows also how the texture coordinates can be computed: the vertices of the 3–D model are backprojected onto the texture image according to the collinearity equation (equations 3.3, 3.4 and 3.5) and transformed in texture coordinates.
In case of an uncalibrated and unoriented texture image the following procedure can be used: few control points are selected manually on the 3–D model (using, e.g., the 3–D point cloud viewer described in section A.2.5) and on the texture image (see figure 5.7). The defined control points are then used to orient the texture image by bundle calibration (section 3.3). Since only few control points are selected, only the external orientation (position and angles) and the focal length can be determined with the calibration and orientation procedures. The results are however sufficiently accurate for texture mapping purposes.

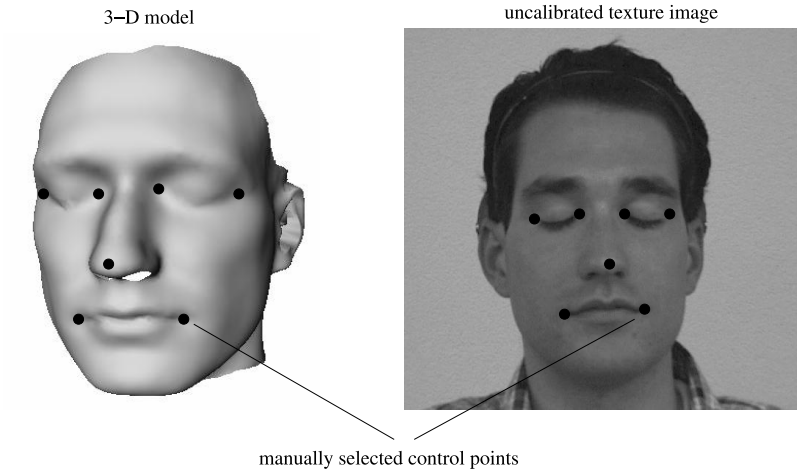

Fig. 5.7 Calibrating and orienting texture images: few control points are selected manually in the 3–D model and in the uncalibrated texture image. External orientation and focal length are then approximately determined by bundle calibration.

5.3 CONSIDERATIONS

The described surface measurement process is flexible and can be applied with different acquisition setups and for different purposes. It was successfully employed for face modeling using images acquired by five CCD cameras (D'Apuzzo, 2002b) (see also section 5.4), for human body modeling using three CCD cameras acquiring video sequences (D'Apuzzo et al., 2000) (see also section 7.7.2), for human body modeling using uncalibrated images acquired by a digital camera (Remondino, 2003), and for the modeling of blood vessels, where three CCD cameras acquired in eight positions twenty-four images (D'Apuzzo, 2001b) (see also section 5.5). In the next section, two of these applications are briefly presented to show the functionality of the proposed method.
The surface measurement procedure is a stand-alone process but it also constitutes part of the surface tracking process (described in chapter 7). In this case, the surface measurement process is performed for each frame of the acquired sequence.

5.4 APPLICATION 1: HUMAN FACE MODELING

The extensively employed methods to produce three dimensional computer models of the human face are laser scanning and coded light based triangulation approaches. The advantage of the presented method over these two techniques is the acquisition of the source data in a fraction of second, allowing the measurement of human faces with higher accuracy and even the possibility to measure dynamic events. Moreover, the developed software can be run on a normal home PC reducing the costs of the hardware.
In the past there was a collaboration for a project intended to evaluate the anatomical changes occurring with maxillo-facial and plastic surgery (Koch et al., 1996; D'Apuzzo, 1998). The goal of this actual work is the development of a portable, cheap and accurate system for the measurement and modeling of the human face (D'Apuzzo, 2002a).

5.4.1 System Setup

Figure 5.8 shows the setup of the used image acquisition system. It consists of five CCD cameras arranged in front of the subject. In case of required high accuracy, texture in form of random pattern can be projected simultaneously from two directions onto the face. The cameras are connected to a frame grabber which digitizes the images acquired by the five cameras at the resolution of 768x576 pixels with 8 bits quantization. A color image of the face without random pattern projection is acquired by an additional color video camera placed in front of the subject. It will be used for the realization of a photorealistic visualisation.

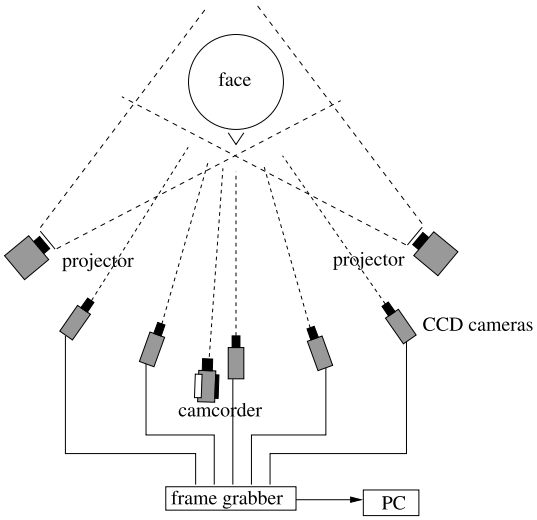

Fig. 5.8 Setup of cameras and projectors.

For calibration and orientation purposes, a calibration field with coded targets was designed specially for the face measurement process (see figure 5.9), so that all the target points could be viewed by the five cameras without occlusions. The camera system is then oriented and calibrated by bundle calibration method, as explained in the section 3.3.

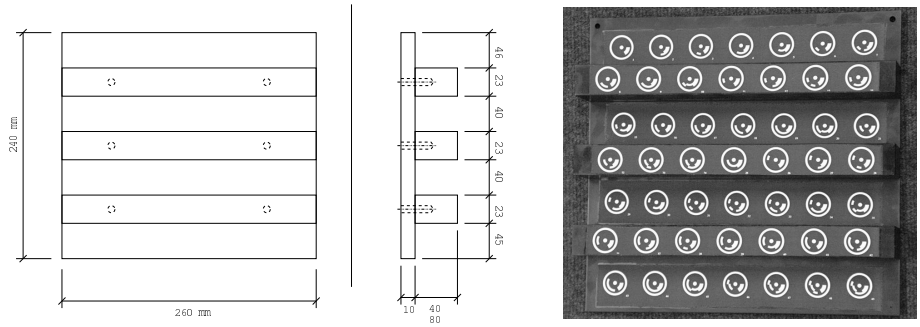

Fig. 5.9 Calibration field with coded targets. Left: construction design, right: an image.

Since the natural texture of the human skin is relatively uniform, the projection of an artificial texture onto the face is required to perform robustly the matching process. A random pattern (see figure 5.10) is preferred over regular patterns to avoid possible mismatches. In fact, regular pattern consist of repeating musters and the automatic matching process may easily produce erroneous correspondences. The resolution of

the projected random pattern has to be adjusted to the image format. It has to result in the acquired images in structures with the size of few pixels (see figure 5.11). The use of two projectors enables a focused texture even on the lateral sides of the face. Some results achieved without the projection of an artificial texture are presented in section 5.4.4. Figure 5.11 shows the result of the random pattern projection on a face and figure 5.12 shows the five images acquired by the cameras.

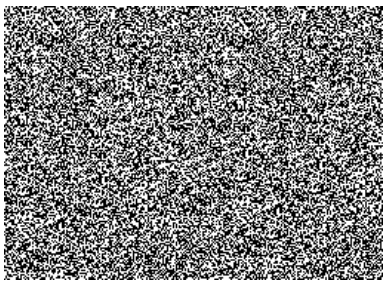

Fig. 5.10 Projected random pattern.

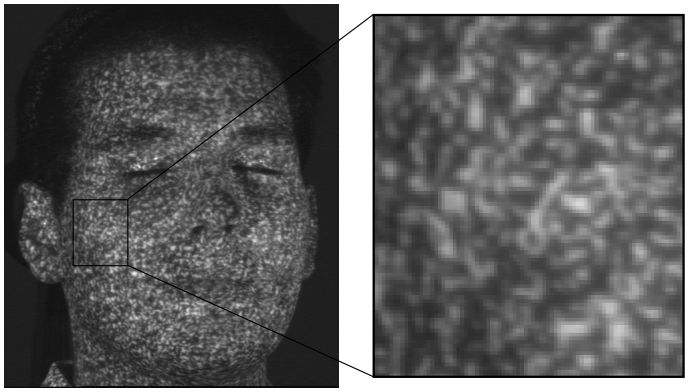

Fig. 5.11 Result of the random pattern projection on a face; right: a detail, in the acquired image the structures of the random pattern have the size of few pixels.

5.4.2 Surface Measurement

Since the human face is a steep surface and both sides of the face are not visible by the same camera, the five acquired images are used as two separate set of triplets, one for each side of the face (see figure 5.12). They are processed independently and at the end, the results are merged into a single data set.

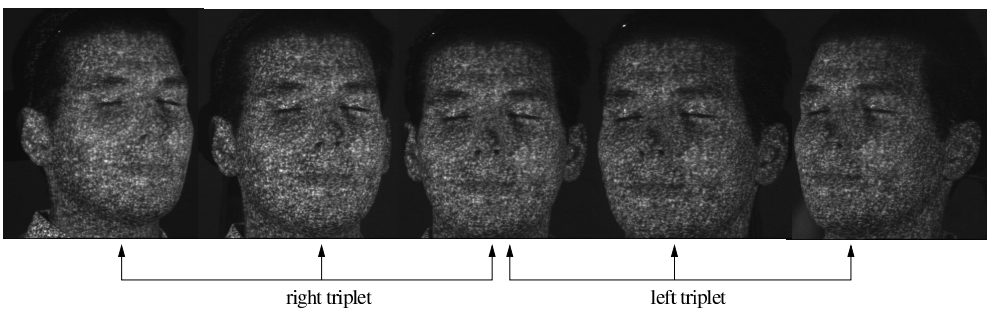

Fig. 5.12 The five images acquired by the cameras. For the processing two triplets are used, one for each side of the face.

The required intervention of the operator for the matching process is reduced to the semi-automated definition of about ten seed points (see section 4.2.1) and to the selection of a contour of the region to be measured (see figure 5.13). The operation can be performed in a couple of minutes, then the process will continue completely automatically.

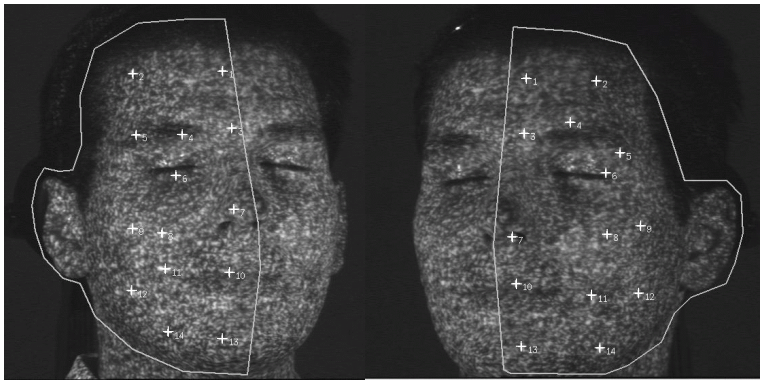

Fig. 5.13 Seed points and contour for the two sides of the face.

Figure 5.14 shows the two sets of matched corresponding points established by the matching process on the two halves of the face. On a Pentium III 600 MHz machine, about 25'000 points are matched on half of the face in about 10 minutes.

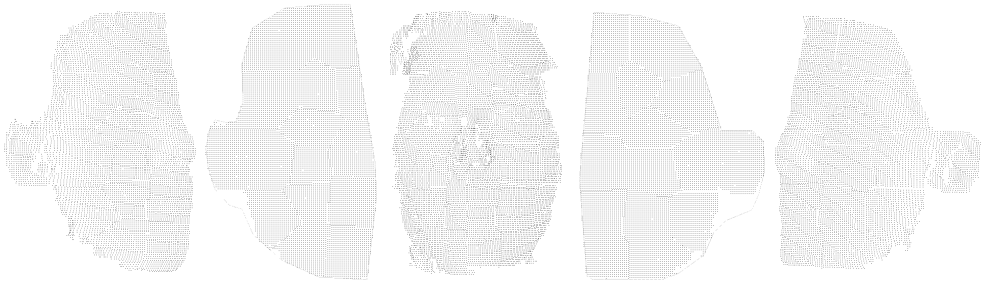

Fig. 5.14 Matched corresponding points in the five images of figure 5.12.

The computation of their 3–D coordinates is achieved by forward ray intersection (see section 5.1.1) and lasts a couple of seconds. The achieved precision ($\hat{\sigma}_i$ of equation 5.6) of the 3–D points is about 0.4 mm in the sagittal direction and about 0.2 mm in the lateral direction. As can be seen in figure 5.15 left, the point cloud is very dense (45'000 points) and some outliers are present. In the center line of the face, the region of overlap of the two joined data set can be observed. However, no relevant discrepancies of the two data set is present. Gaussian filters (Borghese and Ferrari, 2000) are applied to the 3–D point cloud and afterward thinned to reduce the number of points (see figure 5.15 right).

5.4.3 Modeling and Photo Realistic Visualisation

The last step of data processing is the generation of a triangulated surface from the cleaned point cloud and the application of color texture. A meshed surface is generated from the 3–D point cloud by 2.5–D Delauney triangulation.

Fig. 5.15 Left: Measured 3–D point cloud, 45'000 points. Right: after filtering and thinning, 10'000 points.

To achieve a photorealistic visualization, the natural texture acquired by the color video camera is draped over the model of the face. Figure 5.16 shows the surface model, the texture image and two views of the resulting face model with texture. Figure 5.17 shows two other examples of face models.

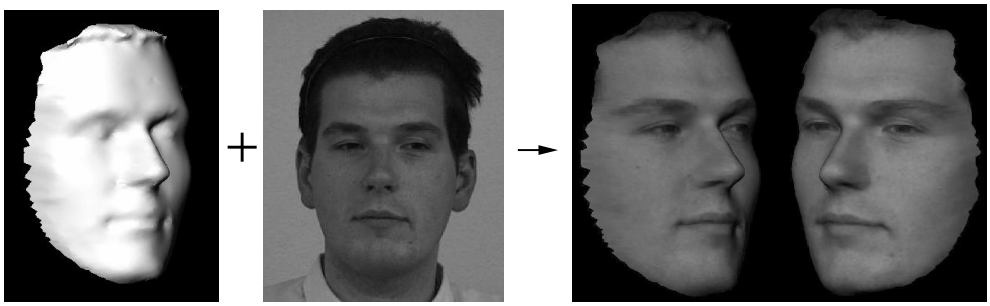

Fig. 5.16 Photorealistic visualisation. Left: shaded surface model, texture image. Right: face model with texture.

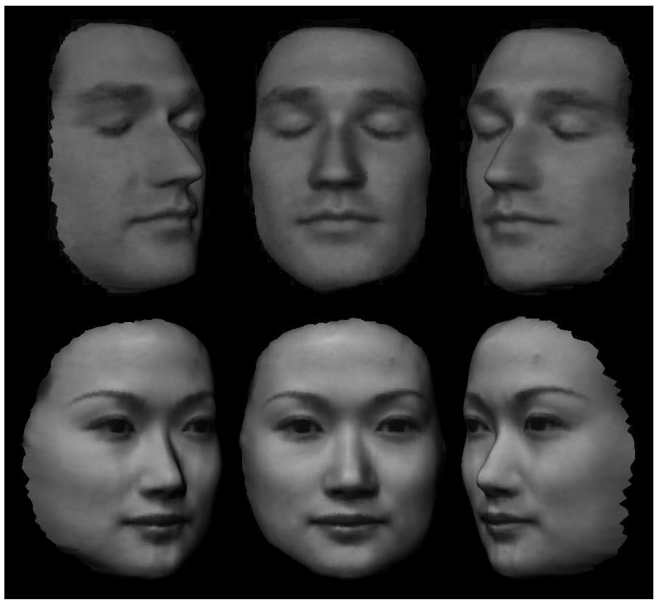

Fig. 5.17 Photorealistic visualisation. Two other examples of face models.

5.4.4 Measurement Without Artificial Texture Projection

The measurement of human faces can be performed also without projecting an artificial texture. In this case, the matching process will perform less robustly. An example is shown in figures 5.18 and 5.19. The acquisition system presented in appendix B was used to acquire five images of a human face (figure 5.18). The five images were used as two independent triplets to perform the matching process, as explained in section 5.4.2. The final result, after forward ray intersection is a point cloud of totally 17'000 points. As can be see in figure 5.19, blunders are present and some regions could not be measured. Indeed, the result is less robust and precise than the case with random pattern projection; the achieved theoretical precision ($\hat{\sigma}_i$ of equation 5.6) is about 2.5 mm in the sagittal direction and about 0.5 mm in the lateral direction.

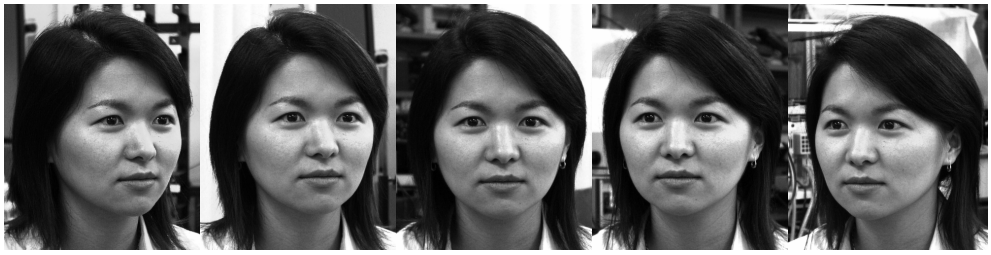

Fig. 5.18 Five images acquired by the cameras without the projection of an artificial texture.

Fig. 5.19 Result of the surface measurement process. 3–D point cloud with radiometric information (each point has a color value).

5.5 APPLICATION 2: MEASUREMENT OF BLOOD VESSEL BRANCHING CASTING

A pilot project was executed in cooperation with the department of radiology of the university hospital Zurich. The aim was to evaluate the accuracy levels of the current techniques used for the visualization and measurement of major blood vessels in the human body. Magnetic resonance (MR), computer tomography (CT) and digital subtraction (DS) angiographies were considered in this work. In the radiology literature, the comparison of different methods can be found. The scientific publications treat the detection of diseases using visualisation tools (Sommer et al., 1996; Brandt-Zawadzki and Heiserman, 1997; Kelekis et al., 1999; Skutta et al., 1999) or describe methods

for the measurement of blood vessels with CT techniques (Rubin et al., 1998), but a quantitative assessment is still missing. The goal of this pilot project was to establish if it was possible to determine the potential accuracy of the three techniques (CTA, MRA, DSA) for the measurement of blood vessels and compare them using unbiased data as reference. The abdominal aorta and its main branches (iliaca communis and iliaca externa) were chosen for this test (see figure 5.20).

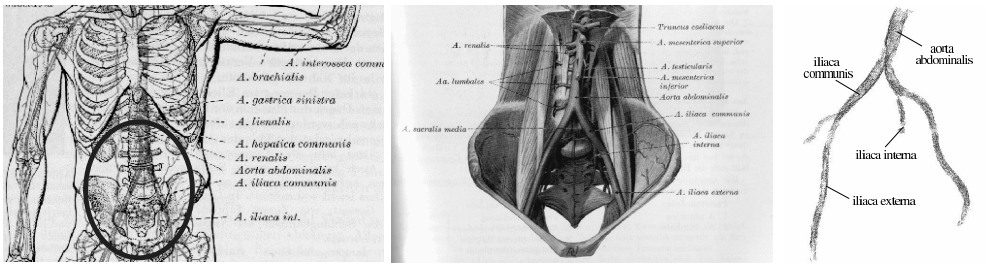

Fig. 5.20 Aorta abdominalis and its main branches: iliaca communis, iliaca externa, iliaca interna (Benninghoff and Gotter, 1961).

MR-, CT-, and DS- angiographies were first performed on a corpse. Then, a casting of the aorta abdominalis and its main branches (iliaca communis, iliaca externa) was prepared. The idea of the project was to measure the casting with optical methods and use the results as unbiased reference for a quantitative comparison between the three techniques.

5.5.1 System Setup and Calibration

The main characteristics of the aortal cast are: (1) the thin (0.5-1.5 cm), elongated (40 cm long) and branched shape, (2) its flexibility and (3) the complete lack of natural texture. For these characteristics the measurement task was rather difficult.
At first an adequate frame was constructed to immobilize the casting and sixteen white spheres were fixed on metal lines for the determination of the external orientation of the images. The rotation of the entire frame containing the blood vessel branching was possible without moving the signalized spheres. The design of the frame allowed an easy replacement of the casting with a new one. Figure 5.21 shows an image of the blood vessel branching casting fixed in the frame.

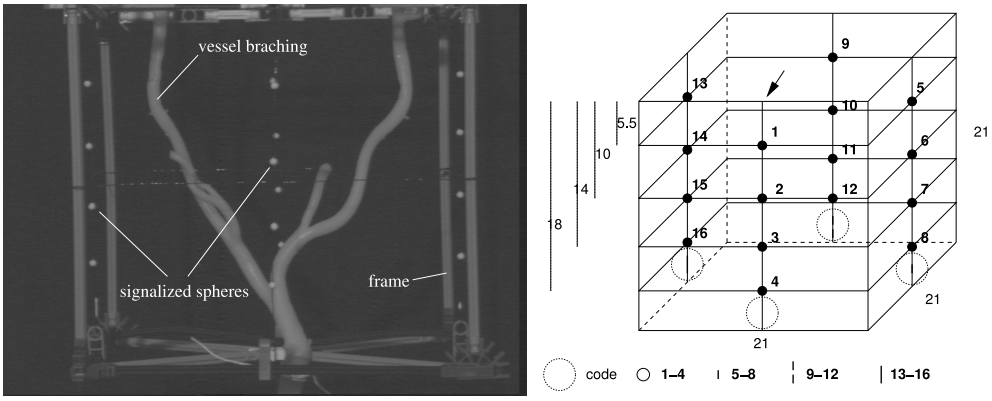

Fig. 5.21 Left: blood vessel branching casting fixed in the frame. Right: configuration of the frame.

Three CCD cameras with 768x576 pixel resolution were used for the image acquisition. Two slide projectors placed at both sides of the cameras projected a random pattern texture. To achieve a complete 360° imaging of the object, the frame was turned in eight positions. At each position, a triplet of images was acquired, resulting a total of twenty four images (figure 5.22).

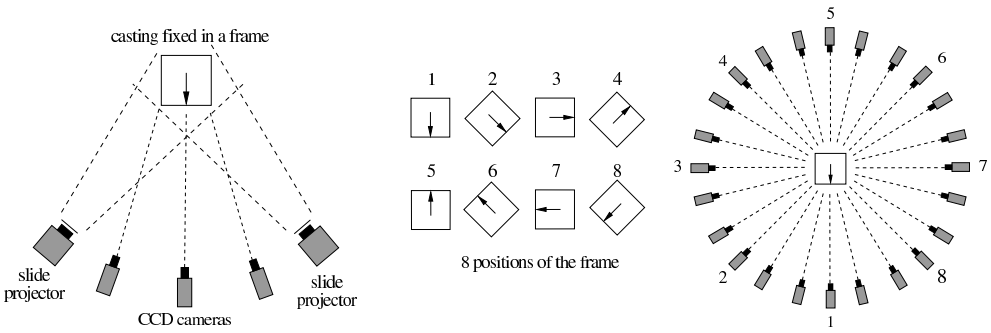

Fig. 5.22 Setup of the acquisition system: three CCD cameras and two slide projectors (left). The frame is turned in eight positions (center), this result in height triplets of images around the object (right).

Figure 5.23 shows a triplet of images for the first position and the images taken by the central camera at the height positions of the frame. As it can be seen, the shape of the casting was very thin and elongated. Its branched form, together with the frame construction, caused relatively wide occlusions (i.e., in the turns number 3 or 7). A large number of images was therefore required to assure the complete coverage of the object.

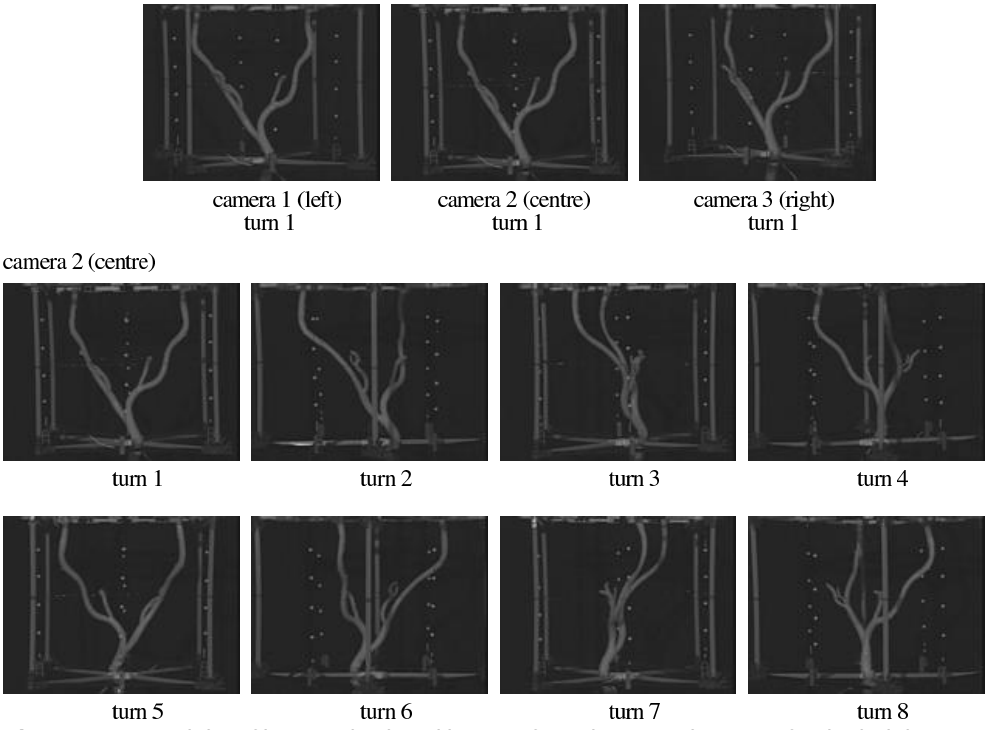

Fig. 5.23 One triplet of images (top) and images from the central camera for the height turns (bottom).

To calibrate the three CCD cameras, a 3–D reference field with coded target points was used (figure 3.4 left). The reference field was placed in the object space and the three CCD cameras acquired a triplet of images. The images were used to perform the calibration (internal orientation and determination of the lens distortion of the three CCD cameras) and to determine the external orientation for the first position. Once the calibration process was concluded, the reference field was replaced by the frame containing the blood vessel branching casting. The frame was turned eight times of about 45 degrees around its axis. This rotation can be considered equivalent to a displacement of the three cameras around the object (figure 5.22). To determine the external orientation at the different positions of the frame, the signalized spheres (figure 5.21) were used as control points. Their image coordinates were measured semi automatically by centroid operators and finally bundle adjustment led to a mean accuracy for the external orientation (position of the cameras) of about 0.8 mm in the three axis directions.

5.5.2 Measurement and Modeling

The matching process is performed independently for each triplet acquired by each turn of the frame. As the blood vessel branching casting had a complete lack of natural texture, the projection of an artificial texture was required to perform successfully the matching process. For this purpose, two slide projectors projected a random pattern from two directions (figure 5.22). Figure 5.24 shows the effects achieved.

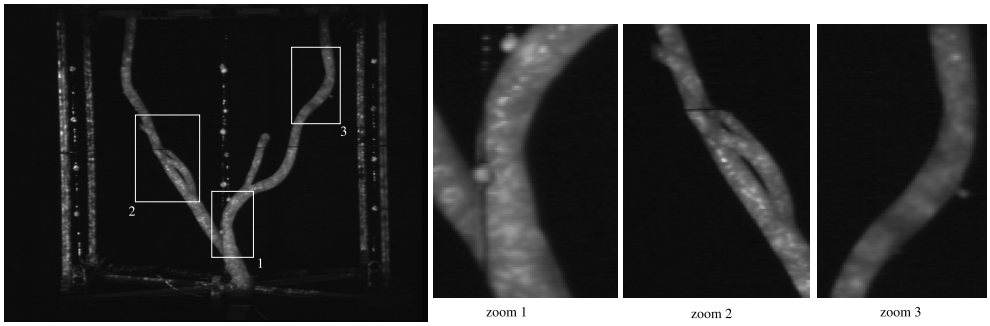

Fig. 5.24 Random pattern projection is required to perform the automated matching process. Left: acquired image of the blood vessel branching with artificial texture projection. Right: zoomed parts showing difficulties: 1 overlaps with signalized spheres, 2 self occlusion, 3 unfocused projection.

As the object was wide, the projected texture was unfocused in some regions of the surface. This undesired effect reduced the accuracy potential of the matching process. Moreover, the shape of the branching was critical: the main branching (iliaca externa) was about 17 pixels thick in the image. For the matching process, a patch size of 11x11 pixels was chosen, therefore only a thin part along the center of the branching could be matched. Occlusions caused by the fixation frame, the signalized spheres and the branching itself, also disturbed the automated matching procedure.
Combining all these effects, the matching process required many manual interventions. However, good matching results were achieved and dense sets of matched points could be determined for all the eight turns.
The 3–D coordinates of the matched points were computed with a mean accuracy of 0.2 mm by forward ray intersection using the known calibration and orientation

data. This process was performed independently for the eight data sets and the results were merged together into a cloud of about 13'000 points (figure 5.25, left). A 3–D filter (see section 5.1.2) was applied to reduce the amount of redundant data (overlap between data sets), to get a more uniform density of the point cloud and to reduce the noise. Figure 5.25 shows the point cloud before (left, 13'223 points) and after the filtering process using two different reduction levels (in the center 8'040 points and right 2'537 points).

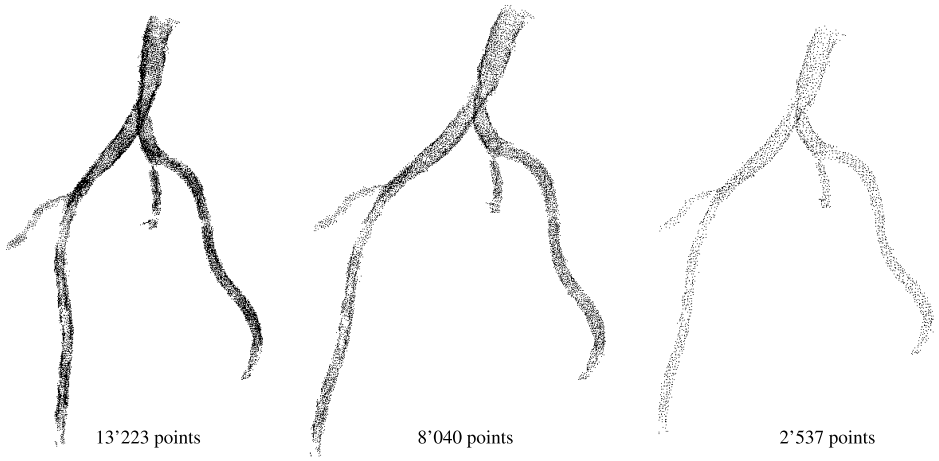

Fig. 5.25 Original 3–D point cloud (left) and two level of filtered data (center, right).

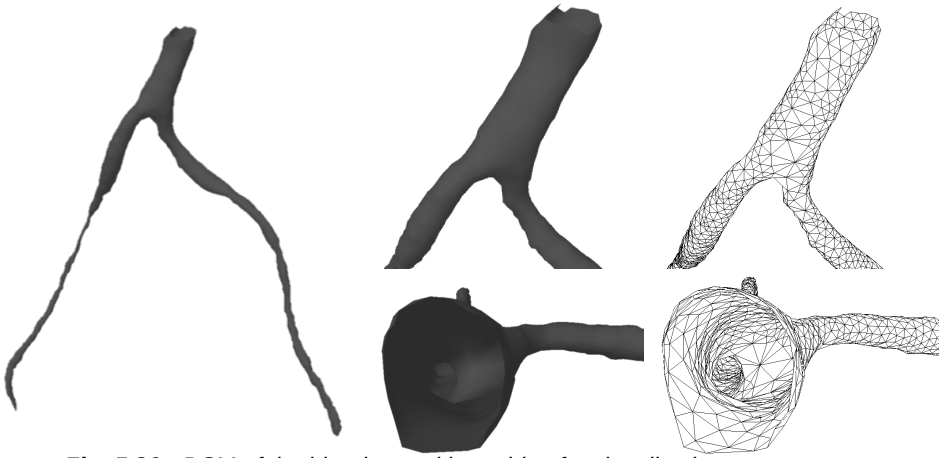

Fig. 5.26 DSM of the blood vessel branching for visualisation purposes.

In this work, the modeling process was intended only for visualisation purposes. The generation of a complete and unique digital surface model was a difficult task. Indeed, the 3–D point cloud was very dense in some regions and poor in others. The areas where the object was very thin caused the mainly problems. For example, the two iliacas interna were not fully measured, because of their extreme thinness and therefore these two parts were excluded from the modeling process.
In order to generate a surface model starting from unorganized 3–D point cloud, a commercial software (Wrap of Geomagic^{TM}) was used. Figure 5.26 shows the results of the modeling process. It works fine in the region of the aorta abdominalis and the iliaca communis, but rather poor at the end of the iliacas externa, where the blood vessels were very thin.