

Validation

This section treats the validation of the proposed surface measurement method. A solid reference object was measured as described in section 5.4 and the results were compared with the measurement achieved by a coordinate measurement machine. Section 6.1 describes the setup for the validation process, i.e., the acquisition system, the reference object and the reference data. Section 6.2 describes the measurement process and section 6.3 the surface comparison. Finally, section 6.4 presents the comparison results and section 6.5 some considerations.

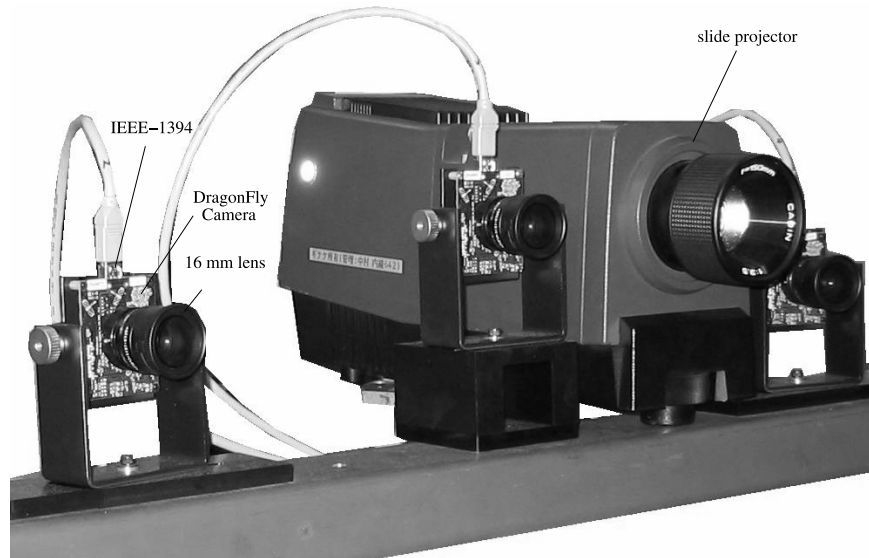


Fig. 6.1 Acquisition system used for validation (see appendix B): three progressive scan CCD cameras (model *DragonFly*) with a 680x480 pixel sensor and IEEE-1394 port. A slide projector is used for the projection of artificial texture onto the surface.

6.1 SETUP

This section describes the setup of the validation process. A multiple camera system as described in appendix B and by D'Apuzzo and Kochi (2003) was used. Additionally, a slide projector was used to project a random pattern onto the surface, as described in section 5.4. For the validation process only three of the five CCD cameras were used. Figure 6.1 shows the image acquisition system.



Fig. 6.2 Reference field used for calibration and orientation of the camera system.



Fig. 6.3 Dummy head used as reference object. Right: with random pattern projection.

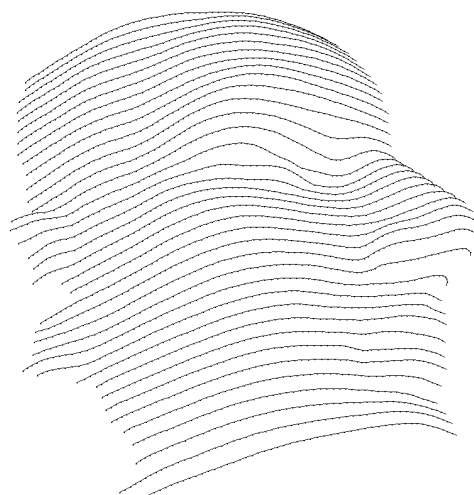


Fig. 6.4 Reference data: 2448 points in 37 profiles, measured by a coordinate measurement machine (*Carl Zeiss UMC 550S*) with an accuracy of 5 μm .

The camera system was calibrated and oriented as described by D'Apuzzo and Kochi (2003) using a reference field with signalized points of known 3-D coordinates (see figure 6.2). The accuracy of the calibration and orientation processes can be expressed according to equation 3.11 and results in this case in $\hat{\sigma}_0 = 0.5 \mu m$. For the used cameras, it corresponds to $1/15^{th}$ pixel.

A dummy head was used as reference object (see figure 6.3). This was firstly measured by a coordinate measurement machine (*Carl Zeiss UMC 550S*) with an object space accuracy of $5 \mu m$ (precision given by the constructor). The measurement results in profiles as shown in figure 6.4. A total number of 2448 points in 37 profiles were measured on a half side of the head.

6.2 SURFACE MEASUREMENT

Image 6.5 shows the three acquired images used for the surface measurement process. The full functionalities of the matching process proposed in chapter 4 were used. Figure 6.6 shows the template image with the defined contour and seed points. Figure 6.7 shows the result of the matching process (matching at every four pixels) and the result after grid regularization (see section 4.3.1).

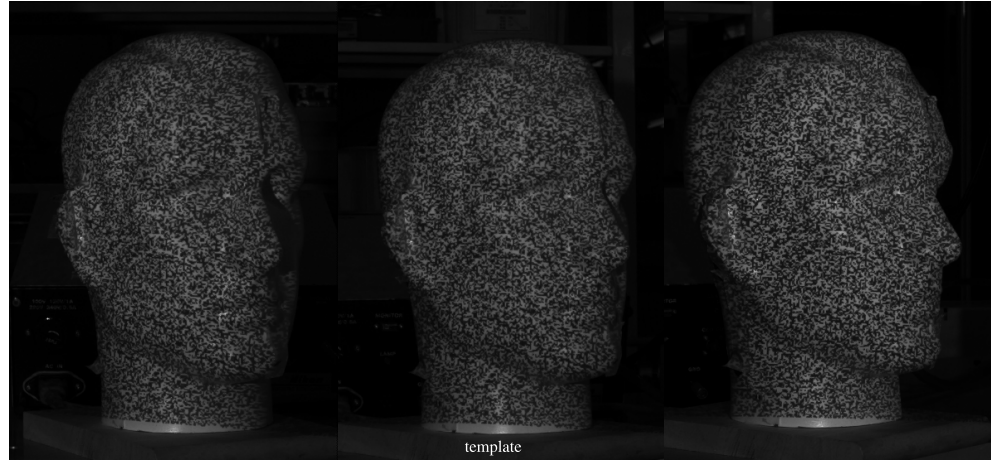


Fig. 6.5 Triplet of the dummy head with random pattern projection for surface measurement.

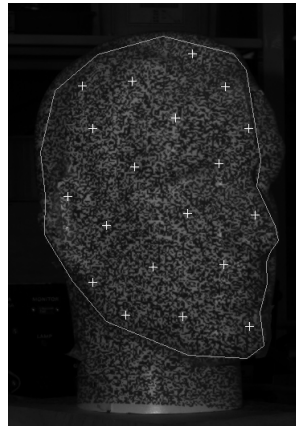


Fig. 6.6 Matching process: template image with defined contour and seed points.



Fig. 6.7 Left: result of the matching process (matching at every four pixels). Right: result after grid regularization.

Six different matching resolutions were used: matching each pixel in the images, matching every two pixels, three pixels, four pixels, six pixels and eight pixels. The forward ray intersection process (see section 5.1.1) results in the point clouds shown in figure 6.8. As can be seen in the figures, the results are consistent and few blunders were present (e.g., at the top of the head, on the tip of the nose). This will then be confirmed by the quantitative analysis of the results (see section 6.4).

The theoretical precision of the computed 3-D point can be expressed as the Helmert's point error σ according to:

$$\sigma^2 = \text{tr}(\mathbf{Q}) = \sigma_x^2 + \sigma_y^2 + \sigma_z^2 \quad (6.1)$$

where \mathbf{Q} is the covariance matrix and $\sigma_x, \sigma_y, \sigma_z$ are the variance factors of the three coordinates according to equation 5.6. These three values express the theoretical precisions in the three axes directions, whereas σ expresses an average point error independent from the orientation of the coordinate system. The results for the six cases are shown in table 6.1. The total number of matched points is listed together with the average theoretical precision $\bar{\sigma}$, its standard deviation and its maximal value. The next three columns list the same values ignoring blunders with a theoretical precision larger than 1.5 mm ($\sigma > 1.5 \text{ mm}$); the last column of the table lists the number of blunders.

Table 6.1 Theoretical precision of the six point clouds. Total number of matched points, average theoretical precision $\bar{\sigma}$ [mm], standard deviation [mm] and maximal value [mm]; same values ignoring blunders with $\sigma > 1.5 \text{ mm}$; number of blunders.

Surface	points	$\bar{\sigma}$	<i>std.dev.</i>	max σ	$\bar{\sigma}_{<1.5}$	<i>std.dev.</i>	max σ	- pt
1 pixel	92216	0.21	0.28	9.12	0.20	0.10	1.50	476
2 pixels	22944	0.19	0.09	2.88	0.19	0.07	1.41	19
3 pixels	10152	0.19	0.08	2.43	0.19	0.07	1.26	5
4 pixels	5722	0.19	0.08	1.76	0.19	0.07	1.39	4
6 pixels	2532	0.20	0.10	2.88	0.19	0.07	1.35	2
8 pixels	1428	0.19	0.07	1.24	-	-	-	-

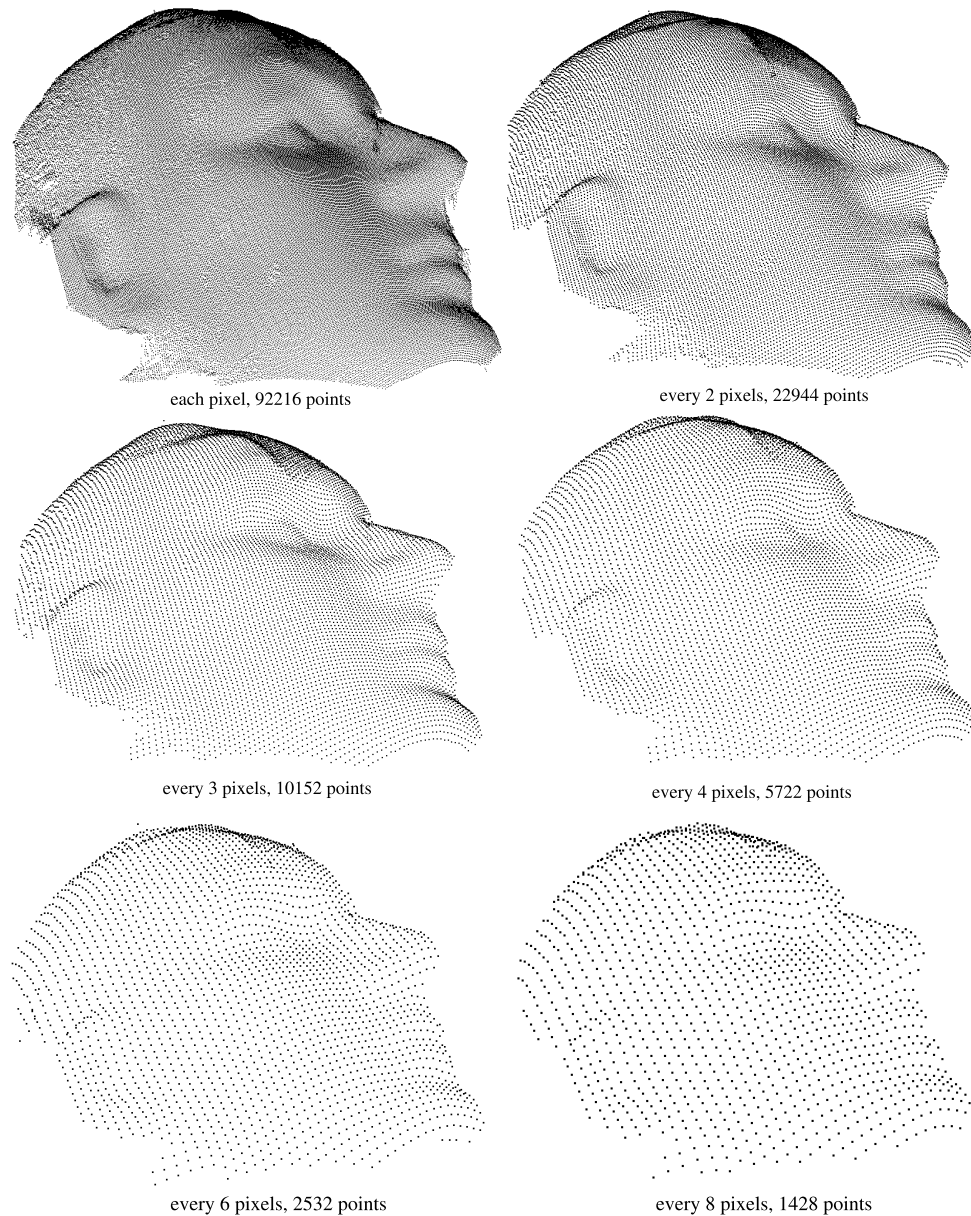


Fig. 6.8 Surface measurement of the reference object: point clouds resulted matching each pixel, every two, three, four, six and eight pixels.

6.3 SURFACE COMPARISON

To compare the measured surface with the reference data two tasks have to be performed. Firstly the two surface data have to be aligned and secondly the distance between the two surfaces has to be defined and determined. Additional problems to be considered for the comparison are: (a) the reference data (profiles) and the measured data (point cloud) have different structures, (b) they have different resolutions, (c) they are defined in different coordinate systems, (d) no common points, planes, or axes can be determined.

6.3.1 Alignment

The adequate solution to align two point clouds with different resolutions and no common points is provided by the ICP (Iterative Closest Point) algorithm proposed by Besl and McKay (1992). The commercial software *Geomagic Studio*TM was used for this process. The average error resulting from the alignment process is 0.19 mm. Figure 6.9 shows the results.

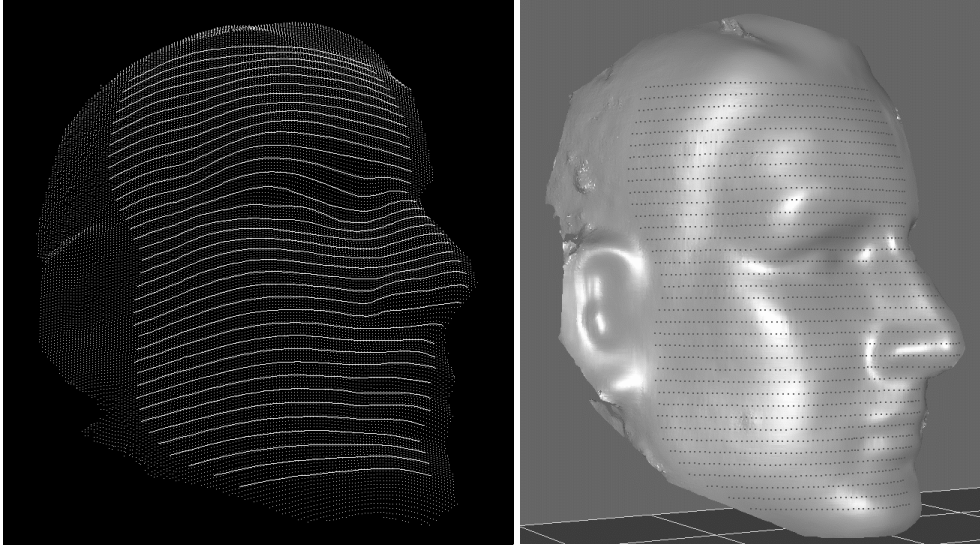


Fig. 6.9 Alignment result, average error is 0.19 mm. Left: visualisation of aligned reference data as profiles and measured data as point cloud. Right: snapshot of the commercial software *Geomagic Studio*TM.

6.3.2 Distance Reference Data to Measured Data

The determination of the distance between the reference data and the measured data is not a trivial task because their resolution is different and no common points, planes or axes exist. Since the measured point cloud (99216 points) has a much higher density than the reference data (2448 points), it was decided to compute for each point of the reference data the distance to the measured point cloud. Figure 6.10 shows the two data set. The figure 6.11 shows the detail of the dotted box of figure 6.10 and explains how the distance of the reference data to the measured data is computed.

The distance of a point in the reference data to the point cloud is computed as follows: (a) determine the three closest points in the point cloud, (b) compute the distance of the reference point to the plane defined by the three points found in the measured data. The determination of the distance of a point to a plane introduces an additional error, i.e., the approximation of a curved surface by a set of triangular facets, as shown in figure 6.12. The error added by this factor ($e1$ or $e2$ in figure 6.12) decreases with the increment of the density of the measured point cloud. For this reason it is more appropriate to compare the reference data with the measured point cloud with highest density. Therefore, the 92216 point cloud achieved by matching each pixel was used for the comparison. The distance in the space of two adjacent points ($l1$ or $l2$ in figure 6.12) is about 0.5 mm for this data. Assuming for the reference object a minimal curvature radius of 5 mm (r in figure 6.12), the maximal error of the distance to the real surface is $e = r \cdot (1 - \cos \sin^{-1} \frac{l}{2r}) = 0.01 \text{ mm}$. The error added by this factor can therefore be considered insignificant.

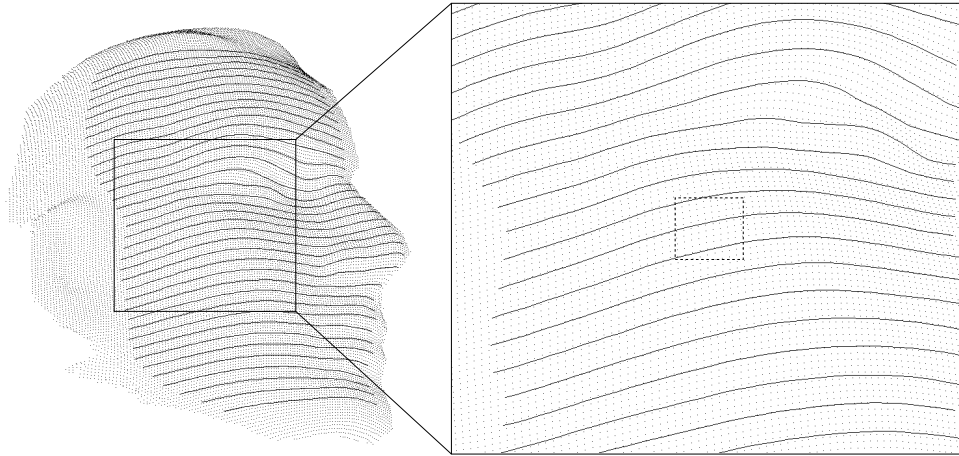


Fig. 6.10 Reference data as profiles and measured data as point cloud. Right: detail (no common points exist and the density of the two data set is different).

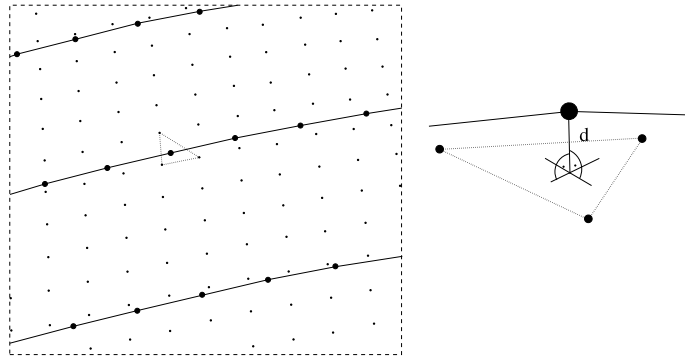


Fig. 6.11 Determination of the distance of the reference data to the measured data. Left: for each point in the reference data, the three closest point in the measured data are determined. Right: the distance d is computed as distance to the plane formed by these three points.

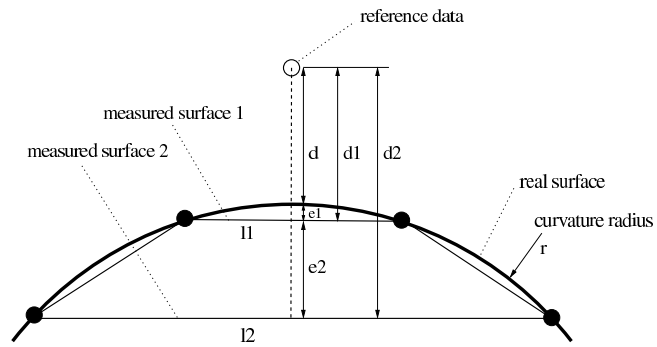


Fig. 6.12 The error caused by approximating the real surface with facets decrease with the increment of the density of the measured point cloud: $e1 < e2$.

6.4 COMPARISON RESULTS

This section will present and discuss the results of the comparison of the reference data and the measured data as well as the comparison of the six different measured point clouds.

6.4.1 Distance Error

The distance of the reference data to the measured point cloud is used as distance error. Table 6.2 summarizes the comparison of the reference data to the point cloud resulted by matching each pixel of the template image (92216 points). Values with sign and absolute values are given. The average absolute error of 0.2 mm can be considered the value describing the precision of the measurement system. Figures 6.13 and 6.14 show graphically the error distribution.

Table 6.2 Distance reference data to measured data (values with sign and absolute values). Average [mm], standard deviation [mm], minimal and maximal values [mm].

	average	std.dev.	min	max
error	-0.02	0.27	-1.13	1.29
absolute error	0.20	0.18	0.00	1.29

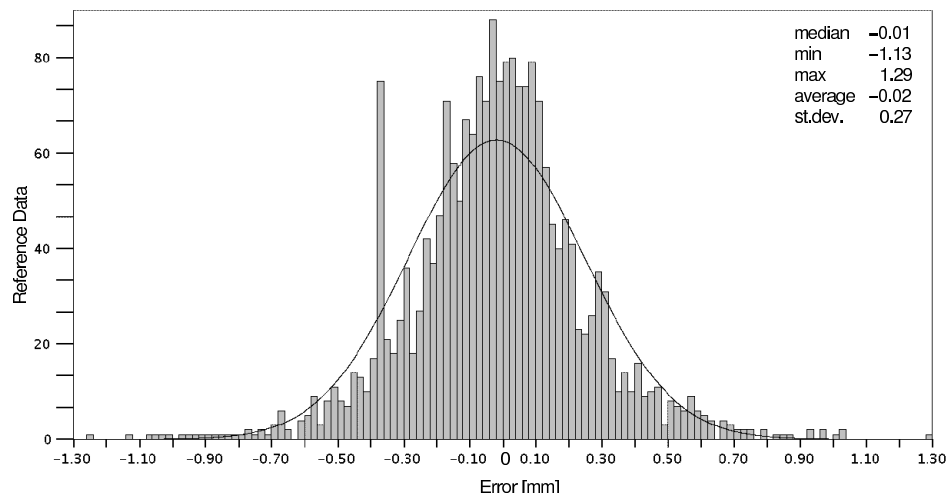


Fig. 6.13 Distance reference data to measured data: error distribution

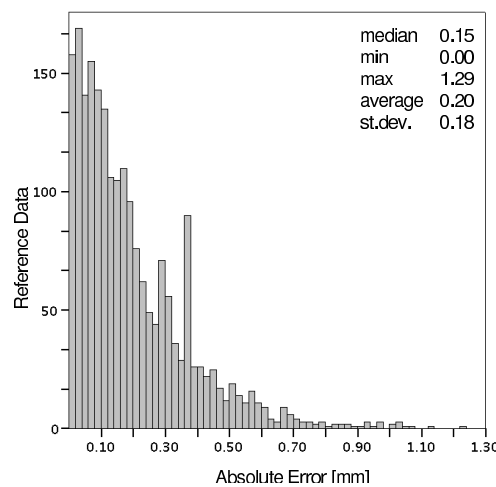


Fig. 6.14 Distance reference data to measured data: absolute error distribution.

Additionally, the commercial software *Geomagic Qualify*TM was used to compare the surface with the reference data. Table 6.3 summarizes the results. They confirm the achievements of the computation of the single distances.

Table 6.3 Absolute distance of the measured surface to the reference data, achieved by the software *Geomagic Qualify*TM. Average [mm], standard deviation [mm], minimal and maximal values [mm].

	average	std.dev.	min	max
absolute error	0.19	0.20	0.00	1.36

Unfortunately, no further numerical values can be obtained from the software. However, a graphical representation of the error distribution can be plotted as shown in the figure 6.15. As can be seen, the larger errors appear in the expected difficult areas, as for example steep surfaces, regions with reflexions and regions where the projected texture was unfocused. These results confirm the assumptions made in chapter 4.

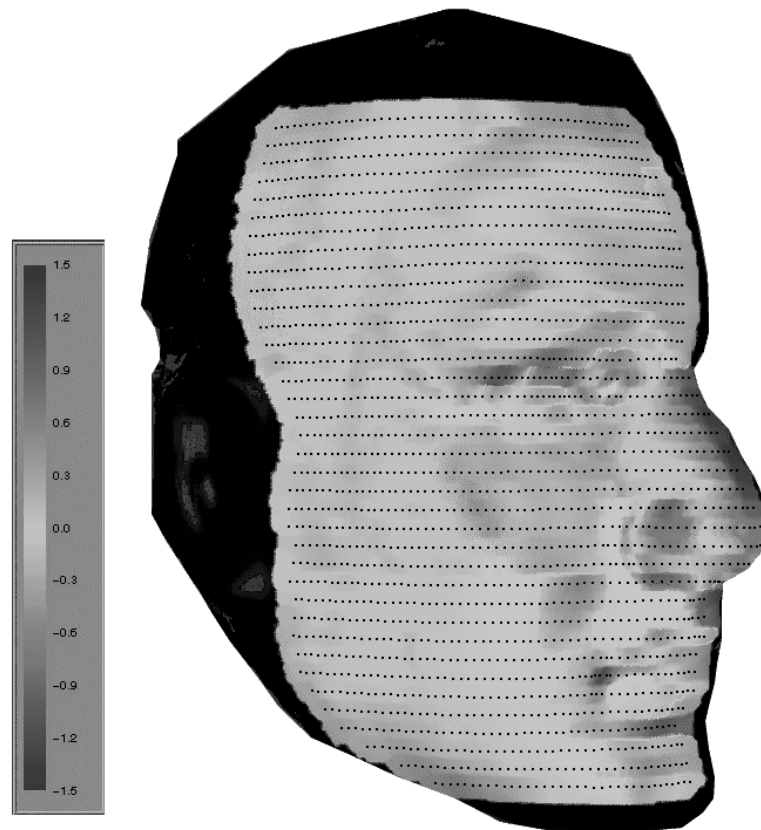


Fig. 6.15 Distance of the measured point cloud to the reference data, from software *Geomagic Qualify*TM.

6.4.2 Comparison of Surfaces

A further interesting analysis of the measured data is the comparison between surfaces resulted by matching with different point densities. The commercial software *Geomagic Qualify*TM was used for this task.

The point cloud with highest density (92216 points) was used as reference data. The results of the comparison are listed in the table 6.4. The average distance of the surfaces remains in all cases below 0.1 mm. Thus, it can be concluded that no relevant accuracy decrease can be observed by matching with different point densities.

Table 6.4 Difference between surfaces resulted by matching with different point densities. The 92216 point cloud was used as reference data. Number of points, average distance [mm], standard deviation [mm], maximal distance [mm].

Surface	points	average distance	std. dev.	maximal distance
2 pixels	22944	0.09	0.11	0.99
3 pixels	10152	0.09	0.11	1.00
4 pixels	5722	0.09	0.11	1.00
6 pixels	2532	0.10	0.13	1.48
8 pixels	1428	0.10	0.14	1.45

6.5 CONSIDERATIONS

The first important consideration to underline is the full consistency of all the validation steps. In fact, the average error of the alignment process (0.19 mm), the theoretical precision of the forward ray intersection (0.20 mm) and the average absolute distance of the reference data to the measured surface (0.20 mm) results in the identical value of 0.2 mm. This value can therefore be considered as real accuracy of the measurement process achieved with the proposed acquisition system. Different accuracy values can however be expected depending on the used acquisition system (different cameras, different image format). The accuracy of the calibration and orientation process also influence the achieved measurement accuracy.

On the other hand, it has to be considered that a dummy head was used for the validation process. The measurement of the human skin presents in fact more difficulties caused by, e.g., its roughness, hairs, reflections. Anyhow, the validation process also proved that the theoretical precision resulting from the forward ray intersection process is consistent with the achieved accuracy of the measurements and can therefore be used as a quality indicator.

Surface Tracking

In section 7.1 the multi-image tracking process is introduced and the method for tracking single points is described in detail. In section 7.2 the process is extended for tracking entire surface parts. Section 7.3 describes the special case of tracking *key-points* in the vector field of trajectories. The software implementation is discussed in section 7.4. The options and parameters to be set are described in section 7.5. Finally, section 7.7 presents results achieved by the proposed method for full body motion capture.

7.1 MULTI-IMAGE TRACKING PROCESS

7.1.1 Tracking in Multi-image Space

The basic idea of the multi-image tracking process is to track corresponding points in the multi-images through the sequence. Therefore, at the end of the process it is possible to compute their 3-D trajectories. The process is based on least squares matching techniques (*LSM*). The least squares matching algorithm introduced in section 4.1 is used to establish the spatial correspondences between the images of the different views (*spatial LSM*) and the temporal correspondences between subsequent frames (*temporal LSM*). Figure 7.1 graphically depicts the principle.

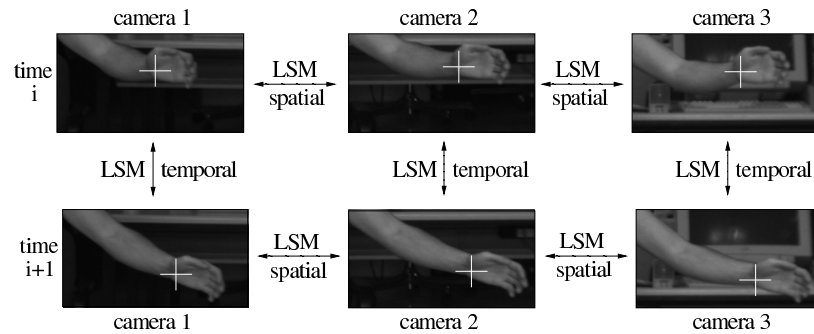


Fig. 7.1 LSM tracking: temporal and spatial correspondences are established with LSM.

The process starts from matched corresponding points in the multi-images and continues with the tracking loop: (1) predicting the position in the next frame, (2) searching the position with the best cross-correlation value and (3) establishing the point in the next frames using least squares matching (*temporal LSM*). These three steps are

7 SURFACE TRACKING

computed independently for the multi-images. The figure 7.2 shows the process graphically.

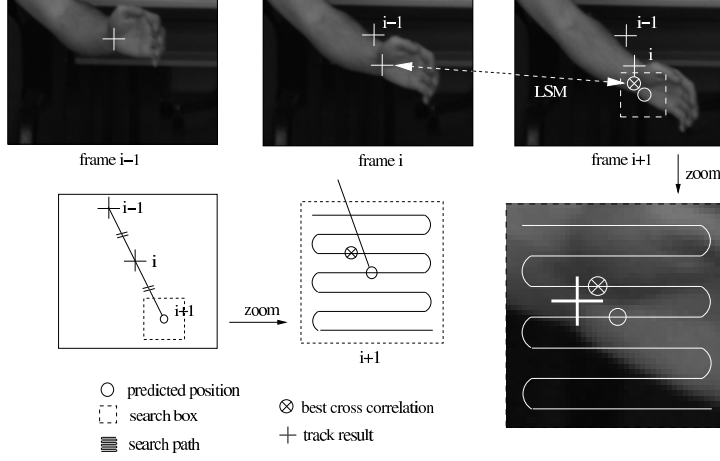


Fig. 7.2 Tracking in image space: *temporal LSM* is applied at the position of best cross-correlation.

The position of the tracked point at time $i+1$ is linearly predicted from the two previous times $i-1$ and i (step 1). A search box is defined around this predicted position in the frame at time $i+1$ and is scanned for the position which has the best value of cross-correlation between the frame at time i and the frame at time $i+1$ (step 2). The least squares matching algorithm is applied at that position and if the result passes the quality control (same as explained in section 4.1.4, without geometrical constraints) it can be considered the exact position of the tracked point in the new frame (step 3). Figure 7.3 shows how the tracking in image space starts. At the beginning of the process, it is not possible to predict the position of the point in the next time step. In this case, the search box has larger sizes and it is centered in the position of the actual time step (image coordinates (x_i, y_i) in figure 7.3). Once the prediction is possible, the search box is smaller and centered in the predicted position (circle in frame $i+2$ in figure 7.3).

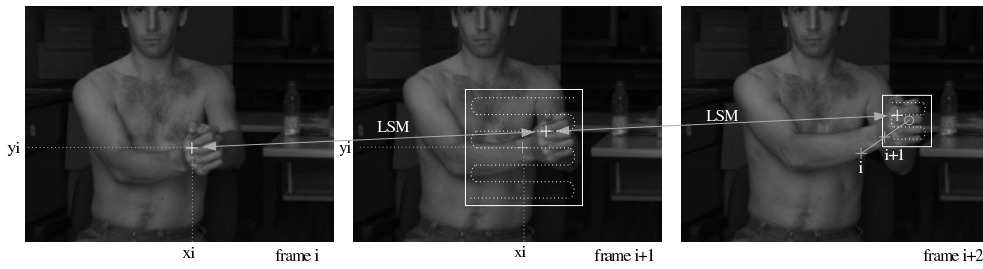


Fig. 7.3 Tracking in image space: at the beginning of the process (when the prediction is not yet possible) the location in the next time step is searched in a larger area centered in the location of the previous frame (image coordinates (x_i, y_i)). Once the point is tracked at least in two frames, the prediction is possible and the search area is therefore smaller and centered in the predicted position (circle in frame $i+2$).

This process is performed independently for the different images. To test the individual results, *spatial LSM* is computed at the positions resulting from the *temporal LSMs* and if no significant differences occur between the two matches, the point is

7.1 MULTI-IMAGE TRACKING PROCESS

considered tracked and the process continues to the next time step. If instead the differences are too large, the step (2) of the process is repeated by searching the value of best cross-correlation in a larger region around the predicted position. LSM is then applied again (step 3) and if the result is rejected again, the tracking process stops. The flowchart in figure 7.4 illustrates all the operations of the multi-image tracking process in detail.

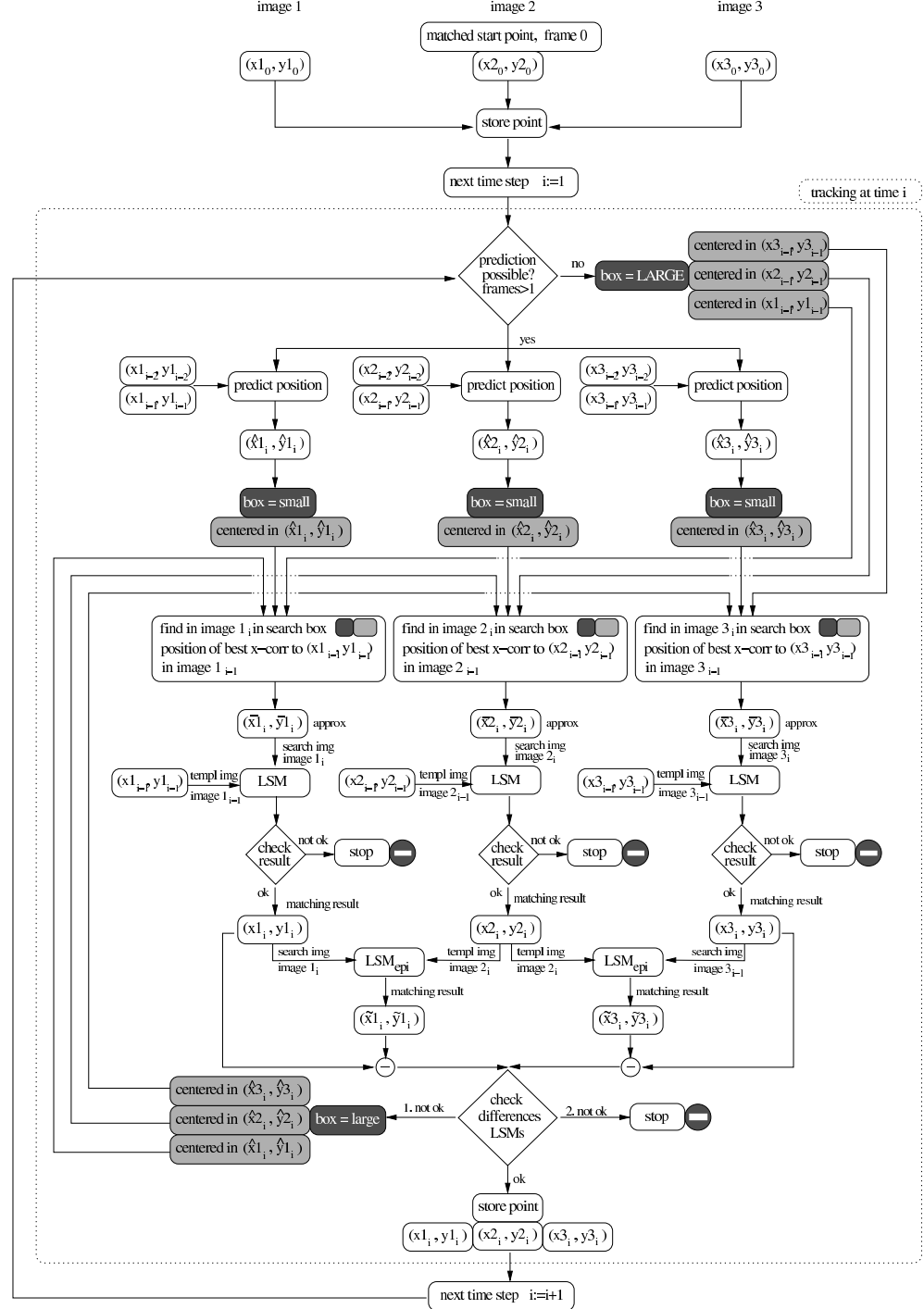


Fig. 7.4 Flowchart of the multi-image tracking process.

7.1.2 3-D Trajectories

The result of the multi-image tracking process applied to a single point are its coordinates in the multi-images through the sequences, thus the 3-D coordinates of the point for each time step can be computed by forward ray intersection (see section 5.1.1), finally resulting in its 3-D trajectory. Its velocity and acceleration can also be determined for each time step. Figure 7.5 shows an example of several 3-D trajectories of points on the surface of a person walking.

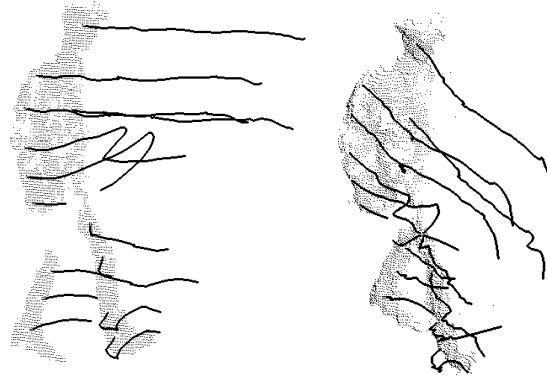


Fig. 7.5 Measured 3-D point cloud of the first frame and 3-D trajectories of single points tracked on a walking person (sequence of figure 7.17).

The proposed process can be used to track well defined points on a human body surface (e.g., signalized target points). However, trajectories of single points are not sufficient to understand and record the motion and movement of a human or the changes of a human body surface part. The next two sections propose a method to extract more complex information from the image sequence.

7.2 TRACKING SURFACE PARTS

Tracking surface parts means tracking simultaneously 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 dense vector field of trajectories.

7.2.1 Extended Algorithm

The requirement for the surface tracking process is completing the matching process on the surface parts of interest for at least the first multi-image set of the sequence. However, it is recommended to perform the process with a defined frequency, e.g., each two multi-image set, for the entire length of the sequence. This can be performed automatically with the process described in section 4.2. The tracking process depicted in section 7.1 is then applied to all the points measured on the surface of the first frames. With this approach, a new problem must be considered: during the sequence, some surface parts can get lost by occlusions and new surface parts can appear. For these reasons, a new functionality has to be integrated in the tracking process: before passing to the next time step, the local density of the data resulted from the tracking process is compared to a threshold. This operation is executed with a defined frequency, e.g., each two frames. In the regions of low density, new points are imported from the

previously computed data, i.e., surface matching process for the multi-image sets. In this way, newly appearing surface parts or lost points are integrated in the tracking process. Figure 7.6 shows the extended flowchart for the surface tracking process. For the determination of the local density of the matched data, the template image is divided into a regular grid whose size can be chosen. The number of matched points is counted for each grid element and if it is lower than a threshold (e.g., 80%), then the grid element is classified as a region with low density.

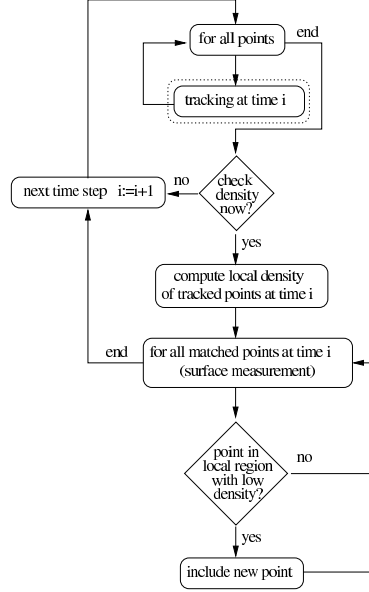


Fig. 7.6 Flowchart: surface tracking (dotted box "tracking at time i", from figure 7.4).

Figure 7.7 shows an example of the result of the surface tracking process applied to a sequence of a walking person acquired by two cameras (sequence of figure 7.17). The tracked points are displayed as trajectories from the previous frame. Figure 7.8 shows a detail of the tracked points, in the frame of time $i+1$ the newly included points can be recognized as single dots, as these points do not have any tracking information yet.

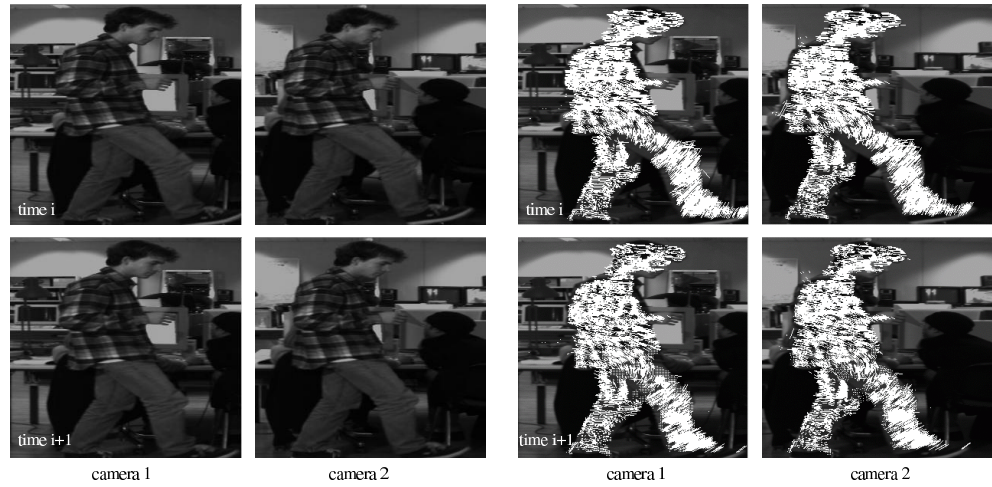


Fig. 7.7 Example of surface tracking. Left: image pairs acquired at time i and $i+1$; right: tracked points displayed as trajectory from the previous frame.

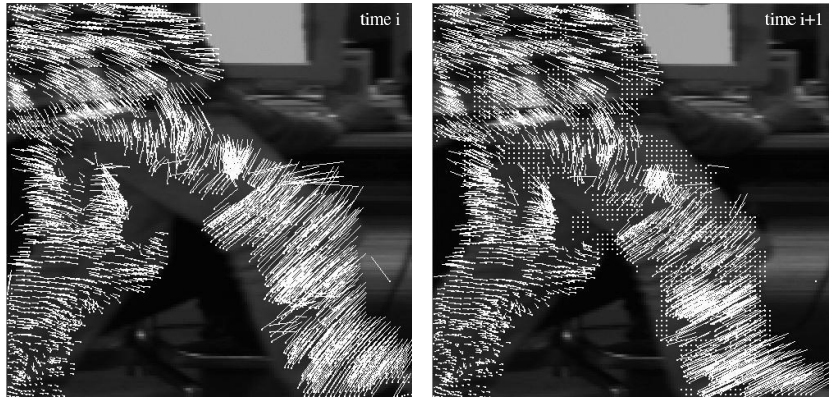


Fig. 7.8 Example of surface tracking. Detail from figure 7.7: in the frame at time $i+1$ (right) the new included points can be recognized as single dots.

7.2.2 Filtering

Tracking entire surface parts and not only good signalized single points can produce undesirable effects. Especially in case of poor texture of the surface to be measured, the tracking process may produce false trajectories. An example is shown in figure 7.9, which displays the computed 3-D trajectories for a part of the arm of a person walking (sequence of figure 7.17).

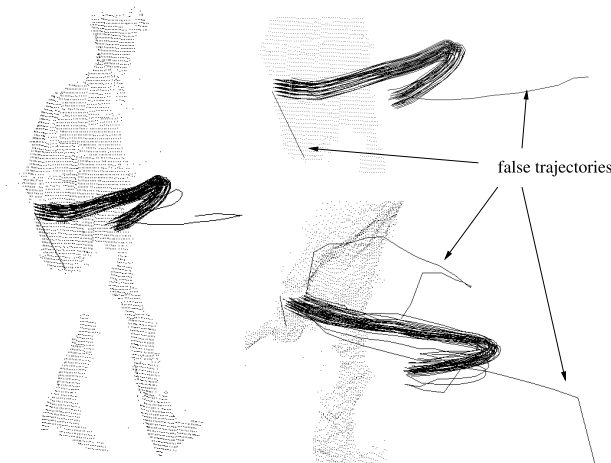


Fig. 7.9 Surface tracking: the false trajectories do not follow the uniform movement.

The false trajectories can easily be recognized because they do not follow the common movement of the majority. The vector field of trajectories can indeed be checked for consistency and local uniformity of the movement. Two filters are applied to the results to delete the false trajectories entirely or to truncate the false part of them.

7.2.2.1 Velocity and Acceleration Filter. The first filter removes large errors using velocity and acceleration thresholds (figure 7.10 left). The filter exploits the physical limitations of humans regarding velocities and accelerations of performed movements. Depending on the situation (e.g., walking, running), the two thresholds

of maximal velocity and acceleration can be defined at the beginning of the process and remain constant during the sequence.

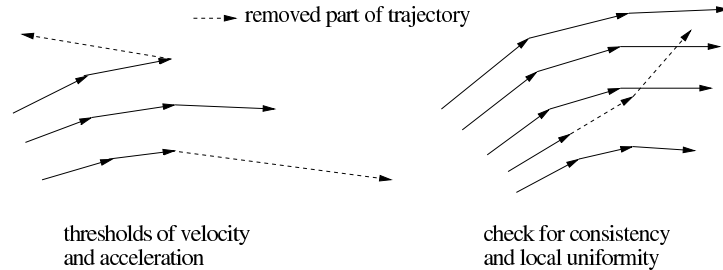


Fig. 7.10 Left: velocity and acceleration threshold filters. Right: uniformity filter.

7.2.2.2 Uniformity Filter. The second filter checks for the local uniformity of the movement, both in space and time (see figure 7.10 right). To check this property, the space is divided into regular voxels and local mean values of the velocity vectors are computed in each voxel for each time step. The single trajectories are compared to the local mean values of the velocity vector and truncated or removed if the differences in magnitude and direction are too large. The level of difference is computed as percentage of difference of the magnitude and as angle to the local mean velocity vector. Figure 7.11 shows graphically the uniformity filtering process and the flowchart in figure 7.12 explains it in detail.

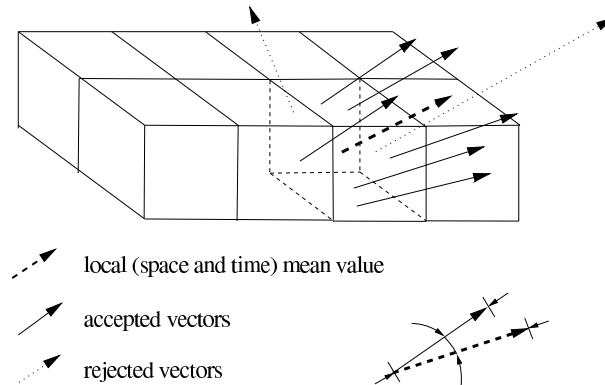


Fig. 7.11 Uniformity filter: the single vectors are compared in magnitude and angle with the mean vector.

7.2.2.3 Problems. After filtering the trajectories, problems may still remain. An ideal trajectory starts from the begin of the sequence and lasts until the end. However, in some cases, depending on the quality of the image sequence, on the type of surface to be tracked and on the movement performed, the result can be a set of broken trajectories with varying lengths (see figure 7.13). This effect is particularly strong when measuring the motion of a whole human body and is caused by occlusions, lack of texture, lost of tracked points and the appearance of new points. In order to solve this problem, the concept of *key-point* is introduced in the next section.

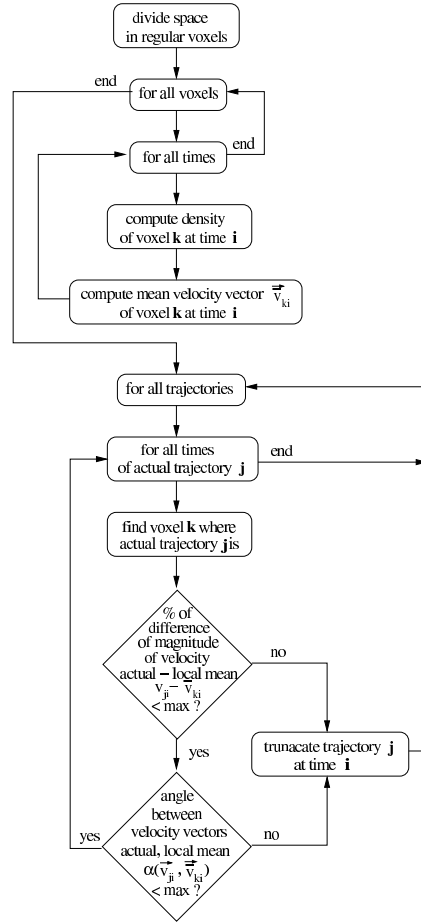


Fig. 7.12 Flowchart of uniformity filter.

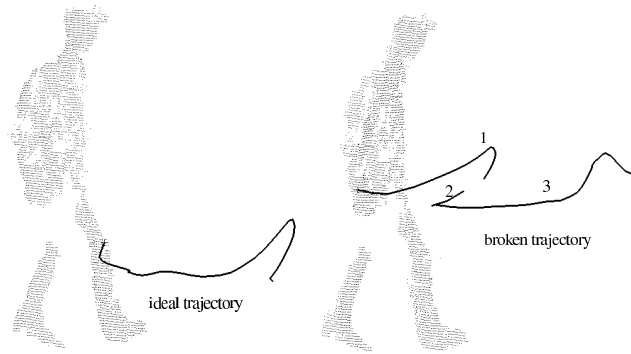


Fig. 7.13 Ideal and broken trajectories, an example.

7.3 TRACKING KEY-POINTS

7.3.1 Definition

The *key-point* is a 3-D region defined in the vectorfield of trajectories, whose size can vary and whose position is defined by its center of gravity. The key-points are interactively defined in a graphical user interface; they can be placed at any time step

and moved in the 3-D space (see figure 7.14). This will be described in detail in section A.2.7.

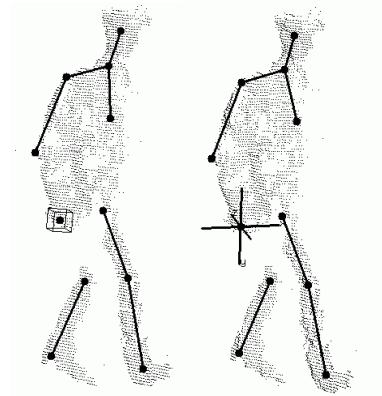


Fig. 7.14 Interactive definition of key-points. Left: definition of the size, right: moving the key-point in the 3-D space.

The key-points exploit the large amount of data resulted from the tracking process and extract a general description of the performed movement: a set of 3-D trajectories of selected key-points results from a dense vectorfield of trajectories.

7.3.2 Tracking

The key-points are tracked in a simple way: the position in the next time step is established by computing the mean value of the displacement of all the trajectories inside the region which define the key-point (see figure 7.15). The tracking process is explained in detail by the flowchart in figure 7.16.

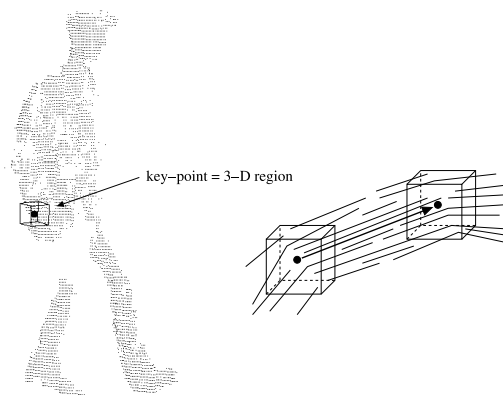


Fig. 7.15 Tracking key-points.

It starts by counting the trajectories inside the 3-D region of the key-point and by computing their average displacement in space to the next time step. This is added to the actual 3-D position of the key-point to determine the position in the next time step. This process is repeated until no trajectories are found in the 3-D region of the key-point.

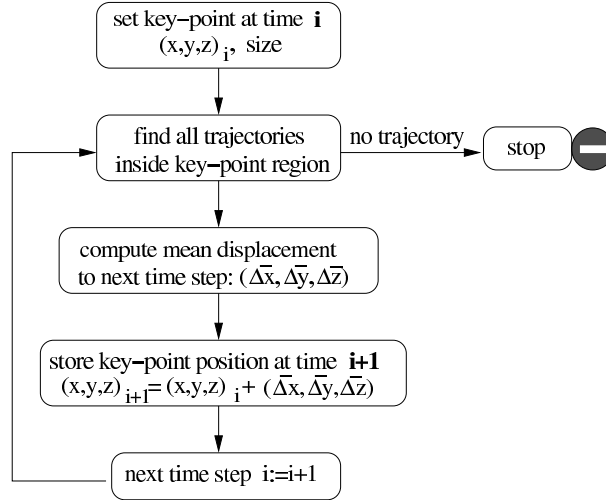


Fig. 7.16 Flowchart of the key-point tracking process.

7.3.3 Considerations

Since the key-points are defined interactively in the graphical user interface, they cannot be placed precisely. Therefore, they can be used to give a qualitative 3-D description of the movement performed by a person and not to measure it precisely. The size of the region to be tracked has an important role: if a small size is chosen, the key-point can be considered with a good approximation to lie on the tracked surface, in this case representing fictitious markers on the human body. If, on the other hand, larger sizes are chosen, the key-point represents a more approximative position of, e.g., joints of the human body. Two examples are given in section 7.7.3.

7.4 SOFTWARE IMPLEMENTATION

The amount of data to take into consideration is one of the problems for the implementation of the multi-image tracking process. A multi-image sequence is in fact composed of $n \cdot k$ images, where n is the number of used cameras and k the length of the acquired sequence, i.e., the number of acquired multi-image sets. The resulting data may also become very large. In fact, the tracked trajectories are composed of dense sets of corresponding points in each multi-image set of the sequence.

For the implementation of the tracking process it is therefore preferable to read in the memory of the PC only the required images (i.e., the multi-images of the previous, the actual and the next time steps) and to store the results directly to a file while keeping in the memory only the used data (i.e., the tracked points in the previous two frames). For this reason, *ring buffers* (with the length of three) were used to store the multi-images and the information about the tracked points. Ring buffers are circular array structures that permit a continuous increment (also negative) of the index still having a fixed size (in this case, three). Therefore, during the entire tracking process, always the same amount of memory will be required.

A second problem to solve is the way to store the results of the tracking process. Since the process proceeds forward with the time, the tracked trajectories are firstly stored if the following format:

```

frame number
  point number  x1 y1    x2 y2    x3 y3
  point number  x1 y1    x2 y2    x3 y3
  point number  ...
  ...
  -1
frame number
  point number  x1 y1    x2 y2    x3 y3
  ...
  -1
frame number
...
-1

```

However, this format is not adequate for post processing such as the computation of velocities and acceleration, filtering, visualization and tracking key-points. The trajectory file is therefore converted into the following more suitable form:

```

point number
  frame number  x1 y1    x2 y2    x3 y3
  frame number  x1 y1    x2 y2    x3 y3
  frame number  ...
  ...
  -1
point number
  frame number  x1 y1    x2 y2    x3 y3
  ...
  -1
point number
...
-1

```

7.5 OPTIONS

Different options and parameters have to be set for the tracking process and for post processing. The tracking process parameters can be divided into four groups: the input data, the LSM parameters, the tracking parameters and the filtering parameters.

7.5.1 Input Data

The required input data are the multi-image sequences, the orientation and calibration information of the cameras and, in case of surface tracking, the set of matched points for the chosen multi-image sets or at least for the first multi-image set.

7.5.2 Least Squares Matching

The parameters regarding the least squares matching are the same used in the automatic matching process of section 4.2.3:

- *used LSM parameters*: 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 x,y* : image patch sizes in pixels.
- *max sigma 0*: threshold of $\hat{\sigma}_0$ (see eq. 4.10) to accept the matching result.
- *max sigma x,y* : thresholds of $\hat{\sigma}_{a_0}$ and $\hat{\sigma}_{b_0}$ (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 (resulting parameters a_0 and b_0).
- *epipolar constrain*: on/off.

As explained in section 4.2.3, some parameters have to be determined case by case. These are the thresholds for the quality control (*max sigma 0*, *max sigma x,y* , *max x,y*) and the image patch sizes for the matching process (*patch size x,y*). No rules or advises can be given regarding these parameters. They strongly depends on the texture in the image and on its format. Their ideal values have to be determined by manually testing LSM on the images. Some examples of parameter sets can be found in appendix C.

7.5.3 Tracking Process

The specific parameters for the tracking process are the following:

- *patch size x,y* : image patch sizes in pixels for the cross-correlation process used to find an approximation before applying LSM (step 2 of algorithm of section 7.1.1).
- *first search size x,y* : sizes of the search region in the images of the next time step at the first tracking frame (figure 7.3, at frame $i+1$ and *box=LARGE* in flowchart of figure 7.4).
- *search size x,y* : sizes of the search region in the images of the next time step (figure 7.3, at frame $i+2$ and *box=small* in flowchart of figure 7.4).
- *large search size x,y* : sizes of the search region in the images of the next time step at the second try of the tracking process, in case of an unsuccessful match (*box=large* in flowchart of figure 7.4).
- *search step x,y* : pixel steps along the search path in the search region. It is not necessary to perform the cross correlation at every pixels along the search path; moreover, the speed of the entire process can be reduced.
- *max diff*: threshold of percentage of differences of the *spatial* and *temporal* LSM to accept the tracking result.
- *tessel size x,y* : sizes of the grid elements in the image for the computation of the local density for the introduction of new points into the tracking process.

- *density threshold*: local density threshold in the grid elements to introduce new points in the tracking process. Depends on the *tessel size* and on the *pixel shift* of the surface matching process (e.g., if *tessel size*=10, *pixel shift*=2, then the total maximal number of points matched in one tessel is 25; the *density thresholds* can then be for example set as 20 which is equivalent to 80%).

Examples of parameter sets for the tracking process can be found in appendix C.

7.5.4 Filtering Process

7.5.4.1 Velocity and Acceleration Filter. The following parameters have to be set:

- *min frame*: minimal length of a trajectory (in frames). Shorter trajectories are removed. Usually, all the trajectories shorter than 3 frames should be removed.
- *max velocity*: threshold of maximal velocity allowed. Trajectories with larger velocities are truncated at that time.
- *max acceleration*: threshold of maximal acceleration allowed. Trajectories with larger accelerations are truncated at that time.

The two thresholds can usually be determined considering the average and the maximal values occurred in the unfiltered data.

7.5.4.2 Uniformity Filter. The following parameters have to be set:

- *voxel size*: size of cubic voxels [mm] used for dividing the object space into local regions.
- *min density*: minimal density in each voxel to be processed. If the number of trajectories in a voxel is too small, there is no sense to check the uniformity of the movement.
- *min velocity*: minimal actual velocity to check difference with the mean value. If the actual trajectory is stationary there is no sense to check the differences with the mean value.
- *max difference magnitude*: threshold for the maximal percentage of the difference of the magnitudes of the velocity vector and the local mean value. The trajectories exceeding the thresholds are truncated at that time.
- *max angle*: threshold for the maximal angle between the actual velocity vector and the local mean velocity vector [deg]. The trajectories with larger values are truncated at that time.

7.6 CONSIDERATIONS

The presented surface tracking process is flexible and can be used under different conditions for different applications. In the next section it is applied to capture the motion of humans without using markers (D'Apuzzo, 2001a). The process of tracking was also applied successfully in two dimensional mode for tracking facial expressions (a short description of the results is given by D'Apuzzo et al. (2000) or in section 7.8)

and for special applications (D'Apuzzo and Willneff, 2001).

The discussion of the results achieved by the tracking process has only qualitative nature in this work. A quantitative analysis was not possible because reference data for the comparison of the results was missing. For future studies of this work, it is strongly recommended to acquire test sequences both with the multi-image acquisition system and with a precise optical motion capture system available commercially (see section 1.4). It would be of great interest to compare the accurate motion capture measurements with the results achieved by the method proposed in this work as 3-D trajectories of key-points.

A second limitation of the description of the results achieved by the proposed method is the very small number of tested datasets. In fact, only two multi-image sequences were used in this work for testing purposes. Anyhow, it was possible to show the flexibility of the proposed method, i.e., the possibility to use different image acquisition systems (two, three cameras) recording the human body entirely or partially and resulting in different types of 3-D data.

7.7 APPLICATION: FULL BODY MOTION CAPTURE

In this section, the proposed tracking process is applied to extract, without using markers, full body 3-D motion information from multi-image video sequences acquired with synchronized CCD cameras. This work was part of a project aimed at developing an highly automated system to model most realistically human bodies from video sequences (Fua et al., 1998; D'Apuzzo and Plänkers, 1999; Plänkers et al., 1999; D'Apuzzo et al., 1999, 2000; D'Apuzzo, 2001a; Fua et al., 2002).

Two different image sequences are processed. Their characteristics and the achieved results are presented in the next sections. The two multi-image sequences are acquired with different camera systems. The former represent a person walking parallel to the camera system and will be called *walking* sequence. The latter represent a more complex movement where only the upper part of the body is acquired and will be called *baseball* sequence.

7.7.1 Image Acquisition System

Two different acquisition system are used for the two different sequences. For the *walking* sequence, two synchronized (interlaced) CCD cameras are used. A sequence of image pairs is acquired with a frame grabber and the images are stored with 768x576 pixels resolution at 8 bit quantization. Since the CCD cameras are interlaced (i.e., a full frame is split into two fields which are recorded and read out consecutively), only the odd lines of the images are processed, avoiding the *saw pattern* effect (see section 2.2.2). This results in the reduction of the image format in the vertical direction by 50 percent (i.e., the effective format of the images is 768x288 pixels).

Figure 7.17, shows some frames of the *walking* sequence. As can be seen in the figure, no markers are signaled on the person and no special conditions are required. The person wears indeed normal clothes and in the background even other persons are moving.

The *baseball* multi-image sequence is acquired by three synchronized progressive scan CCD cameras in a triangular arrangement. A sequence of triplets is stored with 640x480 pixels resolution at 8 bit quantization (bottom, left, right).



Fig. 7.17 Acquired walking sequence (8 of totally 55 processed frames).

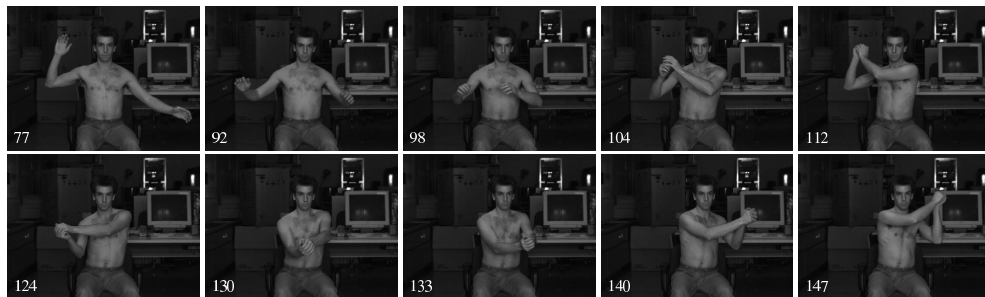


Fig. 7.18 Acquired baseball sequence (10 of totally 71 processed frames).

Figure 7.18 shows some frames of the sequence; the person performs a complex movement of the upper part of the body (resembling vaguely a *baseball* pitch). In order to increase the contrast, the images of this sequence are enhanced using Wallis filtering (Wallis, 1976). Figure 7.19 shows a frame of the sequence before (left) and after (right) Wallis filtering.



Fig. 7.19 A frame (112) of the acquired *baseball* sequence. Left: original image, right: after Wallis filter contrast enhancement.

7.7.2 Surface Measurement

The surface of the visible parts of the human body are measured every second time step of the sequence in the way explained in chapters 4 and 5. The amount of multi-image sets to be processed is very large, therefore, the fully automatic mode is used for the definition of the seed points (section 4.2.1). The manual intervention of the operator is reduced to the definition of the contour of the region to measure. This procedure

7 SURFACE TRACKING

could be skipped, measuring the entire images. However, the time required to process the image-sequence will increase.

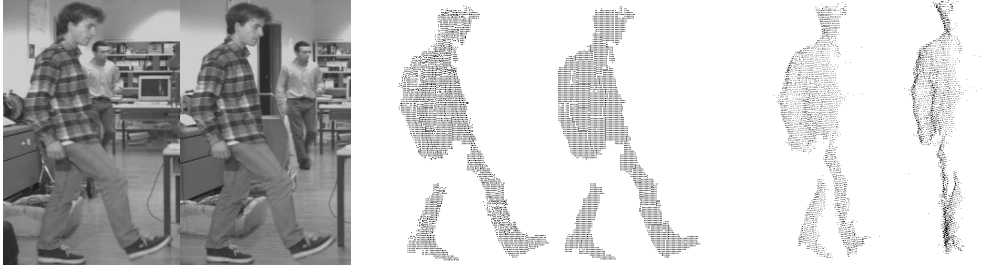


Fig. 7.20 Walking sequence. Left: image pair. Center: matched points. Right: measured 3-D point cloud.

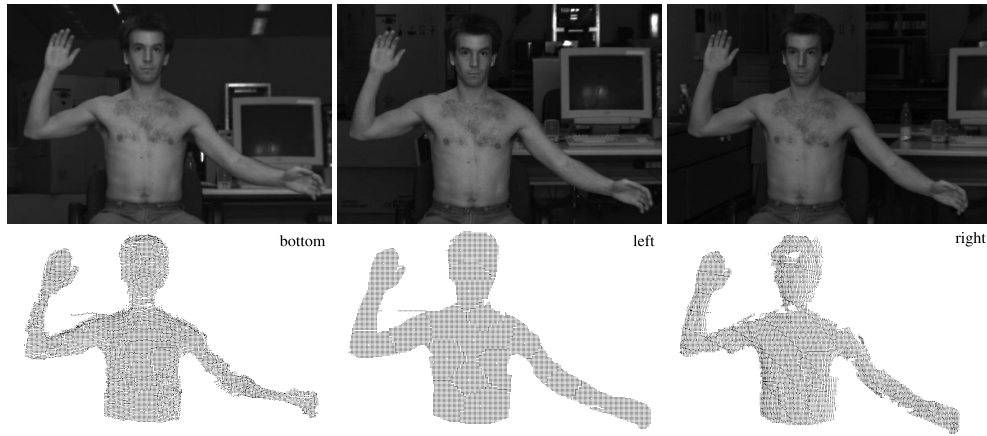


Fig. 7.21 Matching process on the *baseball* sequence. Top: image triplet. Bottom: matched points.

The result of the surface measurement process are 3-D point clouds each two time steps. This data will then be used in the successive surface tracking process. Figure 7.20 shows an example of the matching results achieved on an image pair of the *walking* sequence and the computed 3-D point cloud. Figure 7.21 shows a triplet of the *baseball* sequence with the matched points and figure 7.22 the computed 3-D point cloud.

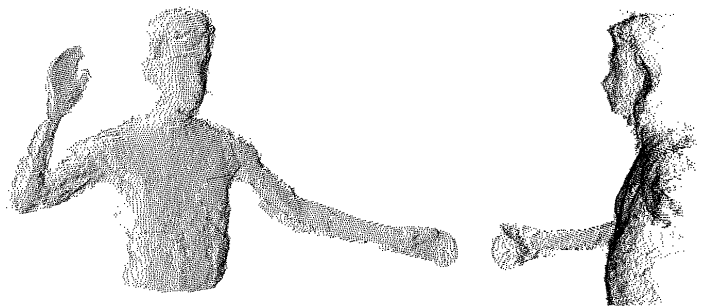


Fig. 7.22 Baseball sequence: Measured 3-D point cloud.

Table 7.1 summarizes some data regarding the two sequences and the matching results and table 7.2 give some hints about the achieved precision of the measurement. Since the images of the *walking* sequence are acquired by only two cameras, the theoretical precision of the computed 3-D point clouds cannot be derived.

Table 7.1 Data regarding sequences and surface matching. Image format [pixel], number of used images, number of processed frames, value of pixel shift for the matching process [pixel], average number of matched points.

Sequence	image format	images	frames	pixel shift	matched points
<i>Walking</i>	768x288	2	55	2	5804
<i>Baseball</i>	640x480	3	71	2	10052

Table 7.2 Theoretical precision (Helmert's point error according to eq. 6.1) of the measured point clouds (first frame of the sequence). Total number of matched points, average theoretical precision $\bar{\sigma}$ [mm], standard deviation [mm] and maximal value [mm]; same values ignoring points with $\sigma > 4.0$ mm; number of ignored points.

Sequence	points	$\bar{\sigma}$	<i>std.dev.</i>	max σ	$\bar{\sigma}_{<4.0}$	<i>std.dev.</i>	max σ	- pt
<i>Baseball</i>	12489	2.6	1.9	10.0	1.9	1.0	4.0	2387

The results achieved on full body are less accurate and more noisy than the results presented for the face measurement application (section 5.4). The reasons are the reduced resolution, because the entire human body has to be imaged and the texture, because the projection of an artificial texture is not allowed by the tracking process. The clothes worn by the person in the *walking* sequence make the surface measurement task even more difficult. However, the goals of the entire process have in this case more qualitative than quantitative nature. Only the approximative extraction of 3-D movements performed by the person is required.

7.7.3 Tracking Process

The completely automatic surface tracking process applied to the *walking* sequence provides totally about 32'000 trajectories for 55 frames. The mean number of trajectories per time step is about 6'000 and the average length of the trajectories is 10 frames. Due to the poor natural texture (mostly on the legs) and self-occlusion the surface tracking process does not produce a dense set of long trajectories. Figure 7.23 shows the tracked trajectories (displayed as displacement to the previous frame) for the frame number 39.

The results of the surface tracking process applied to the *baseball* sequence are about 120'000 trajectories during 71 frames, with an average length of 9 frames and an average of 10'000 trajectories per frame. The larger format of the images of this sequence compared to the previous, is the reason of the larger number of trajectories per frame. The more complex movement and the poorer natural texture are the main factors for the shorter average length of the computed trajectories. Figure 7.24 shows the tracked trajectories for the frame number 115.

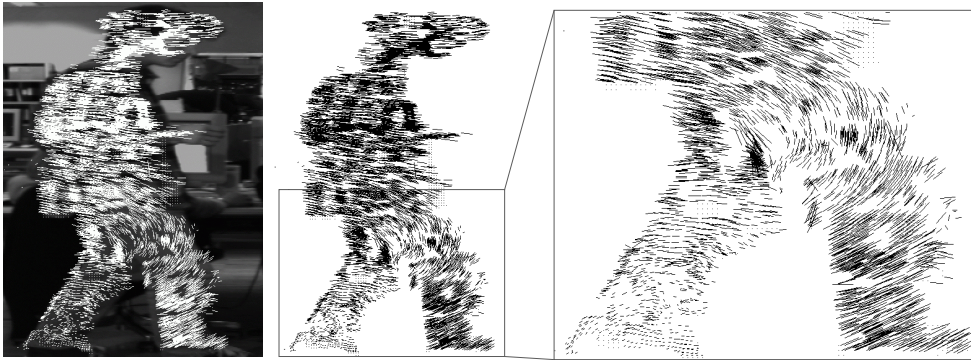


Fig. 7.23 Walking sequence. Surface tracking results (frame 39).

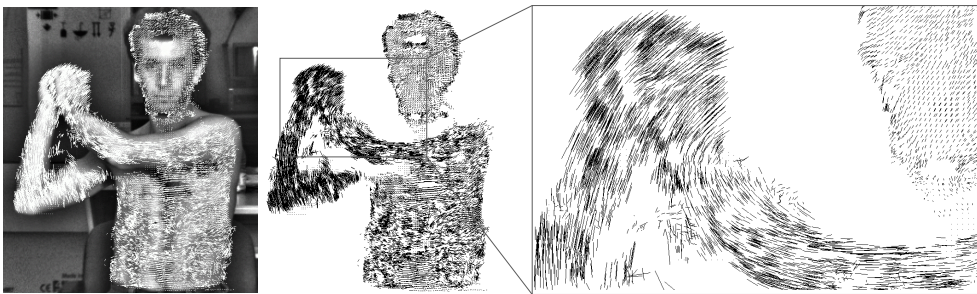


Fig. 7.24 Baseball sequence. Surface tracking results (frame 115).

Table 7.3 summarizes some data regarding the results of the tracking process on the two sequences.

Table 7.3 Data regarding tracking results. Number of processed frames, total number of trajectories, average number of trajectories per frame, average length of the trajectories [frames], total number of trajectories after filtering and average length [frames] after filtering.

Sequence	frames	trajectories	per frame	length	after filtering	length
<i>Walking</i>	55	32008	5804	9.8	22908	12.9
<i>Baseball</i>	71	121292	10052	8.9	70063	14.7

7.7.4 Key-point Tracking

The process of key-points tracking (see section 7.3) is required to extract general 3-D information of the performed movements of both sequences. The key-points are therefore placed in such a way that their trajectories can describe the walking process for the *walking* sequence and the complex upper body movement performed in the *baseball* sequence. For the *walking* sequence, a small set of key-points is placed near the joints: feet, knees, hips, hands, elbows, shoulders, neck, head and bust (see figure 7.25).

For the *baseball* sequence, a small set of key-points is placed near the joints: hands, elbows, shoulders, sides of neck; three are placed on the bottom of the upper body part; three on the face (nose and eyes) and four on the head (see figure 7.26).

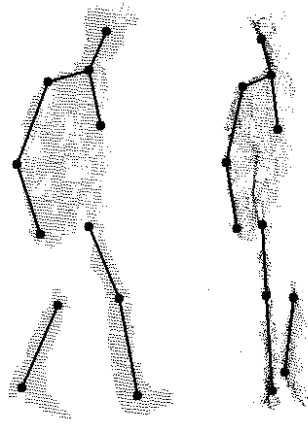


Fig. 7.25 Walking sequence. Key-points tracked on the human body.

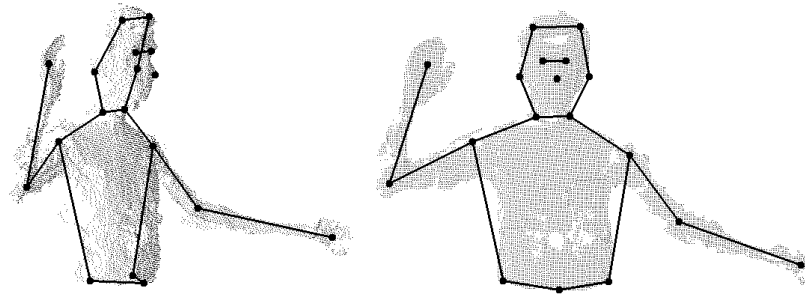


Fig. 7.26 Baseball sequence. Tracked key-points.

The final result of the tracking process for the *walking* sequence is shown in figures 7.27 and 7.28. The size of the region defining the key-points is chosen small enough to assume them lying on the surface of the human body. The results achieved by the tracking process have therefore the same nature of the results of a conventional motion capture system, i.e., trajectories of key points on the body, which can then be used to reconstruct the motion performed by the person. Offset values of the key-points on the body could be defined, to get the joints of the human body.

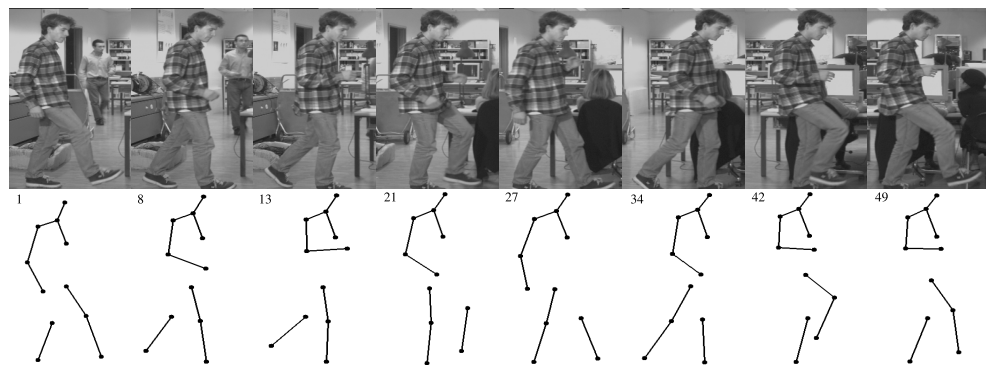


Fig. 7.27 Tracking key-points: some frames of the *walking* sequence and frontal view of the position of the tracked key-points.

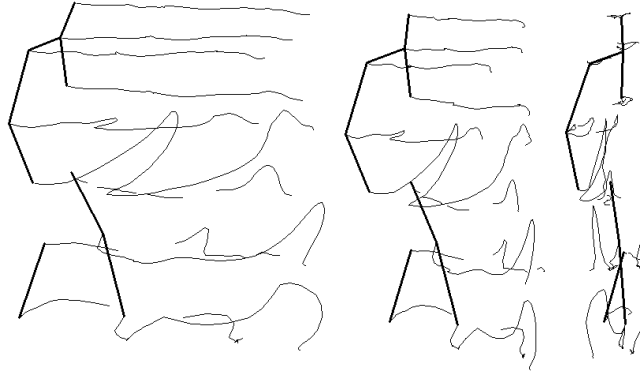


Fig. 7.28 Tracking key-points: Three views (frontal, 1/2 and lateral) of the 3-D trajectories of the tracked key-points in the *walking* sequence.

Because of the complexity of the performed movement of the *baseball* sequence, the size of the region defining the key-point is chosen in this case large enough to guarantee robustness. The key-points represent here an approximation for the joint trajectories. The final result of the tracking process is shown in figure 7.29 together with the corresponding frames. Two views of the 3-D trajectories of the key-points are displayed in figure 7.30 (note that the connections between key-points are intended only for a better understanding of the movement).

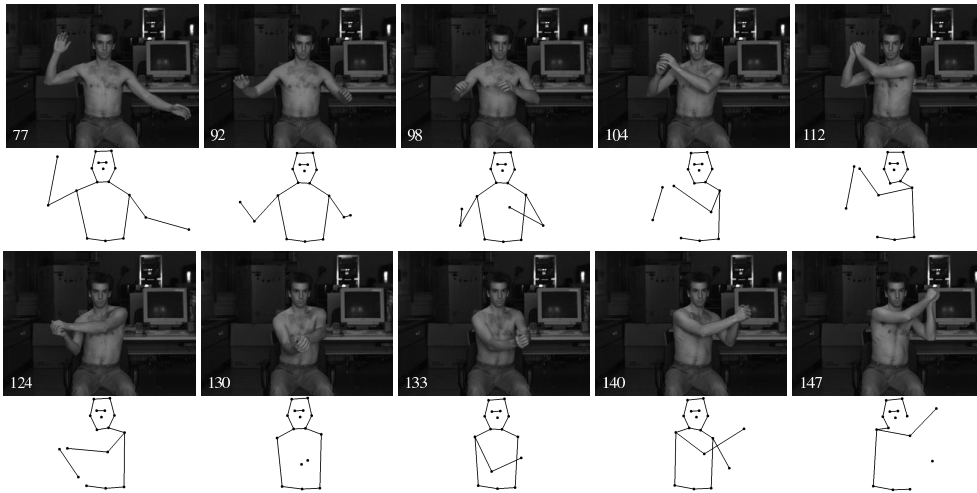


Fig. 7.29 Tracking key-points: some frames of the *baseball* sequence and frontal view of the position of the tracked key-points.

As can be seen, e.g., in the frame number 130 of figure 7.29, some key-points cannot be tracked during the entire sequence because of strong occlusions caused by the complex movement. In this case, the lost key-points have to be defined again in the following time steps. However, the new defined key-points will not be placed exactly at the same position. Anyhow, as explained above, the 3-D trajectories of the key-points serve in this case as an approximative description of the performed movement (as shown in figure 7.30) and therefore the results are acceptable.

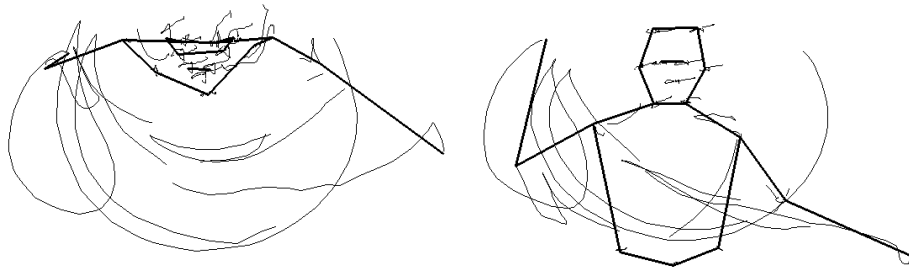


Fig. 7.30 Tracking key-points: view from the top (left) and from the front (right) of the 3-D trajectories of the tracked key-points in the *baseball* sequence.

7.8 APPLICATION: TRACKING IN 2-D

The proposed tracking process is a flexible tool and can also be used in 2-D mode. In this case, the sequence of a single camera, e.g., a camcorder, is processed. The use of a single image sequence can obviously not produce 3-D data but for some cases 3-D information is not required (D'Apuzzo and Willneff, 2001).

The Figure 7.31 shows a simple example of tracking facial expressions, where some points (white crosses on the first image) are tracked through the sequence acquired by a video camcorder. The single point tracking algorithm of section 7.1 was applied using as input the images of a single camera. The tracking was performed in this case only in the image space.

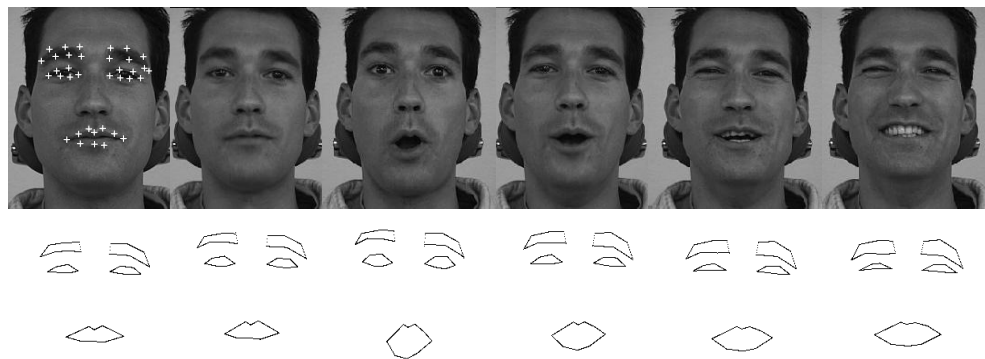


Fig. 7.31 Tracking in 2-D. Some frames of a single camera image sequence (the crosses on the first frame show the tracked points). Bottom: basic animation created by joining the tracked points with lines.

This example underlines the flexibility of the proposed tracking process, which can produce in this case a simple animation by tracking points on the face without using markers.

Conclusions

8.1 SUMMARY OF THE RESULTS

A system and the methods for automated measurement and tracking of the surface of human body parts from multi-image sequences has been presented. 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 gained 3-D data can be of two different types: surface measurement of the interested part in 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 (position, velocity and acceleration).

The entire process is composed of seven steps (four, in case of only 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 multi-image sets and/or multi-image sequences;
3. matching process: establishment of correspondences in the multi-images (at least in the first set);
4. surface measurement: computation of the 3-D point cloud for each matched multi-image set;
5. surface tracking in the multi-image sequences;
6. establishment of a 3-D vectorfield of trajectories (position, velocity, acceleration);
7. tracking key-points in the vectorfield of trajectories.

To demonstrate the multiple functionality of the method, three different applications were 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.

8.1.1 Data Acquisition and Calibration

The accurate measurement of a human body part starts with an adequate acquisition of the required data, i.e., multi-images (multiple images acquired from different positions in the space describing the same scene) in case of static surface measurement

8 CONCLUSIONS

or multi-image sequences (multi-images acquired during a time interval) in case of dynamic surface measurement or surface tracking. In the latter case, the multi-images have to be obviously acquired simultaneously. Nevertheless, also in case of static surface measurement of human body parts, it is recommended to synchronize precisely the multiple cameras because the human body cannot be considered as a static object; in fact, a person always moves slightly unconsciously (e.g., breathing, muscle contractions). The precision of the synchronization of the multiple imaging devices plays therefore an essential role for the accuracy potential of the measurement.

Different systems can be used to acquire multi-image sequences with different levels of quality depending on the used cameras. With decreasing accuracy potential, these are: multiple synchronized machine vision progressive scan CCD cameras, multiple synchronized machine vision interlaced CCD cameras or multiple digital video camcorders synchronized using a clapper. For surface measurement purposes without tracking, only multi-images are required. In this case also the following systems can be used for the acquisition (with decreasing accuracy potential): multiple digital still cameras, a single digital still camera acquiring in different position or multiple digital web cameras.

A precise synchronization of the multiple imaging units is possible only by the two machine vision acquisition systems. Systems based on multiple digital still cameras and systems using digital web cameras can be set up to acquire multi-images simultaneously. However, the cameras cannot be electronically synchronized to each other; the maximum time delay which can occur between the acquisition of the different images is therefore half a frame (i.e., 0.04 seconds). The same synchronization accuracy can be achieved with systems using multiple video camcorders, aligning the audio signals of the different sequences. The systems based on a single camera can evidently not acquire simultaneously multi-images, in some cases the larger time delay can however be acceptable (e.g., when high accuracy of the measurements is not required and if the human body part can be immobilized).

The lens and optical system of the camera also play a role for the quality of the acquired images. Machine vision lenses are preferable because of the smaller distortions. However, consumer market products (e.g., video camcorders, digital still cameras) can also be used for measurement purposes without a great loss of accuracy (D'Apuzzo and Maas, 1999), correcting the larger distortions with a thorough calibration procedure. The tasks of the orientation and calibration processes are the establishment of the position and orientation of the camera sensors in 3-D space as well as the parameters describing the internal geometry of the imaging device and some parameters modeling the distortion caused by the optical system. A thorough determination of all the parameters is a requirement for the accurate measurement using photogrammetric techniques. To calibrate the multi-image camera system, various methods can be used. For simplicity, only two methods were applied in this work: the reference bar method and the reference field method. In the first case, a reference bar with two retroreflective target points is moved through the object space while the camera system acquires multi-image sets. The multiple camera system can then be calibrated by self calibrating bundle adjustment with the only required information of the known distance between the two points on the bar and their automatically measured image coordinates along the sequence. 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 reference field calibration methods use instead a 3-D field

with signalized points whose coordinates in space are known. The bundle calibration method is in this case used and full automation can be achieved using coded target points. A graphical user interface for the latter calibration procedure (code recognition and calibration) was implemented to facilitate the user and to reduce dramatically the time spent on this task. In fact, the most time-expensive problem that has to be solved for the implemented calibration method is the determination of the approximative external orientation; it is especially difficult to assume the correct values for the direction angles of the camera. An effective and fast solution to solve this problem is implemented in a smart way in the graphical user interface: the external orientation of the camera is indeed modified by the user interactively translating, rotating and moving in and out the backprojected 3-D coordinate of the control points (and not the camera itself), without taking care about the external orientation of the camera (position and direction). Using the implemented graphical user interface, the time required to calibrate an image decreases to a couple of minutes (without having any information about the external orientation of the camera). The proposed solution is intuitive and can be used without any difficulties by non-expert user too (Schneeberger et al., 2002).

8.1.2 Matching Process and Surface Measurement

The goal of the implemented automatic matching process is the determination of a dense set of corresponding points in the multi-images covering the entire interested surface part. Moreover, the defined matching strategy was developed especially to reduce to a minimum the required processing time.

The automatic matching process is based on a stereo matcher which determines the position of corresponding points in an image pair. In case of multi-images, the stereo matcher is applied independently to image pairs using the same template point in a unique template image. The results are correspondences in multi-images. Evidently this method is less robust than a complete multi-image matching process taking into account all the images in a single step (Gruen and Baltsavias, 1988; Baltsavias, 1991). Anyhow, the results achieved by the simpler implemented multi-image matching process were enough accurate and no further time was spent for the investigation and implementation of more complex and accurate matching algorithms.

The stereo matcher is built on the adaptive least squares method (Gruen, 1985) with the additional geometrical constraint of the matched point to lie on the epipolar line. Different methods could be used to integrate the epipolar geometry in the least squares matching process. Since for this work, the processing time has to be limited to a minimum, the geometrical constraint is expressed as a single additional observation describing the fact that the position in the search image has to move along the epipolar line.

The quality of the single matching result is evaluated using different indicators as the resulted a posteriori standard deviation of the least squares adjustment, the resulting standard deviation in x and y directions, the displacement from the start position in x and y direction and the distance to the epipolar line.

The developed strategy for the automatic matching process consists of the definition of few seed points and, starting from them, the establishment of a dense and robust set of corresponding points covering the entire interested region. The strategy is entirely developed regarding to the smooth characteristic of the surface of the human body.

8 CONCLUSIONS

However, the surface can present some discontinuities that can be treated separating the different continuous regions by placing the seed points in the correct positions. The same applies in case of overlapping parts.

The seed points can be generated in three different modes: fully automatically, semi-automatically defining them only in one image or defined manually in each image. For static surface measurement using images with good texture, the semi-automatic mode is the preferable and faster than the others. On the other hand, the full automatic mode is adequate for the processing of long sequences, because of the large number of multi-image sets. In special cases where the automatic modes could fail as, e.g. with images with a fine repeated pattern, the time-expensive manual mode has to be used.

Once the seed points are defined, the image is divided into polygonal regions according to which of the seed points is closest. Starting from the seed points, the automatic matching process produces then a dense set of corresponding points in each polygonal region by sequential horizontal and vertical shifts. The time required to cover the entire image depends strongly on the distribution, the location and the number of the defined seed points. The processing time decreases by incrementing the number of seed points. However, the definition of a larger number of seed points requires more time. An ideal compromise has to be chosen.

A very simple method to reduce additionally the processing time is the definition of a region of interest in the template image. It can be given as a sequence of points or defined interactively in the implemented graphical user interface.

Since the automatic matching process has to be used for different purposes (e.g., matching corresponding points in multi-images, target point measurement, tracking process), various options and parameters have to be set. A relevant option that can be chosen 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.

Problems that could not be solved in this work and that still need to be investigated (see suggestions for further work, section 8.2.2) reside in the difficulty to establish image correspondences in dark areas such as eyebrows or areas with lack of texture. The hairs and beard still pose unsolved problems too. The automatic matching process results in fact in these areas very noisy.

A graphical user interface for the automatic matching process was implemented to ease the user and to decrease the total time required to perform the measurement of human body parts. All the required processes can be performed in the GUI: the definition of seed points can be achieved in the different modes (manual, semi-automated, full semi-automated, full automatic) and a ROI can be selected manually as a contour. Additionally, the automatic matching process can be visualized in real-time, allowing an analysis of the implemented search strategy. It also helps the user to get an idea of where the more problematic regions for the matching process are, gaining know-how about where to put the seed points and what their ideal number is. The regularization of grid, the neighborhood filter and the computation of the 3-D coordinates of the matched points are implemented as post-processing procedures.

An important tool is also the implemented 3-D point cloud viewer/editor. Large amounts of points can be displayed with color information (greyscale or RGB) and full navigation (translation, rotation, zoom) is possible. Additionally, stereo visualisation (with red/green glasses) and some basic editing functions are also implemented. Single points can be selected to get information about the point number, the 3-D coordinates and, if available, the color information (as RGB values). Multiple points can be selected in rectangular areas or in a (clockwise) hand drawn region. The selected points can then be deleted or stored as separate data set. This offer a simple and very fast opportunity to remove outliers or unuseful points from the data (e.g., background points or points not interesting for the modeling purposes).

Basic modeling procedures were implemented only to achieve a better visualisation of the surface measurement results. Therefore, detailed modeling aspects are not analyzed nor described in this work. For better qualitative modeling results, commercial software can be used.

A triangulated surface is generated from the 3-D point cloud by 2 1/2-D Delauney triangulation method (IGP software *DTMZ*). The disadvantage of this method is the impossibility to process fully 3-D data. For this reason, only landscape-like surface models can be generated (e.g., face masks). For texture mapping, basic modeling tools were implemented for the generation of VRML files with texture image coordinates for each vertex of the model.

The presented surface measurement process is flexible and can be applied with different acquisition setups and for different purposes. In fact, it was successfully employed for face modeling using images acquired by five CCD cameras (D'Apuzzo, 2002b), for human body modeling using three CCD cameras acquiring video sequences (D'Apuzzo et al., 2000), 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). Moreover, the surface measurement procedure constitutes also part of the surface tracking process. In this case, the surface measurement process can be performed for each multi-image of the acquired sequence.

The advantage of the presented method over the extensively employed methods to measure the surface of the human body such as laser scanning and coded light based triangulation approaches, is the acquisition of the source data in a fraction of second, allowing a surface measurement with higher accuracy and even the possibility to measure dynamic events like movements or changes of the surface.

8.1.3 Tracking Process

The main idea is to track corresponding points in the multi-images through the sequence and compute their 3-D trajectories. The results of the tracking process are the coordinates of a point in the multi-images through the sequence, thus the 3-D trajectory is determined by computing the 3-D coordinates of the point at each time step by forward ray intersection. Velocities and accelerations are also computed. The process is based on least squares matching techniques: the spatial correspondences between the images of the different views and the temporal correspondences between subsequent frames are determined using the least squares matching algorithm. The same problems and possible improvements for the matching process described in sections 8.1.2 and 8.2.2 apply also for the implemented tracking process. Additionally,

8 CONCLUSIONS

the impossibility to project an artificial texture onto the surface to track, increases the noise in the set of matched and tracked points. The reason is the insufficient texture of the human skin.

The proposed process can be used to track well defined points on the human body surface. Trajectories of single points are however not sufficient to understand and record the motion and movement of a human or the changes of the surface of human body parts. For this reason the tracking process was extended to track simultaneously a dense set of points belonging to a common surface. In this case, the result of the tracking process can be considered as a vector field of trajectories. The process of tracking entire surface parts and not only good signalized single points, can however produce false trajectories especially in case of poor texture of the surface to be measured. Nevertheless, these can easily be recognized because they do not follow the common movement of the majority. Filters for checking the local uniformity of the movement in the vector field of trajectories were therefore implemented.

To solve additional problems caused by occlusions, lack of texture, loss of tracked points and appearance of new points, the concept of key-points was introduced. The key-points are 3-D regions defined in the vectorfield of trajectories, whose size can vary and whose position is defined by its center of gravity. They are interactively defined in a graphical user interface and tracked in a simple way: the position in the next time step is computed as the mean value of the displacements of all the trajectories contained in side the 3-D region. The size of the region defining the key-point has an important role: if a small size is chosen, the key-point can be considered with a good approximation to lie on the tracked surface, representing in this case fictitious markers on the human body. If, on the other hand, larger sizes are chosen, the key-point represents a more approximative position of, e.g., joints of the human body.

Graphical user interfaces were developed and implemented for the dynamic visualization of the tracking results, two-dimensionally together with the image sequences as well as three-dimensionally displaying the computed 3-D trajectories in different view modes. All the functionalities for the definition, tracking and visualization of key-points are implemented in the latter graphical user interface.

The presented surface tracking process is flexible and can be used under different conditions for different applications. The work was part of a project aimed at developing an highly automated system to model most realistically human bodies from video sequences (Fua et al., 1998; D'Apuzzo and Plänkers, 1999; Plänkers et al., 1999; D'Apuzzo et al., 1999, 2000; Fua et al., 2002). The process of tracking was also applied successfully in two dimensional mode for tracking of facial expressions (brief description in section 7.8) and for special applications (D'Apuzzo and Willneff, 2001).

8.2 SUGGESTIONS FOR FURTHER RESEARCH

8.2.1 Data Acquisition and Calibration

Concerning the image acquisition systems, further research can be directed toward two different interesting goals: high quality or low cost. For an high quality acquisition system the number of cameras has to be extended, allowing a more complete imaging of the interested human body part (from the front, sideways and from the back). Machine vision progressive scan cameras are the best choice for multi-image

acquisition systems. The larger amount of data (in case of six, nine or even twelve cameras) has to be considered. Probably multiple frame grabbers or multiple acquisition systems have to be used and synchronized together to solve the task.

In the other direction, for demonstration and educational purposes, it would be very interesting to develop a (very) cheap multi-image acquisition system using, e.g., web cameras connected to a portable PC. A precise synchronization of the cameras is not possible and therefore high accuracy of the measurement cannot be achieved. Still, the great demonstration potential of a portable measurement system would be very attractive.

Concerning the two calibration methods applied in this work (reference field and reference bar), full automation is still a field for further investigation and implementation. For the reference field calibration methods, it could be achieved using in a first step a reduced number of coded targets for the determination of approximate values for the external orientation and continue then with the full calibration procedure. For the reference bar method, the achievement of full automation is challenging. The automatic recognition and measurement of the target points on the bar in all the acquired images is not trivial in case of an extension of the number of used cameras. Additionally, an automation in the calibration procedure has also to be implemented, because the used software package (*SGAP* of IGP) is very user unfriendly and long time is actually spent for the preparation of the required input files. Moreover, more steps are required to achieve the final calibration of the camera system.

For the calibration and orientation of multi-image acquisition systems, other methods, such as the point field method or the AustralisTM approach, could be investigated.

8.2.2 Matching Process and Surface Measurement

The implementation of a fast multi-image matching process in an unique least squares process could be an interesting task for further research. The main problem to solve is the definition of a robust and fast method to remove the irrelevant images from the set of images used in the process. Since the cameras are usually placed concentric to the human body part, it is in fact probable that regions of the interested part are not imaged by all the cameras. A smart strategy has therefore to be defined to select the relevant images before entering the least squares procedure.

The use of generic full body models or generic surface models of the interested parts could be integrated in the matching strategy to limit in the images the regions to process. The implementation of an automatic silhouette extraction could also be useful for this task.

A second interesting aspect that could be further investigated is the integration of neighborhood conditions. In this work, a neighborhood filter was implemented to detect false matching results. Furthermore, it could be interesting to use the smooth characteristic of the human body surface to integrate the neighborhood in the least squares process allowing the matching of multiple neighboring points in an unique process (Gruen and Zhang, 2002), making the results more robust and increasing the success rate in the regions with lack of texture. Moreover, new techniques for evaluating the matching results have to be investigated.

It would also be very interesting to compare the results achieved by the proposed method with commercially available measurement systems like laser scanners and coded light approaches. Methods for the comparison of the different data have to

8 CONCLUSIONS

be investigated. Moreover, the acquisition of the required data for the three different surface measurement systems should be performed simultaneously.

8.2.3 Tracking Process

The possible improvements of the result of the tracking process mainly depend on the matching process. The same suggestions described in the previous section can therefore be applied also for the tracking process.

What could be additionally investigated is the implementation of the independently used least squares matching processes (*temporal LSM* and *spatial LSM*) in a unique least squares procedure. Moreover, smart strategies could be integrated in the process to adaptively change parameters of the least squares matching depending on the results achieved.

In case of full body tracking or tracking of articulations, to increase the level of automation of the entire process, an automatic extraction of the silhouette of the human body or the contour of human body parts has to be implemented. A student work (Nef, 2001) investigated methods that could be used to solve this problem and presented preliminary implementations. Lot of work has however still to be done in this field to solve problems as shadow and noise. A thorough illumination of the subject and laboratory conditions, such as a uniform background, could limit the shadow effects as well as the noise caused by other persons or moving objects present in the scene.

Additional cues that could be extracted from the vectorfield of trajectories resulting from the improved tracking process are the extraction of rotation axes and angles of the different articulations. The definition of the key-points could be extended for this purpose.

For the extraction of the more elaborated data, it is strongly recommended to increase the number of the used cameras (to six, nine or twelve), so that also very complex movements can be acquired from the front as well as sideways and from the back. This will pose additional problems for the tracking process which has also to be generalized for the use of a variable number of cameras.

For future studies of this work we also recommend a comparison of the results with reference data. Test sequences should be acquired simultaneously both with the multi-image acquisition system and with a precise optical motion capture system available commercially. It would be of great interest to compare the accurate motion capture measurements with the results achieved by the newly developed method.

Graphical User Interface

A.1 MOTIVATIONS

The development and the implementation of a graphical user interface for the surface measurement and surface tracking processes have different intentions:

- *user friendliness*: to make everything easier for the user;
- *preprocessing time*: to reduce the time required for the preprocessing of the data;
- *visualisation*: for visualisation purposes (matched points, point cloud, trajectories);
- *analysis*: to analyze the defined processes in real-time.

A.2 OPERATING SYSTEM AND IMPLEMENTED GUI

A.2.1 Operating System

All the software was developed under Unix/Linux environment, from the calibration procedure through the matching process and the 3-D point cloud editing, till the visualisation of the tracking results and the definition of the key points. The code was implemented in ANSI C for a general portability.

A user friendly graphical interface (*GUI*) was implemented using the GTK/GDK libraries. All the software can be compiled for Linux and Unix operating systems. Nowadays, GDK/GTK libraries exists also for Microsoft Windows operating system, it is therefore possible to compile the software also under this environment by performing some changes in the code.

A.2.2 GUI for the Calibration Process

A graphical user interface for the calibration procedures (code recognition and calibration) was implemented to facilitate the user and to reduce the time spent on these

A GRAPHICAL USER INTERFACE

tasks. Figure A.1 shows the implemented graphical user interface and the figure A.2 describes the functions of the different menus and tool buttons.

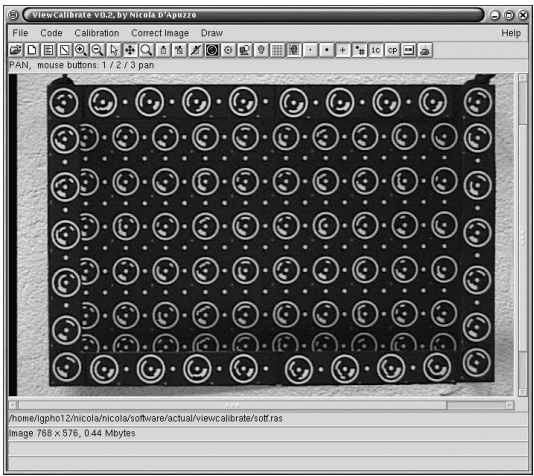


Fig. A.1 Graphical user interface of the code recognition and calibration processes.

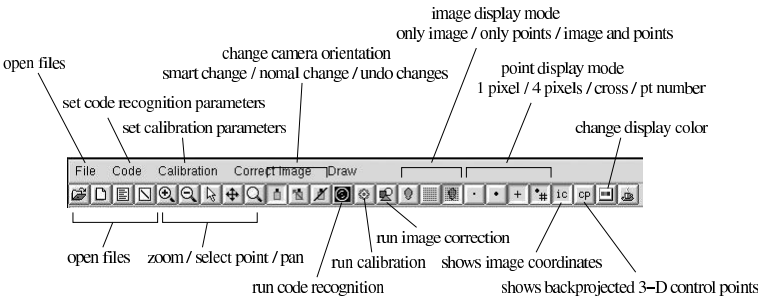


Fig. A.2 Menu line and tool buttons of the graphical user interface of the code recognition and calibration processes.

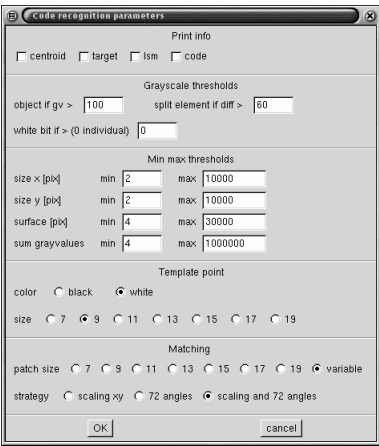


Fig. A.3 Parameters for the automatic recognition and measurement of the coded targets in the image.

A.2 OPERATING SYSTEM AND IMPLEMENTED GUI

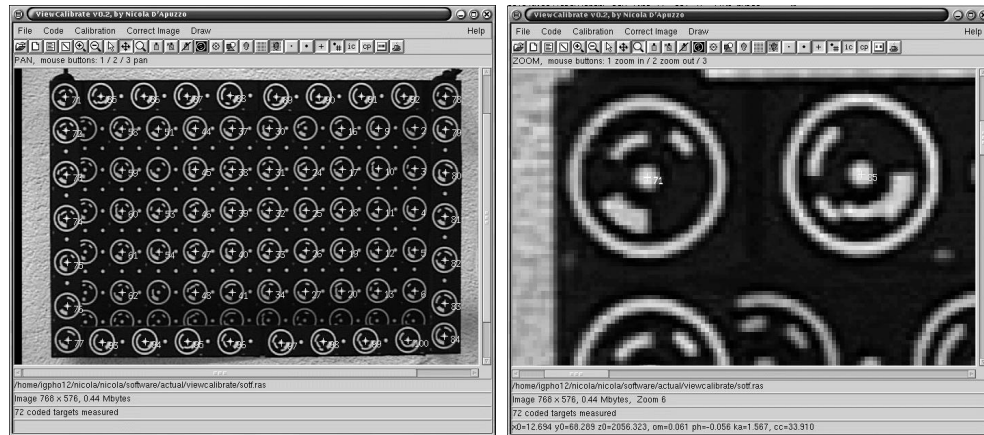
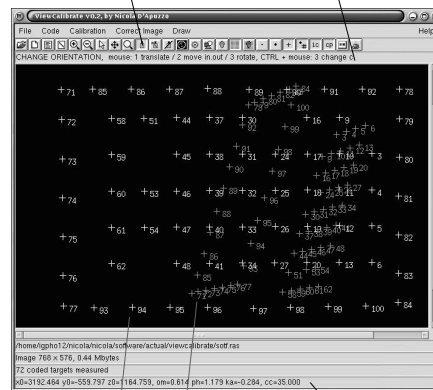


Fig. A.4 Result of the code recognition and target measurement processes.

The calibration process starts with the automatic recognition and measurement of the coded targets in the image (Niederoest, 1996). Figure A.3 shows the parameters to be set for the process and the figure A.4 shows an example. For the details about the process and the parameters, the reader is referred to the publication (Niederoest, 1996).

mouse controls: pan horizontally and vertically, move in and out, rotate 1 and 2, change c
smart change orientation



backprojected control points
measured image coordinates
actual external orientation of the camera

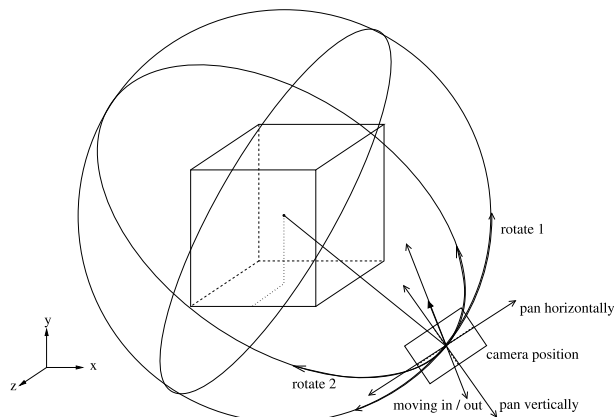


Fig. A.5 Smart change of the orientation of the camera.

A GRAPHICAL USER INTERFACE

The most time expensive problem that the user has to solve for the calibration process applied in this work, is the determination of the approximative external orientation of the camera: position of projection center (x_0, y_0, z_0) and direction $(\omega, \varphi, \kappa)$. What cause the biggest troubles, is the difficulty to assume the correct values for the direction angles. An effective and fast solution to find an approximation of the external orientation is implemented in a smart way in the graphical user interface. The external orientation of the camera is indeed changed by the user interactively by translating, rotating and moving in and out the backprojected 3-D coordinates of the control points (and not the camera itself) (see figure A.5), without taking care about the values of the position and direction of the camera (for the details about how the movement of the camera is computed, see section A.2.5).

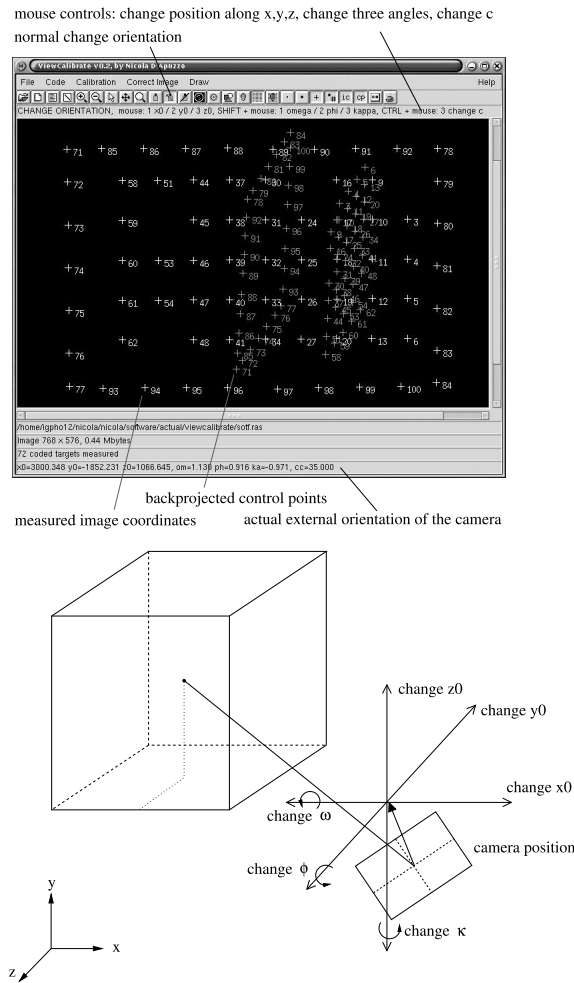


Fig. A.6 Direct change of the orientation of the camera.

If wished, the user can also change directly the position (x_0, y_0, z_0) and the direction $(\omega, \varphi, \kappa)$ of the camera (see figure A.6).

Before executing the calibration process of section 3.3, an approximation of the correct orientation of the camera has to be determined. This is the case when the control points backprojected onto the image have more or less the same disposition as the measured image coordinates (see figure A.7).

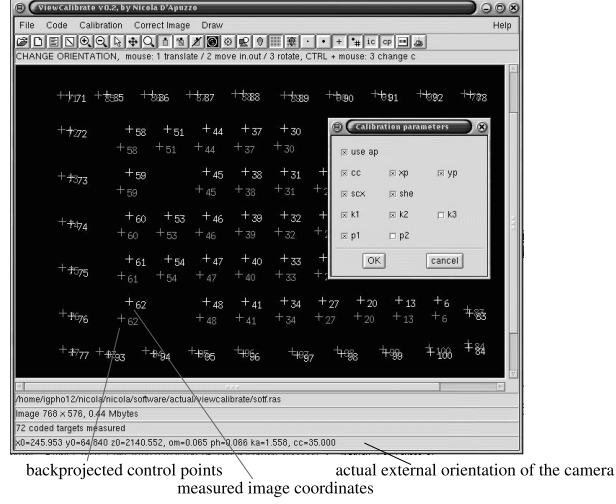


Fig. A.7 Starting the calibration process (the backprojected control points have more or less the same position as the measured image coordinates) and used parameters for the calibration.

Using the implemented graphical user interface, the process of calibrating an image takes a couple of minutes (even without having any information about the orientation of the camera).

The geometrical correction of the images is implemented as additional feature. An undistorted image can be generated by using the distortion parameters ($k_1, k_2, k_3, p_1, p_2, sc, sh$) determined during the calibration process. The flowchart of figure A.8 describes the process in detail.

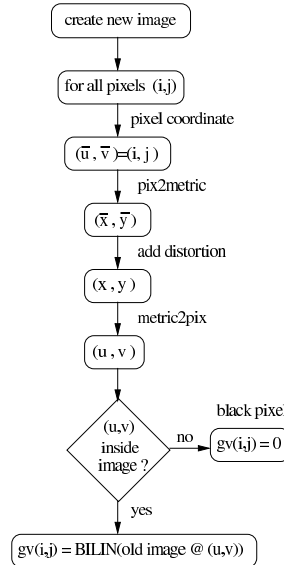


Fig. A.8 Flowchart: correcting the distortion of an image.

A new image with the same resolution is generated. Each pixel in the new undistorted image $((i, j) = (\bar{u}, \bar{v}))$ is firstly transformed from image coordinate to metric coordinates (\bar{x}, \bar{y}) (according to equation 3.6). The distortion terms are then added to the obtained metric image coordinates resulting in $(x, y) = (\bar{x}, \bar{y}) + (\bar{d}x, \bar{d}y)$ (according to equations 3.4 and 3.5). These are then transformed into image coordinates (u, v) (according to equation 3.7), which represent the distorted image coordinates. The gray value of the new image at the pixel (i, j) is therefore computed as bilinear resampling (according to equation 4.7) of the original image at the position (u, v) ; if (u, v) is outside the image limits, the new pixel is set black.

A.2.3 GUI for the Automatic Matching Process

The graphical user interface for the automatic matching process is the core program for the surface measurement process. All the processes described in section 4.2 can be performed in the GUI. As pre-processing, Wallis filter contrast enhancement can be applied on the images before starting the matching process. The definition of seed points can be performed in the different modes (manual, semi-automated and fully automatic) and a ROI can be selected manually as a contour.

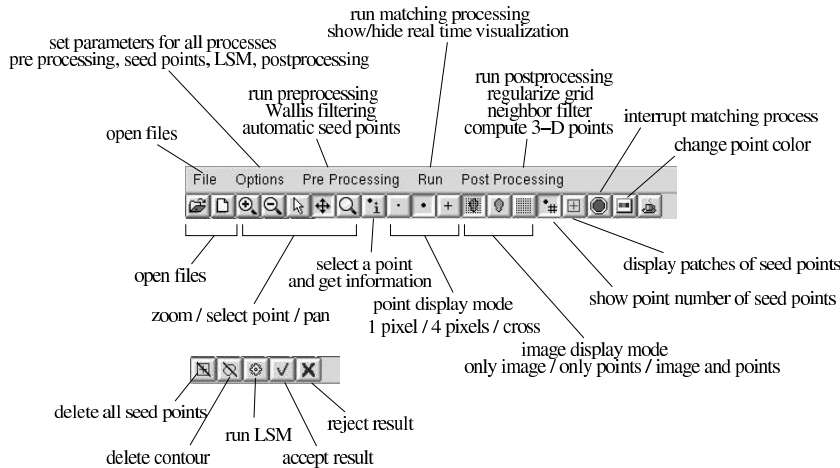


Fig. A.9 Menu line and tool buttons of the graphical user interface for the automatic matching process.

The automatic matching process can be visualized in real time. This functionality plays an essential role for the analysis of the implemented search strategy. It also helps the user to get an idea of where the more problematic regions for the matching process are, gaining know-how about where to put the seed points and what their ideal number is.

The regularization of grid (see section 4.3.1), the neighborhood filter (see section 4.3.2) and the computation of the 3-D coordinates of the matched points (see section 5.1) are implemented as post-processing procedures. Figure A.10 and figure A.11 show three moments of the processing: the definition of seed points, the real time visualization of the matching process and the end result with the computed 3-D point cloud. While the figure A.9 explains the functions of the menu line and the various tool buttons.

A.2 OPERATING SYSTEM AND IMPLEMENTED GUI



Fig. A.10 Graphical user interface of the matching process. Left: definition of seed point. Right: real time visualisation of the matching process.

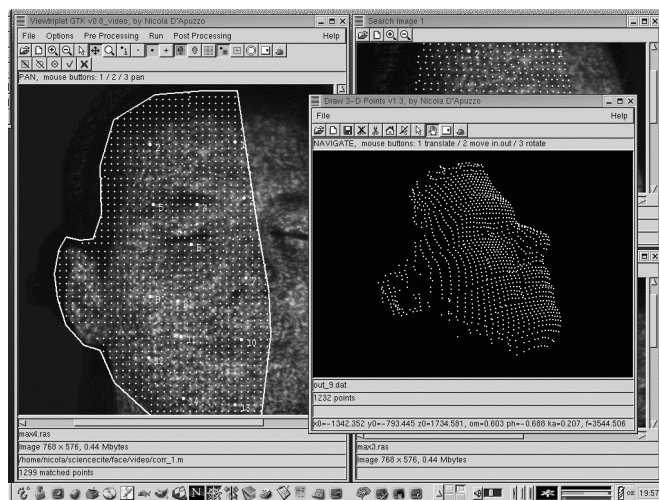


Fig. A.11 Graphical user interface of the matching process after finishing the matching and computed 3-D point cloud.

A.2.4 Matching Result Viewer

The graphical user interface of the automatic matching process provides also a visualization of the results. However, an additional tool was implemented to allow a more accurate analysis of the data. The set of matched points can be displayed alone or over the used images, zooming and selection functions are implemented. Figure A.12 shows an example while the figure A.13 explains the functions of the tool buttons.

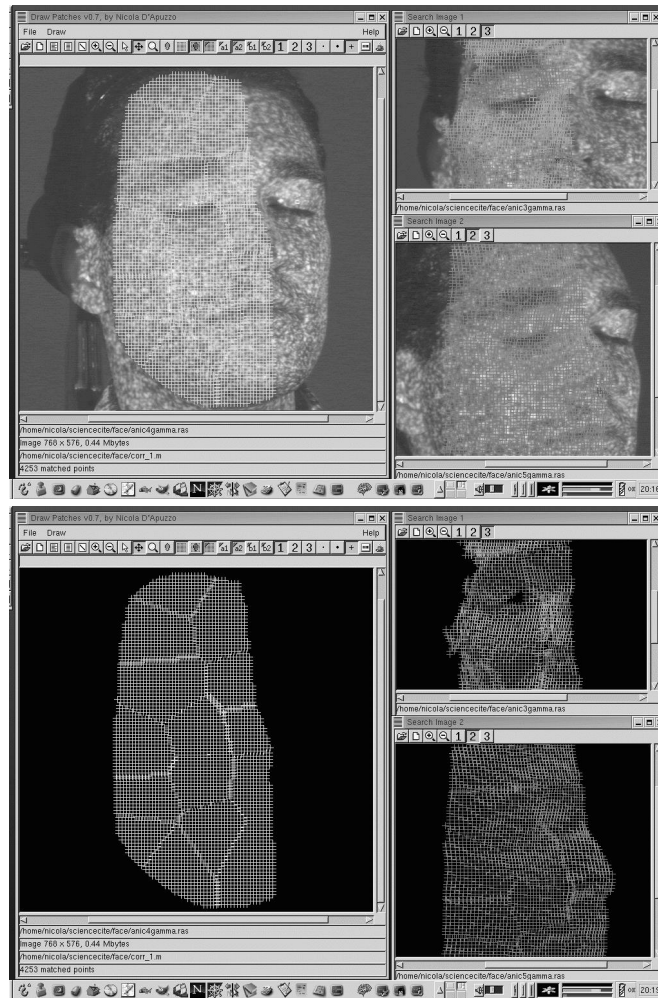


Fig. A.12 Matching result viewer.

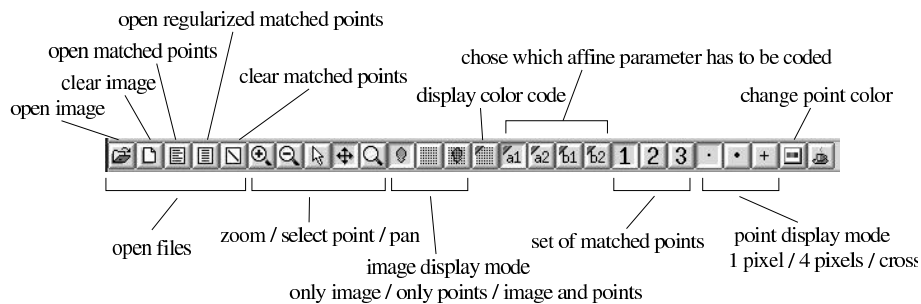


Fig. A.13 Tool buttons of the matching results viewer.

The additional feature of this tool is the possibility to visualize the matched points with different colors according to the resulting values of the parameters of the affine transformation (a_1, a_2, b_1, b_2) . This functionality is interesting to analyze the local uniformity of the transformed patches in the search images.

A.2.5 Point Cloud Viewer/Editor

The implemented 3-D point cloud viewer/editor is completely based on the collinearity equation (equation 5.1), i.e., the points displayed on the graphical user interface are the result of the backprojection of the 3-D point cloud onto a virtual image plane. The actual external orientation of the virtual camera as position X_0, Y_0, Z_0 and rotation angles ω, φ, κ and a very large camera constant f (to avoid large perspective effects) are displayed on the bottom text bar of the viewer (the figure A.14). The color information (greyscale or RGB) can also be displayed (figure A.14 right).

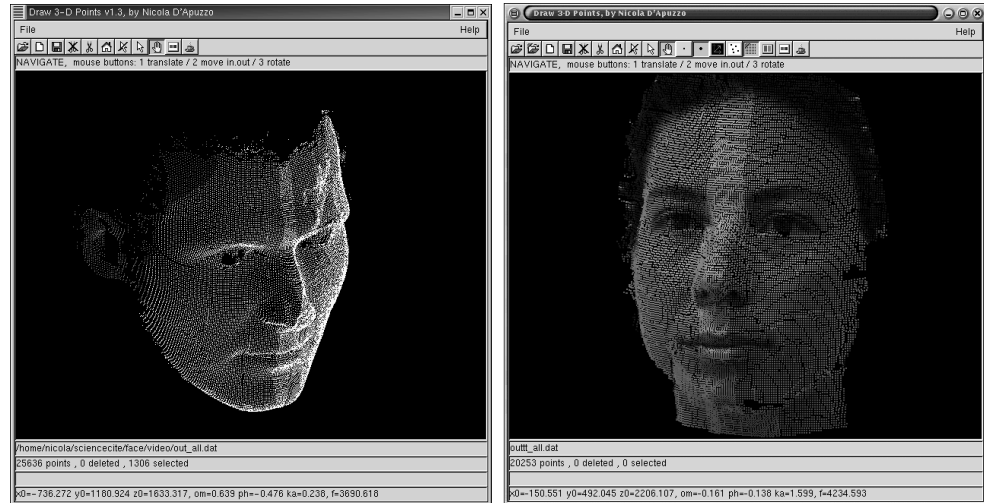


Fig. A.14 Graphical user interface of the 3-D point cloud viewer/editor. Left: simple view. Right: color information of the 3-D point cloud is also displayed.

The figure A.15 explains the functions of the tool buttons.

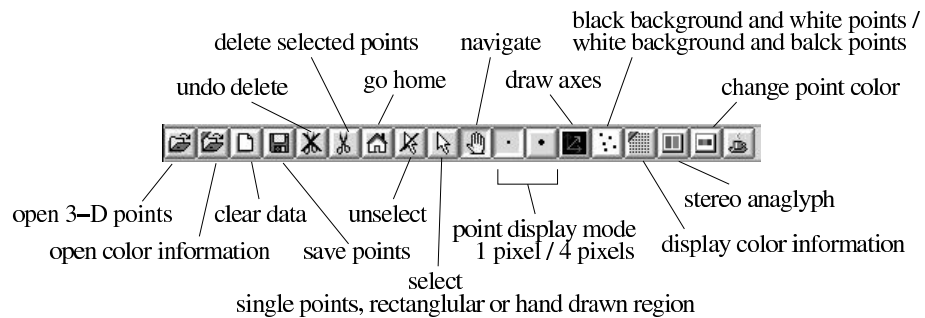


Fig. A.15 Toolbar of the 3-D point cloud viewer/editor.

Figure A.16 describes how the initial position of the virtual camera (that can be also recalled with the *go home* button of the toolbar) is determined. It is oriented along the z-axis (i.e., all rotation angles are equal to 0), pointing to the center of the 3-D point cloud. The distance of the virtual camera to the 3-D point cloud is chosen very large to

avoid strong perspective effects (a defined factor *PERSP* multiplied with the extension of the point cloud in *z* direction Δz); the camera constant f is then determined so that all the 3-D points can be displayed in the maximal size of the viewer window.

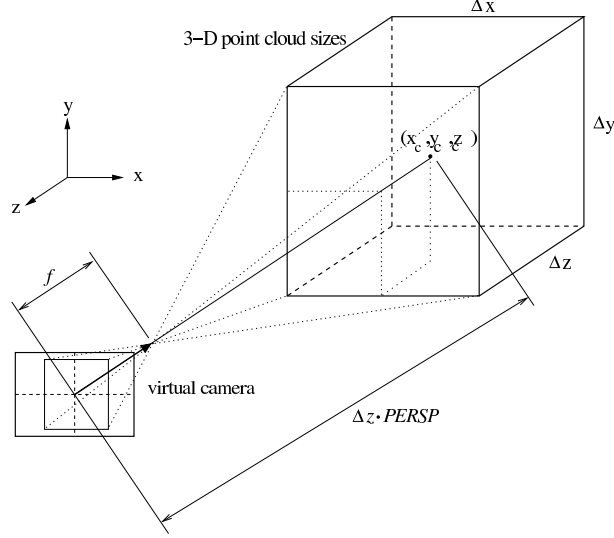


Fig. A.16 Definition of the initial position of the virtual camera for the 3-D point viewer.

The navigation possibilities are: pan in vertical and horizontal direction, zooming in and out and rotate around the center of the point cloud. The figure A.17 explains how the virtual camera position and orientation change, while the 3-D point cloud stays fix.

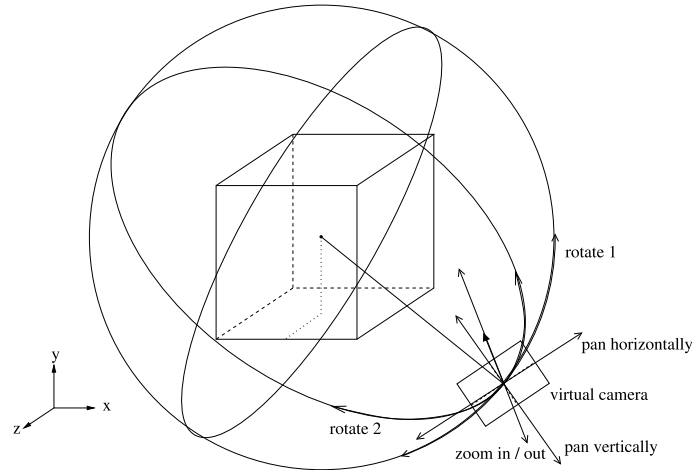


Fig. A.17 Navigation in the 3-D point viewer: zoom in and out, panning horizontally and vertically, rotate 1 and 2. The virtual camera changes position and orientation, while the 3-D point cloud stays fixed.

For **zooming in and out**, the direction of the virtual camera remains unchanged, while its position moves along the view direction according to:

$$\begin{aligned}\Delta X_0 &= r_{13} \cdot \Delta zoom \\ \Delta Y_0 &= r_{23} \cdot \Delta zoom \\ \Delta Z_0 &= r_{33} \cdot \Delta zoom\end{aligned}\tag{A.1}$$

where

$\Delta X_0, \Delta Y_0, \Delta Z_0$	position change of the projection center of the virtual camera,
r_{ij}	elements of the rotation matrix \mathbf{R} defining the orientation of the virtual camera,
$\Delta zoom$	displacement along the view direction of the virtual camera defined interactively by the user.

For **panning horizontally and/or vertically**, the direction of the virtual camera also remains unchanged and the position of the camera moves in the image plane according to:

$$\begin{aligned}\Delta X_0 &= r_{11} \cdot \Delta hor + r_{12} \cdot \Delta vert \\ \Delta Y_0 &= r_{21} \cdot \Delta hor + r_{22} \cdot \Delta vert \\ \Delta Z_0 &= r_{31} \cdot \Delta hor + r_{32} \cdot \Delta vert\end{aligned}\tag{A.2}$$

where

$\Delta X_0, \Delta Y_0, \Delta Z_0$	position change of the projection center of the virtual camera,
r_{ij}	elements of the rotation matrix \mathbf{R} defining the orientation of the virtual camera,
$\Delta hor, \Delta vert$	horizontal and vertical displacements in the image plane of the virtual camera defined interactively by the user.

For the **rotations 1 and 2**, the virtual camera rotates around the center of the point cloud in two directions, the horizontal and vertical directions of the starting location of the virtual image plane (see figure A.17). The position of the projection center and the direction of the virtual camera are computed according to:

$$\begin{aligned}\Delta X_0 &= 2 \cdot d \cdot (r_{11} \cdot \sin \frac{\alpha_1}{2} \cdot \cos \frac{\alpha_1}{2} - r_{13} \cdot \sin^2 \frac{\alpha_1}{2} - \\ &\quad - r_{12} \cdot \sin \frac{\alpha_2}{2} \cdot \cos \frac{\alpha_2}{2} - r_{13} \cdot \sin^2 \frac{\alpha_2}{2}) \\ \Delta Y_0 &= 2 \cdot d \cdot (r_{21} \cdot \sin \frac{\alpha_1}{2} \cdot \cos \frac{\alpha_1}{2} - r_{23} \cdot \sin^2 \frac{\alpha_1}{2} - \\ &\quad - r_{22} \cdot \sin \frac{\alpha_2}{2} \cdot \cos \frac{\alpha_2}{2} - r_{23} \cdot \sin^2 \frac{\alpha_2}{2}) \\ \Delta Z_0 &= 2 \cdot d \cdot (r_{31} \cdot \sin \frac{\alpha_1}{2} \cdot \cos \frac{\alpha_1}{2} - r_{33} \cdot \sin^2 \frac{\alpha_1}{2} - \\ &\quad - r_{32} \cdot \sin \frac{\alpha_2}{2} \cdot \cos \frac{\alpha_2}{2} - r_{33} \cdot \sin^2 \frac{\alpha_2}{2}) \\ r_{11}'' &= r_{11} \cdot \cos \alpha_1 - r_{13} \cdot \sin \alpha_1 \\ r_{12}'' &= r_{12} \cdot \cos \alpha_2 + r_{13}' \cdot \sin \alpha_2 \\ r_{13}'' &= -r_{12} \cdot \sin \alpha_2 + r_{13}' \cdot \cos \alpha_2 \\ r_{21}'' &= r_{21} \cdot \cos \alpha_1 - r_{23} \cdot \sin \alpha_1 \\ r_{22}'' &= r_{22} \cdot \cos \alpha_2 + r_{23}' \cdot \sin \alpha_2 \\ r_{23}'' &= -r_{22} \cdot \sin \alpha_2 + r_{23}' \cdot \cos \alpha_2 \\ r_{31}'' &= r_{31} \cdot \cos \alpha_1 - r_{33} \cdot \sin \alpha_1 \\ r_{32}'' &= r_{32} \cdot \sin \alpha_2 + r_{33}' \cdot \cos \alpha_2 \\ r_{33}'' &= -r_{32} \cdot \sin \alpha_2 + r_{33}' \cdot \sin \alpha_2 \\ \text{with:} \\ r_{13}' &= r_{11} \cdot \sin \alpha_1 + r_{13} \cdot \cos \alpha_1 \\ r_{23}' &= r_{21} \cdot \sin \alpha_1 + r_{23} \cdot \cos \alpha_1 \\ r_{33}' &= r_{31} \cdot \sin \alpha_1 + r_{33} \cdot \cos \alpha_1\end{aligned}\tag{A.3}$$

where

$\Delta X_0, \Delta Y_0, \Delta Z_0$	position change of the projection center of the virtual camera,
r_{ij}	elements of the rotation matrix \mathbf{R} defining the orientation of the virtual camera before the rotation,
r_{ij}''	elements of the rotation matrix \mathbf{R} defining the orientation of the virtual camera after the rotation,
d	distance of the projection center of the virtual camera to the center of the 3-D point cloud,
α_1, α_2	rotations angles in horizontal and vertical directions defined interactively by the user.

Stereo visualisation (with red/green glasses) is also implemented in the GUI (see figure A.19). Figure A.18 describe how it is achieved: two virtual cameras with a default parallax of ($\alpha_{stereo} = 0.1$ [rad]) are defined according to equations A.3 with the two angles $\alpha_1 = \alpha_{stereo}$, $\alpha_2 = 0$. By need, the parallax can be interactively changed by the user. The points backprojected on the two different virtual cameras are drawn in red for the first camera and in green for the second camera (displaced with α_{stereo}). Using red/green glasses, the 3-D point cloud can therefore be viewed in stereo.

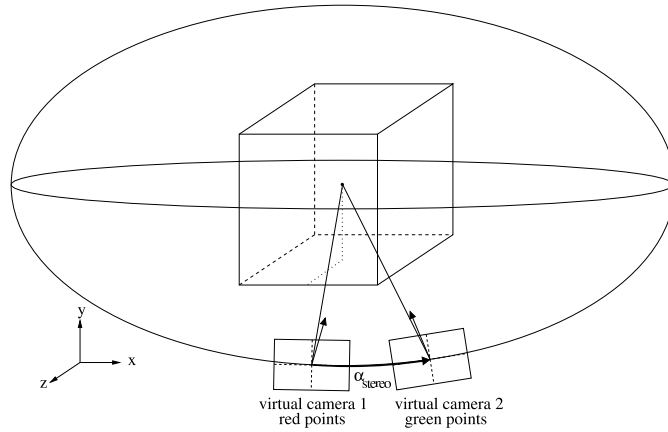


Fig. A.18 Stereo visualisation is achieved by backprojecting the 3-D points onto two virtual cameras with red and green colors (green is for the right eye, red for the left).

Some basic editing functions were implemented in the viewer. Single points can be selected to get information about the point number, the 3-D coordinates and, if available, the color information (as RGB values). Multiple points can be selected in rectangular areas or in a (clockwise) hand drawn region. The selected points can then be deleted or stored as separate data set. This offer a simple but very fast opportunity to remove outliers or unuseful points from the data (e.g., background points or points not interesting for modeling purposes).

Figure A.20 describes the strategy for the selection of points inside a clockwise hand drawn area, which is given in the form of a sequence of points (x_i, y_i) . Firstly the contour is interpolated so that the displacement in y direction is always 1 pixel (or 0 in case of horizontal parts). For all the elements of the contour, is searched the nearest point on the right with the same y, if the contour is moving up and respectively for the nearest point on the left with the same y, if the contour is moving down.

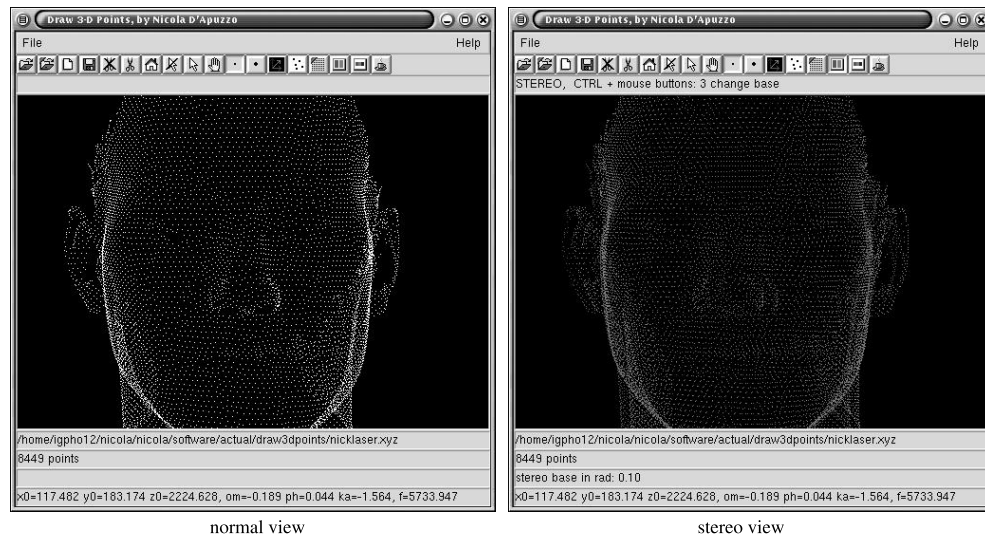


Fig. A.19 Stereo visualisation: left normal view, right: stereo view (red and green points).

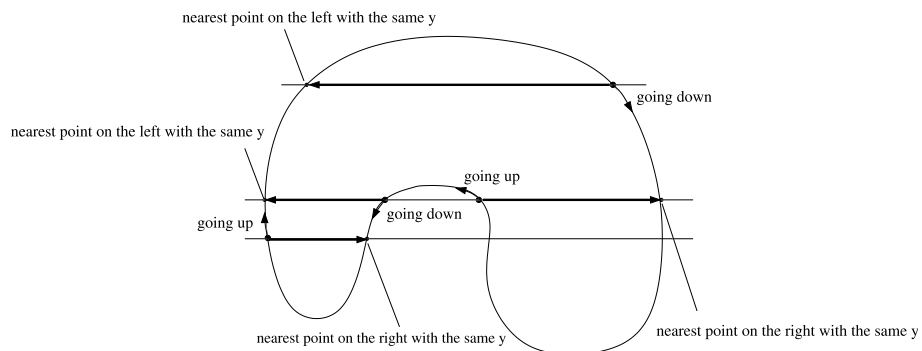


Fig. A.20 Hand drawn clockwise contour. The inside of the region is determined searching on the contour for the nearest point with the same y , on the right if the contour is going up and respectively on the left if it is going down; all the points on the horizontal line between the two found points are inside the region.

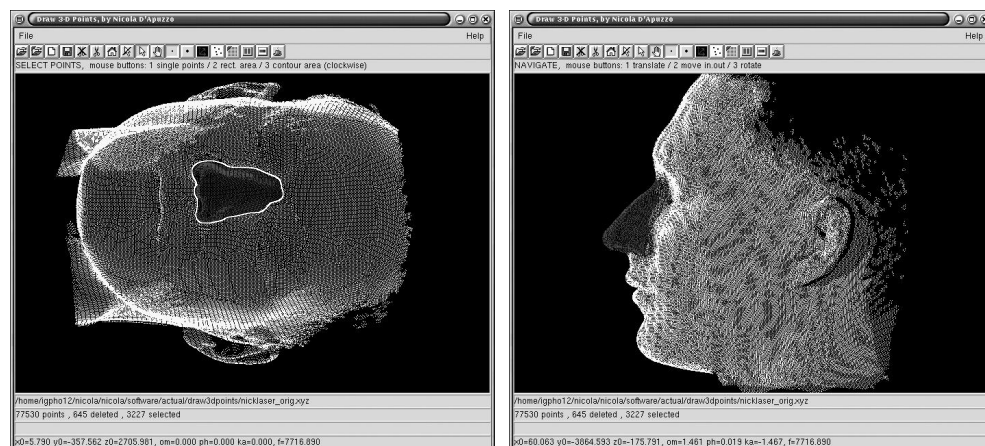


Fig. A.21 Hand drawn point selection (the selected points are drawn in grey). Left: selection, right: after turning the 3-D point cloud.

All the points lying on the segment between the two points are then defined inside the hand drawn region. At the end, an image mask defining the points inside the region (value 1 if the pixel is inside the hand drawn region and value 0 if it is outside) is obtained. For the selection of 3-D points, these are first backprojected onto the image. Their image coordinates are then compared with the image mask and the points having value 1 are selected. In this way, complex 3-D point sets can be selected easily. Figure A.21 shows an example of a hand drawn selection.

A.2.6 2-D Trajectories Viewer

The 2-D trajectory viewer is analog to the matching result viewer described in section A.2.4.



Fig. A.22 Visualization of trajectories. Top: the trajectories are displayed as points on the actual frame; bottom: without the images.

Matched and tracked points can be visualized over the multi-image sequences or alone. The sequence can be browsed and played forward or backward with variable speed. Different display modes exist for the trajectories: they can be displayed as points on the actual frame (*actual points*), as complete trajectories (*all trajs*), as complete trajectories with crosses on the actual frame (*all trajs + actual crosses*) (these two last mode are suitable only for a small number of trajectories) or as points with tail (*trajs with tail*) with variable length. Figures A.22 and A.23 show four screen shots. In the first figure on the left the trajectories are displayed as points on the actual frame, on the top right without the images. In the latter, with a tail the length of one frame, left together with the images and right alone. The figure A.24 explains the functions of the tool buttons.



Fig. A.23 Visualization of trajectories. Top: the trajectories are displayed with a tail the length of one frame; bottom: without the images.

A GRAPHICAL USER INTERFACE

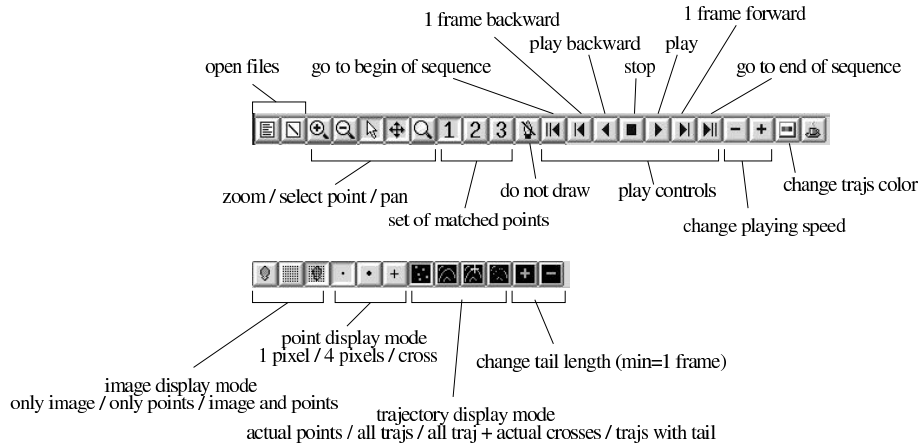


Fig. A.24 Tool buttons of the 2-D trajectories viewer.

A.2.7 3-D Trajectories Viewer / Key-Points Editor

The 3-D trajectories viewer combines the functionalities of the 2-D trajectory (section A.2.6) and the 3-D point cloud (section A.2.5) viewers. The trajectories are in this case displayed in 3-D and a real time navigation (moving the virtual camera) is possible while playing the sequence.

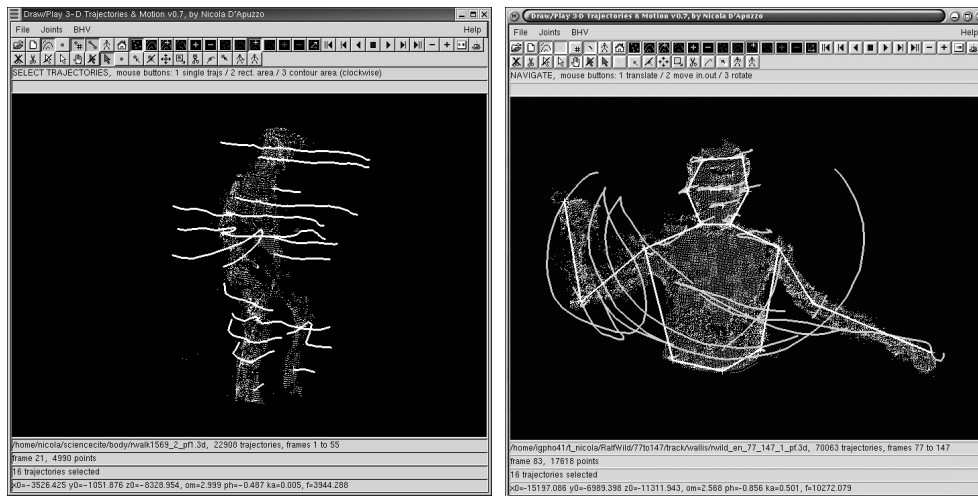


Fig. A.25 Graphical user interface of the 3-D trajectory viewer. Left: visualization of 3-D trajectories. Right: visualization of tracked key-points.

The 3-D trajectories and the trajectories of the key-points can be displayed together in different modes as explained in the previous section: points in the actual frame, complete trajectories, complete trajectories with crosses on the actual frame or as trajectories with tails. Figure A.25 shows two different combined visualizations: on the left, some selected trajectories are displayed entirely while all the trajectories are displayed as points in the actual frame; on the right, the defined key-points connections and the trajectories of the tracked key-points are displayed entirely while all the trajectories are displayed as points in the actual frame. The figure A.26 explains the functions of the various tool buttons.

A.2 OPERATING SYSTEM AND IMPLEMENTED GUI

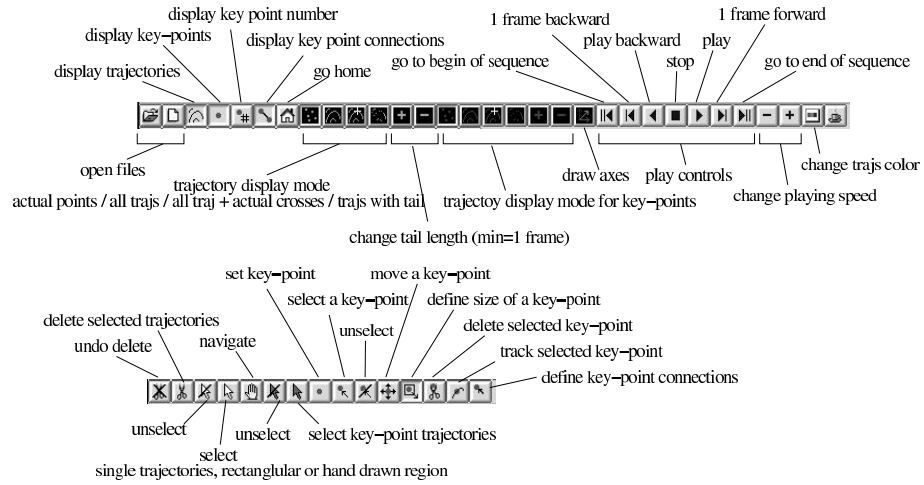


Fig. A.26 Tool buttons of the 3-D trajectories viewer software.

All the functionalities for the definition, tracking and visualization of key-points are implemented in the graphical user interface. The key-points are defined interactively in the vector field of trajectories. A key-point can be easily placed over an existing tracked point at any frame of the sequence, its position in space can then be manually adjusted by moving the key-points along the three axes (figure A.27 right). The size of the region to be tracked (defined as a cube) is also chosen interactively (see figure A.27 left).

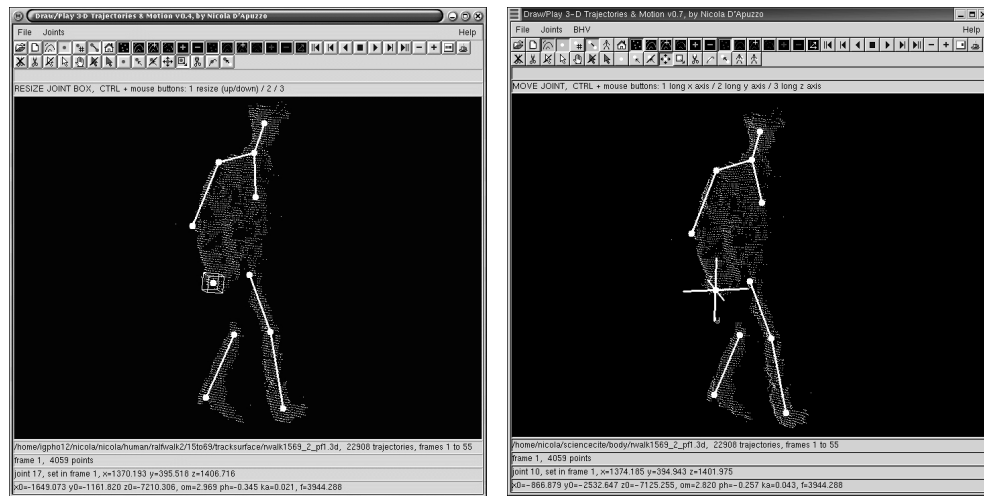


Fig. A.27 Key-point definition. Left: definition of the box size, right: moving the key-point in space.

For a nice visualization and a better understanding of the movement, connections between the tracked key-points can be defined interactively (note that they do not have any physical meaning). Figure A.28 shows an example of visualization of tracked key-points; their trajectories are displayed entirely and the connections are drawn for the actual frame, all the other trajectories are hidden. As can be seen in the figure, the connections make the understanding of the movement easier.

A GRAPHICAL USER INTERFACE

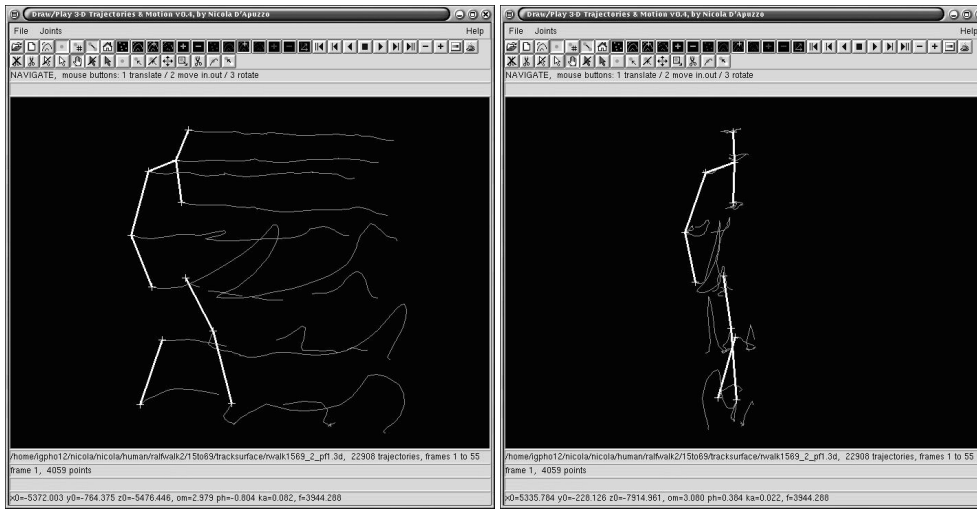


Fig. A.28 Key-point trajectory visualization, two views of the complete 3-D trajectories of the tracked key-points and visualization of the connections at the beginning of the sequence.

IEEE-1394 Camera System

B.1 MOTIVATIONS

A multi-image acquisition system based on five IEEE-1394 progressive scan CCD cameras was developed by myself at the Imaging Laboratory of the R&D Center of TOPCON Corporation in Tokyo, under the project "3-D Human Face Feature Extraction" (D'Apuzzo and Kochi, 2003).

The used cameras are provided with the IEEE-1394 port. Therefore no frame grabber is required for the acquisition of images. They can be read out accessing the IEEE-1394 port.

This appendix describes shortly the camera system and the implemented GUI for controlling the cameras and for acquiring the images.

B.2 MULTI-IMAGE ACQUISITION SYSTEM

The outline of the acquisition system is based upon the face measurement camera system described in section 5.4.

The cameras used in this case are progressive scan CCD cameras provided with a 1/3" sensor of 640x480 pixels and IEEE-1394 port. Through the IEEE-1394 interface, the computer communicates with the camera, allowing digital transmission of images as well as software control of some camera parameters. Figure B.1 shows the camera Dragonfly of Point Grey ResearchTM from the front and from the back.

Different frame rates can be set from 3.75 Hz up to 30 Hz. Other features, such as brightness, exposure, gain, shutter and white balance can be controlled through the IEEE-1394 port. The Dragonfly comes equipped with the ability to perform multiple camera synchronization. When more than one Dragonfly camera is present on the IEEE 1394 bus, the cameras automatically synchronize their acquisition time to within 20 μ s. This feature is particularly important for multi-image acquisition systems.

A c-mount lens with 16mm focal lens was chosen for the camera setup. In this way the complete system can be relatively compact in size (about 1 m wide). The lens are provided with variable iris and the focus can be fixed.

B IEEE-1394 CAMERA SYSTEM

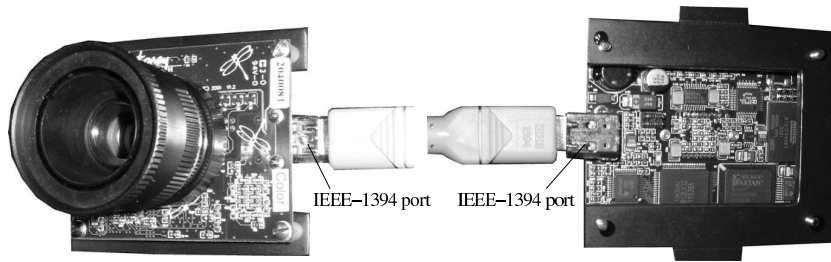


Fig. B.1 Point Grey Research camera, model Dragonfly, with c-mount 16 mm lens. Left: frontal view, right: back view.

Figure B.2 shows the camera setup. The entire system is mounted on a tripod that can move vertically and rotate around its vertical axis. The five cameras can rotate around two axes. Additionally, for the fine tuning of the system, the two external cameras (number 1 and 5) can be displaced sideways and to the front or back. The central camera (number 3) can be moved to the front and back too. Three cameras (number 1, 3 and 5) are mounted over sockets to have a difference in the heights.

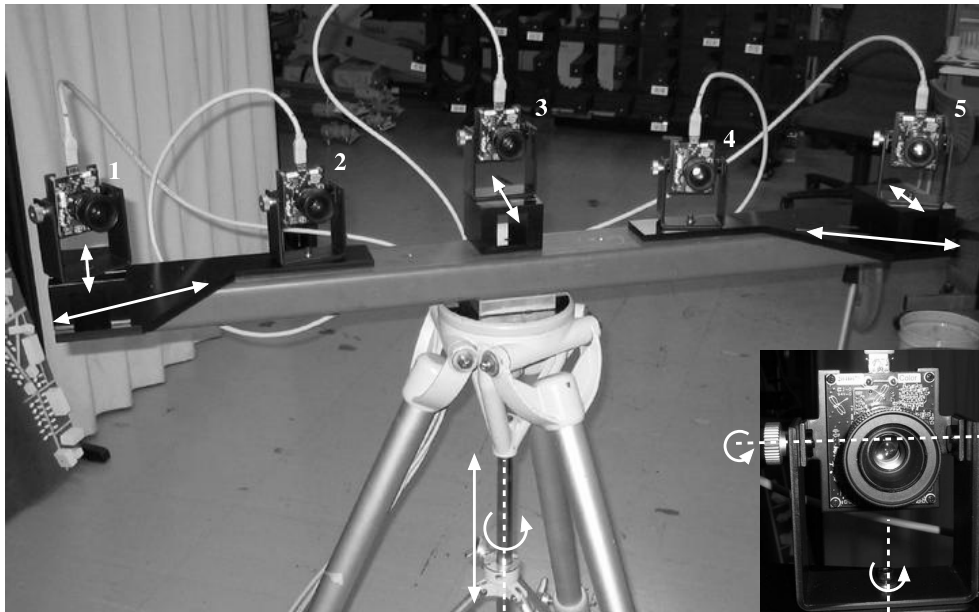


Fig. B.2 Camera setup. The entire system can move up and down and turn around the tripod axis. All the cameras can be rotated around two axes. Additionally, for fine tuning, cameras 1 and 5 can be moved sideways and to the front or back. Camera 3 can also be moved to the front or back.

B.3 IMPLEMENTED GUI

The camera control, the multi-images display and the multi-image acquisition are implemented in the same graphical user interface of the 3-D face feature extraction software. In that way, the user can control all the required processes within an unique software. The images acquired by the five cameras can be viewed in real-time, as shown in figure B.3. Figure B.4 describes the functionalities of different buttons on the toolbar.



Fig. B.3 Graphical user interface of the 3-D face feature extraction software. Live camera display mode.

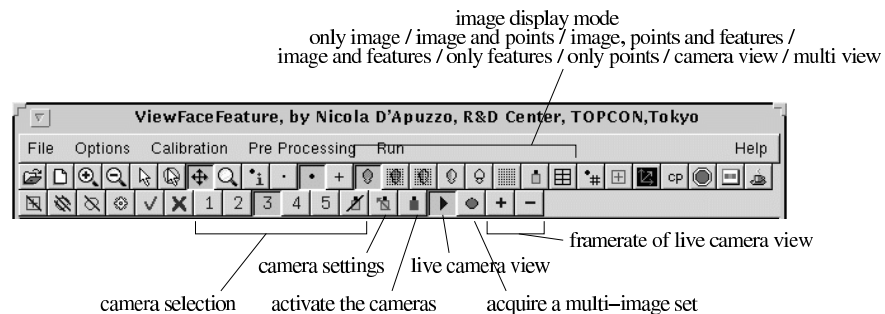


Fig. B.4 Tool buttons of the 3-D face feature extraction software (buttons with no description have same functions as in figure A.13).

As described in the previous section, some features of the cameras can be controlled by software through the IEEE-1394 port. This was implemented in a way that the five cameras have exactly the same features. Figure B.5 shows the camera setting window where information (camera name, port number, node number) about the cameras on the IEEE-1394 bus is displayed and where the user can select the different features (exposure, shutter, gain, brightness).

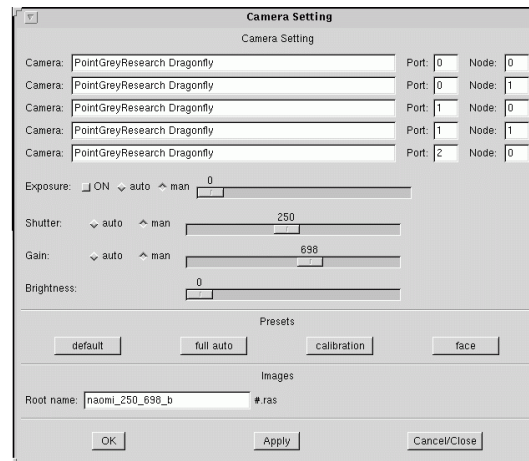


Fig. B.5 IEEE-1395 camera information and camera settings.

B.4 CALIBRATION AND ORIENTATION

A reference field with signalized points was especially designed for the five camera acquisition system to allow a full automatic calibration procedure. The 3-D coordinates of the signalized points were measured photogrammetrically by *PI-3000* system (Kochi et al., 2003) of TOPCON with a precision of 0.2 mm.

The calibration and orientation processes are integrated in the graphical user interface software and the unique task required for the user is to place the calibration field at the indicated position. The signalized points are then fully automatically recognized and measured in the different images by least squares matching. The establishment of the external and internal orientation as well as parameters modeling the lens distortion is performed by bundle adjustment. Figure B.6 shows the calibration field and the full automatic image coordinate measurement process as can be viewed in the graphical user interface.

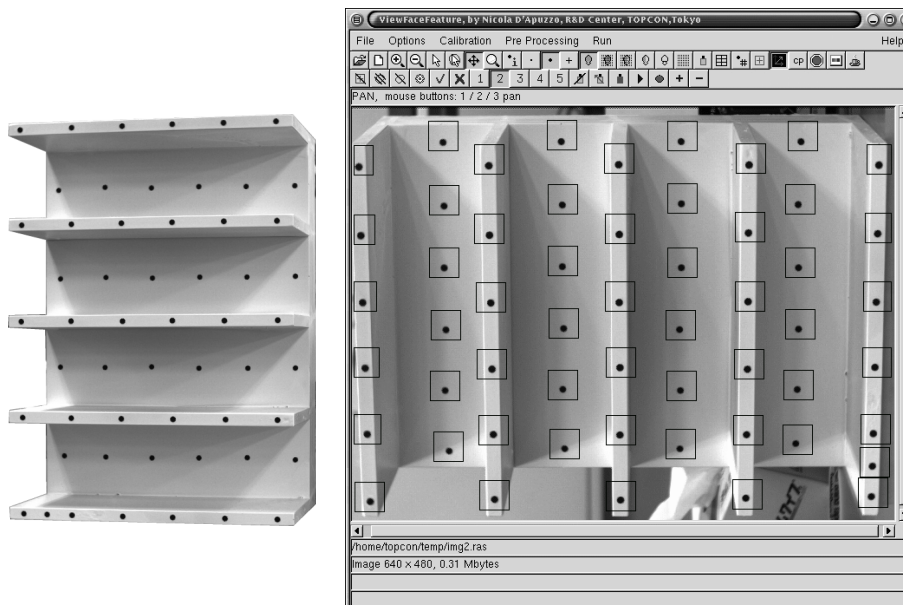


Fig. B.6 Left: calibration field. Right: full automatic measurement of signalized point.

Appendix C

Parameter Sets

This appendix contains some examples of parameter sets for the matching and tracking processes.

C.1 MATCHING PROCESS

Table C.1 lists parameter sets of the matching process for some examples; the names refers to the following figures where are displayed the template images with the used seed points and the contour defining the ROI.

Table C.1 Example of full parameter set for the matching process. The parameter names refer to section 4.2.3.2

Parameter	Max	Naoko	Naomi	Dummy	Walking	Baseball
<i>search shift</i>	1/4	1/2	1/2	1	1/2	1/2
<i>no search shift</i>	1/8	1/4	1/4	1/2	1/4	1/4
<i>bigger patch</i>	yes	yes	yes	yes	yes	yes
<i>close the gaps</i>	yes	yes	yes	yes	yes	yes
<i>used LSM par.</i>	all	all	all	all	all	all
<i>use epipolar constr.</i>	yes	yes	yes	yes	yes	yes
<i>epi weight in P</i>	4n	4n	4n	4n	n	2n
<i>epi patch size</i>	lsm/2	lsm/2	lsm/2	lsm/2	lsm/2	lsm/2
<i>max sigma 0</i>	20.0	20.0	25.0	20.0	25.0	25.0
<i>max sigma x,y</i>	0.18	0.18	0.20	0.15	0.18	0.18
<i>max x,y [pixels]</i>	2.0	2.0	1.5	1.0	2.0	2.0
<i>max x,y small [pixels]</i>	1.0	1.0	0.7	0.5	1.0	1.0
<i>patch size x,y [pixels]</i>	13	13	13	13	13	13
<i>bigger p. size x,y [pixels]</i>	21	17	21	21	19	21
<i>pixel shift [pixels]</i>	10.0	2.0	3.0	1.0	2.0	2.0
<i>voronoi x,y [pixels]</i>	5	5	5	5	5	5
<i>grid size x,y [pixels]</i>	15	15	15	15	15	15

*

C PARAMETER SETS

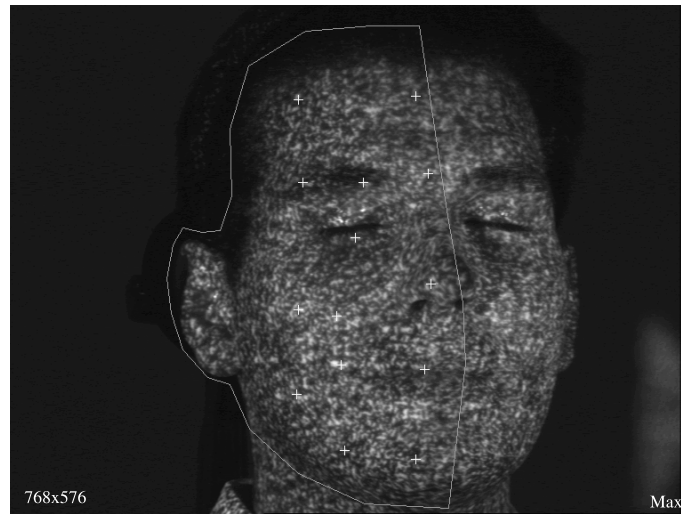


Fig. C.1 Max; face measurement with random pattern projection, image format 768x576 pixels, for demonstration purposes (shift 10 pixels). Template image with contour defining the ROI and seed points.

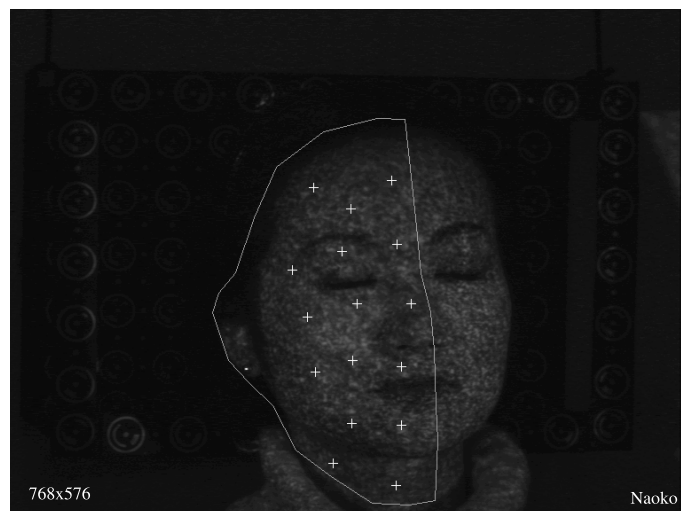


Fig. C.2 Naoko; face measurement with random pattern projection, image format 768x576 pixels, usual face measurement (shift 2 pixels). Template image with contour defining the ROI and seed points.



Fig. C.3 Naomi; face measurement without random pattern projection, image format 640x480 pixels, higher thresholds. Template image with contour defining the ROI and seed points.

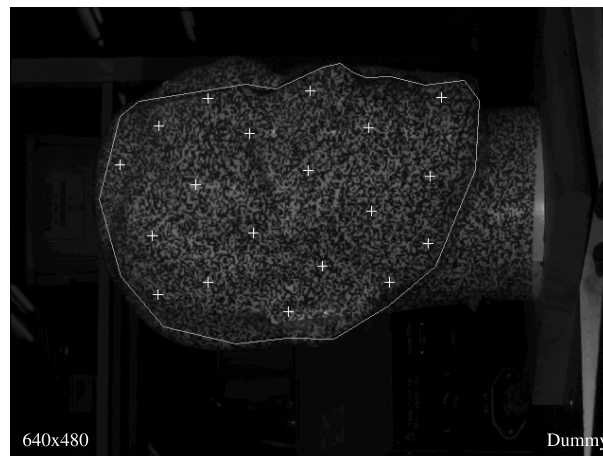


Fig. C.4 Dummy; dummy head measurement with random pattern projection, image format 640x480 pixels, fine measurement (shift 1 pixel). Template image with contour defining the ROI and seed points.

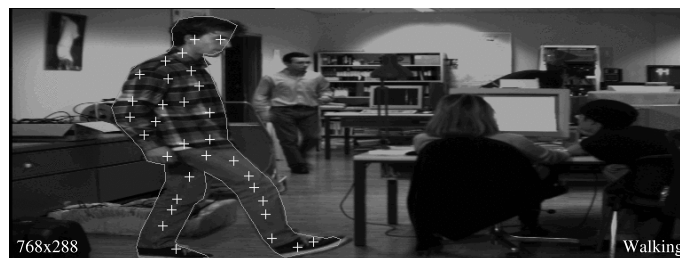


Fig. C.5 Walking; full body dynamic measurement, image format 768x288 pixels, interlaced CCD camera. Template image with contour defining the ROI and seed points.



Fig. C.6 Baseball; full body dynamic measurement, image format 640x480 pixels, progressive scan CCD camera. Template image with contour defining the ROI and seed points.

C.2 TRACKING PROCESS

An example of parameter set for the tracking process is listed below. It refers to the *walking* and *baseball* sequences presented in section 7.7.

Table C.2 Example of full parameter set for the tracking process. The parameter names refer to section 7.5

Parameter	<i>Walking</i>	<i>Baseball</i>
<i>used LSM parameters</i>	all	all
<i>patch size x,y [pixels]</i>	13	13
<i>max sigma 0</i>	25.0	25.0
<i>max sigma x,y</i>	0.25	0.25
<i>max x,y [pixels]</i>	5.0	5.0
<i>cross correlation patch size [pixels]</i>	13	13
<i>first search size x,y [pixels]</i>	101	101
<i>search size x,y [pixels]</i>	11	11
<i>large search size x,y [pixels]</i>	21	21
<i>search step x,y [pixels]</i>	2.0	2.0
<i>max diff [%]</i>	10	10
<i>tessel size x,y [pixels]</i>	10	10
<i>density threshold</i>	23	20

References

- Anai, T., Chikatsu, H., 1999. Development of human motion analysis and visualisation system. *International Archives of Photogrammetry and Remote Sensing*, 32(5-3W12):141–144.
- Anai, T., Chikatsu, H., 2000. Dynamic analysis of human motion using hybrid video theodolite. *International Archives of Photogrammetry and Remote Sensing*, 33(B5):25–29.
- Anil, A., Gabor, B., 2001. Body modeling. Semester work, Computer Sciences Department, EPF Lausanne, Lausanne, Switzerland.
- Badler, N.I., 2000. Animation 2000+. *IEEE Computer Graphics and Applications*, 20(1):28–29.
- Balder, N.I., Palmer, O., Bindiganavale, R., 1999. Animation control for real time virtual human. *Comm. ACM*, 42(8):65–73.
- Baltsavias, E.P., 1991. *Multiphoto geometrically constrained matching*. Ph.D. thesis, Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland, Mitteilungen No. 49.
- Banda, F.A.S., Muller, J.-P., Bhatia, S.N., Bukhary, M., 1992. Automatic generation of facial DEMs. *International Archives of Photogrammetry and Remote Sensing*, 29(B5):893–896.
- Bartl, J., Fira, R., Hain, M., 2001. Inspection of surface by the Moire method. *Measurement Science Review*, 1(1):29–32.
- Benninghoff, A., Gotter, K., 1961. *Lehrbuch der Anatomie des Menschen*. Urban & Schwarzenberg, München, (in German).
- Besl, P.J., McKay, N.D., 1992. A method for registration of 3-D shapes. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 14(2):239–256.
- Beyer, H.A., 1992. *Geometric and radiometric analysis of a CCD-camera based photogrammetric close-range system*. Ph.D. thesis, Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland, Mitteilungen No. 51.
- Bhatia, G., Smith, K.E., Commean, P.K., Whitestone, J., Vannier, M.W., 1994. Design of a multisensor optical surface scanner. *SPIE Proc.*, 2355:262–273.
- Blanz, V., Vetter, T., 1999. A morphable model for the synthesis of 3D faces. In *SIGGRAPH'99 Conf. Proc.*, pp. 187–194.
- Boersma, S.M., 2000. Photogrammetric wound measurement with a three-camera vision system. *International Archives of Photogrammetry and Remote Sensing*, 33(B5):84–91.
- Borghese, A., Ferrari, S., 2000. A portable modular system for automatic acquisition of 3-D objects. *IEEE Trans. on Instrumentation and Measurement*, 49(5):1128–1136.

REFERENCES

- Boulic, R., Fua, P., Herda, L., Silaghi, M., Monzani, J.-S., Nedel, L., Thalmann, D., 1998. An anatomic human for motion capture. In *Proc. of EMMSEC'98*, Bordeaux, France.
- Boulic, R., Thalmann, D., Molet, T., Haug, Z., 1997. An animation interface designed for motion capture. In *Proc. of Computer Animation '97 Conference*, IEEE Press., pp. 77–85.
- Bradt Miller, B., Gross, M., 1999. 3D whole body scans: measurement extraction software validation. In *Proc. of Conference on Digital Human Modeling for Design and Engineering*, Hague, The Netherlands.
- Brandt-Zawadzki, Z., Heiserman, J.E., 1997. The roles of MR angiography, CT angiography, and sonography in vascular imaging of the head and neck. *American Journal of Neuroradiology*, 18(10):1820–1825.
- Brown, D., 1971. Close-range camera calibration. *Photogrammetric Engineering and Remote Sensing*, 37(8):855–866.
- Brunsmann, M.A., Daanen, H., Robinette, K.M., 1997. Optimal postures and positioning for human body scanning. In *Proc. of Int. Conference on Recent Advances in 3-D Digital Imaging and Modeling*, Ottawa, Canada, pp. 266–273, 12-15 May.
- Burnsides, D.B., 1997. Software for visualisation, analysis and manipulation of laser scan images. Paper 3023-15 ASC-97-0378, Armstrong Laboratory, Wright-Patterson Air Force Base, Ohio, USA.
- Certain, A., Stuetzle, W., 1999. Automatic body measurement for mass customisation of garments. In *Proc. of Second Int. Conference in 3-D Digital Imaging and Modeling*, Ottawa, Canada, pp. 405–412.
- Cham, T.-J., Rehg, J.M., 1999. A multiple hypothesis approach to figure tracking. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2, Fort Collins, USA, pp. 239–245.
- Cheung, G.K.M., Kanade, T., Bouguet, J.-Y., Holler, M., 2000. A real time system for robust 3D voxel reconstruction of human motions. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2, South Carolina, USA, pp. 714–720.
- C.N.N., 2001. Facing up to airport security fear.
<http://europe.cnn.com/2001/US/09/28/facial.recognition.QandA>, September 28 (accessed 8.7.2002).
- Commean, P., Smith, K., Vannier, M., 1994. Automated limb prosthesis design. *Visualisation in Biomedical Computing*, 19:493–503.
- Craig, J.J., 1989. *Introduction to robotics mechanics and control*. Addison-Wesley Publishing Company.
- Cyberware, 2003. Head and face color 3D scanner model 3030.
<http://www.cyberware.com/products/psInfo.html> (accessed 8.7.2002).
- Daanen, H., Brunsmann, M.A., Taylor, S.E., Nurre, J.H., 1997. Absolute accuracy of the Cyberware WB4 whole body scanner. In Ellson, R.N., Nurre, J.H. (Eds.), *Three-Dimensional Image Capture*, Proc. of SPIE. Vol. 3023, San Jose, USA, pp. 6–12.
- D'Apuzzo, N., 1997. Automated measurement of human faces. First step: the face mask. In *Proc. of 2nd ARIDA/SGPBF Workshop on Image Sensing and its Applications*, Zurich, Switzerland, October 3.
- D'Apuzzo, N., 1998. Automated photogrammetric measurement of human faces. *International Archives of Photogrammetry and Remote Sensing*, 32(B5):402–407.
- D'Apuzzo, N., 2000. Motion capture by least squares matching tracking algorithm. In *Proc. of AVATARS'2000*. Lausanne, Switzerland, November 30 - December 1.
- D'Apuzzo, N., 2001a. Motion capture from multi image video sequences. In *Proc. of the XVIIIth Congress of the Int. Society of Biomechanics*, Zurich, Switzerland, July 8-13.

- D'Apuzzo, N., 2001b. Photogrammetric measurement and visualisation of blood vessel branching casting: a tool for quantitative accuracy test on MR-, CT- and DS- angiography. In El-Hakim, S.F., Gruen, A. (Eds.), *Videometrics and Optical Methods for 3D Shape Measurement*, Proc. of SPIE, Vol. 4309, San Jose, USA, pp. 34–39.
- D'Apuzzo, N., 2002a. Measurement and modeling of human faces from multi images. *International Archives of Photogrammetry and Remote Sensing*, 34(B5):241–246.
- D'Apuzzo, N., 2002b. Modeling human faces with multi-image photogrammetry. In Corner, B.D., Pargas, R., Nurre, J.H. (Eds.), *Three-Dimensional Image Capture and Applications V*, Proc. of SPIE, Vol. 4661, San Jose, USA, pp. 191–197.
- D'Apuzzo, N., 2002c. Surface measurement and tracking of human body parts from multi image video sequences. *ISPRS Journal of Photogrammetry and Remote Sensing*, 56(5-6):360–375 (Theme Issue "Medical Imaging and Photogrammetry").
- D'Apuzzo, N., 2003. Human body motion capture from multi-image video sequences. In El-Hakim, S.F., Gruen, A., Walton, J.S. (Eds.), *Videometrics VIII*, Proc. of SPIE, Vol. 5013, Santa Clara, USA, pp. 54–61.
- D'Apuzzo, N., Kochi, N., 2003. Three-dimensional human face feature extraction from multi images. In Gruen, A., Kahmen, H. (Eds.), *Optical 3-D Measurement Techniques VI*, Vol. I, Zurich, Switzerland, pp. 140–147.
- D'Apuzzo, N., Maas, H.-G., 1999. On the suitability of digital camcorders for virtual reality image data capture. In El-Hakim, S.F., Gruen, A. (Eds.), *Videometrics VI*, Proc. of SPIE, Vol. 3641, San Jose, USA, pp. 259–267.
- D'Apuzzo, N., Plänkner, R., 1999. Human body modeling from video sequences. *International Archives of Photogrammetry and Remote Sensing*, 33(5-3W12):133–140.
- D'Apuzzo, N., Plänkner, R., Fua, P., 2000. Least squares matching tracking algorithm for human body modeling. *International Archives of Photogrammetry and Remote Sensing*, 33(B5):164–171.
- D'Apuzzo, N., Plänkner, R., Fua, P., Gruen, A., Thalmann, D., 1999. Modeling human bodies from video sequences. In El-Hakim, S., Gruen, A. (Eds.), *Videometrics VI*, Proc. of SPIE, Vol. 3641, San Jose, USA, pp. 36–47.
- D'Apuzzo, N., Willneff, J., 2001. Extraction of metric information from video sequences of an unsuccessfully controlled chimneys demolition. In Gruen, A., Kahmen, H. (Eds.), *Optical 3-D Measurement Techniques V*, Vienna, Austria, pp. 259–265.
- De Carlo, D., Metaxas, D., Stone, M., 1998. An anthropometric face model using variational techniques. In *Proc. of SIGGRAPH '98*, pp. 67–74.
- Dekker, L., Douros, I., Buxton, B.F., Treleaven, P., 1999. Building symbolic information for 3D human body modeling from range data. In *Proc. of the Second Int. Conf. in 3-D Digital Imaging and Modeling*, Ottawa, Canada, pp. 388–397.
- Delamarre, Q., Faugeras, O., 1999. 3D articulated models and multi-view tracking with silhouettes. In *Proc. of the 7th IEEE Int. Conf. on Computer Vision*, Vol. 1, Kerkyra, Greece, pp. 716–721.
- Deutscher, J., Blake, A., Reid, I., 2000. Articulated body motion capture by annealed particle filtering. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2, South Carolina, USA, pp. 126–133.
- Dorffner, L., 1996. A new approach for spatial measurement of dynamical processes. *International Archives of Photogrammetry and Remote Sensing*, 31(B3):135–138.
- Ertaud, J.Y., Savatier, X., Schmidt, W., Thomine, J.M., Dujardin, F.H., 1999. Multivision determination of the volume of human body segments and implementation of a dynamic model. *Human Movement Science*, 18(5):647–654.

REFERENCES

- Foerstner, W., Guelch, E., 1987. A fast operator for detection and precise location of distinct points, corners and center of circular features. In *Proc. of ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*, Interlaken, Switzerland, pp. 281–305, June 2–4.
- Frankowski, G., Chen, M., Huth, T., 2000. Real-time 3D shape measurement with digital stripe projection by Texas Instruments micromirror devices DMD. In Corner, B., Nurre, J. (Eds.), *Three-Dimensional Image Capture and Application III*, Proc. of SPIE, Vol. 3958, San Jose, USA, pp. 90–105.
- Frankowski, G., Chen, M., Huth, T., 2001. Optical 3D in vivo measurement of human skin by prisms. In Corner, B., Nurre, J. (Eds.), *Proc. of the 3rd Int. Workshop on Automatic Processing of Fringe Patterns*, Bremen, Germany, pp. 533–539, September 15–17.
- Frobin, W., Hierholzer, E., 1983. Automatic measurement of body surfaces using rasterstereography. *Photogrammetric Engineering and Remote Sensing*, 49(10):1443–1452.
- Fua, P., 1999. Using model-driven bundle-adjustment to model heads from raw video sequences. In *Proc. of the 7th Int. Conf. on Computer Vision*, Corfu, Greece, pp. 46–53.
- Fua, P., 2000. Regularized bundle-adjustment to model heads from image sequences without calibration data. *Int. Journal of Computer Vision*, 38(2):153–171.
- Fua, P., Gruen, A., D'Apuzzo, N., Plänters, R., 2002. Markerless full body shape and motion capture from video sequences. *International Archives of Photogrammetry and Remote Sensing*, 34(B5):256–261.
- Fua, P., Gruen, A., Plänters, R., D'Apuzzo, N., Thalmann, D., 1998. Human body modeling and motion analysis from video sequences. *International Archives of Photogrammetry and Remote Sensing*, 32(B5):866–873.
- Fua, P., Herda, L., Plänters, R., Boulic, R., 2000. Human shape and motion recovery using animation models. *International Archives of Photogrammetry and Remote Sensing*, 33(B5):253–268.
- Fua, P., Miccio, C., 1997. Fitting sophisticated facial animation models to image data. *Optical 3D Measurement Techniques IV*, pp. 3–10.
- Gaebel, H., Kakoschke, D., 1996. Photogrammetric quantification of changes of soft tissue after skeletal treatment of the facial part of the skull. *International Archives of Photogrammetry and Remote Sensing*, 31(B5):188–193.
- Gaebel, H., Wester-Ebbinghaus, W., Woytowicz, D., Hallbauer, D., Schumpe, G., 1992. Photogrammetric measurement of the human back shape and its relation to the spine. *International Archives of Photogrammetry and Remote Sensing*, 29(B5):862–866.
- Gravila, D.M., Davis, L., 1996. 3-D model based tracking of humans in action: a multi-view approach. In *Proc. Conference on Computer Vision and Pattern Recognition*, San Francisco, USA, pp. 73–80.
- Gruen, A., 1985. Adaptive least squares correlation: a powerful image matching technique. *South African Journal of Photogrammetry, Remote Sensing and Cartography*, 14(3):175–187.
- Gruen, A., Baltsavias, E.P., 1988. Geometrically constrained multiphoto matching. *Photogrammetric Engineering and Remote Sensing*, 54(5):633–641.
- Gruen, A., Baltsavias, E.P., 1989. Automatic 3-D measurement of human faces with CCD-cameras. *Proc. of SPIE*, 1030:106–116.
- Gruen, A., Beyer, H.A., 2001. System calibration through self-calibration. In A. Gruen and T.S. Huang, editors, *Calibration and Orientation of Cameras in Computer Vision*, Information Sciences 34, pp. 163–193.

- Gruen, A., Zhang, L., 2002. Automatic DTM generation from three-line-scanner (TLS) images. *International Archives of Photogrammetry and Remote Sensing*, 34(A3):131–137.
- Hackenberg, L., Liljenqvist, U., Hierholzer, E., Halm, H., 2000. Scanning stereographic surface measurement in idiopathic scoliosis after VDS (ventral derotation spondylodesis). *Zeitschrift fuer Orthopadie und ihre Grenzgebiete*, 138(4):353–359.
- Haritaoglu, I., Harwood, D., Davis, L.S., 1998. W4: Who? When? Where? What? A real time system for detecting and tracking people. In *Proc. of the 3rd Int. Conference on Face and Gesture Recognition*, Nara, Japan.
- Hasegawa, K., Hattori, K., Sato, Y., 1999. A high speed face measurement system. In *Proc. of Vision Interface '99*, Trois-Rivieres, Canada, pp. 196–2002.
- Herda, L., Fua, P., Plaenkers, R., Boulic, R., Thalmann, D., 2000. Skeleton-based motion capture for robust reconstruction of human motion. In *Proc. of Computer Animation 2000*, IEEE CS Press.
- Hilton, A., Gentils, T., 1999. Virtual people: capturing human models to populate virtual worlds. In *IEEE Int. Conf. on Computer Animation*, pp. 174–185.
- Höflinger, W., 1996. Digital stereophotogrammetric solutions for orthodontics. *International Archives of Photogrammetry and Remote Sensing*, 31(B5):247–252.
- Horiguchi, C., 1998. BL (body line) scanner. The development of a new 3D measurement and reconstruction system. *International Archives of Photogrammetry and Remote Sensing*, 32(B5):421–429.
- Ip, H.H.S., Yin, L., 1996. Constructing a 3D individualized head model from two orthogonal views. *The Visual Computer*, 12:254–266.
- Iwai, Y., Ogaki, K., Yachida, M., 1999. Posture estimation using structure and motion models. In *Proc. of 7th IEEE Int. Conf. on Computer Vision*, Vol. 1, Kerkyra, Greece, pp. 214–219.
- Jojic, N., Turk, M., Huang, T.S., 1999. Tracking self-occluding objects in dense disparity maps. In *Proc. of 7th IEEE Int. Conf. on Computer Vision*, Vol. 1, Kerkyra, Greece, pp. 123–130.
- Jones, P.R.M., West, G., Brooke-Wavell, K., 1993. Interrogation of 3D body data for applications in manufacturing industries. In *Application of Computers to Manufacturing Engineering*, Sheffield, USA, pp. 20–25.
- Jordan, P., Willneff, J., D'Apuzzo, N., Weishaupt, M., Wistner, T., Auer, J., 2001. Photogrammetric measurement of deformations of horse hoof horn capsules. In El-Hakim, S.F. and Gruen, A. (Eds.), *Videometrics and Optical Methods for 3D Shape Measurement*, Proc. of SPIE, Vol. 4309, San Jose, USA, pp. 204–211.
- Ju, X., Siebert, J. P., 2001a. Individualizing human animation models. In *Proc. of Eurographics 2001*. Manchester, UK.
- Ju, X., Siebert, P., 2001b. Conforming generic animatable models to 3D scanned data. In *Proc. of 6th Numerisation 3D/Scanning 2001*, Paris, France.
- Kadaba, M.P., Stine, R., April 12–15 2000. Real-time movement analysis: techniques and concepts for the new millenium. In *Proc. of 5th Annual Gait and Clinical Movement Analysis Meeting*, Rochester, USA, April 12–15.
- Kaas, M., Witkin, A., Terzopoulos, D., 1998. Snakes: active contour models. *IJCV*, 1(4):763–768.
- Kelekis, N.L., Semelka, R.C., Worawattanakul, S., Molina, P.L., Mauro, M.A., 1999. Magnetic resonance imaging of the abdominal aorta and iliac vessels using combined 3-D gadolinium-enhanced MRA and gadolinium-enhanced fat-suppressed spoiled gradient echo sequences. *Magnetic Resonance Imaging*, 17(5):641–651.

REFERENCES

- Kinzel, W., Behring, R., 1995. Initializing the recognition of moving persons. *International Archives of Photogrammetry and Remote Sensing*, 30(5W1).
- Koch, R.M., Gross, M.H., Carls, F.R., von Bueren, D.F., Frankhauser, G., Parish, Y.I.H., 1996. Simulating facial surgery using finite element models. In *Proc. of SIGGRAPH'96*, New Orleans, USA, pp. 421–428, August 4-9.
- Kochi, N., Ito, T., Noma, T., Otani, H., Nishimura, S., Ito, J., 2003. PC-based 3D image measuring station with digital camera. An example of its actual application on a historical ruin. *International Archives of Photogrammetry and Remote Sensing*, 34(5W12):195–199.
- Lee, W.-S., Magnenat-Thalmann, N., 2000. Fast head modeling for animation. *Image and Vision Computing*, 18(4):335–364.
- Liu, Z., Zhang, Z., Jacobs, C., Cohen, M., 2000. Rapid modeling of animated faces from video. In *Proc. of the 3rd Int. Conf. on Visual Computing*, Mexico City, Mexico, pp. 58–67.
- Luhmann, T., 2000. *Nahbereichsphotogrammetrie*. Herbert Wichmann Verlag, Heidelberg (in German).
- Maas, H.-G., 1992. Robust automatic surface reconstruction with structured light. *International Archives of Photogrammetry and Remote Sensing*, 29(B5):709–713.
- Maas, H.-G., 1997. Concepts of real-time photogrammetry. *Human Movement Science*, 16:189–199.
- Maas, H.-G., 1998. Image sequence based automatic multi-camera system calibration techniques. *International Archives of Photogrammetry and Remote Sensing*, 32(B5):763–768.
- Mahoney, D.P., 2000. A new track for modeling human motion. *Computer Graphics World*, pp. 18–20, May.
- Mao, Z., Siebert, P., Ayoub, A.F., 2000. Development of 3D measuring techniques for the analysis of facial soft tissue change. In *Proc. of Medical Image Computing and Computer-Assisted Intervention 2000 (MICCAI2000)*, Pittsburgh, USA.
- Marschner, S.R., Guenter, B., Raghupathy, S., 2000. Modeling and rendering for realistic facial animation. In *Proc. of the 11th Eurographics Rendering Workshop*, Brno, Czech Republic.
- Marschner, S.R., Westin, S.H., Lafortune, E.P.F., Torrance, K.E., Greenberg, D.P., 1999. Image-based BRDF measurement including human skin. In *Proc. of Eurographics Workshop on Rendering*, pp. 131–144.
- Marshall, S.J., Gilby, J.H., 2001. New opportunities in non-contact 3D measurement. In *Proc. of National Measurement Conf. 2001*, Harrogate, UK.
- Marshall, S.J., Whiteford, D.N., Rixon, R.C., 2001. Assessing the performance of 3D whole body imaging systems. In *Proc. 6th Numerisation 3D/Scanning 2001 Congress*, Paris, France.
- Matsumoto, Y., Fujimura, K., Kitamura, T., 1999. Cybermodeler: a compact 3D scanner based on monoscopic camera. In Nurre, J.H., Corner, B. (Eds.), *Three-Dimensional Image Capture and Applications II*, Proc. of SPIE, Vol. 3640, pp. 3–10.
- McKenna, P., 1999. Measuring up. <http://www.af.mil/news/airman/0296/measurin.htm> (accessed 8.7.2002).
- Minakuchi, S., Hirano, Y., Sekita, T., Suzuki, T., Kobayashi, K., Nagao, M., 1995. Three dimensional analysis of lip movement by 3-D auto tracking system. *International Archives of Photogrammetry and Remote Sensing*, 30(5W1).
- Mitchell, H.L., 1992. An approach to digital photogrammetry for body surface measurement. *International Archives of Photogrammetry and Remote Sensing*, 29(B5):856–861.

- Motegi, N., Tsutsumi, S., Wakatsuki, E., 1996. A facial growth analysis based on FEM employing three dimensional surface measurement by a rapid laser device. *Okajimans Folia Anatomica Japonica*, 72(6):323–328.
- Narayanan, P.J., Rander, P.W., Kanade, T., 1998. Constructing virtual words using dense stereo. In *Proc. of the Sixth Int. Conf. on Computer Vision*, Bombay, India, pp. 3–10.
- Nebel, J.-C., 2000. Soft tissue modelling from 3D scanned data. In *Proc. of DEFORM2000*, Geneva, Switzerland.
- Nebel, J.-C., Rodriguez-Miguel, F.J., Cockshott, W.P., 2001. Stroboscopic stereo rangefinder. In *Proc. 3rd Int. Conf. on 3D Imaging and Modelling (3DIM2001)*, Quebec City, Canada.
- Nef, M., 2001. Automatic extraction of silhouettes from video sequences. Semesterarbeit, Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland.
- Niederoest, M., 1996. Codierte zielmarken in der digitalen nahbereichsphotogrammetrie. Master's thesis, Institut fuer Geodaesie und Photogrammetrie, ETH Zuerich (in German).
- Ohno, H., Yamamoto, M., 1999. Gesture recognition using character recognition techniques on two-dimensional eigenspace. In *Proc. of 7th IEEE Conf. on Computer Vision*, Vol. 1, Kerkyra, Greece, pp. 151–156.
- Okada, E., 2001. Three-dimensional facial simulations and measurement: changes of facial contour and units associated with facial expression. *Journal of Craniofacial Surgery*, 12(2):167–174.
- Ono, T., 1995. Trunk deformity in scoliosis studied by surface measurement. *Nippon Seikeigeka Gakkai Zasshi Journal*, 69(10):915–926.
- Paquet, E., Rioux, M., 1999. Neferiti: a query by content software for three-dimensional models database management. *Image and Vision Computing*, 17(2):157–166.
- Pighin, F., Hecker, J., Lischinski, D., Szeliski, R., Salesin, D.H., 1998. Synthesizing realistic facial expressions from photographs. In *Proc. of SIGGRAPH '98*, pp. 75–84.
- Plänkers, R., 2001a. *Human body modeling from video sequences*. Ph.D. thesis, Laboratoire d'infographie, EPF Lausanne, Switzerland.
- Plänkers, R., 2001b. Tracking and modeling people in video sequences. *Int. Journal of Computer Vision and Image Understanding*, 81(3):285–302.
- Plänkers, R., Fua, P., D'Apuzzo, N., 1999. Automated body modeling from video sequences. In *Proc. of IEEE Int. Workshop on Modeling People (mPeople)*, Corfu, Greece, September 20.
- Proesmans, M., Van Gool, L., 1996. Reading between the lines. In *SIGGRAPH'96 Conf. Proc.*, pp. 55–62.
- Remondino, F., 2003. 3D reconstruction of static human body with a digital camera. In El-Hakim, S.F., Gruen, A., Walton, J.S. (Eds.), *Videometrics VIII*, Proc. of SPIE, Vol. 5013, Santa Clara, USA, pp. 38–45.
- Richards, J.G., 1999. The measurement of human motion: a comparison of commercially available systems. *Human Movement Science*, 18:589–602.
- Robinette, K.M., Daanen, H., 1999. The Caesar project: a 3-D surface anthropometry survey. In *Proc. of the 2nd Int. Conference on 3-D Digital Imaging and Modeling*, Ottawa, Canada, pp. 380–386.
- Ronsky, J.L., Boyed, S.K., Lichti, D.D., Chapman, M.A., Salkauskas, K., 1999. Precise measurement of cat patellofemoral joint surface geometry with multistation digital photogrammetry. *Journal of Biomechanical Engineering*, 121(2):196–205.

REFERENCES

- Rosales, R., Sclaroff, S., 2000. Inferring body pose without tracking body parts. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2, South Carolina, USA, pp. 721–727.
- Rubin, G.D., Paik, D.S., Johnston, P.C., Napel, S., 1998. Measurement of the aorta and its branches with helical CT. *Radiology*, 206(3):823–829.
- Savatier, X., Ertaud, J.Y., Dujardin, F.H., 2001. Three-dimensional multivision determination of human body segment characteristics. In *Proc. of the XVIIIth Congress of the Int. Society of Biomechanics*, Zurich, Switzerland, 8–13 July.
- Schneeberger, K., Stamm, C., Maetzler, C., Fluehler, H., Lehmann, E., Willneff, J., 2002. Multifrequency ground-based radiometer and in-situ measurements of soil moisture at high temporal resolution. In *9th International Symposium on Remote Sensing*, Proc. of SPIE, Vol. 4879, Crete, Greece.
- Segawa, H., Totsuka, T., 1999. Torque-based recursive filtering approach to the recovery of 3D articulated motion from image sequences. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2, Fort Collis, USA, pp. 340–435.
- Shan, Y., Liu, Z., Zhang, Z., 2001. Model-based bundle adjustment with application to face modeling. In *Proc. of the 8th Int. Conf. on Computer Vision (ICCV01)*, Vol. 2, Vancouver, Canada, pp. 644–651.
- Siebert, J., Marshall, S., 2000. Human body 3D imaging by speckle texture projection photogrammetry. *Sensor Review*, 20(3):218–226.
- Siebert, P., Ju, X., 2000. Realistic human animation using scanned data. In *Proc. of 5th Numerisation 3D/Scanning 2000 Congress*, Paris, France.
- Sitnik, R., Kujawinska, M., 2000. Opto-numerical methods of data acquisition for computer graphics and animation systems. In Corner, B.D., Nurre, J.H. (Eds.), *Three-Dimensional Image Capture and Application III*, Proc. of SPIE, Vol. 3958, San Jose, Usa, pp. 36–43.
- Skutta, B., Fürst, G., Eilers, J., Ferbert, A., Kuhn, F., 1999. Intracranial stenocclusive disease: double-detector helical ct angiography versus digital subtraction angiography. *American Journal of Neuroradiology*, 20(5):791–799.
- Sommer, T., Fehske, W., Holznecht, N., Smekal, A.V., Keller, E., Lutterbey, G., Kreft, B., Kuhl, C., Gieseke, J., Ramadan, D. A., Schild, H., 1996. Aortic dissection: a comparative study of diagnosis with spiral CT, multiplanar transesophageal echography, and MR imaging. *Radiology*, 199(2):347–352.
- Song, Y., Feng, X., Perona, P., 2000. Towards detection of human motion. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 1, South Carolina, USA, pp. 810–817.
- The Imaging Source, 2001a. Introduction to cameras.
http://www.imagingsource.com/resources/background/intro-to-cameras_eng.pdf (accessed 8.7.2002).
- The Imaging Source, 2001b. Introduction to frame grabbers.
http://www.imagingsource.com/resources/background/intro-to-grabbers_eng.pdf (accessed 8.7.2002).
- Thomas, P.R., Newton, I., 1996. Evaluation of low cost digital photogrammetric system for medical applications. *International Archives of Photogrammetry and Remote Sensing*, 31(B5):405–410.
- Tsuruoka, M., Shibasaki, R., Box, E.O., Murai, S., 1995. Biomechanical 3-D analysis of a human sit-to-standing sequence using two CCD video cameras. *International Archives of Photogrammetry and Remote Sensing*, 30(5W1).

REFERENCES

- Tukuisis, P., Meunier, P., Jubenville, C.E., 2001. Human body surface area: measurement and prediction using three dimensional body scans. *European Journal of Applied Physiology and Occupational Physiology*, 85(3-4):264–271.
- Vedula, S., Rander, P., Saito, H., Kanade, T., 1998. Modeling, combining and rendering dynamic real world events from image sequences. In *Proc. of the 4th Conference on Virtual Systems and Multimedia*, Vol. 1, pp. 326–332.
- Wallis, R., 1976. An approach to the space variant restoration and enhancement of images. In *Proc. of Symposium on Current Mathematical Problems in Image Science*, Naval Postgraduate School, Monterey, USA, pp. 329–340.
- Wolf, H., 1996. Structured lighting for upgrading 2D-vision system to 3D. In *Proc. of Int. Symposium on Laser, Optics and Vision for Productivity and Manufacturing I*, Besancon, France, pp. 10–14, June 10-14.
- Youmei, G., 1994. Non-contact 3D biological shape measurement from multiple views. Master's thesis, University of Western Australia.
- Zheng, J., 1994. Acquiring 3-D models from sequences of contours. *IEEE Trans. Patt. Anal. Machine Intell.*, 16(2):163–178.

Index

- Affine transformation, 27, 31, 36, 47, 51, 119
- Automatic matching process, 36, 116
- Backprojection, 60
- Calibration, 19, 22, 69
 - bundle, 24, 60, 62
 - reference bar, 22
 - reference field, 22, 62
- Camcorder, 14, 62, 101
- CCD camera, 9, 62, 68
 - interlaced, 13, 94
 - progressive scan, 10, 13, 94, 129
- Close the gaps, 43, 48–49
- Coded light, 61, 107
- Coded target, 22, 62, 69, 113
- Collinearity equation, 20, 56, 58
- Coplanarity condition, 29
- Corresponding points, 40, 56, 64
- Covariance matrix, 25, 29, 57
- Cross-correlation, 39, 81–82
- Data acquisition, 9
- Delauney triangulation, 59, 64
- Design matrix, 24–25, 28, 31, 56
- Digital still camera, 16
- Digital video camcorder, 16
- Distortion terms
 - decentering, 21
 - radial, 21
- Editing, 58
- Epipolar
 - constant, 31–32, 34
 - constraint, 31–32, 36–37, 47, 49
 - line, 29, 32, 47, 49
 - plane, 29
 - segment, 32–33, 47, 49
- Face modeling, 61
- Filtering, 50, 57, 86
 - neighborhood, 51, 116
 - regularization of grid, 50, 116
- FireWire, 15, 17
- Foerstner interest operator, 39, 46
- Forward intersection, 64, 69, 84
- Frame grabber, 11
- Gaussian filters, 64
- Gauss-Markov model, 25, 29, 57
- Genlock, 11
- Geometrical correction, 115
- GUI, 111
- IEEE-1394, 13, 17, 129
- I-link, 17
- Image coordinate system, 32
- Image patch, 27, 31, 34, 36, 51
- Interlacing, 10, 14
- Key-point, 88, 98, 127
 - tracking, 88, 98
- Laser scanning, 61, 107
- Least squares estimation, 27, 56
- Least squares matching, 27, 81–82
 - spatial, 81–82
 - temporal, 81–82
- Lens distortion, 21
- Linear prediction, 82
- Local uniformity, 51
 - filtering, 87
- Matching process, 64, 69, 84
- Matching strategy, 40
- Metric coordinate system, 32
- Modeling, 59
- Motion capture, 94
- Multi-images, 12, 35
- Object coordinate system, 20
- Observation equation, 24, 27, 31
- Observation vector, 24–25, 31, 56
- Operating System, 111
- Orientation
 - exterior, 20, 23, 69, 114
 - interior, 20, 23
- Patch size, 47, 49
- Point cloud, 55, 119

- Point viewer, 58
- Principal point, 19
- Projection center, 19, 29
- Radiometric correction, 27, 31, 36, 47
- Random pattern, 62, 68–69
- Region of interest, 40, 116
- Resampling
 - bilinear, 28, 58
 - linear, 29
- Residual vector, 25, 29, 57
- Result check, 35
- Ring buffer, 90
- Rotation matrix, 19–20
- Search box, 82
- Search image, 27, 31, 34, 37
- Search region, 46
- Search strategy, 48, 116
- Seed point definition, 116
 - automatic, 39, 46, 95
 - manual, 37, 46
 - semi-automatic, 37, 46–47, 64
- Seed points, 36
- Self calibrating bundle adjustment, 22
- Sensor coordinate system, 20
- Solution vector, 25, 29, 57
- Spatial resection, 24
- Stereo matcher, 27, 34
- Surface measurement, 55, 63, 95
- Surface tracking, 84
- Synchronization, 10–11
 - clapper, 15
 - genlock, 10
- Template image, 27, 31, 34–35, 37
- Texture coordinate, 60
- Texture mapping, 59, 65
- Tracking loop, 81
- Tracking process, 81, 84, 97
- Trajectory, 84, 86, 124
 - vector field, 84, 86, 88
- Transformation
 - affine, 27, 31, 36, 47, 51, 119
 - metric to pixel, 21
 - pixel to metric, 21
- Validation, 71
- Variance factor, 25, 29, 57
- Vector difference, 31
- Video signal, 9
 - CCIR, 9
 - EIA, 9
- Virtual camera, 119
- Visualisation, 64
 - point cloud, 58
- Voronoi tessellation, 40, 49
- VRML, 59
- Wallis filter, 45, 95, 116
- Web camera, 17
- Weight coefficient matrix, 29, 31

Acknowledgments

I would like to thank Prof. Dr. Armin Gruen at the Institute of Geodesy and Photogrammetry of the ETH Zurich for giving me the opportunity to work on interesting projects and for providing me with excellent research environment.

I thank Professor Hans-Gerd Maas who gave me the C-routines from which I started my project and for important advice during the course of my research.

I thank Dr. Pascal Fua and Dr. Ralf Plaenkers from the Laboratoire d'Infographie at the EPF Lausanne for providing me with multi-image sequences.

I thank Mr. Nobuo Kochi of the Imaging Laboratory of TOPCON Corporation in Tokyo for allowing me to continue working on my thesis during my stay in Japan.

I thank Prof. Dr. Hans-Peter Meinzer of the German Cancer Research Center in Heidelberg and Prof. Dr. Henrik Haggrén of the Institute of Photogrammetry and Remote Sensing at the Helsinki University of Technology in Espoo for being coadvisors for this work.

I thank my friends who kindly posed as models for the face measurement studies and all my colleagues and friends in the group of Photogrammetry and Remote Sensing at the ETH Zurich for their assistance in various technical fields and for their friendships.

Special thank are due to my brother Massimo and my sister-in-law Chantal for their support, friendship and for correcting my English during the long writing process.

Finally, I thank from heart my family and my close friends who gave my life richness in so many ways that I can not list them all here.

N. D'A.